Linear Error-correcting Codes

Eric Filiol

ESIEA - Laval
Laboratoire de cryptologie et de virologie opérationnelles
$(C+V)^O$
filiol@esiea.fr

2013 - 2014

Agenda

Agenda

Introduction

From the general introduction on error-correcting codes we have seen that :

- If the minimum distance id $2e + 1$ we can correct up to e errors reliably.
- The number $|\mathcal{C}| = A_q(n, d)$ of codewords is a key parameter.
- Whenever $|\mathcal{C}|$ is large, the coding and decoding process can be very time- and memory-consuming.
- It is then interesting to consider highly structured codes to limit and even avoid these constraints.
- Linear codes are very interesting codes in this respect. It is the most used class of error-correcting codes.
- The key concept is that of vector space and generating matrix.

Notation and Formalization

- We consider an alphabet Σ whose size is a prime power q. In fact, Σ is chosen as the set of distinct elements of the field $\mathbb{F}_q$.
- We consider the vector space $V_n(q)$ of dimension n over $\mathbb{F}_q$. Its elements are vector $x = (x_1, x_2, \ldots, x_n) = x_1 x_2 \ldots x_n$ with $x_i \in \mathbb{F}_q$.

Linear Code

A *linear code* $\mathcal{C}$ over Σ is any subspace of $V_n(q)$. If $\mathcal{C}$ is a k-dimensional subspace with minimal distance d, we note it $[n, k, d]$.

- As a subspace of $V_n(q)$, $\mathcal{C}$ must contain the zero codeword $0000\ldots000$.
- Any k-dimensional subspace of $\mathbb{F}_q$ contains q^k elements.

Generator Matrix

Proposition

Let $\mathcal{C}$ be an $[n, k, d]$-code is an (n, q^k, d)-code.

- So we can fully describe $\mathcal{C}$ with only k linearly independent codewords. We save an enormous amount of space. Coding/decoding time is drastically reduced. We then can define the *generator matrix* of a linear code.

Generator matrix of a linear code

An $[n, k, d]$-code is an $[n, k, d]$-code with a basis $\mathcal{B} = \{b_1, b_2, \ldots, b_k\}$. If $b_i = b_{i1}, b_{i2}, \ldots, b_{in}$ then the $k \times n$ matrix

$$\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ b_{k1} & b_{k2} & \cdots & b_{kn} \end{pmatrix}$$

whose rows are the codewords in $\mathcal{B}$ is called the *generator matrix* G for $\mathcal{C}$.

Generator Matrix and Equivalent Codes

- Suppose that we have a generator matrix G of $\mathcal{C}$ and G' is any other matrix obtained from G by any finite sequence of operations of the following types (SOP) :
  - Permuting rows or permuting columns.
  - Multiplying a row or a column by a non-zero scalar.
  - Adding to a row a scalar multiple of another row.
- By basic linear rules (proof left as an exercice), we have

Generator Matrix and Equivalent Codes

G' is the generator matrix of a code $\mathcal{C}'$ which is equivalent to $\mathcal{C}$.

Left Standard Form of Generator Matrix

Generator Matrix Left Standard Form

Let G be any $k \times n$ generator matrix. Then, by applying a SOP to G it is possible to transform G into a matrix of type $[I_k, M_{k,n-k}]$ where I_k is the $k \times k$ identity matrix. This matrix is called the *Left Standard Form* of G.

- The left standard form of G makes coding and decoding processes very simple (see further).
- The code $\mathcal{C}$ with generator matrix G is equivalent to the code $\mathcal{C}'$ with generator matrix $[I_k, M_{k,n-k}]$.
- Coding with the help of the left standard form is called *systematic coding*.
- Exercices.

Minimum Distance of a Linear Code

- For linear codes, the minimum distance is usually found more easily than for general codes (for which we have to explore $\frac{M.(M-1)}{2}$ distances).
- We define the *weight* $w(x)$ of a vector to be $|\{i \in [1, \ldots, n] | x_i \neq 0\}$.

Theorem

The minimum distance of a linear code $\mathcal{C}$ is the minimum weigth of non-zero vector in $\mathcal{C}$.

- Proof left as an exercice.
- We will see later in the coding process by linear codes how d can be computed more easily.

The Dual (code) of a Linear Code

- We are going to consider a new technique to build new codes from old ones which is specific to linear codes.
- We define the *linear product* of x and y from $\mathbb{F}_q^n$ as $x.y = \sum_{i=1}^{n} x_i y_i$ where the sum and product are taken in $\mathbb{F}_q$
- For any $S \subseteq \mathbb{F}_q^n$ we note $S^\perp$ the set of all strings in $\mathbb{F}_q^n$ that are orthogonal to every string in S. Thus

$$S^\perp = \{x \in \mathbb{F}_q^n | s.x = 0, \forall s \in S\}$$

$S^\perp$ is the *orthogonal complement* of S

Lemma

For any subset S in $\mathbb{F}_q^n$, the set $S^\perp$ is a linear code. The orthogonal complement $\mathcal{C}^\perp$ of any code $\mathcal{C}$ is a linear code called the *dual code* of $\mathcal{C}$.

The Dual (code) of a Linear Code (2)

Proposition

Let $\mathcal{C}$ be a linear $[n, k, d]$-code over $\mathbb{F}_q$, with generator matrix G

1. $\mathcal{C}^{\perp} = \{x \in \mathbb{F}_q^n | x.G^t = 0\}$.
2. $\mathcal{C}^{\perp}$ is a linear $[n, n-k]$-code. In other words, $dim(\mathcal{C}^{\perp}) = n - dim(\mathcal{C})$.
3. We have $(\mathcal{C}^{\perp})^{\perp} = \mathcal{C}$.

- Important : mathematical properties in finite fields can be different from those in $\mathbb{R}$ (for example in $\mathbb{R}$ no vector is orthogonal to itself and hence for any subspace W, $W^{\perp} \cap W = \{0\}$ which is not the case in finite fields).

Definition

A linear code $\mathcal{C}$ is said to be *self-orthogonal* if $\mathcal{C} \subseteq \mathcal{C}^{\perp}$. A linear code $\mathcal{C}$ fr which $\mathcal{C} = \mathcal{C}^{\perp}$ is said to be *self-dual*.

Parity-check Matrix

- By the previous Proposition, we can describe the dual code as the solutions to certain equations. The system $x.G^t$ is called the *parity-check equations* for the code $\mathcal{C}^\perp$.
- A string $x = x_1x_2 \dots x_n \in \mathbb{F}_q^n$ is in the dual code $\mathcal{C}^\perp$ if and only if its components $x_1, x_2, \dots, x_n$ satisfy the parity-check equations for $\mathcal{C}^\perp$.

Definition

A *parity-check matrix* for linear q-ary $[n, k]$-code $\mathcal{C}$ is a matrix P with the property that

$$C = \{x \in \mathbb{F}_q^n | x.P^t = 0\}$$

- *Remark* : no requirement is made about the independence of the rows of P. When independent, P is much smaller.

Parity-check Matrix (2)

- It is easy to construct a parity-check matrix from a generator matrix that is under the left-standard form.

Proposition

The matrix $G = (I_k|B)$ is a generator matrix for an $[n, k]$-code $\mathcal{C}$ if and only if the matrix $P = (-B^t|I_{n-k})$ is a parity-check matrix for $\mathcal{C}$. A vector $x \in \mathbb{F}_q^n$ is a codeword of $\mathcal{C}$ if and only if $P.x^t = 0$.

- Proof left as an exercice.
- The matrix $= (-B^t|I_{n-k})$ is called the *right standard form*.
- Exercices.

Agenda

Encoding Process

- We consider a $[n, k]$-code $\mathcal{C}$ over Σ_q with generator matrix $G = [I_k, A]$.
- The codewords of $\mathcal{C}$ are all the q^k vectors of length n of the form

$$\sum_{k=1}^{k} a_i r_i \qquad a_i \in \mathbb{F}_q$$

  where r_i is the i-th row of G.
- Encoding is then very simple. If the message sequence is $s = (s_1, s_2, \ldots, s_k)$, we encode s by the codeword $c = (c_1, c_2, \ldots, c_n)$ in this way

$$c = s.G$$

Encoding Process (2)

- When G is in the left-standard form, then $c_i = s_i$ for $i \in [1, k]$ (message digits) and bits c_j for $j > k$ are called *parity-check digits*. In this case, this encoding is called *systematic encoding*.
- So encoding can be performed by simple vector-matrix products.
- For a long message $m = m_1 m_2 \dots m_N$, we split it into vectors of size k and we encode each vectors separately.
- Upon reception of codeword c, error-detection is performed by using parity-check bits. They must verify the parity-check equations. It is then possible to determine which bit(s) has been modified during the transmission.
- Exercices.

Agenda

Introduction

- Determining the minimum distance for linear codes is much simpler than for general codes.
- Let us recall that $d(x, y) = w(x - y)$ for all strings x and y in $\mathbb{F}_q^n$. For a linear code, if $c, d \in \mathcal{C}$ then $x - y \in \mathcal{C}$.

Proposition

If $\mathcal{C}$ is a linear code, then $d(\mathcal{C}) = w(\mathcal{C})]$

- *Remark* : this proposition holds only for codes which are additive subgroups of $\mathbb{F}_q^n$.
- To compute d we can
  - Compute all codewords of $\mathcal{C}$ from G (time consuming).
  - Use the parity-check matrix P to determine whether a given string x is in $\mathcal{C}$ or not.

Gilbert-Varshamov Bound for Linear Codes

Proposition

Let P be a parity-check matrix for a linear code $\mathcal{C}$. Then the minimum distance of $\mathcal{C}$ is the smallest integer r for which there are r linearly dependent columns in P.

- The sphere-covering lower bound on $A_q(n,d)$ is defined by $\frac{q^n}{V_q^n(d-1)} \leq A_q(n,d)$. We can improve this bound in the case of linear codes and using the previous proposition.

Gilbert-Varshamov Bound for Linear Codes

There exists a q-ary linear $[n,k,d]$-code if $q^k < \frac{q^n}{V_q^{n-1}(d-2)}$. Thus if q^k is the largest power of q satisfying this inequality we have $A_q(n,d) \geq q^k$.

- Proof can be omitted.

Agenda

Introduction : Cosets

- Wlog we focus on binray codes ($q = 2$). The extension to other alphabet of prime-power size is straightforward.
- We consider a $[n, k]$-code $\mathcal{C}$. Since $\mathcal{C}$ is a subspace of $V(n)$, it is also a subgroup of the additive groupe $(V(n), +)$.

Definition (Coset)

Let x be an arbitrary vector of $V(n)$ and $\mathcal{G}$ any subgroup of the additive groupe $(V(n), +)$. x determines a unique set $x + \mathcal{G}$, called *coset*, as follows :

$$x + \mathcal{G} = \{y \in V(n) | y = x + z \text{ for some z} \in \mathcal{G}\}$$

- Thus $\mathcal{C}$ determines a collection of cosets of $V(n)$.
- Now suppose that $\hat{x}$ is a received vector when some codeword x is transmitted through a noisy channel. We say that e is a *possible error vector* of $\hat{x}$ if there is some codeword $x \in \mathcal{C}$ such that $\hat{x} - x = e$.

Cosets and Minimum-distance Decoding

- The interpretation is pretty simple : a vector e is an error vector associated with a received vector if it could represent a possible pattern of errors in transmission (with respect to the error probability of the channel)

Lemma

If $\hat{x}$ is the received vector, then the set of possible error vectors is the coset of $\mathcal{C}$ that contains the vector $\hat{x}$.

- Proof left as an exercice.
- The minimum-distance decoding consists then in finding an error vector of minimum weight. If we know for each coset, a member of the coset of minimum weight (*coset leader*), we have a basis for the minimum-distance decoding rule.
- The coset leader may be not unique !

Minimum-distance Decoding : Algorithm I

Input: A received vector y.
Input: $\mathcal{C} = \{c_1, c_2, \ldots, c_{2^k}\}$
Step 1 : On receiving y, find a coset leader z_0 of the coset determined by y.
Step 2 : Decode y as the codeword $y - z_0$.

- Step 1 can be very time-consuming.

Proposition

Let P be the parity-check matrix of a code $\mathcal{C}$. Two vectors x and y are in the same coset with respect to $\mathcal{C}$ if and only if

$$Px^t = Py^t$$

- Proof left as an exercice.

Syndrome Decoding : Algorithm II

Syndrome

The *syndrome* of the coset $x + \mathcal{C}$ is the vector $S(x) = Hx^t$.

- Thus for each coset, we have a syndrome and a coset leader. So we can build a *look-up table* which gives the corresponding coset leader for each syndrome.
- We then can speed up the previous algorithm.

Input: A received vector y.
Input: $\mathcal{C}$ with parity-check matrix P.
Step 1(a) : On receiving y, compute its syndrome $P.y^t$.
Step 1(b) : From the above look-up table, read off the corresponding coset leader z_0.
Step 2 : Decode y as the codeword $y - z_0$.

Syndrome Decoding (2)

Theorem

Algorithm II is a minimum-distance decoding scheme for the linear code $\mathcal{C}$.

- Proof left as an exercice.
- The look-up table may be not unique (see exercices).
- Building the look-up table is not straightforward (how to build the different cosets is not obvious from the definition).
- We are going to see how to build it effectively by considering the *standard array* structure for linear codes.
- Let $\mathcal{C}$ be a q-ary $[n, k, d]$-code whose codewords are $c_1, c_2, \ldots c_{q^k}$.

Linear Code Standard Array

- The standard array is the following structure

$$\begin{array}{ccccc} 0 & c_1 & c_2 & \cdots & c_{q^k} \\ f_2 & f_2 + c_1 & f_2 + c_2 & \cdots & f_2 + c_{q^k} \\ f_3 & f_3 + c_1 & f_3 + c_2 & \cdots & f_3 + c_{q^k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ f_{q^{n-k}} & f_{q^{n-k}} + c_1 & f_{q^{n-k}} + c_2 & \cdots & f_{q^{n-k}} + c_{q^k} \end{array}$$

- The first row consists of the codewords in $\mathcal{C}$.
- To form the second row as the coset $f_2 + \mathcal{C}$, we choose f_2 as a string of smallest weight that is not in the first row and build the coset.
- In the same way for the coset $f_i + \mathcal{C}$ we choose f_i as a string of smallest weight that is not in the array yet.
- The elements f_i are by construction the coset leaders

Linear Code Standard Array (2)

Lemma : Standard Array Properties

Let $\mathcal{C}$ be a q-ary $[n, k, d]$-code with standard array A.

1. Every string in $\mathbb{F}_q^n$ appears exactly once in A.
2. The number of rows of A is q^{n-k}.
3. Two strings x and y in $\mathbb{F}_q^n$ lie the same coset (row) of A is and only if their difference $x - y$ is in $\mathcal{C}$.
4. The coset leader has the minimum weight among all strings in its coset.

Proposition

Let $\mathcal{C}$ be a q-ary $[n, k, d]$-code with standard array A. For any string $x \in \mathbb{F}_q^n$, the codeword c that lies on the top of the column containing c is the nearest neighbour to x.

- Proofs left as exercices.

Decoding with Standard Array

- The standard array may be not unique !
- Since any linear $[n, k, d]$-code corrects up to $\frac{d-1}{2}$ errors, it follows that the coset leaders of any standard array for $\mathcal{C}$ must include all strings of weight $\frac{d-1}{2}$ or less.
- The difference $x - c$ between the received string x and codeword c seen as the nearest neighbour is the coset leader for the coset containing x. This coset leader is hence called the *error string*.
- Algorithm II can then be implemented simply by building the standard array to determine the coset leaders f_i and then compute their respective syndrome $S(f_i)$.
- The table whose rows consist in the pair $f_i, S(f_i)$ is called the *syndrome table* for $\mathcal{C}$.

Linear Decoding Performances

- Let $\mathcal{C}$ be a q-ary $[n, k, d]$-code. Let us analyze the performance of the syndrome decoding. We assume that the channel error probability is p.
- If we let w_i denote the number of coset leaders of weight i, then the probability of correct decoding is given by

$$P[\text{correct decoding}] = \sum_{i=1}^{n} w_i p^i (1-p)^{n-i}$$

- In general determining the numbers w_i is difficult. For perfect linear $[n, k, d]$-codes, we have $w_i = \binom{n}{i}$ for $0 \leq i \leq \frac{d-1}{2}$ and $w_i = 0$ for $i > \frac{d-1}{2}$.
- An error during the transmission of a codeword c remain undetected if and only if the error string is a non-zero codeword.
- If A_i denotes the number of codewords of weight i, the probability or an undetected error is given by

$$P[\text{undetected error}] = \sum_{i=1}^{n} A_i p^i (1-p)^{n-i}$$

Agenda

Burst-errors

- We have considered channels assuming that symbol errors were independent of time and each other. This is not fully realistic. To be closer to reality we have to consider burst errors.

Burst-errors

A *burst* in $\mathbb{F}_q^n$ of length b is a string in $\mathbb{F}_q^n$ whose non-zero coordinates confined to b consecutive positions, the first and the last of which must be non-zero.

- The string 0001100100 in $\mathbb{F}_2^{10}$ is a burst of length 5.

Lemma

Let $\mathcal{C}$ be a linear $[n, k, d]$-code over $\mathbb{F}_q$. If $\mathcal{C}$ contains no burst of length b or less, then $k \leq n - b$.

- This comes from the fact that if $\mathcal{C}$ is able to correct any burst of length b or less, then no such burst can be a codeword.

Detecting and Correcting Burst-error

Proposition

If a linear $[n, k, d]$-code $\mathcal{C}$ can detect all burst errors of length b or less, then $k \leq n - b$. Furthermore, there is a linear $[n, n - b]$-code that will detect all burst errors of length b or less.

Proposition

If a linear $[n, k, d]$-code $\mathcal{C}$ can correct all burst errors of length b or less, using nearest neighbour decoding, then $k \leq n - 2b$.

Proposition

If a linear $[n, k, d]$-code $\mathcal{C}$ over $\mathbb{F}_q$ can correct all burst errors of length b or less, using nearest neighbour decoding, then $k \leq n - b + 1 - \log_2[(n - b + 1)(q - 1) + 1]$.

- Proofs can be omitted.

Majority Logic Decoding

- This decoding procedure provides a simple method for decoding linear codes.

Definition

A system of parity-check equations for a linear code is said to be *orthogonal* with respect to the variable x_i provided that x_i appears in every equation of the system but all other variables appear in exactly one equation.

- In this case, suppose that a single error occurs in transmission. If the error is at the i-th position, then x_i is incorrect but all other x_j are not. Hence each equation will be unsatisfied.
- On the other hand if error occurs at position $j \neq i$ then only one equation will be unsatisfied.

Majority Logic Decoding (2)

- More generally suppose that we have r parity-check equations which is orthogonal with respect to x_i. Suppose further that $t \leq \frac{r}{2}$ errors have occured during the transmission.
- If one of the errors occured in position i, then at most $t-1$ equations can be corrected by the remaining errors and so at least $r-(t-1) \leq \frac{r}{2}+1$ equations will be unsatisfied.
- On the other hand, if error occurs at position $j \neq i$, then at most $t \leq \frac{r}{2}$ equations will be unsatisfied.
- Therefore, the i-th position in the received string is in error if and only if the majority of equations is unsatisfied. This is the core principle of *majority-logic decoding*.

Agenda

Conclusion

- Linear codes are very powerful codes.
- The algebraic structure enables simple ncoding and decoding procedures.
- This class of codes is widely used in many application.
- Go now to the computer room to practice with exercices.

Agenda

Essential Bibliography

A few papers are available on the Moodle repository for this lecture.

- Lidl, R. & Niederreiter, H. (1986). *Introduction to Finite Fields and their Applications*. Cambridge University Press.
- MacWilliams, F. J. & Sloane, N. J. A. (1977). *The Theory of Error-Correcting Codes*. North-Holland.
- Pless, V. S. & Huffman, W. C. eds (1998). Handbook of Coding Theory. North-Holland.
- Shannon, C.E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, Vol. 27, pp. 379–423, 623–656.