ENEE626, CMSC858B, AMSC698B

Error Correcting Codes

Instructor: Alexander Barg (abarg@umd.edu) Office: AVW2361

Course goals: To introduce the main concepts of coding theory and the body of its central results.

Prerequisites for the course

The main prerequisite is mathematical maturity, in particular, interest in learning new mathematical concepts. No familiarity with information theory and communications-related courses will be assumed. On the other hand, the students are expected to be comfortable with linear spaces, elementary probability and calculus, and elementary concepts in discrete mathematics such as binomial coefficients and an assortment of related facts. There is no required textbook.

The **web site** http://www.ece.umd.edu/~abarg/626 contains a detailed list of topics, problems, schedule of exams, grading policy, reference books.

Part I. Introduction to coding theory

Plan for today:

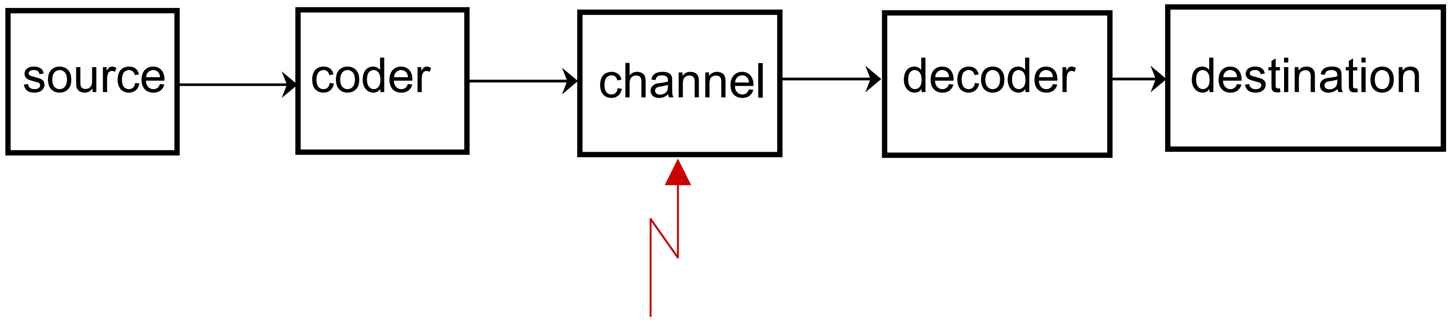
1. Syllabus, logistics
2. Model of a communication system
3. Binary Symmetric Channel
4. Coding for error correction
5. Notation and language

Digital communication: Computer networks, wireless telephony, data and media storage, RF communication (terrestrial, space)

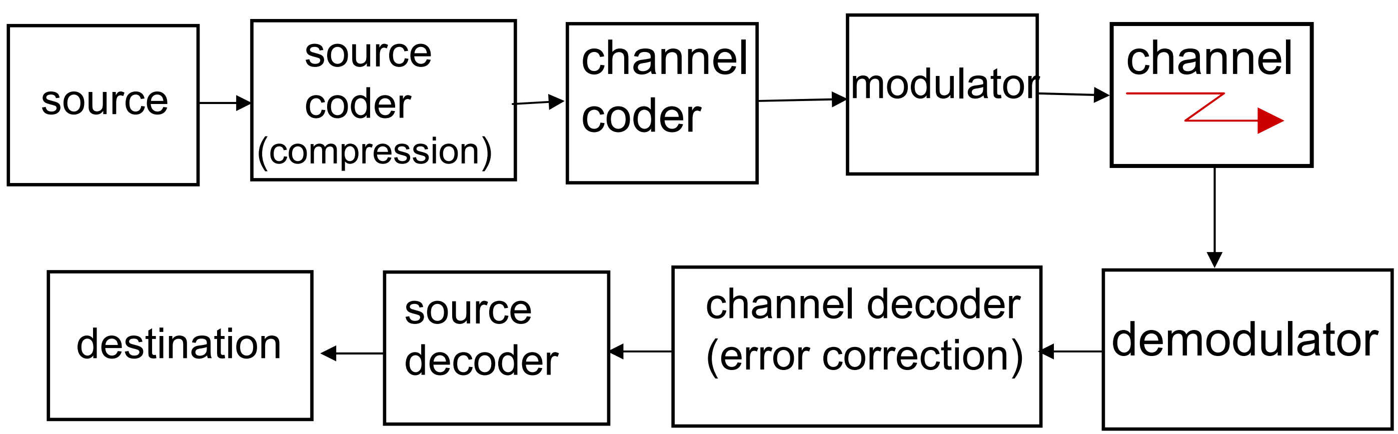
Transmission over communication channels is prone to errors.

background noise, mutual interference between users, attenuation in channels, mechanical damage, multipath propagation, …

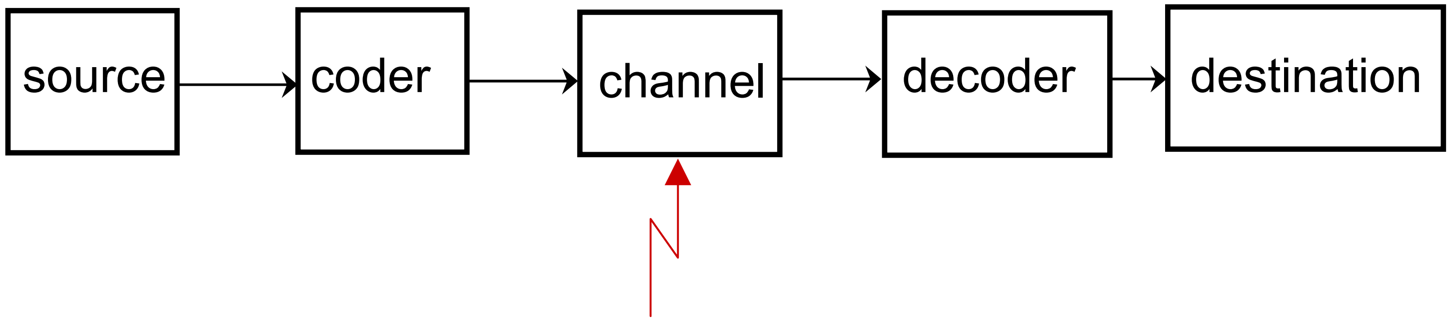
Model of a communication system



more detailed:

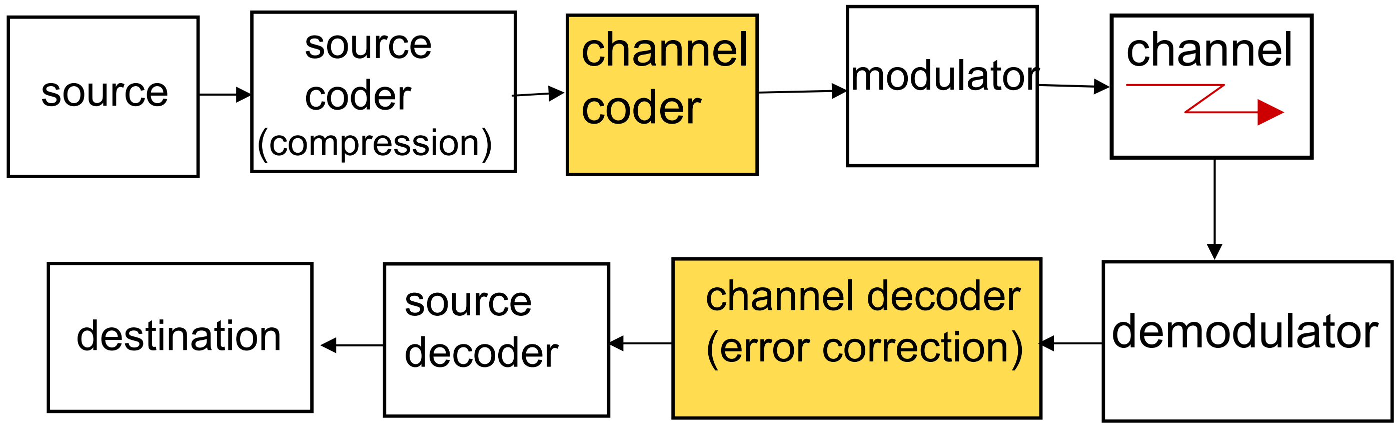


Model of a communication system

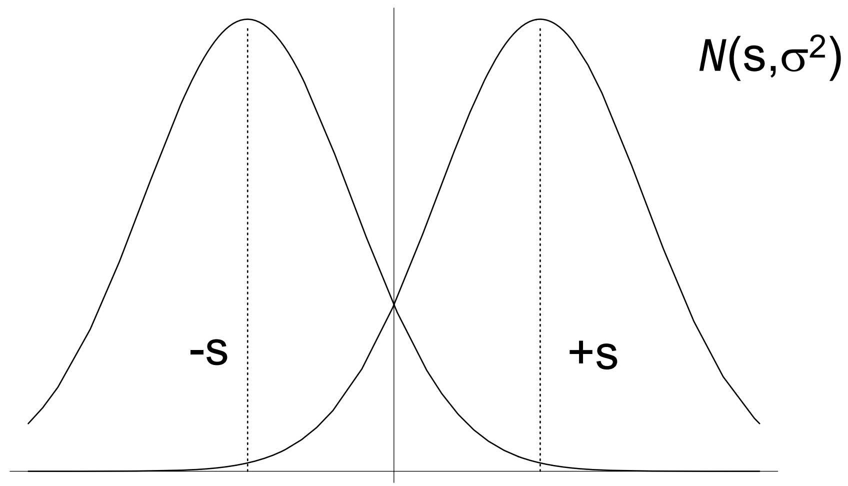


more detailed:

we are interested in



Assume transmission with binary antipodal signals over a Gaussian channel

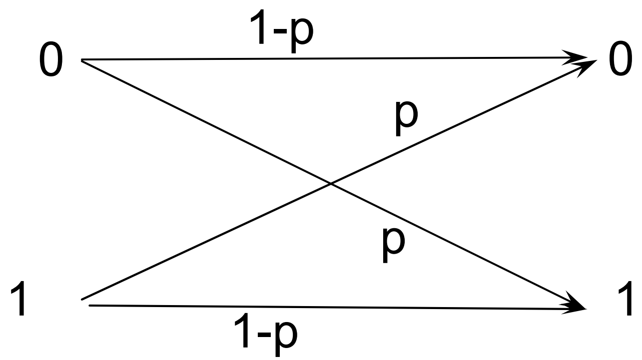


Suppose that the received signal y is decoded as
x=sgn(y) s

The probability of error is computed as

$$p = P(y < 0|s) = P(y > 0|-s) = \Phi\left(\frac{-s}{\sigma}\right) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{-s} e^{-x^2/2\sigma^2} dx$$

Binary Symmetric Channel (BSC)

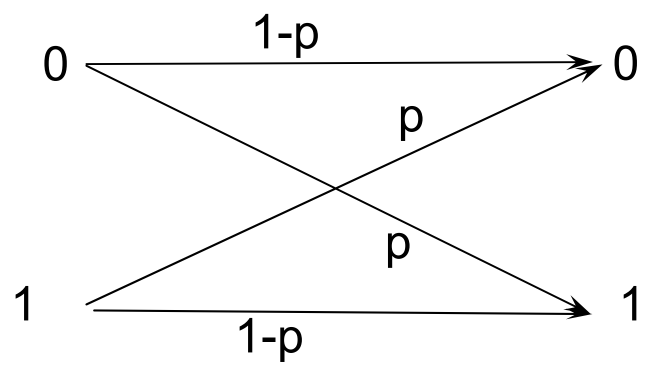


transmissions are independent

p is called the transition (cross-over) probability

Much of coding theory deals with error correction for transmission over the BSC. This will also be our main underlying model.

Binary Symmetric Channel (BSC)

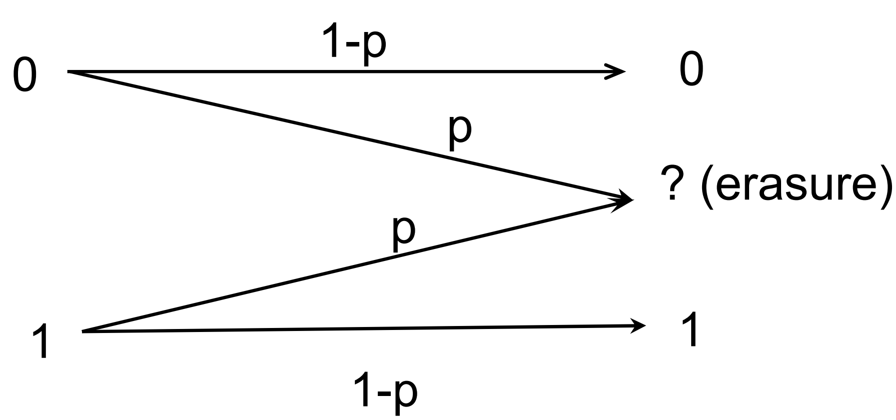


transmissions are independent

p is called the transition (cross-over) probability

Much of coding theory deals with error correction for transmission over the BSC. This will also be our main underlying model.

The erasure channel



Main example:
internet traffic

Messages = binary strings   Ex.: 101

k bits $(m_1, m_2, \ldots, m_k)$ word, vector $m_i \in \{0,1\}$

encoding: message $\longrightarrow$ codeword. purpose: error correction

Example: 2 messages 0,1.

no coding: 0 $\rightarrow$ channel $\rightarrow$ 1 (message lost)

encode 0 $\rightarrow$ 000   C={000,111} – a code

1 $\rightarrow$ 111

000 $\rightarrow$ channel $\rightarrow$ 010

$$\Pr[0|010]=p(1-p)^2\Pr[0]/\Pr[010];\ \Pr[1|010]=p^2(1-p)\Pr[1]/\Pr[010]$$

$$\frac{\Pr[0|010]}{\Pr[1|010]} = (1-p)/p > 1 \text{ if } p<1/2.$$

Thus, if $p<1/2$, $\Pr[0|010]>\Pr[1|010]$.

Conclude: decoding by maximum a posteriori probability (MAP) will recover the message correctly

Definition 1.2: Hamming distance between two vectors $\boldsymbol{x},\boldsymbol{y}$

$$d((x_1,x_2,\ldots,x_n),(y_1,y_2,\ldots,y_n))=|\{i:x_i \neq y_i\}|$$

Transmit $M=2^k$ messages with a code $C=\{\mathbf{x}_1,\mathbf{x}_2,\ldots\mathbf{x}_M\}$

y received from the channel. Decode to

$$\mathbf{x} = \underset{\mathbf{x}_i \in C}{\operatorname{argmin}}\, d(\mathbf{x}_i,\mathbf{y}) \qquad \text{(minimum distance decoding)}$$

(if there are several such **x**, declare an error)

Observation: on the BSC(p), p<½, the probability Pr[**e**] of error $\mathbf{e}=(e_1,e_2,e_3)$ decreases as the # of 1's among e_1,e_2,e_3 increases. Hence, decoding by minimum distance is *equivalent* to MAP decoding

Conclude: it's a good idea in many cases to have codewords far apart

Bits of notation

Finite sets A,B,C,F, ...
The number of elements in A is called the size of A, denoted |A| or #(A).

$\mathbb{F}_2$={0,1} the binary field; F=$\mathbb{F}_2^n$ – n-dim linear space over $\mathbb{F}_2$
x,**y**,... – vectors (often in F) (row vectors); $\mathbf{x}^T$ transpose (column vect.)
0=0^n the all-zero vector; likewise, (0^i1^j...) is a generic shorthand for a vector
$(\mathbf{x},\mathbf{y})=\sum_{i=0}^n x_i\, y_i$ dot product
$d(\mathbf{x},\mathbf{y}) = |\{i: x_i \neq y_i\}|$ Hamming distance
w(**x**) (sometimes wt(**x**)) the weight of **x**, i.e., d(**x**,0)

G,H,A,... matrices

d(C) = the distance of the code C
C[n,k,d] a linear code of length n, dimension k, distance d
C(n,M,d) a code, not necessarily linear, of length n, size M, distance d

Mathematical concepts used in coding theory

The primary language is that of linear algebra.
Linear algebra deals with geometry of linear spaces and their transformations

A linear space L is the most familiar concept, such as $\mathbb{R}^2$, $\mathbb{R}^3$ and the likes
It is formed of a field of constants (e.g., $\mathbb{R}$) and vectors over it
Vectors obey the natural rules:
they can be added to form another vector; they can be stretched by multiplying them by a constant.

To describe L it is convenient to choose a basis (a frame). The number of vectors in the basis is called the dimension of L.
The space does not depend on the choice of the basis although the coordinates of the vectors generally change if one passes to another basis

A subspace M of L can be described by any of its bases or as
a set of solutions of a system of equations (kernel of a linear operator)

The quotient space L/M consists of M and its shifts by vectors from L\M
Linear spaces of coding theory live over finite fields (such as $\mathbb{F}_2$={0,1}).

Reminder (cont'd): binomial coefficients

(a) Permutations: (abc, acb, bac, bca, cba, cab)

n(n-1)(n-2)…2 .1=n! (n factorial)

(b) The number of ways to choose an ordered k-tuple out of an n-set

n(n-1)(n-2)…(n-k+2)(n-k+1)=$(n)_k$

(c) The number of unordered k-tuples out of an n-set.

notation: $$\binom{n}{k} = \frac{(n)_k}{k!}$$

|{**x** ∈ F: wt(**x**)=k}|= $\binom{n}{k}$

1	0
1 1	1
1 2 1	2
1 3 3 1	3
1 4 6 4 1	4
1 5 10 10 5 1	5
1 6 15 20 15 6 1	6
1 7 21 35 35 21 7 1	7

Extend the definition:

$$\binom{x}{k} = \begin{cases} \frac{x(x-1)\ldots(x-k+1)}{k!} & \text{if } k \geq 1 \text{ integer} \\ 1 & \text{if } k = 0 \\ 0 & \text{all other cases} \end{cases} \qquad x \in \mathbb{R}$$

See probl. 12, h/work 1

Operating with binary data

XOR

+	0	1
0	0	1
1	1	0

AND

·	0	1
0	0	0
1	0	1

Notation: $\mathbb{F}_2=\{0,1\}$; $F=(\mathbb{F}_2)^n$

$\mathbf{x}_1=(01101)$, $\mathbf{x}_2=(10101)$
$\mathbf{x}_1 + \mathbf{x}_2=(11000)$
$(\mathbf{x}_1,\mathbf{x}_2)=\sum_{i=1}^n x_{1,i}x_{2,i}$ (dot product)

$(\mathbf{x}_1,\mathbf{x}_2)=0$ or 1 according as #i such that $x_{1,i}=x_{2,i}=1$ is even or odd

Examples of codes:

m_1 000 $\mapsto$ 000000
m_2 001 $\mapsto$ 001111
m_3 010 $\mapsto$ 010110
m_4 011 $\mapsto$ 011001
m_5 100 $\mapsto$ 100101
m_6 101 $\mapsto$ 101010
m_7 110 $\mapsto$ 110011
m_8 111 $\mapsto$ 111100

code $\mathscr{C}$ can correct one error, can be used to transmit $8=2^3$ messages (3 bits of information)

Repetition code {000…00,111…11} k=1

Single parity-check code $\{x_1,x_2,\ldots,x_M\}$ formed of *all* codewords of length n with an even number of ones. $M=2^{n-1}$

n=3: {000,011,101,110}

Goal: construct codes of arbitrary length that correct a given number of errors, equipped with a simple decoding procedure

ENEE626 Lecture 2: Linear codes

1. Linear codes: examples, definition
2. Generator and parity-check matrices
3. Hamming weight
4. Algorithmic complexity

Linear codes

Code $\mathscr{C}$

$\mathbf{m}_1$ 000 $\mapsto$ 000000
$\mathbf{m}_2$ 001 $\mapsto$ 001111
$\mathbf{m}_3$ 010 $\mapsto$ 010110
$\mathbf{m}_4$ 011 $\mapsto$ 011001
$\mathbf{m}_5$ 100 $\mapsto$ 100101
$\mathbf{m}_6$ 101 $\mapsto$ 101010
$\mathbf{m}_7$ 110 $\mapsto$ 110011
$\mathbf{m}_8$ 111 $\mapsto$ 111100

Verify that all the codewords of $\mathscr{C}$ can be computed by multiplying

$\mathbf{x}_i = \mathbf{m}_i\, G$, where

$$G = \begin{pmatrix} 100101 \\ 010110 \\ 001111 \end{pmatrix}$$

$\mathbf{m}_6 G = (101)G = 101010 = \mathbf{x}_6$

Therefore, $\mathscr{C}$ is closed under addition:

$$\mathbf{x}_i + \mathbf{x}_j = (\mathbf{m}_i + \mathbf{m}_j)\, G = \mathbf{m}_k\, G = \mathbf{x}_k \in \mathscr{C}$$

$\mathscr{C}$ is a linear code (a linear subspace of $(\mathbb{F}_2)^n$)

$F=(\mathbb{F}_2)^n$ is a linear space:

- F is an abelian group under addition
- Its unit is the all-zero vector **0**=(00...000)
- Multiplication by scalars is distributive

  $c(\mathbf{x}+\mathbf{y})=c\mathbf{x}+c\mathbf{y}$

  $(a+b)\mathbf{x}=a\mathbf{x}+b\mathbf{x}$

- Multiplication is associative:

  $(ab)\mathbf{x} = a(b\mathbf{x})$

Definition 2.1: A linear subspace of F is called a binary linear code

For instance, the code $\mathscr{C}$ above is linear

Let A be a linear code, k=dim A. A matrix whose rows are the basis vectors of A is called a generator matrix of the code.

G $(k \times n)$-matrix

Example: let n=4, consider 4-dim space F

0000
0001
0010
0011
0100
0101
0110
0111
1000
1001
1010
1011
1100
1101
1110
1111

2-dim subspace $\langle \mathbf{x}_1, \mathbf{x}_2 \rangle = \langle 0001, 0010 \rangle$ ($\langle\ ,\ \rangle$ means linear hull)

$C = \{ \lambda_1 \mathbf{x}_1 + \lambda_2 \mathbf{x}_2,\ \lambda_1, \lambda_2 \in \{0,1\} \}$

Explicitly, C={0000,0001,0010,0011}

$G = \begin{matrix} 0001 \\ 0010 \end{matrix}$

Generally, $|C|=2^k$, where k is the dimension of the code n is called the length of the code.

Consider the code A={00000,11111} of length 5, dimension 1

G=[11111]

(the repetition code).

Single parity-check code B, n=5

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

Definition 2.2: The Hamming weight of a vector $\mathbf{x}=(x_1,...,x_n)$ is defined as

$$w(\mathbf{x})=|\{i : x_i=1\}|$$

Exercise: The sum of two even-weight vectors has even weight.

Thus, the code B is formed of $2^4=16$ vectors of even weight (satisfies an overall parity check)

The parity-check matrix of a code

Consider a code of length 6: $\mathbf{x}=(x_1,x_2,x_3,x_4,x_5,x_6)$
Suppose that

$$\begin{cases} x_1 + x_2 + x_3 + x_4 = 0 \\ x_2 + x_3 + x_5 = 0 \\ x_1 + x_3 + x_6 = 0 \end{cases}$$

Assign any values to x_1,x_2,x_3, solve for x_4,x_5,x_6

Parity-check equations

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

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

Definition 2.3: H is called a parity-check matrix of the code

Another definition of a linear code: $C=\{\mathbf{x} \in F: H\,\mathbf{x}^T=0\}$

Notation: C[n,k] denotes a linear code of length n and dimension k ($0 \leq k \leq n$)

Let C[n,k] be a code. The encoding mapping can be written as

$$\mathbf{x}=(m_1,...,m_k)\,G$$

code vector — message vector — generator matrix k x n

rank (G)=k $\Rightarrow$ there exist k linearly independent columns
Suppose w.l.o.g. that they are columns 1,2,...,k:

$$G=[I_k \mid A]$$, where A is some k x (n-k) matrix

then the code vector that corresponds to $(m_1,...,m_k)$ has the form

$$\mathbf{x}=(m_1,m_2,...m_k,x_{k+1},...,x_n)$$

the message bits show directly in the code vector
In such a situation we say that the code is defined in a systematic form

Proposition 2.1: Any [n,k] linear code can be written in a systematic form

Indeed, take the k columns of G that have rank k; by elementary operations diagonalize this submatrix

Example: The matrix

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

defines the single-parity-check code of length 5 in a systematic form: the last 4 coordinates carry the message, the first coordinate corresponds to the parity check. For instance, the message (1101) is encoded as (11101)

Lemma 2.2: Let $G=[I_k|A]$ be a k x n generator matrix of a code C. Then $H=[A^T|I_{n-k}]$ is a parity-check matrix of C.

Proof: $HG^T=[A^T|I_{n-k}][I_k|A]^T=A^TI_k+I_{n-k}A^T=0$

Note that we can have message symbols in any 4 of the 5 coordinates:

for instance, the matrix $\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$ defines the *same code* as in Example 2.2,

which has been written in a systematic form to show the message bits in coordinates 1,3,4,5.

Encoding in a systematic form

$G=[I_k \mid A]$, A a k x (n-k) matrix with rows $\mathbf{a}_1,\ldots\mathbf{a}_k$

$\mathbf{m}G=(m_1,\ldots,m_k, \mathbf{a})$, where $\mathbf{a}=\sum_i m_i\, a_i$

Let $H=[A^T|I_{n-k}]$ be the p.-c. matrix. The parity check symbols are computed from the equations $H\mathbf{x}^T=0$, where $\mathbf{x}=(m_1,\ldots,m_k,x_1,x_2,\ldots,x_{n-k})$. Thus,

$$
\begin{aligned}
&m_1\, a_{1,1} + m_2\, a_{2,1} + \ldots + m_k\, a_{k,1} + x_1 = 0\\
&m_1\, a_{1,2} + m_2\, a_{2,2} + \ldots + m_k\, a_{k,2} + x_2 = 0\\
&\ldots.\\
&m_1\, a_{1,n-k} + m_2\, a_{2,n-k} + \ldots + m_k\, a_{k,n-k} + x_{n-k} = 0
\end{aligned}
$$

Encoding in a systematic form is easier than in a general form

Definition 2.4: Let $\mathbf{x}_1,\mathbf{x}_2 \in F$. The Hamming distance

$$d(\mathbf{x}_1,\mathbf{x}_2) = \#\{i: x_{1,i} \neq x_{2,i}\}$$

Exercises: 1. Prove that $d(\cdot,\cdot)$ is a metric on F.

2. Prove that d is translation invariant, i.e.,

$$d(\mathbf{x}_1,\mathbf{x}_2)=d(\mathbf{x}_1+\mathbf{y},\mathbf{x}_2+\mathbf{y})$$

where $\mathbf{y} \in F$ is an arbitrary vector.

Take $\mathbf{y}=\mathbf{x}_2$, then $d(\mathbf{x}_1,\mathbf{x}_2)=d(\mathbf{x}_1+\mathbf{x}_2,\mathbf{0})$

Call $d(\mathbf{x},\mathbf{0})$ the weight of $\mathbf{x}$, denoted $wt(\mathbf{x})$

$wt(\mathbf{x})=\#\{i: x_i \neq \mathbf{0}\}$

Definition 2.5: Let C be a linear code. The distance of C is defined as

$$d(C)=\min_{\mathbf{x}_1,\mathbf{x}_2 \in C,\ \mathbf{x}_1 \neq \mathbf{x}_2} d(\mathbf{x}_1,\mathbf{x}_2)$$

Exercise: $d(C)=\min_{\mathbf{x} \in C \setminus \mathbf{0}} wt(\mathbf{x})$

Example: Consider again the code C={0000,0001,0010,0011}

d(C)=1

Notation: We write C[n,k,d] to denote a linear code of length n, dimension k and distance d.

Linear codes are the main subject of coding theory. We can think of a linear code as of a mapping C: $\{0,1\}^k \to \{0,1\}^n$.

Remark: Unrestricted codes. A code is an arbitrary subset $C \subset F$.
The minimum distance of the code is defined as

$$d(C)=\min_{\mathbf{x} \neq \mathbf{y};\, \mathbf{x},\mathbf{y} \in C} d(\mathbf{x},\mathbf{y})$$

We write C(n,M,d) to denote a code of length n, size M and distance d. Unrestricted codes are described by listing all the codewords or describing a way to generate the codewords. There are many interesting theoretical problems related to nonlinear codes. In practical applications, codes are almost always linear because of complexity constraints.

Many ways to describe a linear code

1. A code $\mathscr{C}$ is a row space of its generator matrix G

$$G = \begin{pmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \vdots \\ \mathbf{g}_k \end{pmatrix} \qquad \mathscr{C} = (\textstyle\sum_{i=1}^{k} \lambda_i \mathbf{g}_i)$$

2. A code $\mathcal{C}$ is a null space of its parity-check matrix H.

$$\mathcal{C}=\{ \mathbf{x} \in F: H\, \mathbf{x}^T=0\}$$

A code can have many different generator matrices, many different p.-c. matrices

3. Given a code $\mathcal{C}$ with a parity-check matrix H, consider a bipartite graph $G=(V_1 \cup V_2, E)$, where V_1 are the columns of H, V_2 the rows of H, and $(v_1,v_2) \in E$ iff $H_{v_1,v_2}=1$. This graph is called a *Tanner graph* of the code $\mathcal{C}$.

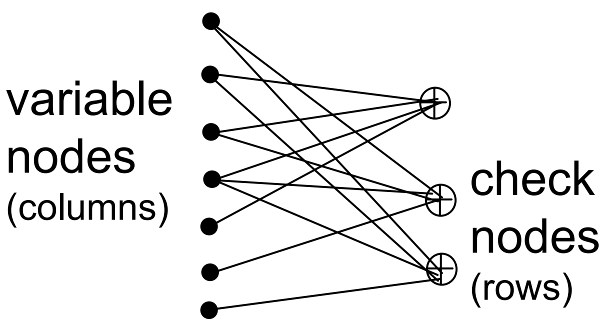
Example: Consider a [7,4,3] code

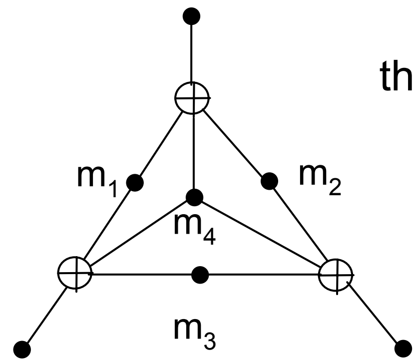
$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad H = \begin{bmatrix} m_1 & m_2 & m_3 & m_4 & & & \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Another p.-c. matrix of $\mathcal{H}_3$:

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

Tanner graph representation





the same graph

An assignment of values to the variable nodes forms a valid codeword if the sum at every check node=0

Complexity of algorithms

An important objective of coding theory is simple processing of data

We shall assume a naive model under which one operation with two binary digits involves a unit cost.

For instance, computing **z=x+y**, where **x,y,z** $\in (\mathbb{F}_2)^n$ has complexity n.
Likewise, computing (**x,y**) takes complexity n+(n-1)
(n multiplications, n-1 additions).

Computing the Hamming distance d(**x,y**) takes n operations.

Suppose we are given a code C(n,M) and a vector **y**$\in (\mathbb{F}_2)^n$, want to find **x**=arg $\min_{\mathbf{z}\in C}$ d(**y,z**). In principle, this can take nM operations. With n growing this becomes prohibitively complex.

We will assume that an algorithm of complexity p(n), where p is some polynomial, is acceptable, an algorithm of exponential complexity is “too difficult” (comparable to exhaustive search).

Notation: Let n$\rightarrow \infty$

f(n)=O(g(n)) $\Leftrightarrow \exists$ const such that f(n)$\leq$ (const)g(n)   Big-O

Examples: Let C be a code of size |C|=M.

1. The complexity of encoding for a linear code.
   Let G be a k x n matrix over $\mathbb{F}_2$, let **m** be a k-vector. The complexity of computing **x**=**m** G is O(k n)=O($\log^2$ M)

2. The complexity of ML decoding is O(nM), No shortcuts are known in general for linear codes.

Coding theory studies families of codes as much as (or more than) individual codes. The primary reason is Shannon's theorem which says that reliable transmission can be achieved at the expense of a growing code length n. Exact formulation and proof given later.

ENEE626 Lecture 3: Linear codes and their decoding

Plan

1. Linear codes over alphabets other than binary
2. Correctable errors
3. Standard array

Nonbinary codes

Nonbinary alphabets. Examples: q=3; q=4.

Ternary alphabet $\mathcal{Q}$={0,1,2} with operations mod 3.   -1=2 mod 3

The set $\mathcal{Q}^n$ forms a linear space $\{x_1,x_2,\ldots,x_{3^n}\}$

000,001,002,010,011,012,020,021,022,100,200,101,….

A ternary linear code C is a linear subspace of $\mathcal{Q}^n$. The concepts defined earlier (generator matrix, parity-check matrix, standard array, etc.) are extended straightforwardly.

C[4,2] $G=\begin{pmatrix} 0 & 1 & 2 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$ $H=\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{pmatrix}$

Distance d(C)=min. # of nonzero coordinates in a nonzero code vector.
Above: C[4,2,2]

Lemma 3.1: If G[I,A] is a generator matrix of a code C then H=[-A^T, I] can be taken as a parity-check matrix. Here A is a kx(n-k) matrix over $\mathcal{Q}$.

Quaternary alphabet. Possibilities: {0,1,2,3} with operations mod 4; but 2.2=0 which may be inconvenient in the study of linear codes. $\mathcal{Q}$={0,1,ω, $\bar{\omega}$ }. Rules of operation:

$+$	0	1	ω	$\bar{\omega}$
0	0	1	ω	$\bar{\omega}$
1	1	0	$\bar{\omega}$	ω
ω	ω	$\bar{\omega}$	0	1
$\bar{\omega}$	$\bar{\omega}$	ω	1	0

.	0	1	ω	$\bar{\omega}$
0	0	0	0	0
1	0	1	ω	$\bar{\omega}$
ω	0	ω	$\bar{\omega}$	1
$\bar{\omega}$	0	$\bar{\omega}$	1	ω

No zero divisors; it is possible to construct a linear space $\mathcal{Q}^n$.

Consider a linear code C with the generator matrix

$$G=\begin{pmatrix} 0 & 1 & 1 & \omega \\ 1 & \omega & \omega^2 & 1 \end{pmatrix}$$

Work out a parity check matrix, distance, parameters [n,k,d]

Elementary properties of linear codes

Definition 3.1: Support of a vector $\mathbf{x}$, $supp(\mathbf{x})=\{i : x_i \neq 0\}$

Thus, $wt(\mathbf{x})=|supp(\mathbf{x})|$

Let $E \subset \{1,2,...,n\}$. For a matrix $H=(\mathbf{h}_1,...,\mathbf{h}_n)$ with n columns let

$$H(E)=\{\mathbf{h}_{i_j},\ i_j \in E\}$$

.

Lemma 3.2: Let $\mathbf{x} \neq \mathbf{0}$ be a codeword in a linear code C with a p.-c. matrix H. Then the columns of $H(supp(\mathbf{x}))$ are linearly dependent. (Example p.4)

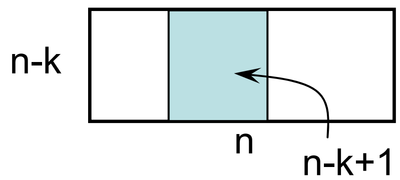
Proof: $H\mathbf{x}^T=\sum_{i \in supp(\mathbf{x})} \mathbf{h}_i = 0$

Theorem 3.3: Let C be a linear code with a parity-check matrix H. The following are equivalent:

1. distance(C)=d
2. every d-1 columns of H are linearly independent. There exist d linearly dependent columns

Corollary 3.4: Let C[n,k,d] be a code. Then $d \leq n-k+1$

Proof: H is an (n-k) x n matrix. Hence any n-k+1 col's are linearly dependent.



Example

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

every 2 col's of H are l.i. (distinct)

$\mathbf{h}_1+\mathbf{h}_2+\mathbf{h}_3=0$ (rk(H({1,2,3})=2<3)

Hence, d(C)=3

For instance,

1110000 is a codeword

Exercise: Let $E \subset \{1,2,...,n\}$. Suppose that rk(H(E))<|E|. Is it true that there is a codeword **x** with supp(**x**)=E? If not, what claim can be made instead?

Correctable errors

Let C[n,k,d] be a code

Definition 3.2: A code C corrects an error vector **e** (under minimum distance decoding) if for any $\mathbf{x} \in C$

$d(\mathbf{x},\mathbf{x}+\mathbf{e}) < d(\mathbf{y},\mathbf{x}+\mathbf{e})$ for all $\mathbf{y} \in C \backslash \mathbf{x}$

(equivalently, $w(\mathbf{e}) < d(\mathbf{y},\mathbf{x}+\mathbf{e})$)

This definition holds for all codes, linear or not

We say that a code corrects up to t errors if it corrects all error vectors $\mathbf{e} \in F$ with $w(\mathbf{e}) \leq t$

Main result:

Theorem 3.5: If $d(C) \geq 2t+1$ then the code corrects every combination of $\leq t$ errors.

Proof: Let $\mathbf{x},\mathbf{y} \in C$, $wt(e) \leq t$

$$2t+1 \leq d(\mathbf{x},\mathbf{y}) \leq d(\mathbf{x},\mathbf{x}+\mathbf{e})+d(\mathbf{y},\mathbf{x}+\mathbf{e}) \leq t+d(\mathbf{y},\mathbf{x}+\mathbf{e}), \text{ so}$$

$$d(\mathbf{y},\mathbf{x}+\mathbf{e}) > t \geq d(\mathbf{x},\mathbf{x}+\mathbf{e})$$

Let C be a code with distance 2t+1. All errors of wt $\leq$ t are correctable. There are errors of weight >t that are not correctable (generally, but not always, some errors of weight >t will be correctable)

For nonlinear codes, an error vector **e** can be correctable for some transmitted codevectors **x** and not correctable for other codevectors

Example**:** C={0000,1110,1100} d=1

x=0000 **e**=0010 correctable

x=1110 the same **e** is not correctable

Definition 3.3: The set of correctable errors for a given code vector **x** is called the Voronoi region of **x**, denoted D(**x**,C)

Let C be a code with distance 2t+1. All errors of wt $\leq$ t are correctable
There are errors of weight >t that are not correctable (generally, but not always, some errors of weight >t will be correctable)

For nonlinear codes, an error vector **e** can be correctable for some transmitted codevectors **x** and not correctable for other codevectors

Example: C={0000,1110,1100} d=1
x=0000 **e**=0010 correctable
x=1110 the same **e** is not correctable

Definition 3.3: The set of correctable errors for a given code vector **x** is called the Voronoi region of **x**, denoted D(**x**,C)

For linear codes the vector is either correctable or not for any transmitted vector of C (Voronoi regions of the codewords are congruent).

Theorem 3.6: The set of correctable errors is the same for any vector of a linear code

Proof: Let **e** be such that $d(\mathbf{x}_1+\mathbf{e},\mathbf{x}_1)<d(\mathbf{x}_1+\mathbf{e},\mathbf{x}_2)$ for all $\mathbf{x}_2 \neq \mathbf{x}_1$

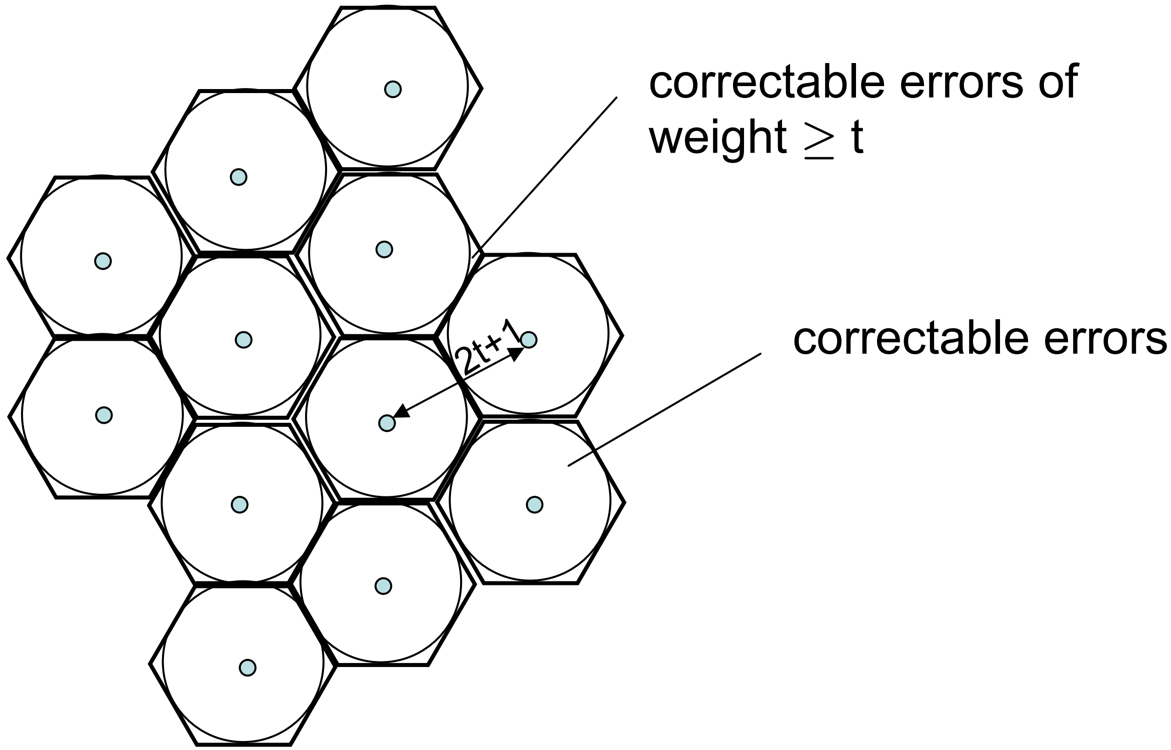
Suppose that $d(\mathbf{x}_3+\mathbf{e},\mathbf{x}_3) \geq d(\mathbf{x}_3+\mathbf{e},\mathbf{x}_4)$ for some $\mathbf{x}_3,\mathbf{x}_4$

Then take $\mathbf{y}=\mathbf{x}_1+\mathbf{x}_3$ so that $\mathbf{x}_1=\mathbf{y}+\mathbf{x}_3$

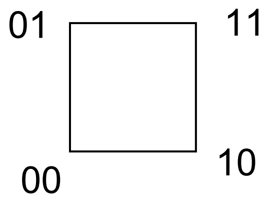
$d(\mathbf{x}_3+\mathbf{y}+\mathbf{e},\mathbf{x}_3+\mathbf{y})=d(\mathbf{x}_1+\mathbf{e},\mathbf{x}_1) \geq d(\mathbf{x}_1+\mathbf{e},\mathbf{x}_4+\mathbf{y})$, where $\mathbf{x}_4+\mathbf{y} \in C$

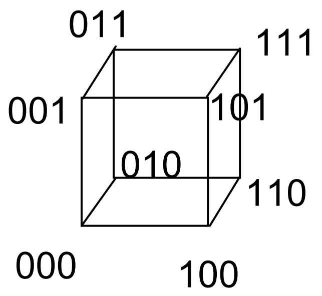
Contradiction

Useful visualization



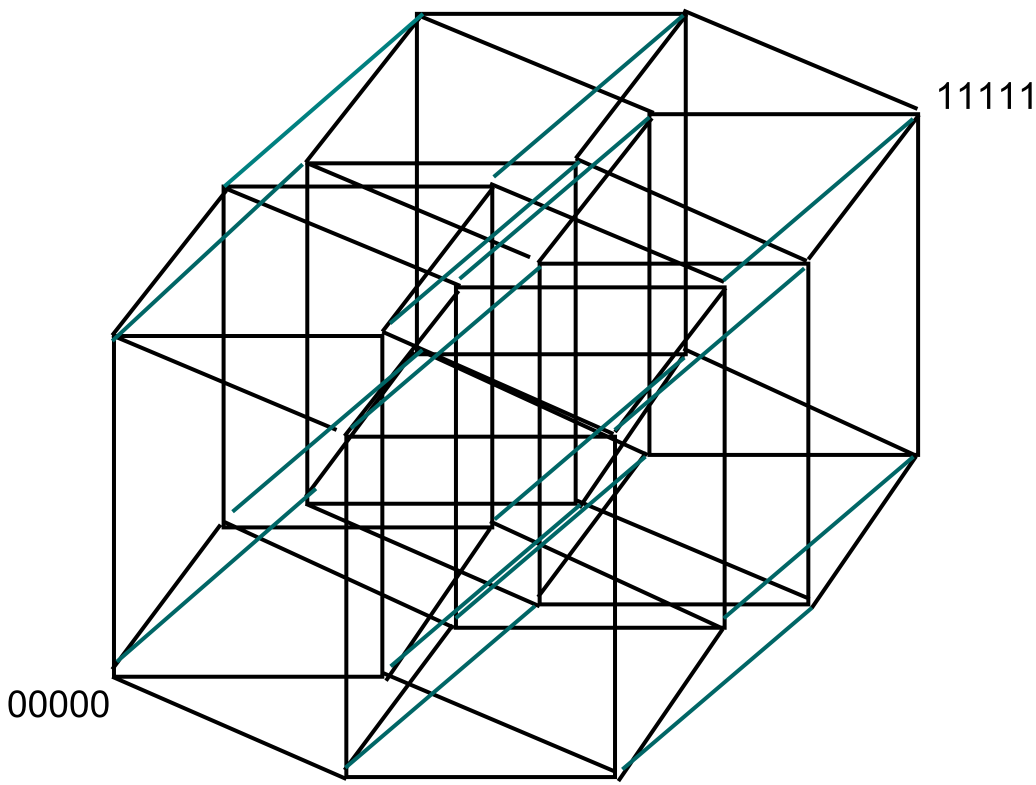
Building geometric intuition: what do spaces $\mathbb{F}_2^n$ look like?



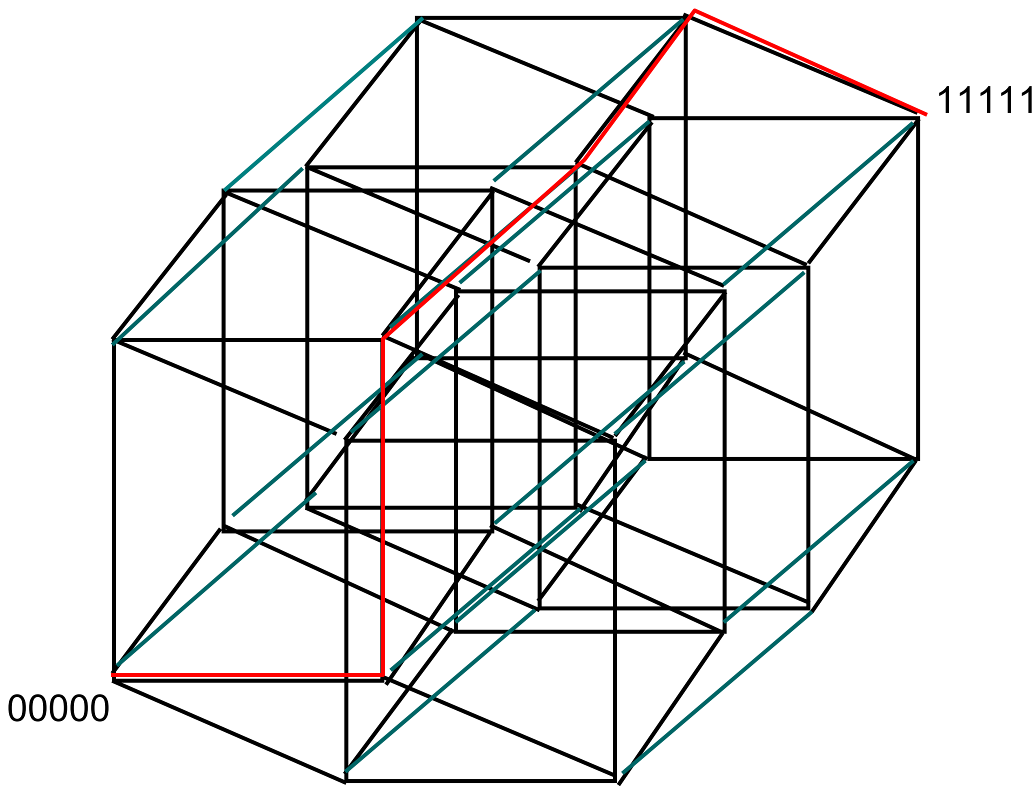


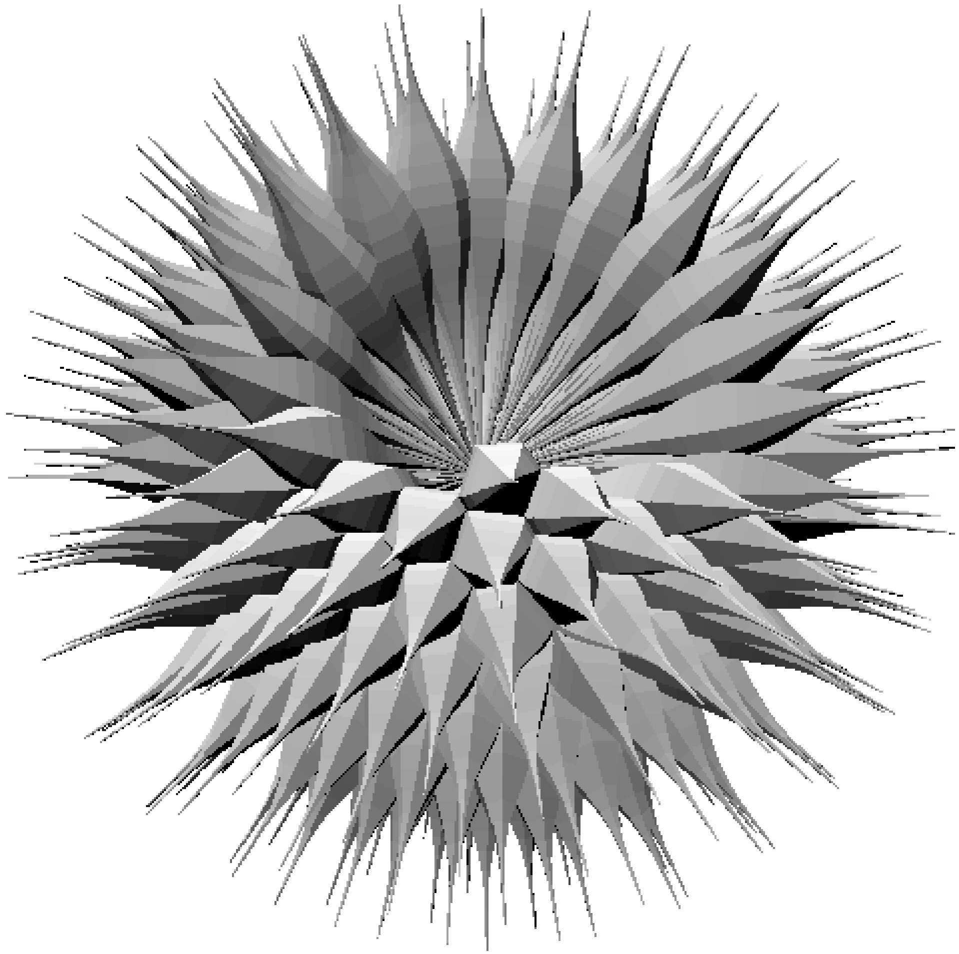
Hamming distance = number of edges in a shortest path in the graph from $\mathbf{x}_1$ to $\mathbf{x}_2$

5-dimensional Hamming cube



5-dimensional Hamming cube





8-dim hypercube projected on R^3

From here onward the codes are again binary.

Given a linear code C, let E(C) be the set of correctable errors

$$\forall_{\mathbf{e}\in E(C)}\ wt(\mathbf{e})<d(\mathbf{e},\mathbf{x}) \text{ for all nonzero } \mathbf{x}\in C$$

Given a vector $\mathbf{x}=(x_{n-1},\ldots,x_1,x_0)\in F$, consider
a binary number $X=\sum_{i=0}^{n-1} x_i\, 2^i$

Definition 3.5: Lexicographic order on F. $\mathbf{x},\mathbf{y}\in F$

$$\mathbf{x} \prec \mathbf{y} \text{ if the binary numbers } X<Y$$

defines a total order on F

$00101 \prec 01010$ etc.

(intuition: that’s how words are ordered in the dictionary, except for us all the words are of equal length)

Example:

00000	10000
00001	10001
00010	10010
00011	10011
00100	10100
00101	10101
00110	10110
00111	10111
01000	11000
01001	11001
01010	11010
01011	11011
01100	11100
01101	11101
01110	11110
01111	11111

increasing order

Standard array for a linear [n,k] code.

Consider the quotient space F/C. Make a 2^{n-k} x 2^k table as follows:
the first row is the codewords with 0 on left, otherwise ordered arbitrarily
Row i begins with the vector of the smallest weight $\mathbf{e}_i$ that is not in rows 0,...,i-1. If there are several possibilities for $\mathbf{e}_i$, we take the smallest one lexicographically

$$
\begin{array}{lllll}
0 & \mathbf{x}_1 & \mathbf{x}_2 & \ldots & \mathbf{x}_{2^k-1} \\
\mathbf{e}_1 & \mathbf{x}_1+\mathbf{e}_1 & \mathbf{x}_2+\mathbf{e}_1 & \ldots & \mathbf{x}_{2^k-1}+\mathbf{e}_1 \\
\mathbf{e}_2 & \mathbf{x}_1+\mathbf{e}_2 & \mathbf{x}_2+\mathbf{e}_2 & \ldots & \mathbf{x}_{2^k-1}+\mathbf{e}_2 \\
\mathbf{e}_3 & \mathbf{x}_1+\mathbf{e}_3 & \mathbf{x}_2+\mathbf{e}_3 & \ldots & \mathbf{x}_{2^k-1}+\mathbf{e}_3 \\
\ldots & & & & \\
\mathbf{e}_{2^{n-k}-1} & \mathbf{x}_1+\mathbf{e}_{2^{n-k}-1} & \ldots & & \mathbf{x}_{2^k-1}+\mathbf{e}_{2^{n-k}-1}
\end{array}
$$

Vectors $0, \mathbf{e}_1, \ldots, \mathbf{e}_{2^{n-k}-1}$ are called coset leaders

Exercise: Cosets are equally sized, pairwise disjoint

Lemma 3.6 (Lagrange's theorem) Let G be a finite group, F its subgroup. Then |G| is a multiple of |F|.

ENEE626 Lecture 4: Decoding of linear codes

Today's topics:

1. Maximum likelihood decoding of linear codes
   - Standard array, syndrome table
   - information sets
   - information set decoding

Theorem 4.1: E(C) = {coset leaders that are unique vectors of the smallest weight in their cosets}

Proof: Exercise

In particular, all errors of weight $\leq \lfloor (d-1)/2 \rfloor$ are unique coset leaders.

Generally, the question of locating all coset leaders is difficult.

Example 4.1:

syndrome	coset leader				
0000	000000	011101	101010	110111	Code
0001	000001	011100	101011	110110	correctable error
0010	000010	011111	101000	110101	
0100	000100	011001	101110	110011	
1000	001000	010101	100010	111111	
1101	010000	001101	111010	100111	
1010	100000	111101	001010	010111	
0011	000011	011110	101001	110100	
0101	000101	011000	101111	110010	not correctable
0110	000110	011011	101100	110001	
1001	001001	010100	100011	111110	
1100	001100	010001	100110	111011	
1111	010010	001111	111000	100101	
1011	100001	111100	001011	010110	
1110	100100	111001	001110	010011	
0111	110000	101101	011010	000111	

recover a p.-c.m

$$H = \begin{matrix} 111000 \\ 010100 \\ 100010 \\ 010001 \end{matrix}$$

Syndrome table

C[n,k]; H parity-check matrix

$\mathbf{x} \in C$ $\quad H\mathbf{x}^T=(000...000)^T$
$\mathbf{y} \notin C$ $\quad H\mathbf{y}^T=\mathbf{s}$

Lemma 4.2: Let $\mathbf{e}_i$ be a coset leader, $\mathbf{y} \in C+\mathbf{e}_i$ be a vector from the same coset. Then $H\mathbf{e}_i^T = H\mathbf{y}^T$

The vector $\mathbf{s}_i=H\mathbf{e}_i^T$ determines the coset uniquely. $\mathbf{s}_i$ is called the syndrome (of this coset).

Definition 4.1: The Syndrome table is an array of pairs

(syndrome, coset leader) (see Example 4.1)

2^{n-k} pairs, total size $(2n-k)2^{n-k}$ bits

Maximum likelihood (ML) decoding
(decoding by minimum distance).

Compute the syndrome of the received vector $s=H\,\mathbf{y}^T$
Decode $\mathbf{y} \to \mathbf{y}+\mathbf{e}$ (coset leader)

Complexity of ML decoding $O(n2^k)$ time complexity
or $O(n\,2^{n-k})$ space complexity to store the syndrome table

Constructing the syndrome table generally is difficult (exhaustive search). Becomes infeasible for large codes.

Error probability of ML decoding for a linear code on a BSC(p):

$P_e(\mathbf{x})$=P(decoding incorrect | $\mathbf{x}$ transmitted) does not depend on $\mathbf{x}$ (Thm. 3.5)

$$P_{correct}=\sum_{i=0}^{n} S_i\, p^i\, (1-p)^{n-i}$$

where S_i= #(coset leaders of wt i that are correctable errors)

General definition of ML decoding

Definition 4.1: Suppose that a code C is used for transmission over a BSC. Let $\mathbf{y} \in \{0,1\}^n$ be a received vector. The maximum likelihood decoding rule is a mapping $\psi: \{0,1\}^n \mapsto C$ such that

$$\psi(\mathbf{y}) = \arg\max_{\mathbf{x} \in C} \Pr[\mathbf{y}|\mathbf{x}]$$ (if there are several solutions, declare an error)

In the case of linear codes, this definition is equivalent to the definition on the previous slide.

Information set decoding

(Another implementation of ML decoding):

Let G[$\mathbf{g}_1,\mathbf{g}_2,\ldots,\mathbf{g}_n$] be a generator matrix of a linear code

$\mathbf{g}_i$ – a binary k-column

Definition 4.2: A subset of coordinates $i_1,i_2,\ldots,i_k$ is called an information set if the columns $\mathbf{g}_{i_1},\mathbf{g}_{i_2},\ldots,\mathbf{g}_{i_k}$ are linearly independent.

Definition 4.3: A code matrix is an $M \times n$ matrix whose rows are the codewords.

A subset $i_1,i_2,\ldots,i_k$ forms an information set if the submatrix of the code matrix with columns with these indices contains all the possible 2^k rows (exactly once each).

Lemma 4.3: A codeword can be recovered from its k coordinates in any information set.

Information set decoding: Input G, **y**, output $\mathbf{c}=\psi_{ML}(\mathbf{y})$

Set **c**=0

Take an information set $(i_1,...,i_k)$, compute the codeword **a** s.t.

$a_{i_j}=y_{i_j}$, $1\leq j \leq k$

If $d(\mathbf{a},\mathbf{y}) < d(\mathbf{c},\mathbf{y})$, set $\mathbf{c} \leftarrow \mathbf{a}$

Repeat for every information set.

Complexity $O\left(n^3\binom{n}{k}\right)$

Recall: Support of a vector $\mathrm{supp}(\mathbf{x})=\{i: x_i \neq 0\}$

Lemma 4.4: Let **e** be a correctable coset leader. The subset $S=\{1,2,...,n\}\setminus\mathrm{supp}(\mathbf{e})$ contains an information set (information set decoding is ML)

Proof: Let $Q=\mathrm{supp}(\mathbf{e})$. $H\mathbf{e}^T=\sum_{i\in Q}e_i\mathbf{h}_i=\mathbf{s}$.

No **e**' with $\mathrm{supp}(\mathbf{e}')\subset\mathrm{supp}(\mathbf{e})$ satisfies $H(\mathbf{e}')^T=s$; hence, $\mathrm{rank}(H(Q))=|Q|$.

$\Rightarrow |Q| \leq n-k$, $|S|\geq k$

Let $\mathbf{x}_1,\mathbf{x}_2\in C$, $\mathbf{x}_1\neq \mathbf{x}_2$ be such that $\mathrm{proj}_S\, \mathbf{x}_1=\mathrm{proj}_S\, \mathbf{x}_2$. Then

$$\emptyset \neq \mathrm{supp}(\mathbf{x}_1+\mathbf{x}_2)\subset Q$$

$(\mathbf{x}_1+\mathbf{x}_2)+\mathbf{e} \in C+\mathbf{e}$ (same coset as **e**) but is of weight smaller than **e**, contradiction.

Example :

$$G = \begin{bmatrix} 010011000 \\ 011100100 \\ 111100010 \\ 111010001 \end{bmatrix}$$

There are $\binom{9}{4} = 126$ 4-subsets of $\{1, 2, \ldots, 9\}$

Subsets {1,2,3,4},{1,2,3,5}, {1,2,3,6},... are information sets

Subsets {3,7,8,9},... are not.

Generally it is difficult to find the number of information subsets of a linear code. Some indication of what to expect is given by considering random matrices.

ENEE626 Lecture 5

Today's topics:

1. Rank of random binary matrices
2. The Hamming code; perfect codes
3. The dual of the Hamming code (the simplex code)

Rank of random matrices

Given a random code, can we perform information set decoding?

Theorem 5.1:

Let G be a random $k \times n$ binary matrix whose entries are chosen independently of each other with $p(1) = p(0) = 1/2$. Let $k = Rn, R < 1$.

Then $\lim_{n \to \infty} \Pr[\text{rk}(G) = k] \to 1$

Proof of part (a):

Number of nonsingluar k x n matrices is

$(2^n-1)(2^n-2)(2^n-2^2)\ldots(2^n-2^{k-1})$

$$\Pr[\mathrm{rk}(G)=k] = \frac{(2^n-1)(2^n-2)(2^n-2^2)\ldots(2^n-2^{k-1})}{2^{nk}} = \prod_{i=0}^{k-1}(1-2^{-n+i})$$

$$> 1-\sum_{i=0}^{k-1}2^{-n+i} = 1-2^{-n+k-1}\left(1+\frac{1}{2}+\cdots+\frac{1}{2^{k-1}}\right)$$

$n \to \infty, \frac{k}{n} = R < 1$

$$= 1-2^{-n(1-R)-1}\cdot 2(1-2^{-k}) \to 1 \qquad \blacktriangle$$

In particular, let k=n. The probability that an n x n matrix over $\mathbb{F}_2$ is nonsingular equals

$\prod_{i=0}^{n-1}(1-2^{-n+i})$

One can prove that this product converges as $n\to\infty$. The limiting value is 0.2889.

The Hamming code

$\mathcal{H}_3[7,4,3]$ is a linear code with the p.-c.matrix

$$H=\begin{pmatrix}0001111\\0110011\\1010101\end{pmatrix}$$ all nonzero 3-columns

$\dim(\mathcal{H}_3)=4$, distance=3

$$G=\begin{pmatrix}1 & & & & 011\\ & 1 & & & 101\\ & & 1 & & 110\\ & & & 1 & 111\end{pmatrix}$$

Syndrome table:

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

$|\text{Coset}|=|\mathcal{H}_3|=16=2^k$

8 cosets $\Rightarrow$ $128=2^7$

All single errors are correctable

$d \geq 3=2\times1+1$

Spheres in F:

$B_t(\mathbf{x})=\{\mathbf{y}\in F: d(\mathbf{x},\mathbf{y})\leq t\}$

$Vol(B_t(\mathbf{x}))$ denotes the volume of $B_t(\mathbf{x})$ (number of points in the ball)

Proposition: $\text{vol}(B_t(\mathbf{x})) = \sum_{i=0}^{t} \binom{n}{i}$

Volume does not depend on the center

Spheres of radius 1 about the c-words of the Hamming code are pairwise disjoint
$vol(B_1)=1+7=8$
total volume of spheres around the codewords$=2^k\, vol(B_1)=16 \times 8=128$
exhausts $\mathbb{F}_2^{\,7}$
Notation: C(n,M,d) a binary code of length n, size M, distance d

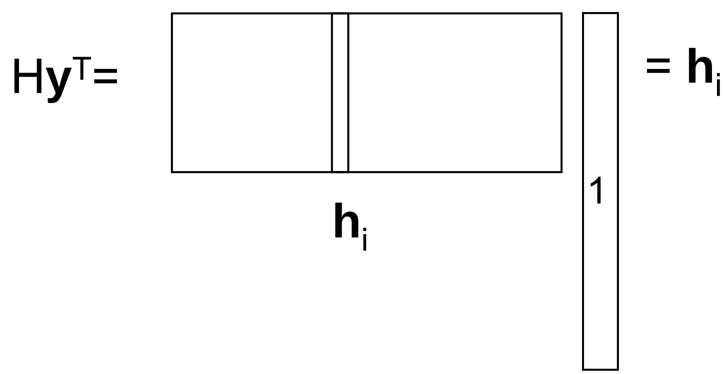
Definition 5.1: Perfect code C(n,M,2t+1)=spheres of radius t about the codewords contain all the poinrs of $\mathbb{F}_2^{\,n}$

$$M \sum_{i=0}^{t} \binom{n}{i} = 2^n$$

Perfect codes are good but rare. Linear perfect codes are all known.

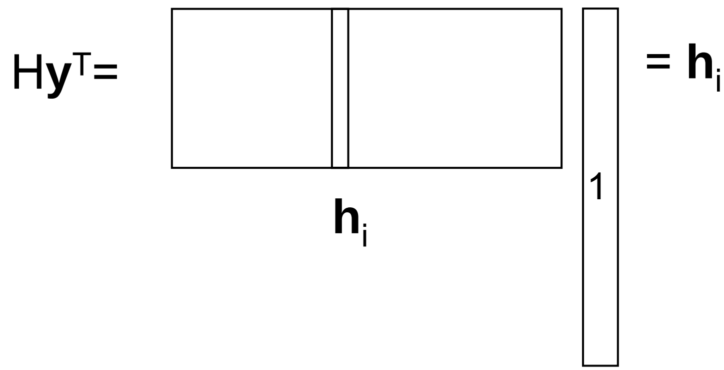
The Hamming code $\mathscr{H}_3$ is a linear 1-error-correcting perfect code.

Generalize: $\mathscr{H}_m[2^m-1,2^m-m-1,3]$ $\mathscr{H}_m$=[all m-columns]
Exercise: compute G_m.
Decoding: correct 1 error. W.l.o.g. assume that we transmit **x**=0
Transmit **x**, receive **y**=(00...010......00)
i



columns ordered lexicographically: then $\mathbf{h}_i$ gives the number of the coordinate in error. To decode, flip that coordinate.

No double, triple, ..., errors are correctable



Message:
to correct 1 error
we need about log n
parity check bits

columns ordered lexicographically: then $\mathbf{h}_i$ gives the number of the coordinate in error. To decode, flip that coordinate.

No double, triple, ..., errors are correctable

Definition 5.2: Let C be a binary linear code. The **dual code** is

$$C^{\perp}=\{\mathbf{x}\in F: \forall_{\mathbf{c}\in C}\ (\mathbf{x},\mathbf{c})=0\}$$

Properties: $C^{\perp}$ is an [n,n-k] linear code generated by H, the p.-c. matrix of C.
Distance of $C^{\perp}$ =? Generally not immediate.

$(\mathscr{H}_m)^{\perp} = S_m[2^m-1, m, (n+1)/2 = 2^{m-1}]$ called the simplex code

a very low-rate code with a very large distance

Exercise: Is $(111...111) \in \mathscr{H}_m$?

Lemma 5.3: $d(S_m) = 2^{m-1}$

Proof: Induction on m

$$G_2 = \begin{bmatrix} 011 \\ 101 \end{bmatrix}, \quad S_2 = \begin{matrix} 000 \\ 011 \\ 101 \\ 110 \end{matrix}$$

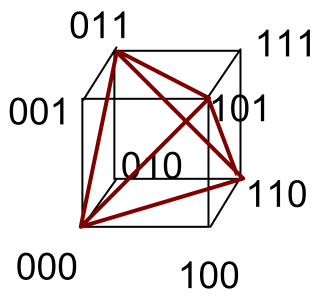
$$G_3 = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ & G_2 & & 0 & & G_2 & \\ & & & 0 & & & \end{bmatrix}$$

induction step

$$S_3 = \begin{array}{|ccc|} & 0 & \\ S_2 & 0 & S_2 \\ & 0 & \\ & 0 & \\ \hline & 1 & \\ S_2 & 1 & \overline{S_2} \\ & 1 & \\ & 1 & \end{array}$$

the bar means negation $1 \to 0$, $0 \to 1$

The term “simplex”



ENEE626 Lecture 6:

1. Weight distribution of the Hamming code.
2. Code optimality, the Hamming and Plotkin bound
3. The binary Golay codes
4. Operations on codes.

Let $A_w = |\{\mathbf{x} \in C: \text{weight}(\mathbf{x}) = w\}|$

Definition 6.1: The vector $(A_0=1, A_1, \ldots, A_w, \ldots, A_n)$ is called the **weight distribution** of the code C.

Clearly, $A_1 = A_2 = \ldots = A_{d-1} = 0$

Theorem 6.1: Let $C = \mathscr{H}_m$.

$$A_3 = \frac{1}{3}\binom{n}{2} = \frac{n(n-1)}{6}$$

$$A_4 = \frac{1}{4}\left(\binom{n}{3} - A_3\right)$$

Proof: Let $\text{wt}(\mathbf{x}) = 2$, then $\exists$ unique $\mathbf{c} \in C$ with $\text{d}(\mathbf{c}, \mathbf{x}) = 1$ (C is perfect); so $\text{wt}(\mathbf{c}) = 3$.
3 different $\mathbf{x}$ give rise to $\mathbf{c}$. So $A_3 = \frac{1}{3}\binom{n}{2}$.
Similarly, let $\text{wt}(\mathbf{x}) = 3$, then either $\mathbf{x} \in C$ or $\exists$ unique $\mathbf{c} \in C$ with $\text{d}(\mathbf{c}, \mathbf{x}) = 1$, so $\text{wt}(\mathbf{c}) = 4$. Hence $A_3 + 4A_4 = \binom{n}{3}$. QED

$$A_4(\mathcal{H}_{m,\text{ext}}) = A_3(\mathcal{H}_m) + A_4(\mathcal{H}_m) = \frac{2^{m-2}(2^m - 1)(2^{m-1} - 1)}{3}$$

In principle, such recurrences can be used to compute the next weight coefficients in $\mathscr{H}_m$, but there is a more efficient way (MacWilliams' theorem, lect.7)

Interlude: The Hat Problem

$n=2^m-1$ people are given hats one each, either red or blue.
At the same time they all walk into a room and see the hats of everyone else except their own. Then they guess **simultaneously** the color of their own hats (if unsure they can pass). If those who do not pass **all** make a correct guess, the entire group win $1 each, otherwise they lose $1 each.

They can follow a pre-arranged strategy. Is there a strategy that will win in more than 50% of color deals in the long run?

(Was popular a few years ago; The New York Times ran a front-page article)

Definition 6.2: A code of length n with M codewords and distance d is called optimal if there does not exist an (n,M+1,d) code.

Theorem 6.2: The Hamming code is optimal.

Proof: Let C[n,k,d] be a code, then

$$2^k \,\mathrm{vol}(B_{\lfloor \frac{d-1}{2} \rfloor}) \leq 2^n \text{ (the \textbf{Hamming bound})}$$

In particular, for $\mathcal{H}_m$, $t = 1$ and $2^k(n+1) = 2^{2^m-m-1} \cdot 2^m = 2^n$, so the bound is met with equality.

Generally, if C is optimal, $C^{\perp}$ is not always optimal. However, this is true for S_m

Theorem 6.3 (the Plotkin bound) Let C[n,k,d] be a linear code. Then

$$k \leq \log_2 \frac{2d}{2d-n}$$

Proof: Consider the $(M = 2^k \times n)$ code matrix. The total $\sharp$ of 1's in it is $\leq nM/2$ There are $M-1$ nonzero rows in the matrix, so the average weight of a nonzero row is $\bar{w} \leq \frac{nM}{2(M-1)}$. Also $d \leq \bar{w}$. QED

In the $[2^m-1,m,2^{m-1}]$ simplex code, 2d/(2d-n)=(n+1)/(n+1-n)=n+1=M

The Plotkin bound

It is also true for unrestricted codes, by the following argument.
Let C(n,M,d) be a code. Compute the average distance between x,y∈ C.
Let λ_i be the # of 1's in the ith column of the code matrix.

$$\sum_{\mathbf{x},\mathbf{y}\in C} \mathrm{d}(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n} 2\lambda_i(M-\lambda_i) \leq \sum 2\frac{M}{2}(M-\frac{M}{2}) = \frac{nM^2}{2}$$

$$M(M-1)d \leq \frac{nM^2}{2}$$

$$M \leq \frac{2d}{2d-n}$$

The Golay code: another binary perfect code

There exists a code $\mathcal{G}_{23}$[23,12,7] that corrects 3 errors

Verify that $\mathcal{G}_{23}$ is perfect

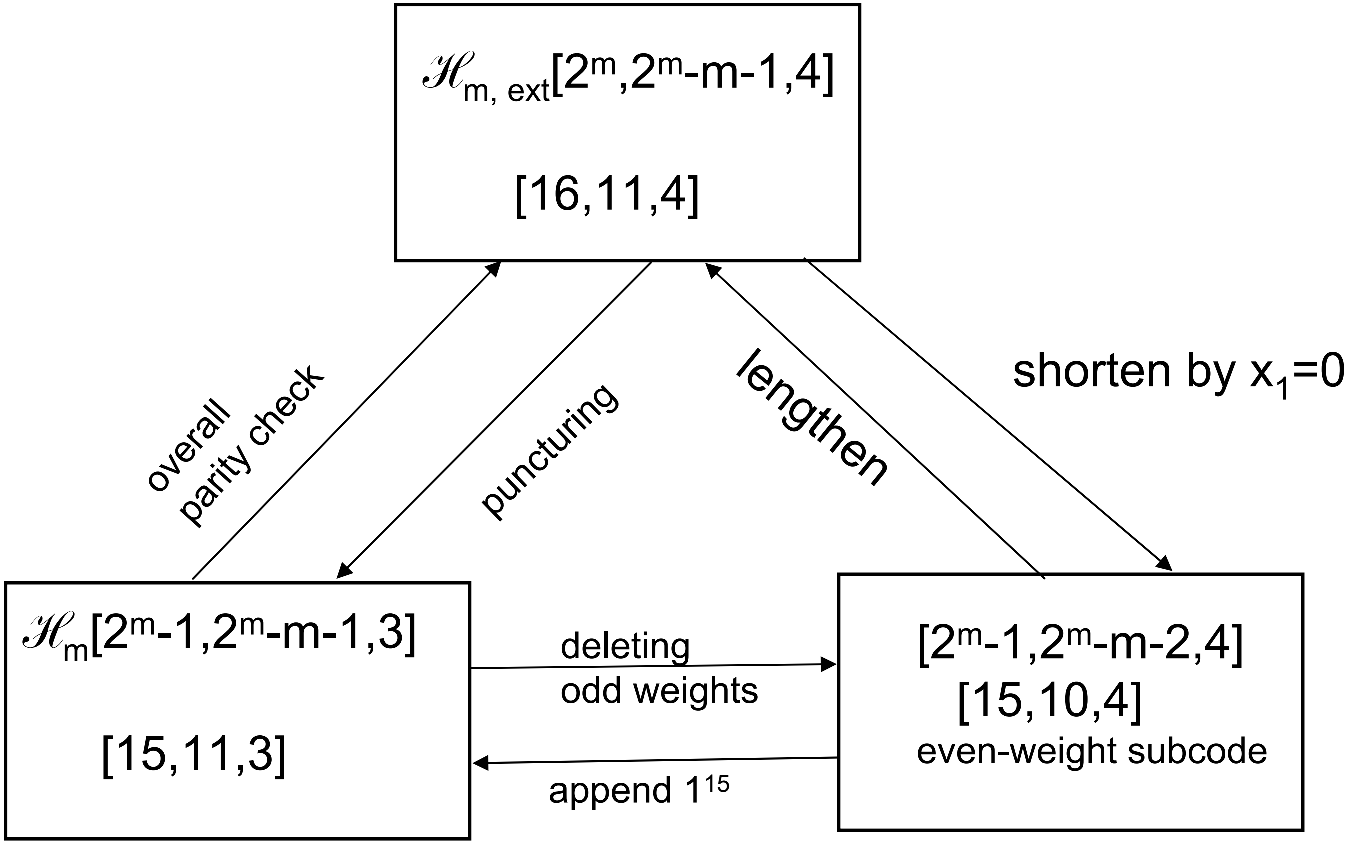
$$2^{12}(1+23+\binom{23}{2}+\binom{23}{3}) = 2^{23}$$

Let $\mathcal{G}_{24}[24,12,8] = \mathcal{G}_{23,\text{ext}}$ The code $\mathcal{G}_{24}$ is self-dual: $\mathcal{G}_{24} = \mathcal{G}_{24}^{\perp}$.

Both codes have a number of other remarkable properites (see the reference books)

The only other binary linear perfect codes that exist are trivial:
[n,n,1] ($n \geq 1$), [2m+1,1,2m+1] ($m \geq 1$)
Moreover, the only possibility for a nonlinear code to be perfect is that its parameters coincide with the parameters of $\mathcal{H}_m$

Operations on codes



Operations on codes: Definitions

Let $C[n,k,d \geq 2]$ be a linear code.
Assume that the code (matrix) does not contain all-zero columns

- **Puncturing** $\mathbf{x} \mapsto \text{proj}_{\{1,\dots,n\}\setminus i}\, \mathbf{x}$ (projection)

$C[n,k,d] \to C'[n-1,k,\geq d-1]$

- **Shortening** $C[n,k,d] \to C'[n-1,k-1,\geq d]$

Lemma 6.4 (Lagrange's theorem). A column in the code matrix contains 2^{k-1} 0's and 2^{k-1} 1's.
To shorten C, take 2^{k-1} codevectors with a 0 in coord. i, remove the rest of C, delete that coordinate.

- **Even weight subcode** $C[n,k,d=2t+1] \to C'[n,k-1,d+1]$

delete all odd-weight codewords

- **Adding overall parity check** $C[n,k,d=2t+1] \to C_{ext}[n+1,k,2t+2]$

C_{ext} is called the extended code

Exercise. Let C be optimal. Is C_{ext} also optimal?

- **Lengthening** $C[n,k,d] \to C'[n+1,k+1]$

add an overall parity check; append the vector 1^{n+1} to the basis of C_{ext}

More ways to create a new code from known codes

|u|u+v| construction. Let $A[n,k_1,d_1]$ and $B[n,k_2,d_2]$ be binary linear codes.

$$C=(|\mathbf{u}|\mathbf{u}+\mathbf{v}|,\ \mathbf{u} \in A,\ \mathbf{v} \in B)$$

Lemma 6.5: C is a $[2n,k_1+k_2,\min(2d_1,d_2)]$ code

Proof: Let $\mathbf{c} \in C$, $\mathbf{c} \neq 0$, $\mathbf{v}=0$, then $wt(c) \geq 2d_1$

On the other hand, if $\mathbf{v} \neq 0$, then

$$wt(\mathbf{c})=wt(\mathbf{u})+wt(\mathbf{u}+\mathbf{v}) \geq wt(\mathbf{u})-wt(\mathbf{u})+wt(\mathbf{v})=wt(\mathbf{v}) \geq d_2$$

(triangle inequality $wt(\mathbf{x}+\mathbf{y}) \leq wt(\mathbf{x})+wt(\mathbf{y})$)

Example: Let $A=S_{m,ext}$, $A[2^m,m+1,2^{m-1}]$

$B[2^m,1,2^m]$

Then $C[2^{m+1},m+2,2^m]=S_{m+1,ext}$

ENEE626 Lecture 7: Weight distributions. The MacWilliams theorem

Weight distributions
Bhattacharyya bound
The MacWilliams theorem
Fourier transform

Weight distributions

C a linear code, $A_w = |\{\mathbf{x} \in C, wt(\mathbf{x}) = w\}|$

$(A_0, A_1, \ldots, A_n)$ weight distribution of a linear code C

Define the generating function of weights (the weight enumerator)

$$A(x,y) = \sum_{i=0}^{n} A_i x^{n-i} y^i$$

$\mathscr{H}_3[7,4,3]$

i	0	1	2	3	4	5	6	7
	1	0	0	7	7	0	0	1

$A(x,y) = x^7 + 7x^4y^3 + 7x^3y^4 + y^7$

$\mathscr{S}_3[7,3,4]$

i	0	1	2	3	4	5	6	7
	1	0	0	0	7	0	0	0

$A^{\perp}(x,y) = x^7 + 7x^3y^4$

The weight enumerator of the code dual to C will be denoted by $A^{\perp}(x,y)$; $A^{\perp}(x,y) = \sum_i A_i^{\perp} x^{n-i} y^i$,

Motivation to study weight distributions

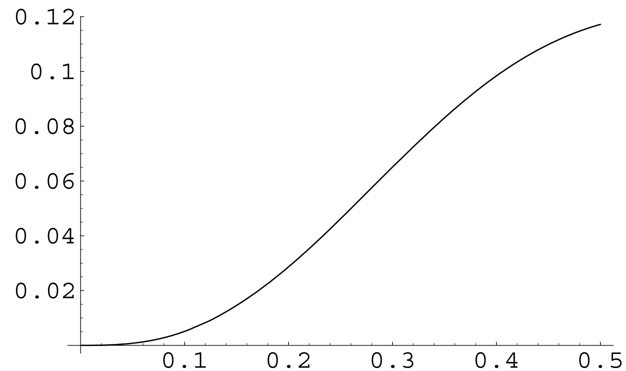
1. The perfect code theorem from last lecture is proved using general properties of weight distriburions.

2. Error detection. Suppose an [n,k,d] linear code C with weight enumerator A(x,y) is transmitted over a binary symmetric channel BSC(p) and used for error detection. Namely, the received vector is tested for being a code vector; if not, an error is declared. The probability of undetected error equals

$$P_{ud}(C)=\sum_{i=1}^{n} A_i p^i (1-p)^{n-i} = A(1-p,p)-(1-p)^n$$

For instance, let C be the [7,4,3] Hamming code $\mathcal{H}_3$.

$P_{ud}(\mathcal{H}_3)$



p

Motivation to study weight distributions

3. Error prob. of ML decoding. Suppose an [n,k,d] linear code with weight enumerator A(x,y) is transmitted over a binary symmetric channel BSC(p) and decoded by Max-likelihood (syndrome decoding). Let $P_e(\mathbf{c})$ be the probability of error conditioned on transmitting the codeword **c**;

$$P_e(C):=2^{-k} \sum_{c\in C} P_e(\mathbf{c})$$

Then

$$P_e(C) \le A(1, 2\sqrt{p(1-p)}) - 1 \quad \text{(Bhattacharyya bound)}$$

Proof. Suppose that the transmitted vector is 0 (does not matter);
Let D(0) be the Voronoi region of 0. Let $P_{e,\mathbf{c}'}(0)$=Pr(decode to $\mathbf{c}'$|0)

$$P_e(0) = \sum_{\mathbf{c}'\in C\setminus 0} P_{e,\mathbf{c}'}(0)$$

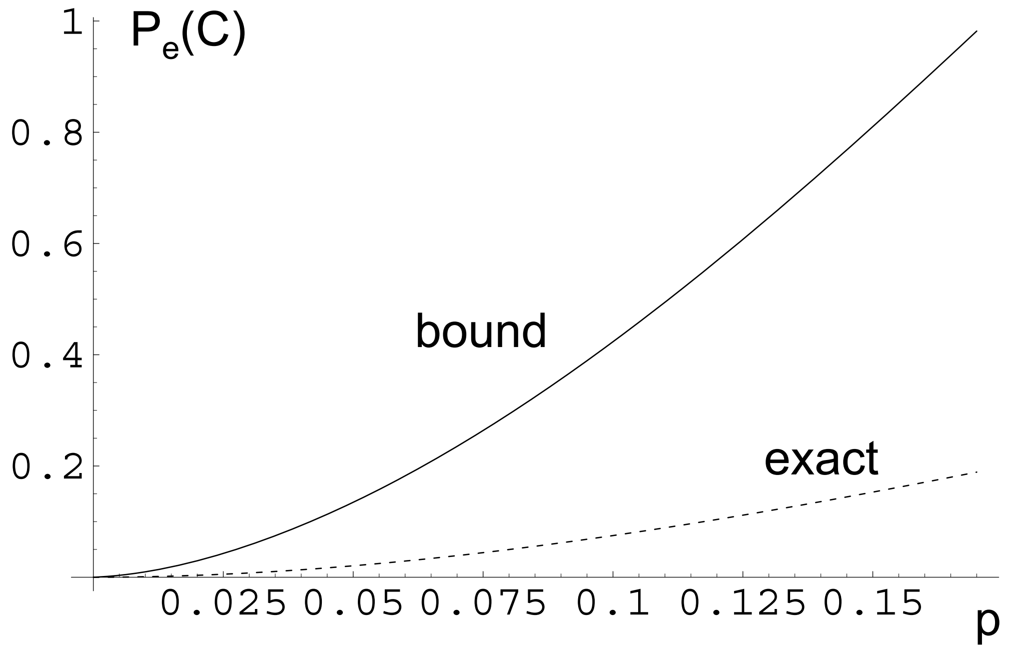
$$= \sum_{\mathbf{c}'\in C\setminus 0} \sum_{\mathbf{y}\in D(\mathbf{c}')} P(\mathbf{y}|0) \le \sum_{\mathbf{c}'\in C\setminus 0} \sum_{\mathbf{y}\in D(\mathbf{c}')} \sqrt{P(\mathbf{y}|0)P(\mathbf{y}|\mathbf{c}')}$$

$$= \sum_{\mathbf{c}'\in C\setminus 0} \sum_{\mathbf{y}\in D(\mathbf{c}')} \prod_{i=1}^{n} \sqrt{P(y_i|0)P(y_i|c_i)} \le \sum_{\mathbf{c}'\in C\setminus 0} \prod_{i=1}^{n} \sum_{y=0}^{1} \sqrt{P(y|0)P(y|c_i')}$$

$$= \sum_{\mathbf{c}'\in C\setminus 0} (2\sqrt{p(1-p)})^{\mathrm{wt}(\mathbf{c}')} = \sum_{w=1}^{n} A_w (2\sqrt{p(1-p)})^{w} \qquad \blacktriangle$$

Example: The [6,2,3] code C from Example 4.1

correctable coset leaders $S_0=1$; $S_1=6$; $S_2=6$
weight distribution: $A_3=A_4=A_5=1$
Bhattacharyya bound: $P_e(C)=\gamma^3(1+\gamma+\gamma^2)$, $\gamma=2(p(1-p))^{1/2}$
Exact value: $P_e(C)=1-((1-p)^6+6p(1-p)^5+6p^2(1-p)^4)$



Note: there are better bounds for $P_e(C)$ for large p

Main result about the weight distributions

Theorem 7.1:(MacWilliams) $A^{\perp}(x,y)=2^{-k} A(x+y,x-y)$

So $A(x,y)=2^{-n+k} A^{\perp}(x+y,x-y)$

Example: compute the weight enumerator of $\mathscr{H}_3$ from the w.e. of $\mathscr{S}_{3:}$

$A^{\perp}(x+y,x-y)=(x+y)^7+7(x+y)^3(x-y)^4=8x^7+56\, x^4y^3+56x^3y^4+8y^7$

$=2^{-7+4} A(x,y)$

Let $f(x_1, x_2, \ldots, x_n)$ be a function

E.g., $f(x_1, x_2, x_3) = x_1 + x_2 x_3; f(011) = 1$

Let $(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} x_i y_i$ be the dot product

Definition 7.1: The *Fourier (Hadamard) transform* of f

$$\widehat{f}(\mathbf{u}) = \sum_{\mathbf{v} \in \mathbb{F}_2^n} (-1)^{(\mathbf{u},\mathbf{v})} f(\mathbf{v})$$

Lemma 7.2: Let $C[n, k]$ be a linear code. Then

$$\sum_{\mathbf{u} \in C^{\perp}} f(\mathbf{u}) = \frac{1}{2^k} \sum_{\mathbf{u} \in C} \widehat{f}(\mathbf{u})$$

Proof:

$$\begin{aligned}
\sum_{\mathbf{u} \in C} \widehat{f}(u) &= \sum_{\mathbf{u} \in C} \sum_{\mathbf{v} \in \mathbb{F}_2^n} (-1)^{(\mathbf{u},\mathbf{v})} f(\mathbf{v}) = \sum_{\mathbf{v} \in \mathbb{F}_2^n} f(\mathbf{v}) \sum_{\mathbf{u} \in C} (-1)^{(\mathbf{u},\mathbf{v})} \\
&= \sum_{\mathbf{v} \in C^{\perp}} f(\mathbf{v}) \sum_{\mathbf{u} \in C} (-1)^{(\mathbf{u},\mathbf{v})} + \sum_{\mathbf{v} \notin C^{\perp}} f(\mathbf{v}) \sum_{\mathbf{u} \in C} (-1)^{(\mathbf{u},\mathbf{v})} \\
&= |C| \sum_{\mathbf{v} \in C^{\perp}} f(\mathbf{v})
\end{aligned}$$

Proof [of the MacWilliams theorem]: take in the lemma $f(\mathbf{u}) = x^{n-\mathrm{wt}(\mathbf{u})}y^{\mathrm{wt}(\mathbf{u})}$
Let $\mathbf{u} = (u_1, \ldots, u_n), \mathbf{v} = (v_1, \ldots, v_n)$

$$\widehat{f}(\mathbf{u}) = \sum_{\mathbf{v}\in F} (-1)^{u_1v_1+\cdots+u_nv_n} \prod_{i=1}^{n} x^{1-v_i}y^{v_i} = \sum_{v_1=0}^{1}\sum_{v_2=0}^{1}\cdots\sum_{v_n=0}^{1}\prod_{i=1}^{n}(-1)^{u_iv_i}x^{1-v_i}y^{v_i}$$

$$= \prod_{i=1}^{n}\sum_{z=0}^{1}(-1)^{u_iz}x^{1-z}y^{z} = \prod_{i=1}^{n}(x+(-1)^{u_i}y) = (x+y)^{n-\mathrm{wt}(\mathbf{u})}(x-y)^{\mathrm{wt}(\mathbf{u})}$$

Then

$$\sum_{\mathbf{x}\in C^{\perp}} f(\mathbf{x}) = \frac{1}{2^k}\sum_{\mathbf{y}\in C}\widehat{f}(\mathbf{y})$$

$$\sum_{\mathbf{x}\in C^{\perp}} x^{n-\mathrm{wt}(\mathbf{x})}y^{\mathrm{wt}(\mathbf{x})} = \frac{1}{2^k}\sum_{\mathbf{y}\in C}(x+y)^{n-\mathrm{wt}(\mathbf{y})}(x-y)^{\mathrm{wt}(\mathbf{y})}$$

$$\sum_{w=0}^{n} A_w^{\perp}x^{n-w}y^{w} = \frac{1}{2^k}\sum_{w=0}^{n}A_w(x+y)^{n-w}(x-y)^{w}$$

$$2^kA^{\perp}(x,y) = A(x+y, x-y)$$

∎

Nonbinary codes

Let C be a linear code of length n over $\mathbb{F}_q$

(means that $\mathbf{x},\mathbf{y}\in C \Rightarrow a\mathbf{x}+b\mathbf{y}\in C$)

For instance, $\mathbb{F}_3=\{0,1,2\}$ with operations mod 3

Definition 7.3. Let $\mathbf{x}=(x_1,x_2,\ldots,x_n)$ be a vector. The Hamming weight $wt(\mathbf{x})=|\{i: x_i \neq 0\}|$. The Hamming distance

$$d(\mathbf{x},\mathbf{y})=wt(\mathbf{x}-\mathbf{y})$$

The weight distribution of the code C

$$(A_0,A_1,\ldots,A_n)$$

The weight enumerator $A(x,y)=\sum_{i=0}^{n} A_i x^{n-i}y^i$

Definition 7.4: The dual code $C^{\perp}=\{\mathbf{y}\in (\mathbb{F}_q)^n : \forall_{x\in C} (\mathbf{x},\mathbf{y})=0\}$

where $(\mathbf{x},\mathbf{y})=\sum_{i=1}^{n} x_i y_i$ (operations in $\mathbb{F}_q$)

Theorem 8.4 (MacWilliams): $A^{\perp}(x,y)= q^{-k} A(x+(q-1)y,x-y)$

Both proofs carry over to the general case