ERROR-CORRECTING CODES

P. VIJAY KUMAR
Course Outline:
Lectures 1-15

Basics of Block Codes (2)

Mathematical Preliminaries: groups, rings and fields (3)

Mathematical Preliminaries: vector spaces (3)

Linear Codes (3)

Bounds on Code Size (2)

Standard Array Decoding (2)
Course Outline:
Lectures 16-42

Convolutional Codes (5)

Generalized Distributive Law (8)

Low-Density Parity-Check Codes (7)

Finite Fields (4)

Cyclic Codes (3)

The End! The End! The End! The End!
Lec 1: Course Overview & Basics

- Basics of Binary Codes
- Linear Binary Codes
- Convolutional Codes
Generalized Distributive Law

LDPC codes
(low density parity check)

Turbo codes

Finite Fields

Cyclic codes:
Reed-Solomon codes
BCH codes
Channel Model

\[
\begin{align*}
T_x & \quad 1 - \epsilon \\
\epsilon & \quad 1 - \epsilon \\
R_x & \quad 1
\end{align*}
\]

Binary Symmetric Channel (BSC)

Additive White Gaussian Noise Channel

\[n + s = z\]
$F_2 = \{0, 1\}$ arithmetic is modulo 2

<table>
<thead>
<tr>
<th>+</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**Addition (XOR)**

<table>
<thead>
<tr>
<th>.</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Multiplication (AND)**
\[
\begin{align*}
\mathbb{F}_{2}^{n} &= \left\{ \begin{bmatrix} x_{1} \\ x_{2} \\ \vdots \\ x_{n} \end{bmatrix} \mid x_{i} \in \mathbb{F}_{2} \right\} \\
\text{Eq. } n &= 2 \\
\mathbb{F}_{2}^{n} &= \left\{ \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \right\} \\
\text{In general } \left| \mathbb{F}_{2}^{n} \right| &= 2^{n}.
\end{align*}
\]
**Definition** The Hamming weight $W_H(x)$ of a vector $x \in F_2^n$ is the number of non-zero components in $x$.

*Example* $n = 3$, $x = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$, $W_H(x) = 2$. 

*Hamming Weight*
Properties

(i) \( \omega_H(x) \geq 0 \) with equality holding iff (if and only if) \( x = 0 \)

(ii) \( \omega_H(x + y) \leq \omega_H(x) + \omega_H(y) \)

Eq. \( x = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad y = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad z = x + y \quad n = 5 \quad H = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \)

\( \omega_H(z) = 4 \leq \omega_H(x) + \omega_H(y) \)

\( = 4 + 2 = 6. \)
Define

\[ x \odot y = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \odot \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 y_1 \\ x_2 y_2 \\ \vdots \\ x_n y_n \end{bmatrix} \]

(Schur

or componentwise

product)

\[ w_H (x + y) = w_H (x) + w_H (y) \]

\[ \begin{pmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{pmatrix} \quad \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \]

\[ x \odot \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = -2 w_H (x \odot \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}) \]

\[ w_H \left( x + \frac{1}{3} \right) = 4 + 2 - 2 = 4 \]
**Hamming Distance**

**Definition** The Hamming distance $d_H(x, y)$ between two vectors $x, y \in \{0, 1\}^n$ is defined by:

$$d_H(x, y) = \sum_{i=1}^{n} (x_i \neq y_i)$$

**Example**

- $x = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$
- $y = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$
- $x + y = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

$$d_H(x, y) = \sum_{i=1}^{3} (x_i \neq y_i) = 4$$
Properties

(i) \( d_H (x, \Xi) \geq 0 \) with equality holding if \( x = \Xi \)

(ii) \( d_H (x, \Xi) = d_H (\Xi, \Xi) \)

(iii) Triangle Inequality

\[
\forall x, \Xi \in \mathcal{X} \quad \forall \eta \in \mathcal{X} \quad d_H (x, \Xi) \leq d_H (x, \eta) + d_H (\eta, \Xi)
\]
Binary (Block) codes

\[ \text{Defn. A binary block code of length } n \text{ is simply any subset of } \mathbb{F}_2^n. \]

The elements of the code are called code words.
Parameters of a code $C$:

1. Size of a code $C = |C| = \text{# of code words in the code}$

2. Rate $R$ of $C = \frac{\log |C|}{n}$

Where $n$ is the block length of the code.

(meaning that $C \leq \mathbb{F}^n$)
3. block length \( n \) of the code itself

4. the minimum distance \( d_{\text{min}}(C) \) definition

\[
2 \\
\min \left\{ d_H(x, y) \mid x, y \in C, \quad x \neq y \right\}
\]

\[d_{\text{min}}(C) \overset{\Delta}{=} \min \left\{ d_H(x, y) \mid x, y \in C, \quad x \neq y \right\}\]
REFERENCES – ERROR-CORRECTING CODES

P. VIJAY KUMAR


**Book Chapter**


Lecture 2: Example codes and their parameters.

Example: The repetition code. (All of our example codes will have block length $n = 7$)

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

Parameters:

(i) $\text{size} = 2$

(ii) $\text{rate} = \frac{\log_2 |R|}{n} = \frac{1}{7}$

(iii) $d_{\text{min}}(C) = 7$
Eq 2. Single Parity Check Code (SPC Code) 

\[
C \equiv \left\{ \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix} \mid \sum_{i=1}^{k} x_i = 0 \right\} 
\]

\[
\text{Code Parameters}
\]

(i) \(\text{size} = 2\) 
(ii) \(\text{rate} = \frac{6}{7}\)

(iii) \(d_{\min} (C) = 2\)

\[
\text{Eg} \ d \begin{bmatrix} 0 \\ \vdots \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ \vdots \\ 0 \end{bmatrix}
\]
Fig 3. The Hanning code of block length $n = 7$.

\begin{align*}
\phi_4 &+ m_0 + m_1 + m_2 + \phi_6 = 0 \\
m_0 + m_1 + m_2 + m_3 + \phi_5 = 0 \\
m_0 + m_1 + m_3 + \phi_6 = 0
\end{align*}

(iii) $d_{\text{min}}(C) = ?$

can show that

code parameters:

(i) code size $= 2^4 = 16$

(ii) rate $= \frac{4}{7}$ (Exercise)
Definition  A \((t_c, t_d)\) code is a code in which any combination of \(\leq t_c\) errors can be detected and corrected and any combination of \(t\) errors, \(t_c < t \leq t_d\) can be detected as an uncorrectable error.

(Note: in the pair, we will always assume that \(t_d > t_c\).)

Theorem 1 A binary block code \(C\) is a \((t_c, t_d)\) code if and only if:
\[ d_{\min}(C) \geq t_c + t_d + 1 \]

**Proof (if part)** assume that \( d_{\min} \geq t_c + t_d + 1 \).

Adopt the following decoding algorithm:

Let \( j \) be the received vector.

Let for any vector \( a \in \mathbb{F}_2^n \) define

\[ B(a, \varepsilon) = \left\{ \xi \in \mathbb{F}_2^n \left| d_{\mathbb{F}_2}(a, \xi) \leq \varepsilon \right. \right\} \]
If $B(y, t_c)$ contains a codeword $x$, then we will declare $x$ to be the transmitted codeword.

If not, we will declare that an uncorrectable number of errors have occurred.

Note: It is not possible for $B(y, t_c)$ to contain more than one codeword.

If $d_H(y, x_i) \leq t_c$
and \( d_{H} (\bar{y}, \bar{x}_2) \leq t_c \)

\( \implies (by \ the \ \Delta \ inequality) \) that

\[
d_{H} (\bar{x}_1, \bar{x}_2) \leq 2t_c \leq t_c + t_d
\]

\[
< t_c + t_d + 1
\]

(contradiction)

\[
d_{H} (x, \bar{x}) \leq t_c + t_d + 1 < t_c + t_d + 1
\]

\[
= d_{\text{min}} (\mathcal{L})
\]
suppose there is no code word to be found in $B(s, t_c)$.

We (the decoder) will then declare an uncorrectable error.

Clearly the decoder will be correct since the only way it could possibly go wrong is if there was a correctable error, i.e., a codeword within Hamming distance $t_c$ of $s$ but this is impossible by our initial assumption that the ball $B(s, t_c)$ was empty.
If (if the only if part)

To show: that $\ell_{\min} \geq t_c + t_d + 1$ is necessary.

Suppose not $\Rightarrow \ell_{\min} < t_c + t_d + 1$

\[
\ell_{\min} = t_c + t_d - \ell
\] (say)

\[\ell \geq 0\]

$\Rightarrow \exists$ a pair $(x_1, x_2)$ in $P$ (there exists)

such that $d_H(x_1, x_2) = t_c + t_d - \ell$.

\[\text{(received)}\]

vector

\[\frac{1}{x_1} \rightarrow \frac{1}{y} \rightarrow \frac{1}{x_2}\]
The situation above presents the receiver with a dilemma that cannot be resolved for the case when \( y \) is the received vector.

Thus when \( d_{\text{min}} < t_c + t_d + 1 \),

a code cannot be a \((t_c, t_d)\) code.  ///

Example a). Repetition code

\[ d_{\text{min}} = 7 \quad (\text{seen earlier}) \]

\[ t_c + t_d + 1 \leq 7 \]

\[ (t_c = 0, t_d = 6) \quad (t_c = 1, t_d = 5) \quad (t_c = 2, t_d = 4) \quad t_c = t_d = 3 \]
Eq 6: The single parity-check code.
\[ d_{\text{min}} = 2 \]
\[ t_c + t_d + 1 \leq d_{\text{min}} \]

Eq (c): Hamming code.
\[ d_{\text{min}} = 3 \]
\[ t_c + t_d + 1 \leq d_{\text{min}} \]

\( t_c = 0, t_d = 1 \)

\( (t_c = 0, t_d = 2) \)

\( (t_c = 1, t_d = 1) \)
A group \((G, \cdot)\) is a set \(G\) along with an operation \(\cdot\) under which:

(i) \(a, b \in G, \ a \cdot b \in G\) Closure

(ii) \(a, b, c \in G, \ a \cdot (b \cdot c) = (a \cdot b) \cdot c\) Associative

(iii) there exists an element \(e\) in \(G\) s.t. (such that) \(a \cdot e = e \cdot a = a\) for all (\(\forall\)) \(a \in G\) Identity Element

(iv) for every \(a\) in \(G\), there exists an element (called \(a^{-1}\)) s.t. \(a \cdot a^{-1} = a^{-1} \cdot a = e\) Inverse

\[ a \circ a^{-1} = \circ^{-1} \cdot a = e \]

pronounced "a inverse"
Furthermore in the case of Abelian groups, we also have that

\[(v) \quad a \cdot b = b \cdot a \quad \forall a, b \in G\]  

**Commutative Property**

**Note:** Abelian groups are also called commutative groups.

\[\begin{align*}
|\mathbb{F}_2| &= n \quad \text{(modulo 2 addition)} \\
\text{Eq.: } &\left(\mathbb{F}_2^+\right) \quad \text{(component-wise)} \\
&n = 3 \quad \text{and} \quad G = \left\{1, i, \bar{i}\right\} \\
&\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}
\end{align*}\]

Can verify that the axioms are satisfied:

0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} is the identity element.
\[
a = \begin{bmatrix}
0 \\
1
\end{bmatrix} \quad a^{-1} = ? \quad a^i = \begin{bmatrix}
0 \\
1
\end{bmatrix} = a \quad !!
\]

This is an example of an Abelian group.

\[\text{Consider } (\mathbb{Z}_n, +) \text{ modulo } n \text{ addition.}\]

\[\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}\]

\[\text{Identity } = 0 \quad a^{-1} = (n-a)\]

Abelian group.
Derived Properties

(i) the identity element is unique

- Suppose not and suppose that $e_1$ and $e_2$ were both identity elements

\[ e_1 + e_2 = e_1 \]

\[ e_2 = e_1 = e_2 \]

(ii) every element has a unique inverse.

Suppose both $c$ and $b$ are inverses of $a$:

\[ c a b = c (a b) = c (c b) = c b = b \]

\[ \therefore b = c \]
(iii) \((ab)^{-1} = b^{-1}a^{-1}\) (Exercise)

(iv) \((a^{-1})^{-1} = a\) (Exercise)

(v) Cancellation law holds:

i.e., if \(ca = cb\) \(\Rightarrow\) \(a = b\)

\[
\text{if } c' (ca) = c' (cb) \\
\Rightarrow (c'c)a = (c'c)b \\
\Rightarrow ca = cb \\
\Rightarrow a = b
\]

(vi) \(a^m = \underbrace{a \cdot a \cdot a \ldots a}_{m \text{ times}}\)
Lec 4: Subgroups and Equivalence Relations

(a further example of a group)

$\mathbb{Q} \rangle \times (\mathbb{Z}/n\mathbb{Z})$

nonzero elements in the set of integers modulo $n$

$\mathbb{Z}_n^* = \{1, 2, 3, \ldots, n-1\} = \mathbb{Z}_n \setminus \{0\}$
$\mathbb{Z}_6 \times (\mathbb{Z}_6, \cdot) \subset \mathbb{Z}_6^{\times} = \{1, 2, 3, 4, 5\}$

1. CLOSURE
2. ASSOCIATIVE
3. IDENTITY ELEMENT
4. INVERSE
5. COMMUTATIVE (for Abelian groups only)

$2 \cdot 3 = 0 \ (\text{mod} \ 6)$ violates closure
$\therefore$ not a group!
Eq. \( \left( \mathbb{Z}_p^*, \cdot \right) \) \( p = \text{prime} \)

\begin{align*}
\text{CL} & \checkmark \quad \text{Assoc} & \checkmark \quad \text{El.} & \checkmark \quad \text{Inverse} & \checkmark \\
\Rightarrow & \quad a \cdot b = 0 \quad \text{(mod } p) \quad \Rightarrow & a \cdot b = \text{Rcm} \left\{ \frac{a \cdot b}{\phi} \right\} \\
\text{Commut.} & \checkmark & \quad \Rightarrow & \quad \phi \mid a \cdot b & \quad \phi \mid a \quad \text{on} \\
\phi \mid b \\
\Rightarrow & \quad a = 0 \quad \text{mod } \phi \quad \text{on} \quad \text{cyc} \\
& \quad b = 0 \quad \text{mod } p
\end{align*}
do inverses exist in $\mathbb{Z}_0^*$?

$\phi = 5$

$\left( \frac{1}{2} \right)^{-1} = ?$

$2^\phi = \{1, 2, 3, 4, 5\}$

\[
\begin{cases}
2 \cdot 1 &= 2 \\
2 \cdot 2 &= 4 \\
2 \cdot 3 &= 6 = 1 \pmod{5} \\
2 \cdot 4 &= 8 = 3 \pmod{5}
\end{cases}
\]

$\Rightarrow 3 = 2^{-1}$ !!!

note: all entries on the RHS above are forced to be distinct since
by cancellation:

$$2a = 2b \implies a = b$$

(else \(2(a-b) = 0\) impossible)

A similar proof can be used to show that every element \(\mathbb{Z}_p^\times\) has an inverse and hence \((\mathbb{Z}_p^\times, \cdot)\) is a group under multiplication.
**Subgroups**

**Defn** A subgroup \((H, \cdot)\) of a group \((G, \cdot)\) is a subset \(H \subseteq G\) such that

\((*) (H, \cdot)\) is a group by itself.

**Eq 1** \(H = G\)   clean

**Eq 2** \(H = \{e\}\)

easy to check that \((\{e\}, \cdot)\) is a subgroup.
(called trivial examples).

Testing for a subgroup

**Brute force:**

- **CL**
- **ASSOC**
- **I.E**
- **INVERSE**

**Better:**

Instead of saying $(H, \cdot)$ is a subgroup

$f((g, \cdot))$ we will simply say that $H$ is a subgroup of $G$. 
Lemma: \( H \subseteq G \) is a subgroup if\
\[ a, b \in H \implies a \cdot b^{-1} \in H \]

**Proof:**

CL

Assoc

I.E.

Inv

Setting \( a = b \) \(\implies a \cdot b^{-1} = e \) => \( e \in H \)

Setting \( a = e \) \(\implies b^{-1} \in H \)

**Closure:** Given \( a, b \in H \), we know that \( b^{-1} \in H \)

\[ = a (b^{-1})^{-1} \in H \]
there is a further simplification in the case of groups containing a finite number of elements:

**Lemma** If \( H \) is a finite subset of \( G \), then \( H \) is a subgroup of \( G \) iff:

\[ a, b \in H \implies a \cdot b \in H \]
Proof (left as an exercise!)

Hint: given $a \in \mathbb{N}$, consider $a, a^2, a^3, \ldots$.

This is an infinite sequence, yet $\mathbb{N}$ is finite. Use this!
**Equivalence Relation**

**Defn.** A relation $R$ on a set $A$ is a subset of $A \times A$ (Cartesian product of $A$ with itself), i.e., $R \subseteq A \times A$.

If $(a, b) \in R$ we will write $a \sim b$.

**Notation:**

$$E_b = \{ a \in A \mid (a, b) \in R \}.$$
(E_b is the set of all elements in X that are related to b via R)

**Definition:** A relation R is said to be an equivalence relation on X provided:

(i) \(a \sim a\) REFLEXIVE

(ii) \(a \sim b \Rightarrow b \sim a\) SYMMETRY

(iii) \(a \sim b, b \sim c \Rightarrow a \sim c\) TRANSITIVE
Claim: If $R$ is an equivalence relation, then if $a, b \in X$, then either $E_a = E_b$ or else $E_a \cap E_b = \emptyset$ (the empty set).
(we will sketch the proof using a figure:)

\[ E_a \quad \cap \quad E_b \]

Suppose \( j \in E_a \setminus E_b \)

Suppose \( x \in E_a \cup E_b \)
\[ \exists y \sim x, \ x \sim b \Rightarrow y \sim b \]
\[ \Rightarrow \ j \in E \]
\[ \text{a contradiction!} \]

(sets of a subgroup)

Let \((a, \cdot)\) be a group and \((H, \cdot)\) be a subgroup of \(G\).

Let us define \( a \cdot b = a \cdot b^{-1} \cdot e_H \)
\[ R = \{ (a, b) \in G \times G \mid a \cdot b^{-1} \in H \} \]

**Claim:** This is an equivalence relation.

- **Reflexive**
  \[ a a^{-1} = e \in H \]

- **Symmetric**
  \[ b a^{-1} = c, c \in H \]
  \( \text{(since } H \text{ is a subgroup) } \]
  \[ \therefore a a^{-1} = b a^{-1} \]

- **Transitive**

**Symmetry:**
\[ a b = b a \]
\[ \Rightarrow a b^{-1} \in H = b a^{-1} \in H \]
but since \((a b^{-1})^{-1} = ba^{-1}\), and

\(H\) is a subgroup \(\Rightarrow ba^{-1} \in H\)

i. Yes!

\[
\text{Trans. } a b, \quad b a c \Rightarrow a a c ?
\]

\[
\begin{align*}
\downarrow & \quad \downarrow \\
abla b^{-1} & \in H \quad \quad \quad \quad \quad \quad bc^{-1} \in H
\end{align*}
\]

\[\therefore (a b^{-1})(b c^{-1}) \in H\]

\[= a \overline{b^{-1}b} c^{-1} \in H\]

\[= ace c^{-1} \in H\]

\[= ac^{-1} \in H \Rightarrow a a c\]
We saw last time that an equivalence relation partitions the set.
\((G, \cdot)\) group

\((H, \cdot)\) subgroup

\(a, b \in G\) and if \(ab^{-1} \in H\)

Saw last time that this is an equivalence relation. \(G\)
\[ a b^{-1} \in H \implies a b^{-1} = h \in H \]
\[ \implies a = h b, \ h \in H \]
\[ \therefore a \sim b \iff a \in H b \quad \text{where} \]
\[ H b = \{ h \cdot b \mid h \in H \} \]
\[ \therefore E_b = H b \]
Subsets of \( G \) of this form are called core sets of \( H \) in \( G \).
Eq

\((G, \cdot) = \left( \mathbb{Z}_6, + \right)\)

\((H, \cdot) = \left( \left\{ 0, 2, 4 \right\}, + \right)\)

(check that this is a subgroup - Exercise!)

\(a \sim b \iff ab^{-1} \in H\)

the equivalence classes are all of the form:

\(H \cdot b \iff H + b\)

\(H = \left\{ 0, 2, 4 \right\}\)
\[ b = 0 \implies H + b = H \text{ itself} \]
\[ b = 1 \implies H + 1 \]
\[ 2 \left\{ h + 1 \mid h \in H \right\} \]
\[ = \left\{ 1, 3, 5 \right\} \]

\[ \text{Coset: } \]
\[ H + 0 \]
\[ H + 1 \]

\[ \left\{ 0, 2, 4 \right\} \]
\[ \left\{ 1, 3, 5 \right\} \]
Eq 2 (of partitioning into equivalence classes)

\[(G, \cdot) = (\mathbb{F}_2, +)\]

\[(H, \cdot) = (\mathbb{C}, +)\]

where \(\mathbb{C}\) is the even parity code \(\text{spc code}\).

\[
\begin{array}{c}
\text{even parity} \\
\text{coset class}
\end{array}
\]

\[
\begin{array}{c}
\text{odd parity} \\
\text{coset class}
\end{array}
\]
verify that this is the partitioning that results.

Claim: Let \((G, \cdot)\) be a group and \((H, \cdot)\) be a subgroup. Let \(a, b, c \in H\) such that \(ab^{-1} \in H\). Then there is a 1-1 correspondence between the elements of different equivalence classes \(\{\text{cosets of } H \text{ in } G\} \cap a \cdot H\) (i.e., there is a 1-1 correspondence between the \(\{\text{cosets of } H \text{ in } G\} \cap a \cdot H\)).
in particular, when \( H \) is a finite subgroup, any two equivalence classes are of the same size.

**Proof** we know that all equivalence classes are of the form \( H + b \), \( b \in G \).

Let \( H + a \), \( H + b \) be two distinct equivalence classes.
we define $\phi$ via:

$$\phi (h + a) = h + b \quad \forall h \in H$$

Clearly $\phi$ is onto.

To show 1-1, assume to the contrary that
\[ \phi(h_1 + a) = \phi(h_2 + a) \]

i.e., \( h_1 + b = h_2 + b \)

\[ \Rightarrow \text{by cancellation} \]

that \( h_1 = h_2 \)

\[ \therefore h_1 + a = h_2 + a \]

\[ \therefore \text{the map is 1-1.} \]
Rings

Definition: A ring \((R, +, \cdot)\) is a set \(R\) together with two operations \(+\) and \(\cdot\) (addition and multiplication respectively) satisfying:

(i) \((R, +)\) is an Abelian group under addition
(ii) \( a, b \in \mathbb{R} \Rightarrow a \cdot b \in \mathbb{R} \)  
(CLOSURE UNDER MULTIPLICATION)

(iii) \( a \cdot (b + c) = a \cdot b + a \cdot c \)  
\((a + b) \cdot c = a \cdot c + b \cdot c\)  
(DISTRIBUTIVE LAWS)

(iv) \( a \cdot (b \cdot c) = (a \cdot b) \cdot c \)  
ASSOCIATIVITY UNDER MULTIPLY.
Ring

with identity

Division ring

Ring

Commutative
Ring

Integral domain

Field

IR, ℂ, ℍ, ℂ₂
Definitions

- A **commutative ring** is a ring in which multiplication is commutative, i.e.,
  \[ a \cdot b = b \cdot a \quad \text{for all} \quad a, b \in \mathbb{R} \]

- A **ring with identity** is a ring with multiplicative identity (which we will call 1), i.e.,
  (the identity under addition is written)
  \[ a + 0 = a = 0 + a \]
An integral domain is a commutative ring that has no zero divisors, i.e.,
\[ a \cdot b = 0 \text{ for } a, b \in R \]
if and only if \( a = 0 \) or \( b = 0 \).

A division ring is a ring with identity in which every nonzero element in \( R \) has
a multiplicative inverse, i.e.,
\[ a \in R, \ a \neq 0 \]
\[ \Rightarrow \exists a^{-1} \in R \text{ s.t.} \]
\[ a \cdot a^{-1} = a^{-1} \cdot a = 1. \]
(note that additive inverse of a is written as \(-a\))

- A field is a commutative division ring, i.e.,
  - it is a ring with 1
  - non zero elements have inverses

\[ \Rightarrow \] in a field \(F\),

\((F, +)\) is an Abelian group
\((F^*, \cdot)\) is also an Abelian group
Examples of rings

- the real numbers \( \mathbb{R} \)
- the complex numbers \( \mathbb{C} \)
- the finite field \( \{0, 1\} \)

\[
\begin{align*}
0 + 1 &= 1 \\
0 + 0 &= 0 \\
1 + 0 &= 1 \\
\text{etc}
\end{align*}
\]
Summary of Lecture 5:

- Equivalence classes defined via cosets:
  - Proof that it is an equivalence relation
  - The nature of the equivalence class $E_b = Hb$
  - Examples:
    - Integers modulo 6 and even subset
    - Even parity check code
  - Elements in different cosets can be placed in 1-1 correspondence

- Rings and Fields
  - Axioms of a ring
  - Ring with identity
  - Commutative ring
  - Integral domain
  - Division ring
  - Examples: where do we place them?
Rings and Fields
  - Examples: where do we place them?

Vector Spaces
  - Axioms
  - Examples
  - Derived properties

Subspaces
  - Definition
  - Example 1: plane in $\mathbb{R}^3$
  - Test for a subspace
  - Further examples: repetition code and spc code

Definition of a linear code
  - Show how the test applies to the Hamming code (nullspace of a matrix)
  - Point out that as far as subsets of $\mathbb{F}_2^n$ are concerned,
\[ \mathbb{F}_2 = \{0, 1\} \text{ field } (\mathbb{F}_2, +, \cdot) \]

can verify that the set \( \{0, 1\} \) along with the operations defined as above forms a field.
Further examples of rings

1) \( \mathbb{R}, \mathbb{C}, \mathbb{F}_2 \) fields

2) \( \mathbb{Z} \) - the set of all integers \{ integral domain \}

\( (\mathbb{Z}, +, \cdot) \)

3) \( 2\mathbb{Z} = \{ 2z \mid z \in \mathbb{Z} \} \) even integers

\( m \times n \)

4) \( R^{m \times n} = \{ \text{the set of all} \ (m \times n) \text{ matrices} \} \text{ over the reals} \)

\( (R^{m \times n}, +, \cdot) \)
5) \( \mathbb{F}[x] \rightarrow \) the set of all polynomials in the indeterminate \( x \) over \( \mathbb{F} \)

\[
\mathbb{F}[x] = \left\{ \sum_{k=0}^{d} a_k x^k \mid a_k \in \mathbb{F}, \ d \geq 0, \ \text{is an integer} \right\}
\]

(degrees)

(\( \mathbb{F}[x], +, \cdot \))

6) \( (\mathbb{Z}/6, +, \cdot) \) addition and multiplication are carried out modulo 6
$2 \cdot 3 = 0 \therefore$ not an integral domain
Vector Spaces

**Defn** A vector space \((V, +, F, \cdot)\) is a set \(V\) of vectors, a field \(F\) of scalars, and two operations:

- \(\cdot \Rightarrow\) vector addition
- \(\cdot =\) scalar multiplication

s.t.

1. \((V, +)\) is an Abelian group
(ii) \( c \cdot v \in V \), \( c \in F \)  \( \forall v \in V \)  \( \text{Closure under scalar multiplication} \)

(iii) \( c_1(c_2 \cdot v) = (c_1c_2) \cdot v \)  \( \text{Associativity} \)

(iv) \( (c_1 + c_2) \cdot v = c_1 \cdot v + c_2 \cdot v \)  \( \text{Distributive Laws} \)

(v) \( 1 \cdot v = v \)  \( \text{Multiplicative identity in } F \)

Examples
(i) \( \mathbb{R}^n = (\mathbb{R}^n, +, \mathbb{R}, \cdot) \) 

\[ \mathbf{v} = \begin{bmatrix} 1 \\ 3 \\ 2 \end{bmatrix} \]

Eq \( n = 3 \) 

(ii) \( \mathbb{H}^n = (\mathbb{H}^n, +, \mathbb{H}, \cdot) \) 

(iii) \( \mathbb{M}_{m \times n} = (\mathbb{M}_{m \times n}, +, \mathbb{R}, \cdot) \) 

(iv) \( \mathbb{R}^n = (\mathbb{R}^n, +, \mathbb{R}, \cdot) \)
Derived Properties

(i) $0 \cdot v = 0$

\[ \downarrow \]

the additive identity in the group $(\mathbb{V}, +)$

\[ \uparrow \]

the additive identity in the field $\mathbb{F}$

\[ \uparrow \]

\[ \text{pf} \]

\[ (1 + 0) \cdot v = 1 \cdot v + 0 \cdot v = v + 0 \cdot v \]

\[ \text{adding } -v \text{ to both } v, v + 0 \cdot v \]

we see that \[ 0 \cdot v = 0 \]

(ii) $c(0) = 0, c \in \mathbb{F}$

\[ \text{pf} \]

\[ c \left( 0 + v \right) = c \circ 0 + c \circ v \]
\( (A) \)

\[ \because \ 0 \leq c \leq 2 \]

\[ \therefore \ (A) \leq c \leq 2 \]

\[ \because \ (A) + c = 0 \to c = 0 \]

\[ \because \ (A) + (c) = 0 \to c = 0 \]

\[ \because \ (A) = 0 \]

\[ \because \ (A) = 0 \]

\[ \because \ (A) = 0 \]

\[ \therefore \ 0 = c = 0 \]

\[ \therefore \ 0 = c = 0 \]

\( (iii) \)

\[ c = 0 \quad \text{iff} \quad \text{either} \ c = 0 \quad \text{or} \quad v = 0 \]

\[ \text{If} \ c = 0 \quad \text{then} \quad \text{done} \]

\[ \text{If} \ c \neq 0 \quad \text{consider} \]

\[ c' = (c \cup) = c' \quad (0) = 0 = 0 \]
\[(\xi, \eta, \zeta) \mapsto \eta\]
\[\xi \mapsto \eta\]
\[\eta \mapsto \eta\]
\[
(iiv) \quad (-1) \eta = -\eta
\]
\[
\text{Proof left as an exercise!}
\]
Subspaces

**Definition** A subspace of a vector space \((V, +, \cdot, F)\) is a subset \(W\) of \(V\) such that \((W, +, \cdot, F)\) is also a vector space.

**Example** In \(\mathbb{R}^3\), possible subspaces are:

- \(\mathbb{R}^3\)
- \(\{0\} \subset \mathbb{R}^3\)
- Any line through \(\{0\}\)
- Any plane through \(\{0\}\)
Test for the presence of a subspace?

\((V, +, \cdot)\) \(W \subseteq V\)

\((W, +, \cdot) \Rightarrow \text{is this a subspace?}\)

To test whether or not "\(W\) is a subspace of \(V\)" it is sufficient to check that:

\[ x + c \cdot y \in W \]

\(\forall x, y \in W \), \(c \in \mathbb{F}\)

how does this
lec 7: Linear Codes, linear independence

Summary

- Example rings and their classification

- Vector spaces
  - 5 examples
  - Derived properties

- Subspaces
  - Test for a subspace
Test for the presence of a subspace?

\((V, +, \cdot, \cdot) \quad W \leq V\)

\((W, +, \cdot) \Rightarrow \text{is this a subspace?}\)

To test whether or not \(W\) is a subspace of \(V\) it is sufficient to check that:

\[\frac{x + c}{y} \in W\]

and \(c \in F\)
\[ x + c \cdot y \in W \]

setting \( c = 1 \) \implies \{ \text{closure under} \] vector addition

setting \( y = x \) and \( c = -1 \)

\[ x + (-1) \cdot x \in W \] identity
\[ \implies \] \( 0 \in W \) element

setting \( x = 0 \), \( c = -1 \)
ensures that \( (-1) \cdot y \in W \)
\( \cup \) inverse is present
\[ \Rightarrow -y \in W \]
\[ x = 0 \Rightarrow y \in W \]

The remaining 5 axioms follow simply from the fact that 
\[ W \subseteq V \].
\[ E^q (V, +, F) \cdot \) = \left( \frac{F}{2} \right) + \left( \frac{F}{2} \right) \]

\[ (\omega, +, 1F, \cdot) = \left( \frac{\mathbb{C}}{2} \right) + \left( \frac{\mathbb{C}}{2} \right) \]

even parity or

special code.

Note that \( x \in \mathbb{C} \) if

\[
\begin{align*}
\frac{1}{t} x &= 0 \\
\frac{1}{t} x &\geq 0 \\
\frac{1}{t} &\geq 1 \\
\frac{1}{t} &= 1
\end{align*}
\]

If \( x, y \in \mathbb{C} \)

\[
\implies \begin{cases} 
\frac{1}{t} x = 0 \\
\frac{1}{t} y = 0 \\
\frac{1}{t} c \in \mathbb{C}
\end{cases} \implies \frac{1}{t} (x + c y)
\]
\[ \begin{pmatrix} t \\ x \end{pmatrix} + c \begin{pmatrix} t \\ 0 \end{pmatrix} = 0 + c \cdot 0 = 0 \]

Thus \( R \) is a subspace of \( \mathbb{F}_2^2 \).
Note: that for the particular case when \( F = F_2 \) the only non-zero scalar is 1.

\[ x + cj \in W \]

the test reduces to

\[ x + j \in W \]
It follows from this that there is no distinction between

subspaces \( (\mathbb{R}, +, \mathbb{F}_2, \cdot) \) of

\[ \bigwedge^n (\mathbb{F}_2, +, \mathbb{F}_2, \cdot) \]

and of subgroups

\( (\mathbb{R}, +) \) of

\[ \bigwedge^n (\mathbb{F}_2, +) \]
Define a linear code of block length \( n \) is any subspace of \( \mathbb{F}_2^n \) (of \( (\mathbb{F}_2, +, \cdot) \))

It follows from this that every such linear code is also a subgroup. For this reason,
linear codes are also called group codes.

Eq 1: We have already seen that the SPC code is a linear code.

Eq 2: the repetition code:

\[ R = \begin{bmatrix} 0 & \cdots & 0 \end{bmatrix}, \begin{bmatrix} \vdots \end{bmatrix} \]
Clearly this is a linear code.

Eg 3  Hamming code

\[ m_0 + m_1 + m_2 + p_4 = 0 \]
Thus the Hamming code is precisely the set of all code words \( C \) satisfying \( H C = 0 \).
\[ x \in \mathbb{F} \implies Hx = 0 \]
\[ j \in \mathbb{R} \implies Hj = 0 \]

\[ \therefore H \left( x + cj \right) = Hx + cHj = 0 \]

\[ \therefore \text{the Hamming code is a linear code.} \]
Note: Given a matrix $H$, the collection of all vectors $x$ such that $Hx = 0$ is called the nullspace of $H$:

$$
\text{null space of } H = \left\{ x \mid Hx = 0 \right\}.
$$
It follows from this that the null space of any binary $m \times n$ matrix is a linear code.
**LINEAR INDEPENDENCE**

**Def.** The vectors \( \{ \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n, \ldots \} \) are said to be linearly independent if

\[
\sum_{j=1}^{n} \lambda_i \mathbf{x}_j = 0
\]

is possible iff

\[
\lambda_1 = \lambda_2 = \cdots = \lambda_n = 0.
\]
Example: In $\mathbb{R}^3$, any collection of 3 vectors in $\mathbb{R}^3$ which do not lie on a plane containing the origin are linearly independent.
Example 2

\[ A = \begin{bmatrix}
1 & 0 & 2 & 1 & 3 \\
0 & 0 & 6 & 1 & 7 \\
0 & 0 & 0 & 0 & 5 \\
0 & 0 & 0 & 0 & 0
\end{bmatrix} \]

Row reduced echelon matrix

- The nonzero rows are linearly independent
- The columns containing the pivots are linearly independent.
Example vector space:

\[ \left( \mathcal{F}[x], +, \cdot \right) \]

- \[ d_1 = x + a \quad a \in \mathcal{F} \]
- \[ d_2 = x^2 + b x + c \quad b, c \in \mathcal{F} \]
- \[ d_3 = x^5 \]

Can be verified that these 3 polynomials are linearly independent.
Spanning

**Defn** A set \( \{ x_1, x_2, \ldots, x_n \} \) is said to span a vector space \( V \) if every vector \( \mathbf{z} \in V \) can be expressed in the form:

\[
\mathbf{z} = \sum_{j=1}^{n} c_j x_j \quad (c_j \in \mathbb{F})
\]
\[ x \geq 1 \text{ is an integer} \]

\[ z = \sum_{j=1}^{2} c_j x_j \]

\[ \text{a linear combination of} \quad \{ x_1, x_2 \} \]
Lec 8: Spanning & Basis

Recall
- test for a subspace
- linear code definition
  - examples
- linear independence
  - examples
- spanning
**Defn.** The space spanned by a set \( \{ x_1, x_2, \ldots, x_n \} \) is the set of all (finite) linear combinations of the form:

\[
\sum_{j=1}^{2} \lambda_j x_j \quad \lambda_j \in \mathbb{F}
\]

where \( \mathbb{F} \) is an underlying field of the vector space.
(the word space is used since this collection actually forms a vector space)

**notation:**

\[ W = \langle \xi_1, \xi_2, \xi_3, \ldots \rangle \]

space spanned by the \( \xi \)s.
Qn: What is the space spanned by

\[ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \]

Setting: \((F_2^7, F_2, F_2)\)
Ans \{ the above collection of 6 vectors form a basis for the single parity check code. \}
Basis

Defn. A basis for a vector space \( (V, +, F, \cdot) \) is a collection \( \{v_1, v_2, \ldots, v_n\} \) of vectors such that:

1. The set is linearly independent
2. The set spans \( V \)
(ii) The set spans the vector space 

Example: \((\mathbb{R}^3, +, \cdot)\) vector space

Basis: \(\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}\) called the standard basis for \(\mathbb{R}^3\)
more generally, any collection of 3 vectors in $\mathbb{R}^3$ which do not lie on a plane through the origin is a basis.

Note: As can be seen from the example, a given vector space can have multiple bases.
Example vector space

\((F[x], +, F, \cdot)\)

Basis \(\Rightarrow \{1, x, x^2, x^3, \ldots\}\)

**Defn** A *finite-dimensional vector space* is any vector space possessing a basis consisting of a finite \(n\) elements.
Thm Let \((V, +, F, \cdot)\) be a finite-dimensional vector space. \((\text{f.d.})\)

Then any two bases for \(V\) must contain the same number of elements.

Pf The proof will make use of the following 2 lemmas:
Lemma 1: If a vector space $V$ has a basis consisting of $m$ elements, then any collection of $n > m$ elements is a linearly dependent set.
Lemma 2: If a vector space $V$ has a basis consisting of $n$ elements, then any collection of $m < n$ elements cannot span the space.
from these two lemmas, it follows that a basis is simultaneously

a) a maximal linearly independent set

b) a minimal spanning set.

H (of theorem). Let

\[ \{x_1, x_2, \ldots, x_m\} \quad \text{AND} \quad \{y_1, y_2, \ldots, y_n\} \]

be two bases for the vector space V.
Since \( \{ e_i \} \) from a basis and

\[
\text{the } \{ f_j \} \text{ are a linearly independent set, it follows that } n \leq m.
\]
Since \( \{ \alpha_i \}_{i=1}^m \) form a basis and
\[
\sum_{j=1}^{n} \beta_j \sum_{i=1}^{m} \alpha_i \]
span the vector space \( V \), it follows that \( n \geq m \).

\[ \therefore n = m \]
**Definition**

The dimension $k$ of a f.d. vector space $\langle V, +, F, \cdot \rangle$ is simply the number of elements in any basis for $V$. 
**Definition** The dimension of a linear code \( C \) of block length \( n \) is simply its dimension as a subspace of the vector space \((\mathbb{F}_2^n, +, \cdot)\).
The repetition code:

\[
\begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}, \quad \begin{bmatrix}
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1
\end{bmatrix}
\]

basis \Rightarrow \begin{bmatrix}
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1
\end{bmatrix}

\therefore \text{the dimension} = 1.
Notation

An $[n, k, d]$ code signifies a block code of length $n$, dimension $k$ and minimum distance $d$.

$[n, k]$ code signifies a block code of length $n$, dimension $k$. 
An \((n, M, d)\) code signifies a block code of length \(n\), size \(M\), and minimum distance \(d\).

\((n, M)\) code signifies a block code of length \(n\), size \(M\).
Eg. the single parity check code:

\[ \begin{bmatrix} 7, 6, 2 \end{bmatrix} \text{ code} \]

basis:

\[
\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right\}
\]
Lec 9: The Dual Code

Recap:
- spanning
- basis
- examples
- any two bases for a given vector space contain the same # of elements
- Dimension
- A vector space
- Of a linear code
- Examples

Note: A given vector space can have more than one basis.

E.g. \( \mathbb{R}^3 \) - The standard basis
\[
\begin{align*}
\{ & \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \}
\end{align*}
\]
also, any three vectors not on a plane also form a basis

(b) Given a basis \( \{ \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n, \ldots \} \) for a vector space \( V \), every vector has a unique expansion as a linear combination of elements of the basis.
Suppose

\[ X = \sum_{j=1}^{n} c_{ji} \cdot x_{ij} = \sum_{k=1}^{s} a_{ik} \cdot \lambda_{ik} \]

\[ \Rightarrow \sum_{j=1}^{n} c_{ij} \cdot x_{ij} - \sum_{k=1}^{s} a_{ik} \cdot \lambda_{ik} = 0 \]

But by the linear independence

of the \( \{x_{ij}\} \) it follows that

this can happen iff
\( n = 5 \), \( \xi = \{ x, y \} \) and \( \xi = \{ x, y, z \} \) are the same and the coefficients are the same.

\[
\begin{align*}
\mathbb{E}_q & \quad \mathbb{R}^3 \\
- & \quad x = \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}
\end{align*}
\]

Standard basis:

\[
\begin{align*}
\vec{x}_1 & = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, & \vec{x}_2 & = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, & \vec{x}_3 & = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\end{align*}
\]
\[ X = 1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + 4 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \]

This coefficient set is unique.

---

Basis for the 3 example codes

\[ \text{Eq } 1 \text{ (the repetition code)} \]
Eq 2  Single parity-check code

\[ \text{basis} = \begin{bmatrix} \vvdots \end{bmatrix} \]

\[ \begin{bmatrix} \vvdots \\ \vvdots \\ \vvdots \end{bmatrix} \]

\[ \begin{bmatrix} \vvdots \\ \vvdots \\ \vvdots \\ \vvdots \\ \vvdots \\ \vvdots \end{bmatrix} \]

the collection of six vectors below:
Eq 3  The Hamming code.
\[
\begin{bmatrix}
m_0 & m_1 & m_2 & m_3 & \phi_4 & \phi_5 & \phi_6 \\
\end{bmatrix}
\]

\[
= \begin{bmatrix}
m_0 & m_1 & m_2 & m_3 \\
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 \\
\end{bmatrix}
\]

The rows of \( \mathbf{G} \) form a basis for the Hamming code.

\( \mathbf{G} \) (this is a generator matrix for the Hamming code).
The Dual Code

Define let $R$ be an $[n,k]$ code. The dual $R^\perp$ of $R$ is defined by:

$$R^\perp = \left\{ y \in \mathbb{F}_2^n \mid \frac{x^t y}{x^t x} = 0 \text{ for all } x \in R \right\}$$
Example: If $C$ is the repetition code

$$C = \left\{ \begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots
\end{array} \right\},$$

clearly, the dual of this code is the spec code.

i.e., $\left( C_{\text{rep}} \right)^\perp = C_{\text{spec}}$
Qn: What is $\mathbb{R}^+_{\text{spec}}$? i.e.,

what is \( \left( \left( (\mathbb{R}_{\leq b})^+ \right)^+ \right)^+ \)?

Ans: \( (\mathbb{R}^+)^+ = \mathbb{R} \) always!
**GENERATOR MATRIX**

**Defn.** Let $R$ be an $[n, k]$ code.

Then any $(k \times n)$ matrix whose rows form a basis for $R$ is called a generator matrix of $R$.

**Note:** A code can in general, have more than one generator matrix.
Eq 1. The Hamming code
(provided earlier)

Eq 2 (x∈X ≤d ≤c)

\[ G = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \]
\[ \begin{bmatrix} 7 & 6 \end{bmatrix} \]

\[ G = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \]

\((6 \times 7)\)

\((k \times n)\)
Lemma  The nullspace of a generator matrix $G$ for the code $R$ is precisely the dual code.

\[ \text{null space of } G \]

Proof: Let $x$ be in the null space of $G$.

\[ \eta(G) = \left\{ x \mid Gx = 0 \right\} \]

null space.
clearly from the definition of the dual code it follows that if $\mathbf{y} \in \mathbb{R}^t$, then $\mathbf{y} \in \eta(\mathbf{u})$.

On the other hand, if $\mathbf{x} \in \eta(\mathbf{u})$, then
\[ x^t \mathbf{g} = 0 \]

If \( x \in \mathbb{R}^n \), \( x = \sum_{i=1}^{m} \mathbf{a}_i \mathbf{g}_i \)

\[
\begin{align*}
x^t \mathbf{c} &= x^t \left( \sum_{i=1}^{m} \mathbf{a}_i \mathbf{g}_i \right) \\
&= \sum_{i=1}^{m} \mathbf{a}_i \left( x^t \mathbf{g}_i \right) \\
&= \sum_{i=1}^{m} \mathbf{a}_i \left( \mathbf{a}_i \mathbf{g}_i \right) \\
&= \sum_{i=1}^{m} \mathbf{a}_i \mathbf{a}_i^t \\
&= 0
\end{align*}
\]

\[ x^t \mathbf{c} = 0 \quad \Rightarrow \quad x \in \mathbb{R}^n \]
**Parity-Check Matrix**

A parity-check matrix for a linear code \( R \) is any generator matrix \( mx \) for the dual code \( R^⊥ \).

**Example**

Let \( R \) be the \( 3 \times 4 \) code. The dual code \( R^⊥ \) is the repetition code having generator matrix \( mx \).
\[ H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix} \]

Note that \( x \in \mathbb{F} \) iff \( \sum_{i=1}^{n} x_i = 0 \)

\[ (\exists) \sum_{i=1}^{n} x_i = 0 \]

this is precisely the parity condition satisfied by code words in \( \mathbb{F} \)

hence the name.
Thm \((\mathbb{R}^+)^+ = \mathbb{R}\).

(Sketch) Let \(H\) be a gen \(m \times n\) \((\mathbb{R}^+)^+\).

\[ \eta(H) = (\mathbb{R}^+)^+ = \mathbb{R} \]
Recap:

(i) Uniqueness of representation
- What is a basis?

(ii) Examples of the 3 codes

(iii) Dual code definition
- Example

(iv) The generator matrix
\[ (v) \quad \eta(c) = \mathbb{R}^1 \]

\[ (vi) \quad \text{the parity-check matrix} \]

---

**Definition:** The rowspace of a \( m \times n \) matrix \( A \) is the set of all linear combinations of the rows of \( A \).
Eq.

\[ G = \begin{bmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 1 
\end{bmatrix} \]

nullspace of \( G \) is the spa code.
Thm (fundamental theorem of linear algebra)

If \( A \) is an \((m \times n)\) matrix,

then

\[
\text{rank}(A) + \dim \left( \text{null space}(A) \right) = n
\]

Also for any given matrix \( A \),

\[
\text{rank}(A) = \dim \left( \text{row space}(A) \right)
\]
Then it follows from ① and ② that if \( A \) is \((m \times n)\), then

\[
\dim(\text{rowspace}(A)) = \eta
\]

\[
+ \dim(\text{null space}(A))
\]
Thm \quad (\mathbb{R}^+)^+ = \mathbb{R}

\textbf{Prf.} \quad \text{Let } H \text{ be a generator matrix for the dual code.}

\text{It follows from an earlier lemma that the null space of } H \text{ is } (\mathbb{R}^+)^+. 
Consider the equation:

\[
\begin{bmatrix}
\frac{1}{h_1} & \frac{1}{h_2} & \cdots & \frac{1}{h_{n-k}} \\
\frac{t}{h_1} & \frac{t}{h_2} & \cdots & \frac{t}{h_{n-k}} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{t}{h_1} & \frac{t}{h_2} & \cdots & \frac{t}{h_{n-k}} \\
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
\vdots \\
x_{n-k} \\
\end{bmatrix} = 0
\]

\[H \in \mathbb{R}^{(n-k) \times n}\]

Note: the nullspace of \( H \)
contains the original code since every row of \( H \) is a codeword in the dual code and hence has zero as the value of the inner product with \( x \), i.e.,

\[
\dim \left( n(H) \right) = (R^{-1})^T
\]

but \( \text{rank}(H) = n-k \)

i.e. \( \dim (n(H)) = n-(n-k) = k \)
but $\dim (R) = k$

It follows that

\[ n(H) = R = (R^\perp)^\perp \]

**Corollary** Every code $R$ is precisely the null space of its parity-check matrix
Goal: finding means to identify a f.c. mx for a given role C.

Lemma 1: If $H$ is an $(n-k \times n)$ matrix of rank $(n-k)$ and

$$C = \eta(H),$$

then $H$ is a valid f.c. mx for $C$. 

\( \begin{bmatrix}
\vdots \\
\hat{r}_1^t \\
\hat{r}_2^t \\
\vdots \\
\hat{r}_{n-k}^t \\
\end{bmatrix}
\begin{bmatrix}
\hat{x}_1 \\
\hat{x}_2 \\
\vdots \\
\hat{x}_{n-k} \\
\end{bmatrix} = 0 \quad \text{(4)}
\)

\( H \)

*the proof follows from noting that* 

\( \text{the rows } \hat{r}_i \text{ of } H \text{ belong to the dual code from (4) and form a basis for the code } \mathbb{C}^\perp \text{ since*} \)
\[
\text{rank } (H) = (n-k).
\]

Lemma 2. If the \((n-k \times n)\) matrix \(H\) has rank \((n-k)\) and satisfies
\[
H G^T = [0]
\]
where \(G\) is any generator matrix \(G\) of the \([n, k]\) code \(C\), then \(H\) is a valid f.c. \(n \times k\) matrix of the code \(C\).
\textbf{Proof:} \[
\begin{bmatrix}
    h_1^t \\
    \vdots \\
    h_n^t
\end{bmatrix}
\begin{bmatrix}
    g_1 \\
    \vdots \\
    g_k
\end{bmatrix}^T = [0]
\]

It follows that
\[
h_i^t g_j = 0 \quad \text{for all } i, j
\]
hence \( R \) is contained in \( \nu(\Pi) \).

But since \( \dim (R) = \dim (\nu(\Pi)) \)
\[ R = \eta(H) \]

and hence by Lemma 1, \( H \) is a valid P.C. \( m \times k \) of \( R \).

**Eq.** If \( \mathbf{G} = \begin{bmatrix} \mathbf{I}_k & \mathbf{P} \\ \mathbf{I} \end{bmatrix} \) \( (k \times k) \) identity \( m \times \)

then \( \mathbf{H} = \begin{bmatrix} \mathbf{P}^T & \mathbf{I}_{n-k} \end{bmatrix} \)
is a valid f.c. n x f R.

This is because:

\[ H C_T = \begin{bmatrix} p^T & I_{n-k} \end{bmatrix} \begin{bmatrix} I_k & p^T \end{bmatrix}^T \]

\[ = \begin{bmatrix} p^T & I_{n-k} \end{bmatrix} \begin{bmatrix} I_k \\ -p^T \end{bmatrix} = p^T + p^T \]

\[(n-k \times k) \]

Also it is clear that rank \((H)\)

\[= (n-k). \]
Img repetition code.

\[ G = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \end{bmatrix} \]

\[ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \end{bmatrix} \]

\[ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \end{bmatrix} \]

\[ H = \begin{bmatrix} P^T & In-k \end{bmatrix} \]

\[ H = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \]

and this is clearly a valid \( G \).
m x d thepetition code.
A generator matrix $G$ is said to be a systematic generator matrix for an $[n,k]$ code if it can be expressed in the form:

$$G = \begin{bmatrix} I_k & P \\ \vdots & \vdots \end{bmatrix}.$$
Qn: Does every code possess a systematic generator matrix?

Ans: No. Let $g$ be any generator matrix of the code $C$.

\[
g = \begin{bmatrix}
    g_1 & g_2 & \cdots & g_k & g_{k+1} & \cdots & g_n
\end{bmatrix}
\]

\[
= \begin{bmatrix}
    h_1 \\
    h_2
\end{bmatrix}
\]
then since any other gen \( \mathbf{m} \times \mathbf{R} \) can be obtained through taking linear combinations of \( \mathbf{g} \) above, it follows that \( \mathbf{R} \) has a systematic gen \( \mathbf{m} \times \mathbf{R} \) iff

\[
\text{rank } (\mathbf{g}) = k.
\]
Example: Let $R$ be the specific code.

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

$$G_1, \quad G_2$$

($k \times k$)  \quad ($6 \times 6$)
Since $h_1$ above has rank = 6, it follows that the Spc code does possess a systematic gen $mx$.

A second reason for being interested in a systematic gen $mx$ is consider the map between message vector $m$ and code words $c$ given by $f$.
\[ m \xrightarrow{\mathcal{C}} c \]

\[ m^t \mathcal{G} = c \]

\[ m^t \begin{bmatrix} I_k & \mathcal{P} \end{bmatrix} = \begin{bmatrix} m^t \mid m^t \mathcal{P} \end{bmatrix} \]

The message symbols are explicitly present in the code symbols.
Defn. Two codes $R_1$ and $R_2$ of block length $n$ are said to be equivalent if there is a mapping $\phi : R_1 \rightarrow R_2$ in 1-1 and onto fashion with the further property that the mapping $\phi$ corresponds to a coordinate permutation.
Recall

- defined namespace
- FT LA
- \((\mathbb{R}^n)^+ = \mathbb{R}\)

Two lemmas helpful in finding a p.c. m x t
- Systematic generator of $x$

- How find $H$ in this case

- Why of interest

- When do systematic generator matrices exist?
From the discussion in the last lecture, it follows that every linear code $R$ is equivalent to a second linear code $R'$ which possesses a systematic generator matrix.
Minimum distance of a linear block code

Thm: The min. dist. \( d_{\text{min}} \) of a linear block code \( C \) is equal to the minimum Hamming weight \( w_{\text{min}} \) of a non-zero code word.
If \( \leq \) have \( W_H(\leq) \leq W_{\text{min}} \).

Then \( d_H(\leq, 0) = W_{\text{min}} \).

\[ \vdash d_{\text{min}} \leq W_{\text{min}} \quad \square \]

On the other hand, let \( \leq_1, \leq_2 \in \mathcal{R} \) be such that

\[ d_H(\leq_1, \leq_2) = d_{\text{min}} \]
\[ W_A \left( \sum \frac{c_1}{c_2} \right) = \varphi_{\text{min}} \in \mathbb{R} \text{ since } \varphi \text{ is linear!} \]

\[ \Rightarrow \]

\[ \varphi_{\text{min}} \leq \varphi_{\text{min}} \quad \ldots \quad (2) \]

\[ \text{i. from (1) and (2),} \]

\[ \varphi_{\text{min}} = W_{\text{min}} \]
Eq \quad \mathcal{R} \text{ repetition code}

\mathcal{R} = \begin{array}{c}
\vdots \\
| \hspace{1cm} | \hspace{1cm} | \\
| \hspace{0.5cm} | \hspace{0.5cm} | \\
| \hspace{0.25cm} | \hspace{0.25cm} | \\
\vdots \\
\end{array}

\text{i. } w_{\text{min}} = 7 = d_{\text{min}}
Eg \( \mathcal{L} = \{ \mathcal{E} \} \) s.t.

\[
\mathcal{L} = \left\{ \mathcal{E} \mid \sum_{i=1}^{n} c_{i} = 0 \right\}
\]

i.e., \( \omega_{\text{min}} = 2 = \delta_{\text{min}} \)
\[ G = \begin{pmatrix}
0 & 1 & 2 & 3 & 4 & 5 & 6 \\
1 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 \\
\end{pmatrix} \]

\[ H = \begin{bmatrix}
\gamma^T \\
\mathbb{I}_{n-k}
\end{bmatrix} \]
\[
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 1 \\
\end{pmatrix}
\begin{pmatrix}
h_1 \\
h_2 \\
h_3 \\
h_7 \\
\end{pmatrix}

= 
\begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
\end{pmatrix}
\]

\[ H \leq 0 \iff \sum_{i=1}^{2} c_i h_i = 0 \]
$d_{\text{min}} = 3$ for the Hamming code

by inspection of the p.c.

max.

Deaf. Given an $(n-k \times n)$ p.c. $mxH$

and $[n, k]$ linear block code $C$, we define the parameter $s$ to
be the largest integer s.t. any 2 columns of $J$ are linearly independent.

**Thm** $d_{\min} = s + 1$

It follows from the observation that the existence of a codeword $c$ of Hamming weight $W$
the existence of a linear dependence relation amongst corresponding columns of the f.c. mx H.

\[ s = 2 \text{ in the case of the Hamming code, hence } \delta_{\text{min}} = 3. \]
General Hamming code.

Defn. Let $n \geq 2^r - 1$ be an integer, set $n = 2^r - 1$. Then a Hamming code of length $n$ is any code possessing a generator matrix of size $(r \times (2^r - 1))$ all of whose columns are nonzero and distinct.
Clearly it follows that the general Hamming code has parameters

\[ [n = 2^r - 1, \ k = 2^{r-1} - 2, \ 3] \]
Thm (Singleton bound)

\[
\rho_{\text{min}} \leq n - k + 1
\]

for any \([n,k]_R\) code \(R\).

\[\text{Def. } H \implies \text{rank } (H) \leq n-k\]

\[
\begin{pmatrix}
(n-k) \\
(n-k) \\
\end{pmatrix}
\]

\[\implies s \leq (n-k)\]
Defn A code whose $d_{\min}$ achieves the Singleton bound with equality is called a Maximum Distance Separable (MDS) code.

Eg 1 (general) repetition code.

\[
\left[ \begin{array}{c} n, 1, n \end{array} \right]
\]

$n \geq d_{\min}$, $d_{\min} = n - k + 1$ MDS
Eg 2 (general) spec code:

\[
\begin{bmatrix}
n, (n-1), 2 \\
n, k, d_{\min}
\end{bmatrix}
\]

\[
d_{\min} = n - k + 1 \quad \therefore \text{ MDS}
\]

Unfortunately these are the only possible families of binary MDS codes. (These two classes of codes are sometimes called
Bounds on the size of a code

Hamming Bound

Thm (Hamming Bound) The size $M$ of an $(n, M, d)$ code $C$ is upper bounded by:
\[ |R| \leq \frac{2^n}{t} \leq \binom{n}{a} \quad \text{for } a = 3 \]

where \( t = \left\lfloor \frac{L \min - 1}{2} \right\rfloor. \]
If \( \leq_1 \leq_2 \) are role works, then
Lec 12: Bounds on the size of a code

Recall

- $d_{\min} = w_{\min}$

- $d_{\min} = s + 1$

- Examples

- General Hamming code

- Singleton bound & MDS codes

- Proof of the Hamming Bound
Hamming Bound

\[ M \leq \frac{2^n}{\sum_{t=0}^{\frac{d_{\text{min}} - 1}{2}} \binom{n}{t}} \]
\( \text{Pf.} \)
\[
F \leq n \leq \frac{F}{2}
\]
\[
\left\{ B(c, t) \left| c \in c \subseteq C \right. \right\}
\]

But these balls are disjoint:
$\mathcal{B}(\leq, t) \cap \mathcal{B}(\leq', t) = \emptyset$

It follows that

$2^n \geq M \mid \mathcal{B}(\leq, t)\mid$

$= M \sum_{i=0}^{t} \left( \begin{array}{c} n \\end{array} \right)_{i}$

$\Rightarrow M \leq \frac{2^n}{\sum_{i=0}^{t} \left( \begin{array}{c} n \\end{array} \right)_{i}}$
Defn. A perfect code is a code that satisfies the Hamming bound with equality, i.e.,

\[ |C| = M = \frac{2^n}{\sum_{i=0}^{t} \binom{n}{i}} \]
Consider the repetition code for odd values of block length $n$:

Parameters: $[n, 1, n]$

$$M \leq \frac{2^n}{\sum_{i=0}^{t} \binom{n}{i}}$$
\[ \sum_{i=0}^{n} \binom{n}{i} = 2^n \quad \text{(well known)} \]

When \( n \) is odd, since
\[ \binom{n}{i} = \binom{n}{n-i}, \]
it follows that
\[ \sum_{i=0}^{n-1} \frac{\binom{n}{i}}{2^i} = 2^{n-1} \]
\[
M \leq \frac{n}{2^{n-1} - 2}
\]

\[\therefore\] all such repetition codes are perfect!

Exercise: Verify that the single-par code is not perfect in general.
Eq 2 The general Hamming code is always perfect!

Proof parameters:

\[ n = 2^k - 1, \quad k = 2^k - 1 - n, \quad d = 3 \]

\[ t = 1 \]

\[ M \leq \frac{2}{\sum_{i=0}^{n} \binom{n}{i}} = \frac{2}{1 + (2^n-1)} \]
\[ = 2^{2-1-n} \quad \text{perfect} \]

The Hamming bound is also called the sphere-packing bound.
Golay's observation:

If a linear code is perfect.

\[ \frac{1}{2} = M = 2 \]

\[ \sum_{i=0}^{t} \binom{n}{i} = 2^{n-k} \]

(2)
Eq 3. The Golay code

\[ n = 23, \quad d = 7, \quad t = 3 \]

\[ \sum_{\eta=0}^{3} \binom{23}{\eta} = \binom{23}{0} + \binom{23}{1} + \binom{23}{2} + \binom{23}{3} \]

\[ = 1 + 23 + 23 \cdot \frac{22}{2} + 23 \cdot \frac{11}{2} \cdot \frac{22}{2} \cdot \frac{1}{2} \cdot \frac{1}{3} \cdot \frac{1}{16} \cdot \frac{1}{1} \]
\[ \begin{align*}
&= 1 + 23 + 253 + (23)(77) \quad \frac{17+1}{3} \\
&= 1771 \\
&= \begin{array}{c}
253 \\
23 \\
1 \\
\end{array} \\
&\quad \frac{20+8}{=2} ! \\
\end{align*} \]

This numerical calculation led Gole to construct a perfect code:

\[ \begin{bmatrix}
23, 12, 7 \\
\end{bmatrix} \]
— now called the Golay code.

Gilbert - Varshamov (lower) bound (GV)

Thm (GV bound) The maximum possible size $M$ of a code of length $n$ and minimum distance $d$ satisfies:
\[ m \geq \frac{2^n}{t^{d-1}} \sum_{i=0}^{t^n} \binom{n}{i} \]

\underline{Pf. (via a greedy algorithm and construction).}
As long as
\[ M \left[ \sum_{\ell=0}^{d-1} \binom{n}{\ell} \right] < 2 \]
we can always enlarge the code to size \((M+1)\) while maintaining a min. distance \(d\).

At the stopping point we will have

\[ M > \frac{2}{d-1} \sum_{i=0}^{n} \binom{n}{i} \]

and this is the C.V. bound.
An approach to attaining reliable communication features:

(i) long codes are employed

(ii) bounded-distance decoding (BDD),
A bounded-distance decoder is one which, when given a received word, examines the ball \( B(\overline{x}, t) \) and declares \( \overline{c} \) (the decoded codeword) to equal \( \overline{s} \) if \( \overline{s} \) is the only codeword in the ball. (Else, gives up.)

\[ t = \left\lfloor \frac{d-1}{2} \right\rfloor \]
Note: by reliable communication, we communicate with negligible probability of error (virtually error-free).

A code word of large block length $n$
The probability that $k$ code symbols are corrupted is given by:

$$
\binom{n}{k} (1-\epsilon)^k \epsilon^{n-k}
$$

(comes from the binomial distribution)

When $n$ is large, this distribution tends to become Gaussian with parameters.
mean \in n \in \mathbb{N}, \quad \text{standard deviation} \quad = \sqrt{n \in (1 - \epsilon)}
Lec 13: Asymptotic Bounds

Recap

- Hamming bound
- perfect codes
- Gilbert - Varshamov bound
- an approach to achieving reliable communication.
\[ P_n \left( k \mid \text{success} \right) = \binom{n}{k} \epsilon^k (1-\epsilon)^{n-k} \]

\[ \text{Gaussian distribution} \]
\[ \eta \left( \eta \epsilon, \eta \epsilon (1-\epsilon) \right) \]

\[ \sigma = \text{std devn} = \sqrt{n \epsilon (1-\epsilon)} \]
\[ n \in \pm \sqrt{n \in} \quad \text{(constant)} \]
Conclusion: Long codes make the error pattern more predictable and hence more correctable.

Since there are no errors, we will use a code $d$

$$d = 2n - \epsilon$$
Define: given $0 < d < 1$, let 
\[ d = \lceil ns \rceil \] and let $M(n, s)$ be the largest possible size of a block code of length $n$ and minimum distance $d$.

Set:

\[
R(s) = \limsup_{n \to \infty} \left[ \frac{\log \left( M(n, s) \right)}{n} \right]
\]
$d = \text{ fractional minimum distance}$

Qn: How does $k(d)$ vary with $d$?

Note

Hamming bound:

$$M \leq \frac{2^{t}}{t!} \binom{n}{\frac{n}{2}}$$

$\hat{a} = 0$
Gilbert–Varshamov bound:

\[ M \geq \frac{2}{d - 1} \sum_{n=0}^{\infty} \binom{n}{n} \]

It can be shown that the Hamming and GV bounds imply that

\[ 1 - H_2(s) \leq R(s) \leq 1 - H_2\left(\frac{s}{2}\right) \]
For $0 \leq \theta \leq 1$,

\[
H_2(\theta) = \theta \log \frac{1}{\theta} + (1-\theta) \log \frac{1}{1-\theta}
\]

binary entropy $H_2$. 

![Graph of $H_2(\theta)$](image)
McEliece–Rodemich–Rumsey–Welch (MRRW) bound (upper bd)

$R(s)$ (natc)

Region where the best codes lie
\[ 1 - H_2(\delta) \leq R(\delta) \leq 1 - H_2\left(\frac{\delta}{2}\right) \]

(Our 3DD philosophy causes us to see)

\[ \delta = \delta = \frac{2n \varepsilon}{n} = 2 \varepsilon \]

\[ 1 - H_2(2\varepsilon) \leq R(\delta) \leq 1 - H_2(\varepsilon) \] \hspace{1cm} \text{Gurv} \hspace{1cm} \text{Hamming}

On the other hand, Shannon tells us that

\[ \max_k \frac{a}{c} = c = 1 - H_2(\varepsilon) \]
Conclusion: Our combination of long block length and 3DD causes us to require the use of long block codes that achieve the Hamming bound to achieve channel capacity.
Minimum Probability of Error Decoder

\[ \mathcal{E} = \{ \text{event that a code word is erroneously decoded} \} \]

\[ \mathcal{E}^c = \{ \text{event that is correctly decoded} \} \]

\[ P_n(\mathcal{E}^c) = \sum_{\hat{n}=1}^{M} P_n(\hat{\mathcal{E}}^c) \left( \frac{1}{C} F_n(\mathcal{E}^c | \hat{\mathcal{E}}^c) \right) \]

\[ M = |\mathcal{E}| \]

receive vector

= \sum_{\hat{n}=1}^{M} P_n(\hat{\mathcal{E}}^c) \left( \frac{1}{C} F_n(\mathcal{E}^c | \hat{\mathcal{E}}^c) \right)
\[ M = \sum_{n} p_n \left( \leq \zeta \right) \sum_{n} p_n \left( \exists \xi \right) \mathbb{I}_{H_1} \left( \xi \right) \]

\[ \lambda = 1 \quad \forall j \in F_2 \]

\[ \mathbb{I}_{H_1} \left( \varepsilon \right) \begin{cases} 1 & \varepsilon \in H_1 \\ 0 & \text{else} \end{cases} \]
\[ \sum_{j \in \mathbb{F}_2^n} \sum_{i=1}^{M} p_n (e_i) p_n (\Xi_i | e_i) \mathbb{I}_{H_i} (\Xi) \]

to maximize the probability of correct decisions, the decoder assigns

\[ \hat{x} \text{ to } H_i \iff p_n (e_i) p_n (\Xi_i | e_i) \geq p_n (e_j) p_n (\Xi_j | e_j) \]

\[ j \neq i \]

\[ 1 \leq i, j \leq M \]
Typically, all codewords are equally likely, i.e., $p_i(\mathbf{e}_i) = \frac{1}{M}$, all $i$.

in which case the probability of correct decision is maximized by selecting

$$j \in A \iff \\frac{p_i(\mathbf{y}_i)}{p_i(\mathbf{x}_i) + \frac{2}{M}}$$

A decoder that uses this rule for making decisions is called a MLD.
(maximum-likelihood decoder)

Note: If all codewords are equally likely then the MLD will also minimize code word error probability.

Note: In case of ties, one flips a coin.
Lemma Over a BSC, MLD reduces to minimum distance decoding (MDD)

\[ d_H (\frac{1}{2}, \epsilon_n) = d \]

\[ \ln \left( \frac{1}{\epsilon} \right) = (1 - \epsilon)^n - d \]

\[ = (1 - \epsilon) \left( \frac{\epsilon}{1 - \epsilon} \right)^d \]
If \( t << 1 \), \( \frac{\epsilon}{1-\epsilon} \)

\[ \Rightarrow p_n \left( \frac{\epsilon}{1-\epsilon} \right) \text{ is maximized} \]

by minimizing

\[ p_H \left( \frac{\epsilon}{1-\epsilon} \right) \]
Lec 14 Standard Array Decoding

Recap

- Asymptotic bounds
- Min prob Jensen decoder
- Maximum likelihood decoder
- Minimum Hamming distance decoder

MLD
MDD
The MDD chooses \( \hat{z} \) (the decoded code word) such that (s.t.)

\[
d_H (y, \hat{z}) \text{ is a minimum with } \hat{z} \in \mathcal{C} \text{ code}
\]
\[ y + x_1 = e_1 \]
\[ y + x_2 = e_2 \]

Thus the decoding (MDD) algorithm can equivalently be phrased as follows:
Step 1 form the set

\[
\{ y + \mathcal{E} \mid \mathcal{C} \subseteq \mathcal{C} + \mathcal{E} \}
\]

Step 2 Let \( \hat{\mathcal{C}} \) be the element in \( \mathcal{C} + \mathcal{E} \) having least Hamming weight.

Step 3 the decoded code word \( \hat{z} \) is then given by
\[ \hat{c} = \hat{y} + \hat{\varepsilon} \]

Note that (i) \( J + R \) is a coset of the subgroup \( R \) of \( \mathbb{F}_2^n \).

(ii) and hence the decoder action is only a function to the coset \( J R \) to which \( y \) belongs and not to \( J \) itself.
Eq

The equation is:

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

And:

$$[n, k, l] = [4, 2, 2]$$

$$G = \begin{bmatrix}
1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0
\end{bmatrix}, \quad H_1 = \begin{bmatrix}
1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0
\end{bmatrix}$$

3 other

(0, 0, 0, 0)

for

$$I_2, 4$$

and

$$2$$
\[ H = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \]

\[ n-k = 2 \]

<table>
<thead>
<tr>
<th>C</th>
<th>0000</th>
<th>1010</th>
<th>0101</th>
<th>1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001 + C</td>
<td>0001</td>
<td>1011</td>
<td>0100</td>
<td>1110</td>
</tr>
<tr>
<td>0010 + C</td>
<td>0010</td>
<td>1000</td>
<td>0111</td>
<td>1101</td>
</tr>
<tr>
<td>0011 + C</td>
<td>0011</td>
<td>1001</td>
<td>0110</td>
<td>1100</td>
</tr>
</tbody>
</table>

\[ \text{Eq. of decoding} \quad J = 0111 \]

MDD algorithm:

\[ J + \hat{e} = 0111 + 0010 = 0101 \]
Defn. The syndrome $S$ associated to a received vector $\mathbf{r}$ is given by:

$$S = H \mathbf{r} \bmod (n-k \times \mathbb{Z})$$

**Example:**

$\mathbf{r} = 0111$  
$H = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$

$$S = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$
Lemma: There is a 1-1 correspondence between the cosets $\mathcal{E} + R$ and syndromes $s \in \mathbb{F}_{2}^{n-k}$.

Proof: $\phi: y + R \rightarrow \{j | j = s\}$

Is $\phi$ well-defined?

Suppose $y' \in y + R$

$\Rightarrow y' = y + s, \quad s \in \mathbb{F}_2$
\[ \text{Size} = 2 \]

\[ n-k \]

\[ J + R \]

\[ J + R \]

\[ \phi \text{ is well-defined.} \]

\[ 2 \text{ cosets} \]
Suppose \( \phi (\overline{z} + R) = \phi (\overline{z_1} + R) \)

\( \Rightarrow \) \( H \overline{z} = H \overline{z_1} \)

\( \Rightarrow \) \( H (\overline{z} + \overline{z_1}) = 0 \)

\( \Rightarrow \) \( \overline{z} + \overline{z_1} \in R \)

\( \Rightarrow \) \( \overline{z} + \overline{z_1} = \overline{c}, \quad c \in \mathbb{R} \)

\( \Rightarrow \) \( \overline{z_1} = \overline{z} + \overline{c}, \quad c \in \mathbb{R} \)

\( \Rightarrow \) \( \overline{z_1}, \overline{z} \) define the same
This leads to the following simple implementation of the MLD algorithm (syndrome decoding):

\[ S = H^\top J \]
Step 2

Use table look up to determine \( \hat{z} \) (the closest leader associated to syndrome \( s \))

Step 3

decode to:

\[
\hat{c} = \hat{z} + \hat{e}
\]
Performance Analysis via the Standard Array

<table>
<thead>
<tr>
<th></th>
<th>1010</th>
<th>0101</th>
<th>1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>1010</td>
<td>0101</td>
<td>1111</td>
</tr>
<tr>
<td>0001</td>
<td>1011</td>
<td>0100</td>
<td>1110</td>
</tr>
<tr>
<td>0010</td>
<td>1000</td>
<td>0111</td>
<td>1101</td>
</tr>
<tr>
<td>0011</td>
<td>1001</td>
<td>0110</td>
<td>1100</td>
</tr>
</tbody>
</table>

The goal in performance analysis is to determine the probability of error.
associated to the code.
Recap

- introduced the standard array
- defined the syndrome
- laid down the steps involved in carrying out SAD
Performance Analysis via the Standard Array

<table>
<thead>
<tr>
<th>0000</th>
<th>1010</th>
<th>0101</th>
<th>1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>1011</td>
<td>0100</td>
<td>1110</td>
</tr>
<tr>
<td>0010</td>
<td>1000</td>
<td>0111</td>
<td>1101</td>
</tr>
<tr>
<td>0011</td>
<td>1001</td>
<td>0110</td>
<td>1100</td>
</tr>
</tbody>
</table>

\[ \hat{e} = \text{true error} \]

The goal in performance analysis is to determine the probability of error.

The corrected vector is shown as the first column.
Lemma 1. The received vector and the error pattern \( \mathbf{e} \) belong to the same coset of the code and hence share the same syndrome.

If \( \mathbf{j} = \mathbf{c} + \mathbf{e} \), \( \mathbf{e} \in \mathbb{F} \)

\( \mathbf{j} \) and \( \mathbf{e} \) belong to the same coset \( \mathcal{J} \).
\[ H(\bar{y}) = H(c + \varepsilon) = -1|\varepsilon| \]

and hence \( \bar{y} \), \( \varepsilon \) share the same syndrome.

Suppose \( \varepsilon \) to be the true error pattern. The decoder computes \( H(y) = H\varepsilon = 0 \).

Let \( \hat{\varepsilon} \) be the coset leader associated to syndrome \( \varepsilon \).
Decoder computes:

\[ J + \hat{e} = \hat{L} + \hat{e} + \hat{\epsilon} \]

**Residual error pattern**

It follows then that in the standard array \( J \), the residual error vector \( \hat{e} \) is the vector in the table at the head of the column to which \( \epsilon \) belongs.
It follows that the only error patterns that the code is able to correct are
precisely the error patterns corresponding to the cost leaders!!

<table>
<thead>
<tr>
<th></th>
<th>0000</th>
<th>1010</th>
<th>0101</th>
<th>1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>1010</td>
<td>0101</td>
<td>1111</td>
<td></td>
</tr>
<tr>
<td>0001</td>
<td>1011</td>
<td>0100</td>
<td>1110</td>
<td></td>
</tr>
<tr>
<td>0010</td>
<td>1000</td>
<td>0111</td>
<td>1101</td>
<td></td>
</tr>
<tr>
<td>0011</td>
<td>1001</td>
<td>0110</td>
<td>1100</td>
<td></td>
</tr>
</tbody>
</table>

Hence the prob. of codeword error is given by:

\[
P_{cwe} = 1 - \left\{ (1 - \epsilon)^4 + 2(1 - \epsilon)^3 \epsilon^2 + (1 - \epsilon)^2 \epsilon^2 \right\}
\]
\[
\begin{array}{c|cccc}
C & 00 & 10 & 01 & 11 \\
\hline
0000 & 1010 & 0101 & 1111 \\
0001 & 1011 & 0100 & 1110 \\
0010 & 1000 & 0111 & 1101 \\
0011 & \text{[Red]} & 0110 & 1100 \\
\end{array}
\]

<table>
<thead>
<tr>
<th>\text{message}</th>
</tr>
</thead>
<tbody>
<tr>
<td>\text{residual}</td>
</tr>
<tr>
<td>\text{error patterns!}</td>
</tr>
</tbody>
</table>

\[
\forall \mathbf{c} \in C \quad \mathbf{c}^T \mathbf{c} = c^T c
\]

\[
\mathbf{e} = 1001 \quad \hat{\mathbf{e}} = 0011
\]

\[
\therefore \mathbf{e} = 1001 + 0011 = 1010
\]

\[
\Rightarrow \mathbf{m}_1 \text{ is decoded erroneously}
\]

\[
\mathbf{m}_2 \text{ is decoded correctly}
\]
\[ \hat{c} = c + \varepsilon + \varepsilon = c + \varepsilon \]

<table>
<thead>
<tr>
<th></th>
<th>0000</th>
<th>1010</th>
<th>0101</th>
<th>1111</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>1011</td>
<td>0100</td>
<td>1110</td>
<td></td>
</tr>
<tr>
<td>0010</td>
<td>1000</td>
<td>0111</td>
<td>1101</td>
<td></td>
</tr>
<tr>
<td>0011</td>
<td>1001</td>
<td>0110</td>
<td>1100</td>
<td></td>
</tr>
</tbody>
</table>

It follows that the probability that \( m_1 \) is erroneously decoded while \( m_2 \) is correctly decoded is given by:

\[
= (1-\varepsilon)\varepsilon + 2(1-\varepsilon)^2 \varepsilon^2 + (1-\varepsilon)^3 \varepsilon^3
\]
Note: The residual error pattern $\tilde{\mathbf{e}}_i$ is independent of the transmitted codeword and this is what enables this analysis to be carried out.
Convolutinal Codes

\[
\begin{align*}
\{u_k\}_{k=0}^{\infty} & \quad \rightarrow \quad D \quad \rightarrow \quad + \quad \rightarrow \quad \{v_k\}_{k=0}^{\infty} \\
\{v_k\}_{k=0}^{\infty} & \quad \rightarrow \quad D \quad \rightarrow \quad + \quad \rightarrow \quad \{v_L\}_{L=0}^{\infty}
\end{align*}
\]

Initially both are set to 0

\[(1) \quad V_k = u_k + u_{k-1} + u_{k-2}\]

\[(2) \quad V_L = u_L + u_{L-2}\]
<table>
<thead>
<tr>
<th>$u_k$</th>
<th>$V_k^{(1)}$</th>
<th>$V_k^{(2)}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Input: $(u_{k-1}, u_{k-2})$

Output: $(V_k^{(1)}, V_k^{(2)})$
Convolutional codes belong to the class of tree codes that are:

1. Finite memory
2. Linear
3. Time invariant
\[
\begin{bmatrix}
  u_0 & u_1 & u_2 \\
\end{bmatrix}
\begin{bmatrix}
  11 & 10 & 11 \\
  11 & 10 & 11 \\
  11 & 10 & \cdots \\
  \vdots & \ddots & \vdots \\
\end{bmatrix}
= 
\begin{bmatrix}
  v_0 & v_1 & v_2 \\
  v_0 & v_1 & v_2 \\
  \vdots & \ddots & \vdots \\
\end{bmatrix}
\]

Semi-infinite generator matrix.
Field of formal power series $\mathbb{F}(\!(x)\!)$

over the scalar field $\mathbb{F}$

$\mathbb{F}(\!(x)\!):= \left\{ \sum_{k=-d}^{\infty} a_k x^k \mid a_k \in \mathbb{F}, d \geq 0 \right\}$

It is clear that all field axioms are satisfied with the possible exception of the multiplicative inverse.
\[ E^g \ (\mathbf{1} + \mathbf{D}^3)^{-1} = \mathbb{I}^g (1 + d^2)^{-1} \]
Recap

- completed performance analysis of the SAD

- convolutional codes
  - encoder
    - semi-infinite generator matrix
— formal power suits
\[ F((D)) = \left\{ \sum_{k=-1}^{\infty} a_k D^k \mid a_k \in F \right\} \]

\( (F((D)), +, \cdot) \)  

\( (F((D)), +) \) Abelian group

\[ 1 + D + D^3 + D^7 + \ldots + D^{19} \]

\[ + \left( D^4 + D^5 + D^6 + D^7 + \ldots \right) \]

\[ = 1 + D + D^3 + D^4 + D^5 + D^6 + D^8 + D^{19} + \ldots \]
\( (F(\langle x \rangle), \cdot) \Rightarrow \text{closure} \)

\( \ast \cdot \cdot = 1 \)

\( \text{associative} \)

\( \text{commutative} \)

\( \text{inverse} \)?
Computing inverses in $F((D))$:

$$(D + D^3)^{-1} = \frac{1}{D + D^3} = \frac{1}{D(1 + D^2)} = D^{-1} \frac{1}{1 + D^2}$$

(Aside: $\frac{1}{1 + g(D)}$ polynomial in D)
that is divisible by $D$, then

$$\frac{1}{1 + g(D)} = 1 + g(D) + [g(D)]^2 + [g(D)]^3 + \ldots$$

If

$$\frac{1}{(1 + g(D)) (1 + g(D) + [g(D)]^2 + [g(D)]^3 + \ldots)} = 1$$
\[
\frac{D^{-1}}{1 + D^2} = D^{-1}(1 + D^2 + D^4 + D^6 + \ldots)
\]

\[
= D^{-1} + D + D^3 + D^5 + \ldots
\]

---

Goal: Describe the convolutional encoder in terms of a polynomial generator matrix (PGM):

\[
G(D) = \begin{bmatrix} 1 + D + D^2 & 1 + D^2 \\ 1 + D + D^2 & 1 + D^2 \end{bmatrix}
\]

(1 x 2)
\[ V_k = u_k + u_{k-1} + u_{k-2} \quad \ldots \quad (1) \]

\[ V_L = u_k + u_{k-2} \quad (2) \]
Define:

\[ U(D) = \sum_{k=0}^{\infty} u_k D^k \]

Input power series

\[ V(D) = \sum_{k=0}^{\infty} v_k D^k \]

\[ V(D) = \sum_{k=0}^{\infty} (1)^k v_k D^k \]

\[ V(D) = \sum_{k=0}^{\infty} (2)^k v_k D^k \]

Converting the time-domain input-output relation given in (1) to the \( D \)-transform
domain, we get:

\[
\begin{bmatrix}
\nu(D)^{(1)} & \nu(D)^{(2)} \\
\end{bmatrix}
\]

\[
= \nu(D) \begin{bmatrix}
1 + D + D^2 & 1 + D^2 \\
\alpha(D) & (1 \times 2)
\end{bmatrix}
\]
Finite-State Machine Description

- **Past 2 Symbols**
  - $00 \rightarrow u_{k-2}$
  - $00 \rightarrow u_{k-1}$

- **Input**
  - $0$
  - $1$
<table>
<thead>
<tr>
<th>In</th>
<th>State</th>
<th>(V^{(1)})</th>
<th>(V^{(2)})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>00</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>00</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>10</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>01</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>11</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Lec 17: The Viterbi Decoder

Recap:
- Completed our discussion on formal power series and polynomials leading to the polynomial generator matrix of the code.
- Code:
  - State diagram of the encoder viewed as a FSM
  - Trellis diagram
Input
If \( \sum_{k=0}^{M-1} \sum_{k} v_k \) \( v_k \) \( k \leq N \)

is the output of the convolutional code encoder, then the associated path segment metric is given by:

\[
\delta \left( \sum_{k=0}^{M-1} \sum_{k} v_k \right) \sum_{k} x_k \left( x_k \right)_{k=0}^{\infty}
\]
Similarly, an edge (or branch) in the trellis associated to code symbols

\[
\left( \begin{array}{c} v_k^{(1)} \\ v_k^{(2)} \end{array} \right) = d \left( \{ v_k^{(1)} \} \cup \{ v_k^{(2)} \} \right)
\]

— in the Viterbi decoding process, with each node at each node level, we associate a survivor, where by
survivor at a node, we mean the path from the start node to that node having least path metric.
Recap

- labelled the trellis diagram
- explained the operation of the Viterbi decoder
Note: by adding a string of zeros at the tail of the message sequence, in order to bring the encoder back to the all-zero state, we have suffered a small loss in rate:

- If the convolutional code is rate $\frac{1}{n}$, and has memory $\nu$ (number of shift registers on the input line), then the loss in rate
per message sequence of length $N$ is given by:

$$\frac{1}{n} - \frac{1}{n} \left[ \frac{N - \nu}{N} \right]_{\text{actual}}$$
Example of catastrophic error propagation (CEP)

\[ g(D) = \begin{bmatrix} 1 + D & 1 + D^2 \end{bmatrix} \]

\[
\begin{bmatrix}
V^{(1)}(D) & V^{(2)}(D)
\end{bmatrix} = U(D) \begin{bmatrix} 1 + D & 1 + D^2 \end{bmatrix}
\]

Suppose input \( U(D) = \sum_{k=0}^{\infty} u_k D^k = \frac{1}{1 + D} \)

\( \Rightarrow \{ u_k \}_{k=0}^{\infty} = 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ - \ - \ - \ - \)
Then

\[
\begin{bmatrix}
V^{(1)}(D) \\
V^{(2)}(D)
\end{bmatrix}
= \frac{1}{(1+D)}
\begin{bmatrix}
1+D & 1+D^2 \\
1+D & 1+D
\end{bmatrix}
\]

\[
= \begin{bmatrix}
\frac{1+D}{1+D} & \frac{1+D^2}{1+D} \\
1+D & 1+D
\end{bmatrix}
\]

\[
\begin{bmatrix}
\{V_k^{(1)} \} \\
\{V_k^{(2)} \}
\end{bmatrix} \Leftarrow \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}
\text{channel corrupted}
\]

\[
\begin{bmatrix}
\{V_k^{(1)} \} \\
\{V_k^{(2)} \}
\end{bmatrix} \Leftarrow \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}
\]
An encoder matrix $G(D)$ for a convolutional code is said to have catastrophic
error propagation if there is some input \( u(d) \) with \( \infty \) Hamming weight that generates an output

\[
\begin{bmatrix}
  v(1) \\
  v(2) \\
  \vdots \\
  v(n)
\end{bmatrix}
\]

whose Hamming weight is finite.

This terminology stems from the observation that if the encoder has CEP, then a finite \# of channel
Errors can cause an $\#j$ message.

Symbol errors.

\[ G_1(D) = \begin{bmatrix} 1 + D + D^2 & 1 + D^2 \\ \end{bmatrix} \]

No CEP.

\[ G_2(D) = \begin{bmatrix} 1 + D & 1 + D^2 \\ \end{bmatrix} \]

Has CEP.
Thm. A necessary and sufficient condition on the generator $m_x$ of a rate $\frac{1}{n}$ convolutional code to avoid CEP is that:

$$g_c = \left( g_1(d) \ g_2(d) \ \ldots \ g_n(d) \right)$$

$$= D^{\ell} \ \ell \geq 0$$

where $h(d) = \left[ g_1(d) \ g_2(d) \ \ldots \ g_n(d) \right]$
is the PAM of the code.

GCD - quick review

\[ \gcd(27, 63) = ? \]

\[ 63 \div 27 = 2 \quad \text{with a remainder of} \quad 9 \]

\[ 27 \div 9 = 3 \]

\[ 9 \div 0 = \]
<table>
<thead>
<tr>
<th>Remainder</th>
<th>63</th>
<th>27</th>
<th>Quotient</th>
</tr>
</thead>
<tbody>
<tr>
<td>63</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>27</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>-2</td>
<td>3</td>
</tr>
</tbody>
</table>

$\text{gcd}(63, 27) = 9 = 63 \cdot 1 + (-2) \cdot 27$
### Eq (polynomial case)

<table>
<thead>
<tr>
<th>Remainder</th>
<th>$D^2 + D + 1$</th>
<th>$D + 1$</th>
<th>Quotient</th>
</tr>
</thead>
<tbody>
<tr>
<td>$D^2 + D + 1$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$D^2 + 1$</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$D$</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$1$</td>
<td>$D$</td>
<td>$D + 1$</td>
<td>$D$</td>
</tr>
</tbody>
</table>

$\gcd$ \[\begin{array}{c|c|c}
1 & 0 & 0 \\
0 & 1 & 1 \\
\end{array}\]

Since $\gcd(D^2 + D + 1, D + 1) = 1$,

there is no CEP (by the theorem).
\[ G(D) = \begin{bmatrix} 1+D & 1+D^2 \end{bmatrix} \]

<table>
<thead>
<tr>
<th>Remainder</th>
<th>D^2+1</th>
<th>D+1</th>
<th>Quotient</th>
</tr>
</thead>
<tbody>
<tr>
<td>D^2+1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>D+1</td>
<td>0</td>
<td>1</td>
<td>D</td>
</tr>
<tr>
<td>D+1</td>
<td>1</td>
<td>D</td>
<td>1</td>
</tr>
</tbody>
</table>

Note: \( gcd \neq D^2, D+1, 2 \) and hence, this code has CEP.
Finite-State Machine Description

Diagram:

- States: LID, LD, LI, 00, 01
- Transitions:
  - From LID to 00: Input = 0
  - From LID to 01: Input = 1
  - From 00 to LD: Input = 0
  - From 00 to LI: Input = 1
  - From LD to LI: Input = 0
  - From LD to LD: Input = 1
  - From LI to 00: Input = 0
  - From LI to 01: Input = 1

Past 2 Symbols:

- u_{k-1}
- u_{k-2}
Lec 19  Path Enumeration

Recap:

- CEP in rate \( \frac{1}{n} \)
- Convolutional codes
  - GCD computation
    - Polynomials
  - Path enumeration
Finite-State Machine Description

Diagram of a finite-state machine with states and transitions labeled.
Modified State Diagram

LID -> LID

LID -> LD

LD -> LD

START

END

\[ e_1 = \sum_{i} \text{message symbols} \]

\[ i_2 = e_2 = W_{41} \quad \text{(input)} \]

\[ D^2 e_3 = e_3 = W_{41} \quad \text{(output)} \]
Our interest is in computing for the END state the power series

\[ A(L, D, I) = \sum_{i,j,k} a_{ijk} L^i D^j I^k \]

END

\[ \# \text{ of paths in the modified state diagram } \]

\[ \text{path length } i, \text{ whose associated input (message) sequence has Hamming weight } j \text{ and } \]

\[ \text{whose associated output sequence } \]
has Hamming weight $k$.

Powerset series such as $A (l, I, D)$ are also called generalising functions.
\[
\begin{bmatrix}
A_{10} \\
A_{11} \\
A_{01}
\end{bmatrix}
= \begin{bmatrix}
0 & 0 & LID \\
LID & LID & 0 \\
LD & LD & 0
\end{bmatrix}
\begin{bmatrix}
A_{10} \\
A_{11} \\
A_{01}
\end{bmatrix}
+ \begin{bmatrix}
LID^2 \\
0 \\
0
\end{bmatrix}
\]

\(A_{10}\) is abbreviation for \(A_{10}(L, I, D)\).

Also: \(A_{\text{START}}(L, I, D) = 1\)
and

\[ A_{\text{END}} (L, I, D) = LD^2 A_{01} (L, I, D) \]

\[
\begin{bmatrix}
1 & 0 & -LI \\
-LID & 1-LID & 0 \\
-LI & -LID & 1
\end{bmatrix}
\begin{bmatrix}
A_{10} \\
A_{11} \\
A_{01}
\end{bmatrix}
= 
\begin{bmatrix}
LD^2
\end{bmatrix}
\]

\[ x = A_0 y + b \]

\[ \Rightarrow \left[ I - A \right] y = b \]
Solving using (name's rule) yields:

\[ A_{01}(L_1, L_2, D) = \frac{L^2 I D^3}{1 - L_1 D - L_2 D} \]

\[ A_{EH} = L D^2 \quad A_{01} = \frac{L^3 I D^5}{1 - L_1 D(1 + L)} \]
Note: Any expression of the form

\[
\frac{1}{1 - g(L,I,I,d)}
\]

can be expanded as:

\[
\frac{1}{1 - g(L,I,I,d)} = 1 + g(L,I,I,d) + g^2(L,I,I,d) + g^3(L,I,I,d) + \cdots
\]

with no constant term.
\[ A_{\text{END}} = L D^2 A_0 = \frac{L^3 I D^5}{1 - L I D (1 + L)} \]

\[ = L^3 I D^5 \left\{ 1 + L I D (1 + L) + L^2 I^2 D^2 (1 + L)^2 + \ldots \right\} \]

How many paths have length 4?

(Our interest is in $L^4$ terms)

\[(L^3 I D^5) L I D = L I^2 D^6.\]
General rate \((k/n)\) convolutional code

\[
E g \rightarrow g (D) = \begin{bmatrix}
1+D & 1 & 1+D \\
0 & 1+D & 0 \\
p & q & m \\
(2 \times 3)
\end{bmatrix}
\]

**Encoding:**

\[
\begin{bmatrix}
U^{(1)} (D) \\
U^{(2)} (D)
\end{bmatrix}
\rightarrow g (D)
\]

\[
= \begin{bmatrix}
V^{(1)} (D) \\
V^{(2)} (D) \\
V^{(3)} (D)
\end{bmatrix}
\]
Memory of the convolutional encoder
\[ a(\theta) = \begin{bmatrix} g_{11}(\theta) & \cdots & g_{1n}(\theta) \\ \vdots & \ddots & \vdots \\ g_{k1}(\theta) & \cdots & g_{kn}(\theta) \end{bmatrix} \]

\[ \mathcal{D} = \frac{1}{k} \sum_{i=1}^{k} \max_{1 \leq j \leq n} \{ \gamma_{ij} \} \]

\[ \gamma_{ij} = \deg(g_{ij}(\theta)) \]

In the example:

\[
\begin{bmatrix}
1+\theta & 0 & 1+\theta \\
0 & 1+\theta & 0 \\
\theta & 1+\theta & 0
\end{bmatrix}
\]
\[
[V_i] = \begin{pmatrix}
1 & 1 \\
1 & -\infty
\end{pmatrix}
\]

\[\forall = \max \{1, 1, 1\} + \max \{1, 1, -\infty\}\]

\[= 1 + 1 = 2.\]
General I/O relation:

\[ V = \sum_{t} \sum_{\hat{\alpha}=1}^{\infty} \sum_{m=0}^{\infty} u_{t,m} \delta_{m} \]

Where \( g_{ij}(\hat{\alpha}) \triangleq \sum_{m=0}^{\infty} f_{m} \delta_{m} \)

\[ V(\hat{\alpha}) \triangleq \sum_{t=0}^{\infty} V_{t} \delta_{t} \]
Recap

* path information enumeration
  using generating function techniques

* General rate $k/n$ convolutional code - PG/M derivation
Thm (CEP) A n.a.s.c. on the PGM

e a general, rate $\frac{k}{n}$ convolutional code to not have CEP is that:

$$\gcd \left( \Delta_1(d), \ldots, \Delta_n(x) \right) = 1 \quad \forall n \geq 0$$

where

$$\left\{ \Delta_p(d) \right\}_{p=1}^n$$

is the collection of
Determinants of the \( \binom{n}{k} \times k \times k \) submatrices of \( A(\mathcal{D}) \)

\[
\begin{bmatrix}
1+\mathcal{D} & 1 & 1+\mathcal{D} \\
\mathcal{D} & 1+\mathcal{D} & 0
\end{bmatrix}
\]

\( k = 2 \quad n = 3 \)

\( (2 \times 3) \)

\( \binom{n}{k} = \binom{3}{2} = 3 \)
\[ \Delta_1(D) = \begin{vmatrix} 1+D & 1 \\ D & 1+D \end{vmatrix} = (1+D)^2 + D \]

\[ \Delta_2(D) = \begin{vmatrix} 1+D & 1+D \\ D & 0 \end{vmatrix} = D(1+D) \]

\[ \Delta_3(D) = \begin{vmatrix} 1 & 1+D \\ 1+D & 0 \end{vmatrix} = (1+D)^2 \]
Can be verified that

\[ \text{gcd} \left( 1 + D + D^2, \ D + D^2, \ 1 + D^2 \right) = 1 \]

Viterbi decoding over the AWGN channel

Channel Model:

\[ s \in \{ \pm \sqrt{E} \} \]

\[ j = s + n \]

\[ n \sim N(0, \frac{N_0}{2}) \]
\( \frac{E}{N_0} \) is a measure of the SNR on the channel.

As in the case of the BSC, it can be shown that when all codewords are equally likely, then the decoder that minimizes the probability of codeword error is the ML decoder.

Choose that codeword that
maximizes \[ \phi \left( \frac{y}{c} \right) \]

\[ c^2 \]

where \[ S_i = (-1)^i \sqrt{E} \]

Symbol transmitted

\[ \text{Symbol transmitted over the AWGN channel} \]

In the convolutional code:

\[ z(t) = \left( v^{(1)}_t, v^{(2)}_t, \ldots, v^{(n)}_t \right), \quad t = 0, 1, \ldots, N-1 \]
\[ \phi(z \mid c) = \prod_{t=0}^{N-1} \prod_{j=1}^{N} \phi(J_{t}^{(j)} \mid v_{t}^{(j)}) \]

where

\[ \phi(J_{t}^{(j)} \mid v_{t}^{(j)}) = \frac{1}{\sqrt{2\pi(N_{0}/2)}} \exp \left( -\frac{1}{2} \frac{(J_{t}^{(j)} - s_{t}^{(j)})^2}{N_{0}/2} \right) \]

\[ s_{t}^{(j)} = \sqrt{\text{E} \left( E_{t}^{(j)} \right) } \]

Clean that MLD calls for
\[
\sum_{t=0}^{n-1} \sum_{j=1}^{m} \left[ \tilde{y}_{t,j} - \tilde{s}_{t,j} \right]^2
\]

\[
\sum_{t=0}^{n-1} \left[ \tilde{y}_{t} - \tilde{s}_{t} \right]^2 - 2 \sum_{t=0}^{n-1} \tilde{y}_{t} \tilde{s}_{t} + \sum_{t=0}^{n-1} \tilde{s}_{t}^2
\]

and thus MLD is aimed at maximizing the inner product.

\[
E
\]
\[
\sum_{E} \sum_{j=1}^{n} \sum_{t=0}^{t-1} (-1)^{t} \phi_{j}(p) \sum_{i=1}^{N-1}
\]

need to maximize

- Each code word is associated to a path
- The associated inner product is called the path metric
- Each path metric is the sum of branch metrics.
- Decoding proceeds exactly as in the case of the BSc except of the fact that our branch metrics are now real numbers (the or -ve) as opposed to the integers in the case of the
BSC

we are looking to maximize path metrics (and not seeking the path with minimum metric as in the case of the BSC).
Upper bound on the bit error probability

Turns out that the generalizing function $A_{\text{END}}(L, \text{I.I.D})$ derived can be used to provide an upper bound on the probability of bit error incurred while
employing the Viterbi decoder:

Case (i) BSC channel:

\[ P_{be} \leq \frac{1}{k} \sum_{A \in \text{END}(L, I, D)} \left( \frac{L}{I}, I, D \right) \]

BIT ERROR PROB.

\[
\begin{array}{c c c}
0 & 1-\epsilon & 0 \\
\epsilon & 1 & 1-\epsilon \\
1 & 1-\epsilon & 1
\end{array}
\]

\[ D = 2 \sqrt{\epsilon(1-\epsilon)} \]
Case (ii) A WGN channel:

\[ P_{bc} \leq \mathcal{Q} \left( \sqrt{\frac{2E_I \Delta f_{\text{free}}}{N_0}} \right) \exp \left( -\frac{E_d \Delta f_{\text{free}}}{N_0} \right) \]

\[ \times \frac{1}{k} \sum_{i=1}^{k} A \left( L, I, D \right) \left( L = 1 \right) \left( I = 1 \right) \]

\[ D = \exp \left( -\frac{E}{N_0} \right) \]
lec 21  The Generalized Distributive Law

Recap

* SC EP in general, rate k/n
* Convolutional code.
* Viterbi decoding over the AWGN channel
* Expressions to compute bit error probability of the Viterbi decoder.
Clarification: $d_{\text{face}} = ?$

$d_{\text{face}}$ is the minimum distance between a pair of distinct codewords (of $\infty$ length) in the convolutional code.

Can be computed from the power series $A_{\text{EHC}}(l, I, D)$ as follows:
Set \( L = I = 1 \) to get \( A_{\text{END}} (1,1,1) \).

In this power series, \( \text{face} \) is the smallest exponent \( D \).

Eq \(~\overline{\quad}\) \[ A_{\text{END}} (L, I, D) = \frac{L^3 \text{ID}^5}{1 - L \text{ID} (1+L)} \]

\[ \therefore A_{\text{END}} (L=1, I=1, D) = \frac{D^5}{1 - 2D} \]

\[ = D^5 \left( 1 + 2D + 4D^2 + 8D^3 + \cdots \right) \]

\[ \boxed{d \text{ face} = 5} \]
Eq \quad a(b + c) = ab + ac

\begin{align*}
\downarrow & \\
1 \text{ multiplication} & \rightarrow \\
1 \text{ addition} & 2 \text{ multiplications} \\
& 1 \text{ addition}
\end{align*}
\[ \alpha(x, \omega) = \sum_{j, l} f(x, j, \omega) g(x, l) \]

\[ \beta(j) = \sum_{x, \omega, z} f(x, j, \omega) g(x, z) \]

Variables \( x, j, \omega \) take on values from a common alphabet \( \mathcal{A} \) of size \(|\mathcal{A}| = 3\).
the functions $f(\cdot)$, $g(\cdot)$ are real valued.

\[ \alpha(x, w) = \sum_{y_1, z_1} f(x, y_1, w) g(x, z_1) \]

# of computations required

\[ = q \sum_{q} \left( q^2 + (q - 1) q^2 \right) \]

\[ = 2 q^2 - q^2 \]
\[ \beta (f) = \sum_{x \in Z} f(x, y, w) g(x, z) \]

# of computations

\[ = q \left\{ q^3 + (q^3 - 1) \right\} \]

\[ = 2q^3 - q. \]

Invoking the distributive law:

\[ d(x, w) = \sum_{j \in Z} f(x, j, w) g(x, z) \]
\[
\mathcal{L}(x, w) = f'(xw) \cdot g'(x)
\]

\[
f' = q^2(q-1) \quad g' = q(q-1)
\]

\[
\lambda = q^2
\]

Total \# of computations

\[
= q^3 - q^2 + q^2 - q + q^2
\]

\[
= q^3 + q^2 - q
\]
\[ \beta'(\mathbf{x}) = \sum_{x \in \mathbf{X}} \sum_{w} f'(x \cdot w) \cdot g'(x \cdot z) \]

\[ = \sum_{x} \left( \sum_{w} f'(x \cdot w) \right) \cdot \left( \sum_{z} g'(x \cdot z) \right) \]

\[ = \sum_{x} f'(x \cdot y) \cdot g'(x) \]

**Total # of Computations**

\[ = q^2 (q-1) + q (q-1) + q (q + q - 1) \]

\[ = q^3 - q + q^2 - 3 + 2q^2 - q \]

\[ = q^3 + 2q^2 - 2q \]
DETOUR: Semirings

Defn. A semiring \((R, +, \cdot)\) is a set \(R\) along with 2 operations \((+, \cdot)\) under which the following properties hold:

1. Under addition: \((R, +)\)
   - Closure
   - Associative
   - Identity EL.

   Under multiplication: \((R, \cdot)\)
   - Associative
   - Commutative
(note that the additive inverse is not required)

(ii) Under multiplication \((R, \cdot)\)
must satisfy:

- **Closure**
- **Identity**
- **Associative**
- **Commutative**

(note that multiplicative inverses need not exist)

(iii) \((R, +, \cdot)\) must satisfy the
distributive law:

\[ a (b + c) = ab + ac. \]
Every commutative ring with identity is a semiring.

In particular, every field is a semiring.

\[ \mathbb{R} \subset \mathbb{F}_2 \]

\[ \mathbb{F}[x] \]

\[ \mathbb{Z} \]
Eq \( (\mathbb{R}, +, \cdot) = ([0, \infty), +, \cdot) \)

\[
\begin{array}{c}
0 \\
1 \\
\mathbb{R}
\end{array}
\]

\[
\begin{array}{c}
(R, +) \\
(R, \cdot)
\end{array}
\]

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓
- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓
DIST LAW
lec 22  The M P F Problem

recap

- defce
- \{\text{distributive law} \}
- \# \text{ of computations}
- \{\text{semiring} \}
- examples.
Further examples of semiring

Eq 3
$\text{Eq 3 } (R, \text{ MAX }, \cdot) = (\mathbb{R}, \text{ MAX }, \cdot) \cup (\mathbb{R}, \cdot) \cup (\mathbb{R}, \text{ MAX }, \cdot) \cup (\mathbb{R}, \cdot) \cup (\mathbb{R}, \cdot)$

- $\text{CL}$ ✓
- $\text{ASSOC}$ ✓
- $\text{ID. EL}$ ✓
- $\text{COMM}$ ✓
identity under MAX

$$\text{MAX} \{ e, a \} = e$$

$$e = 0$$

DISTRIBUTIVE LAW?

$$a (b + c) = ab + ac$$

$$a \left( \text{MAX} \left\{ b, c \right\} \right) = \text{MAX} \left\{ ab, ac \right\}$$
\( 1 \in_{\text{Eq}^+} (R, \text{MIN}, +) = (-\infty, \infty], \text{MIN}, +) \)

\( \text{SUM} \)

\( \text{SEMIRING} \)

\( (R, \text{MIN}) \)

\( (R, +) \)

- CL
- ASSOC
- ID. EL
- COMM
- CL
- ASSOC
- ID. EL
- COMM
\[ \min \{ e, a \} = a \quad e = +\infty \]

**Distributive Law?**

\[ a (b + c) = ab + ac \]
\[ a + \min \{ b, c \} = \min \{ a+b, a+c \} \]

Yes!
THE MPF PROBLEM
(marginalize a product function)

Setting:

\[ S = \{ 1, 2, \ldots, n \} \]

\[ X_S = \{ x_1, x_2, \ldots, x_n \} \]

\[ x_i \in A \quad \text{alphabet} \]
$|A_{i,j}| = q_i$

Subsets $S_j$ of $S$, $1 \leq j \leq M$

$S_j = \{ i_1, i_2, \ldots, i_{n_j} \} \subseteq S$

$X_{S_j} = \{ x_{i_1}, x_{i_2}, \ldots, x_{i_{n_j}} \}$

→ LOCAL DOMAINS
$A_{s_j} = \mathcal{S}$ alphabet from which $x_{s_j}$ is drawn

$|A_{s_j}| = v_{s_j}$

associated with each local domain

$x_{s_j}$ is a local kernel

$z_j (x_{s_j})$
$x : X \times J \rightarrow \mathbb{R}$

$\text{seming}$

Global kernel:

$$\beta (x_s) = \prod_{j=1}^{M} \alpha_j (x_{s_j})$$

$j$th objective function:

$$1 \leq j \leq M$$
\[
\beta_j (x_{sj}) = \sum \beta (x_s) \\
x \in S_c \subseteq S \setminus S_j
\]
\[ p(x_1) = \sum_{x_2} p(x_1, x_2) \]
Eq 1 (The 8-dimensional Walsh Hadamard Transform)

\[ F(x_4, x_5, x_6) = \sum_{x_1 x_2 x_3} f(x_1, x_2, x_3) (-1)^{x_1 x_4} (-1)^{x_2 x_5} (-1)^{x_3 x_6} \]
Alphabets $A_i$:

$$A_i = \{0, 1\}$$

Same for all $1 \leq i \leq M$

$|A_i| = 2$
This is thus an example of the MPF problem.

\[ S = \{1, 2, 3, 4, 5, 6\} \quad X = \{x_1, x_2, x_3, x_4, x_5, x_6\} \]

\[ R = \text{field of all real numbers} = \mathbb{R} \]

\[ S_1 = \{1, 2, 3\} \quad X_{S_1} = \{x_1, x_2, x_3\} \]

\[ S_2 = \{1, 4\} \quad X_{S_2} = \{x_2, x_4\} \quad x_1 \text{, } x_4 \]

\[ \varphi_1(x_{S_1}) = \sigma(x_1 x_2 x_3) \quad \varphi_2(x_{S_2}) = (-1) \]
\[ S_3 = \{ 2, 5 \} \quad x \quad S_3 = \{ x_2, x_5 \} \]
\[ S_4 = \{ 3, 6 \} \quad x \quad S_4 = \{ x_3, x_6 \} \]
\[ S_5 = \{ 4, 5, 6 \} \quad x \quad S_5 = \{ x_4, x_5, x_6 \} \]
\[ \chi_3 \left( x_{S_3} \right) = (-1) \]
\[ \chi_4 \left( x_{S_4} \right) = (-1) \]
\[ \chi_5 \left( x_{S_5} \right) = 1 \]
\[
\begin{bmatrix}
F(000) \\
F(001) \\
F(111)
\end{bmatrix}
= \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \[-1em]
\end{bmatrix}
\times
\begin{bmatrix}
f(000) \\
f(001) \\
f(010) \\
f(011) \\
f(111)
\end{bmatrix}
\]

\[\text{Walsh Hadamard matrix}\]
Eq. 2: \[
\begin{bmatrix}
7 & 4 & 2
\end{bmatrix}
\]
linear block code

\[
\begin{bmatrix}
1 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 1
\end{bmatrix}
\]
(3 x 7)
BSC:

\[
\begin{align*}
0 & \quad 1 - \epsilon \\
\epsilon & \quad \downarrow \\
1 & \quad 1 - \epsilon \\
\end{align*}
\]

\[
0 \times x \in \mathbb{C} \quad \downarrow \\
\epsilon \quad \downarrow \\
1 \quad \text{modulo-2 addition}
\]
Goal: Formulate the maximum likelihood codeword decoding problem as an example of MPE computation.
Lec 23

Further Examples of the MPF Problem

Recap

* completed discussion on semi-rings
* defined the MPF problem
* Examples
  — Fast Walsh Transform
Eq 2: \([7, 4, 2]\) linear block code

parity: 
check: 
\(n \times k \times d\)

\[
H = \begin{bmatrix}
1 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 1
\end{bmatrix}
\]

\((3 \times 7)\)

Our interest is in ML codeword decoding of this block code.
Define

$$F_\epsilon (x, \xi) = \max_{x_1 - x, \dot{\omega} - 1} \phi \left( \frac{y}{\omega} \right)$$

$$x_{\dot{\omega} + 1} - x_7$$

$$x \in \mathcal{C}$$
We will compute $F_i(x_i)$ for $i = 1, 2, \ldots, 7$ for $x_i \in \{0, 1\}$.

<table>
<thead>
<tr>
<th></th>
<th>$F_1(0)$</th>
<th>$F_1(1)$</th>
<th>$\Rightarrow$</th>
<th>$2$</th>
<th>$\boxed{8}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$F_2(0)$</td>
<td>$F_2(1)$</td>
<td></td>
<td>$1$</td>
<td>$\boxed{8}$</td>
</tr>
<tr>
<td></td>
<td>$F_3(0)$</td>
<td>$F_3(1)$</td>
<td></td>
<td>$\boxed{8}$</td>
<td>$6$</td>
</tr>
<tr>
<td></td>
<td>$F_4(0)$</td>
<td>$F_4(1)$</td>
<td></td>
<td>$2$</td>
<td>$\boxed{8}$</td>
</tr>
<tr>
<td></td>
<td>$F_5(0)$</td>
<td>$F_5(1)$</td>
<td></td>
<td>$-1$</td>
<td>$\boxed{8}$</td>
</tr>
<tr>
<td></td>
<td>$F_6(0)$</td>
<td>$F_6(1)$</td>
<td></td>
<td>$\boxed{8}$</td>
<td>$4$</td>
</tr>
<tr>
<td></td>
<td>$F_7(0)$</td>
<td>$F_7(1)$</td>
<td></td>
<td>$6$</td>
<td>$\boxed{8}$</td>
</tr>
</tbody>
</table>

The decoded code word is $[110\ 1101]$.
$$F_\hat{\epsilon} (x_\hat{\epsilon}) = \max_{x_1, \ldots, x_{\hat{\epsilon} - 1}} \left. p(y | \mathbf{x}) \right|_{\mathbf{x} \in \mathbb{R}}$$

$$= \max_{x_1, \ldots, x_{\hat{\epsilon} - 1}} \left. p(y | \mathbf{x}) \right|_{\mathbf{x} \in \mathbb{R}}$$
\[ \chi_{\mathbb{R}}(x) = \begin{cases} 1 & x \in \mathbb{R} \\ 0 & \text{else} \end{cases} \]

\[ \chi_{\mathbb{R}_2}(x) = \chi_{124}(x) \chi_{346}(x) \chi_{457}(x) \]

\[ H = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \]
\[
\chi_{124} (x) = \chi_{124} (x_1 x_2 x_4) = \begin{cases} 1 & x_1 + x_2 + x_4 = 0 \\ 0 & \text{else} \end{cases}
\]

\[
\chi_{3456} (x) = \chi_{3456} (x_3 x_4 x_5 x_6) = \begin{cases} 1 & x_3 + x_4 + x_5 + x_6 = 0 \\ 0 & \text{else} \end{cases}
\]

\[
\chi_{457} (x) = \chi_{457} (x_4 x_5 x_7) = \begin{cases} 1 & x_4 + x_5 + x_7 = 0 \\ 0 & \text{else} \end{cases}
\]
\[
F_{x_i}(x_{\hat{i}}) = \max_{\hat{i}} \prod_{\hat{i}=1}^{2} \phi \left( \frac{x_{\hat{i}} - \mu_{\hat{i}}}{\sigma_{\hat{i}}} \right).
\]

\[
\max_{x_1, \ldots, x_{\hat{i}-1}, x_{\hat{i}+1}, \ldots, x_T} \chi_{124} \left( \eta_1, \eta_2, \eta_4 \right).
\]

\[
\chi_{346} \left( \eta_3, \eta_4, \eta_6 \right).
\]

\[
\chi_{457} \left( \eta_5, \eta_6, 2 \eta_T \right).
\]
**Local Domains**

\[ \{ x_i \}_{1 \leq i \leq 7} \]

**Local Kernels**

\[ \phi \left( \{ x_i \}_{1 \leq i \leq 7} \right) \]

\[ \{ x_1, x_2, x_4 \} \times \{ x_1, x_2, x_4 \} \]

\[ \{ x_3, x_4, x_6 \} \times \{ x_3, x_4, x_6 \} \]

\[ \{ x_4, x_5, x_7 \} \times \{ x_4, x_5, x_7 \} \]
Eq. ML code-symbol decoding of the [7, 4, 2] code, proportional to

\[
\mathcal{P}(z, y) = \sum_{x_i} \mathcal{P}(x_i) \mathcal{P}(z, y | x_i)
\]

\[
x_i \in \mathbb{R} \quad y \in \mathbb{R}
\]

\[
\mathcal{P}(x_i) \mathcal{P}(z, y | x_i) \quad \mathcal{P}(x_i) \mathcal{P}(z, y | x_i) \quad \mathcal{P}(x_i) \mathcal{P}(z, y | x_i)
\]

\[
x_i \in \mathbb{R} \quad y \in \mathbb{R}
\]
\[ \sum_{\mathbf{n}} \prod_{j=1}^{\ell} p(J_j | n_j) \chi_{124} (n_1, n_2, n_4) \chi_{346} (n_3, n_4, n_6) \chi_{457} (n_4, n_5, n_7) \]

It is clear that this is once again an instance of the MPF problem. The computation this time however, is carried out in the sum product...
seming. The local domains as well as local kernels remain exactly the same.

\[ E_g \{ SML \text{ codeword decoding of convolutional codes.} \]
Finite State Machine Description

- Past 2 symbols
  - \( u_{k-2} \)
  - \( u_{k-1} \)

- Transition States:
  - Input = 0
    - 00 \rightarrow 00
    - 10 \rightarrow 10
    - 01 \rightarrow 01
  - Input = 1
    - 00 \rightarrow 00
    - 10 \rightarrow 10
    - 01 \rightarrow 01
\[ \{ u_k \} \rightarrow \text{Convln. Encoder} \rightarrow \text{Channel} \rightarrow \{ (1) \ (2) \ (n) \} \]

Assume a rate \( \frac{1}{n} \) for simplicity (convln code).

Goal: ML code word decoding, i.e.,
identifying the codeword $\mathbf{v}$ such that $p(\mathbf{y} | \mathbf{v})$ is a maximum. which is equivalent to identifying the message vector $\mathbf{y}$ which is such that $p(\mathbf{y} | \mathbf{v})$ is a max.
$\{u_k\}_{k=0}^{N-1}$

state sequence $\{s_k\}_{k=0}^{N-1}$

output sequence $\{z_k\}_{k=0}^{N-1}$

each $z_k$ is an $n$-tuple $z_k = (z_{k1}, \ldots, z_{kn})$

(In the example below, $N=4$)
Recap

- Cast the ML codeword decoding problem of the $[7, 4, 2]$ code as an MPF problem.
- Did the same for ML code-symbole decoding.
If the $[7,4,2]$ code began ML codeword decoding of a rate $\frac{1}{n}$ convolutional code
The above graph (known as a directed acyclic graph (DAG))

aka a Bayes network
tells us how to factor the joint
prob. dis. in:

\[
\prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{4}
\]

\[
\prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{4}
\]

\[
\prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{4}
\]

\[
\prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{4}
\]

\[
\prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{3} \prod_{i=0}^{4}
\]
\[ F_i(u_i) = \max_{u_i} \phi\left( \frac{y_i}{u_i} \right) \quad 0 \leq i \leq 3 \]

\[ \lambda \max_{u_i} \phi\left( \frac{y_i}{u_i} \right) \]

\[ = \max_{u_i} \phi\left( \frac{y_i}{u_i}, x \right) \quad \text{since all message vectors} \quad y_i \text{ are equally likely} \]
\[
\max_{\{u_i\}} \prod_{i=0}^{3} p(s_{i}) \prod_{i=0}^{3} p(u_i) \prod_{i=0}^{3} p(s_{i+1} | s_i, u_i)
\]

\[
\prod_{i=0}^{3} p(u_i) \quad 0 \leq i \leq 3
\]
We have thus formulated the ML code word decoding of a ConvIn code as an MPE problem.
The first step in attempting to solve (using the ADL) the MPF problem is to organize the local domains into a form of graph known as a junction tree.

Defn. A tree is a connected
graph in which there are no cycles.
Note: In any tree we have:

\[ \text{the \# of nodes} - \text{\# of edges} = 1 \]

In any tree there is a unique path between any two distinct nodes.
**Defn.** In the setting of the MRF problem, a junction tree is a graph whose nodes are in 1-1 correspondence with the local domains \( \{ X_\gamma \}_{\gamma=1}^m \) and where edges are drawn between nodes in such a way that
(a) The graph is a tree

(b)
For every node $X_{\bar{s}}$ on the unique path lying between nodes $X_{s_i}$ and $X_{s_e}$, it must be that $s_i \subseteq \mathcal{S} \wedge s_e$.
Eq 2 \[ [7, 4, 2] \text{ linear block code} \]

```
parity
- check
m x
```

\[ H = \begin{bmatrix}
1 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 \\
\end{bmatrix}
\]

(3 x 7)

Our interest is in \underline{ML codeword decoding} of this block code.
Local Domains

\( \{ x_i \}_{1 \leq i \leq 7} \)

Local Kernels

\( \phi \left( \tilde{x}_i \mid x_i \right) \)

\( \{ x_1, x_2, x_4 \} \subset \chi_{124} \left( x_1, x_2, x_4 \right) \)

\( \{ x_3, x_4, x_6 \} \subset \chi_{346} \left( x_3, x_4, x_6 \right) \)

\( \{ x_4, x_5, x_7 \} \subset \chi_{457} \left( x_4, x_5, x_7 \right) \)
Note: An alternative defn. of a "tree is a tree which when projected onto each individual variable also yields a tree."
projection onto $x_1$
projection onto x4
Qn: How does one construct a \textit{maximal tree}?

\textbf{Ans}: By constructing a maximal \textit{w.t spanning tree}

\textbf{Thm}: If $G$ is a graph whose nodes correspond to the local domains of an MPE problem, and which
is also a tree, then it must be that

\[ \text{edge weight} \leq \text{node weight} - n \]

\[ n \]

where \( n = \# \text{ of variables} \)

\[ = |s|. \]
Recap

* Formulating the decoding of a convolutional code as an MPF problem

* Defined "junction tree"
  - Example
  - Stated theorem
Then if $y$ is a graph whose nodes correspond to the local domains, and which
is an MEF problem, and which

\[ \frac{\partial}{\partial t} + \text{something} \]
is also a tree, then it must be that

\[
\text{edge weight} \leq \text{node weight} - \eta
\]

where \( \eta = \# \text{ of variables} \)

\( = |s| \).
If. The weight of a node associated to a local domain

\[ x_{s_i} = |s_i| \]

The edge weight of an edge connecting nodes associated to \( x_{s_i} \) and \( x_{s_j} \) is \( |s_i \cap s_j| \).
Node weight of the graph

\[ = \sum \text{sum of the node weights of the nodes in the graph.} \]

Edge weight of the graph

\[ = \sum \text{sum of the edge weights of edges in the graph.} \]
Thus edge wt = 9 ≤ node weight - 7 = 16 - 7 \nes \node weight of the graph = 16
edge weight of the graph = 9
\[ EW(y) = NW(y) - 1 \]

\[ \sum_{\text{edge}} \sum_{\text{weight}} \sum_{\text{node}} \sum_{\text{weight}} \]

\[ \sum_{\text{weight}} \]

\[ \Rightarrow \quad \# \text{ of edges} = \# \text{ of nodes} - 1 \]

\[ \# \text{ in } y_1 \] 

\( \text{(this is true since } y_1 \text{ is a tree) } \)
\[ NW(y_4) = 4 = \text{# of nodes in } y_4 \]
\[ EW(y_4) = 3 = \text{# of edges in } y_4 \]
\[ \therefore EW(y_4) = NW(y_4) - 1 \]
**Pf.** When \( y \) is in the tree we have

\[
E(W(y)) = \sum_{i=1}^{\hat{n}} E(W(y_i))
\]

\[
= \sum_{i=1}^{\hat{n}} (NW(y_i) - 1)
\]

\[
= \sum_{i=1}^{\hat{n}} NW(y_i) - \hat{n}
\]

\[
\therefore \boxed{E(W(y)) = NW(y) - \hat{n}} \quad (1)
\]
Diagram with nodes and connections:

- Node 1 connected to node 2 and node 3.
- Node 2 connected to node 5.
- Node 3 connected to node 346.
- Node 346 connected to node 6.
- Node 457 connected to node 5.
- The diagram includes numerical values:
  - Node 1: 4
  - Node 2: 4
  - Node 3: 4
  - Node 346: 4
  - Node 457: 4

Equation: $y_i = y_j = 4$
When \( y \) is not a \( j \)n tree, then at least one of the \( y_i \) will fail to be a tree (will be the union of \( >2 \) trees instead) and hence
\[
E W(y_i) < NW(y_i) - 1
\]

If we now proceed to argue as when deriving (1) we would end up with
\[ E_W(y) < NW(y) - \eta \]

The theorem follows.
This suggests that if the local domains can be organized into a \textit{tree}, then that \textit{tree} represents a maximal \textit{ut spanning tree} for the collection of local domains.

\underline{A maximal \textit{ut spanning tree} (MST) can be constructed using Prim's}
greedy algorithm which we will now illustrate.
\[ \alpha(x, w) = \sum_{j, g} \sum f(x \mid f, w) g(x_{12}) \]

\[ \beta(f) = \sum_{x, w, z} f(x, f, w) g(x_{12}) \]

\[ \frac{LD}{\left\{ x \mid f, w \right\}} \]

\[ \frac{LK}{\sum f(x \mid f, w)} \]

\[ \frac{LD}{\left\{ x \mid f \right\}} \]

\[ \frac{LK}{1} \]

\[ \frac{LD}{\left\{ x \mid f \right\}} \]

\[ \frac{LK}{w \times f \in A} \]

\[ |A| = 9 \]
\[ NW(y) = 8 \]
\[ EW(y) = 4 \]
\[ \text{diff} = 4 \]
\[ = \# \text{ of variables} \]

Thus this is the in fact of this example
Aside

It turns out that an edge connecting nodes $X_{s_i}$ and $X_{s_j}$ incur a cost $= q_s + q_{s_i} - q_{s_j}$. (2)
Eq 2 (The 8-dimensional Walsh–Hadamard Transform)

\[ F(x_4, x_5, x_6) = \sum_{x_1, x_2, x_3} f(x_1, x_2, x_3)(-1)^{x_1 x_4 + x_2 x_5 + x_3 x_6} \]
\[
\begin{align*}
S_1 & \ni \{x_1, x_2, x_3\} \\
S_2 & \ni \{x_1, x_4\} \\
S_3 & \ni \{x_2, x_5\} \\
S_4 & \ni \{x_3, x_6\} \\
S_5 & \ni \{x_4, x_5, x_6\}
\end{align*}
\]
\[
\text{LD} \quad \{x_1, x_2, x_3\}
\]

\[
\text{NW}(y) = 12 \quad \text{diff} = 8
\]

\[
\text{EW}(y) = 4 \quad \# \text{ of variables}
\]
Lec 26  
Message passing on the 
\text{In Tree}

Recap

* completed proof showing that a \text{jn tree} (if the local domains can be organized into a \text{jn tree}) is necessarily a maximal \text{spanning tree (MST)}
we would invoke Prim’s (greedy)
algorithm to construct a MST

- [Eq] - pair of simple computations
- Walsh transform

Eq 3  [7, 4, 2] code

\[
H = \begin{bmatrix}
11010000 \\
00110100 \\
00011001
\end{bmatrix}
\]
\[ \Phi(x_4 | \Xi) \propto \sum_{x_4} \prod_{j=1}^{\Xi} \phi(j | x_j) \chi_{124} (x_1 x_2 x_4) \chi_{346} (x_3 x_4 x_6) \chi_{457} (x_4 x_5 x_7) \]
Step 1: In solving (where possible) the MPF problem is to organize the local domains into a join tree.

Step 2: Pass messages along the edges of the join tree in accordance with some schedule.
$h_{ij}(x_{s_i} \cap x_{s_j}) = \sum_{x_{s_i} \setminus s_j} d_i(x_{s_i})$

$$\prod_{k \in N_i \setminus j} h_{ki}(x_{s_k} \cap x_{s_i})$$

$N_i$ = set of neighbours of $x_{s_i}$
Note: the message $h_{ij}(x_{s_i}, n_{s_j})$ passed on by a leaf node is simply the marginalization of its local band.
Eq 3. \([7_1, 4_1, 2]\) code

\[
H = \begin{bmatrix}
1 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 1
\end{bmatrix}
\]

xml codeword \(x\) = \((0, 1, 1, 1, 1, 0, 0)\)

\(e\) = \((0, 0, 0, 1, 0, 0, 0)\)

\(y\) = \((0, 1, 1, 0, 1, 0, 0)\)
Assume a BSC

\[ p(Y_1 | X_1) = \begin{bmatrix} p(Y_1 | x_1 = 0) \\ p(Y_1 | x_1 = 1) \end{bmatrix} \]

This is typical in that it is a vector representation of the fn.
\[ J = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} \]

\[ p(J_1|x_1) = \begin{bmatrix} p(J_1|x_1 = 0) \\ p(J_1|x_1 = 1) \end{bmatrix} = \begin{bmatrix} 1 - \xi \\ \xi \end{bmatrix} \]

Scaling by \((1-\xi)\) yields:

\[ \sigma = \begin{pmatrix} \xi \\ \xi \end{pmatrix} \]
In the case of the single-vertex problem, there is just a single objective function to be computed, so one orients all edges in the jun tree towards the corresponding local domain.
\[
\begin{bmatrix}
2 \Theta \\
1 + \theta^2
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 \\
0
\end{bmatrix}
\]

\[
\sum_x \phi(j_1 | x_1) \phi(j_2 | x_2)
\]

\[
x_1 \times x_2
\]

\[
\chi_{124} (x_1, x_2, x_4)
\]

\[
g(x_4)
\]
\[ \beta_4(x_4) = \begin{bmatrix} (2\theta)^3 \\ \theta (1+\theta^2)^3 \end{bmatrix} = \begin{bmatrix} 8 \theta^3 \\ 0 \end{bmatrix} \]

\[ \theta = \frac{\varepsilon}{1+\varepsilon} \]

\[
\begin{bmatrix}
2\theta \\
1 + \theta^2
\end{bmatrix}
\]

\[
\begin{bmatrix}
2\theta \\
1 + \theta^2
\end{bmatrix}
\]
Lec 27

Convolutional codes

Recap

- Final example of joint
  construction

- Message passing
  - Eq $[7,5,2]$ code decoding
A node is ready to compute its objective $f_i$ once it has received messages from all of its neighbors.
At this stage the node computes its objective function simply by computing the product of the incoming messages and the local kernel.

\[ \beta_j(x_j) = \alpha_j(x_j) \prod_{k \in N_j} \psi_{kj}(x_{s_k}, x_j) \]
\[ \beta_4(x_4) = \left[ (2\theta)^3 \right] \approx \left[ \theta (1+\theta^2)^3 \right] = \left[ 8\theta^3 \right] \]

\[ \theta = \frac{\varepsilon}{1-\varepsilon} \]
\[
\begin{bmatrix}
2\theta \\
1 + \theta^2
\end{bmatrix}
\begin{bmatrix}
2\theta \\
1 + \theta^2
\end{bmatrix}
\begin{bmatrix}
2\theta \\
1 + \theta^2
\end{bmatrix}
\begin{bmatrix}
2\theta \\
1 + \theta^2
\end{bmatrix}
\begin{bmatrix}
1 \\
0
\end{bmatrix}
\]

Schur product

\[
= \begin{bmatrix}
8\theta^3 \\
\theta(1 + \theta^2)^3
\end{bmatrix}^2
\begin{bmatrix}
8\theta^3 \\
0
\end{bmatrix}
\begin{bmatrix}
1 \\
0
\end{bmatrix}
\]

\[
\beta_4(x_4) \propto \beta\left(x_4\mid \frac{y}{2}\right)
\]
Since $q << 1$ (typically) the ML code-symbol decoder will decode $X$ to $1$.

From the example and the in the property it is apparent why marginalization at intermediate stages of message passing as determined by
the HDL is justified.

ML codeword decoding of the \([7,4,2]\) code using the HDL
\[ F_i(x_i) = \max \prod_{\hat{i}=1}^{\tau} \phi(j_{\hat{i}}|n_{i_{\hat{i}}}) \cdot \alpha x_i \]

\[ \max \chi_{124} (\eta_1, \eta_2, \eta_3) \]

\[ \chi_{346} (\eta_3, \eta_4, \eta_5) \]

\[ \chi_{457} (\eta_4, \eta_5, 2\eta) \]

\[ \text{Eq: decode } [7, 4, 2] \text{ code } \overline{J} = (01100) \]
\[ g(x_a) = \max_{x_1, x_2} \log \left( \frac{p(y_1 | x_1) \cdot p(y_2 | x_2) \cdot x_{12} \cdot (x_1 \times x_2 \times a)}{\theta} \right) \]
\[ g(0) = \max \left\{ 1.0, 0.1 \right\} = 0 \]

\[ g(1) = \max \left\{ 1.1, 0.0 \right\} = 1 \]

\[ L \]

\[ \{ x_j \} \quad j = 1 \]

\[ \{ x_1, x_2, x_4 \} \]

\[ \{ x_3, x_4, x_6 \} \]

\[ \sum_{j=1}^{L} \phi(x_j | x_j) \]

\[ \chi_{124} (x_1, x_2, x_4) \]

\[ \chi_{3+6} (x_3, x_4, x_6) \]
\[ F_4(x_4) = \begin{cases} \emptyset^3 & x_4 = 0 \\ \emptyset & x_4 = 1 \end{cases} \]

i. Seven the ML code word decoder
\[ \sum_{\text{all } s} X_s = 1 \]
for more details on the scheduling please see

The Generalized distributive law, S. M. Aji, R. J. McEliece
IEEE Trans. Infom. Theory
March 200
The complexity of the single-vertex implementation of the CDL

$$\sum_{\text{edges}} \left( q_i + q_j - q_k \right) \left( S_i S_j S_k \right)$$

(Additions \& multiplications)
Turns out that the complexity involved in computing the objective at all nodes is \( \Omega(n \log n) \) above by

\[
\log \left( \frac{\text{single-vertex complexity}}{\text{complexity}} \right).
\]
ML code - symbol decoding of a convolutional code
we consider the same convolutional
as before with the difference that we
are now interested in ML code-symbol
decoding.

\[ p(\eta_k | \mathcal{I}) \propto p(\eta_k, \mathcal{I}) = \sum \rho(\left\{ \eta_k \right\}_{k=0}^{3}, \mathcal{I}) \]

\[ = \sum_{\eta_k} \sum \left( \left\{ \eta_k \right\}_{k=0}^{3}, \left\{ \mathcal{I}_k \right\}_{k=0}^{3} \right) \]
= \sum_{n_k} \sum_{s_k} \sum_{s_{k+1}} \frac{3}{11} \phi(s_0) \phi(u_k) \phi(s_{k+1} | s_k, u_k) \phi(s_k | s_{k-1}, u_k)
Recap

Eg: Decoding \([f_{14}, 2]\)

(ML code - Symbol Decoding)

ML Code - Symbol Decoding

Of the convolutional code

(BCJR Algorithm)

formalizing the final step in the

CML - computing the objective function
ML code word decoding of the \([7, 4, 2]\) code

- Complexity (in terms of the number of operations needed) of implementing the ADL
  - Single-vertex
  - Bound for the all-vertex version of the ADL
ML code - symbol decoding of a convolutional code
we consider the same convolutional
as before with the difference that we
are now interested in ML codec-sym.
decoding.

\[ p(u_k | \mathbf{z}) \propto p(u_k, \mathbf{z}) = \sum_{\mathbf{z}} p\left(\{s_k\}_{k=0}^3, \mathbf{z}\right) \]

\[ = \sum_{u_k} \sum_{\mathbf{z}} p\left(\{s_k\}_{k=0}^3, [u_k]_3, \{j_k\}_{k=0}^3, \mathbf{z}\right) \]
= \sum_{n,k} \sum_{l} \phi(s_0) \left( \frac{3}{11} \phi(u_k) \phi(s_{l+1} \mid s_l, u_k) \right)_l \\
\overline{LD} \begin{cases} u \end{cases} \begin{cases} s_0 \end{cases} \begin{cases} s_{n+1} \end{cases} \begin{cases} s_n \end{cases} \phi(u_n) \quad 0 \leq i \leq 3

\overline{L} \begin{cases} \overline{k} \end{cases} \phi(s_0) \phi(s_{l+1} \mid s_l, u_k) \quad 0 \leq i \leq 3
\[ f_1(s_1) = \sum_{s_0, u_0} p(s_0) p(u_0) p(s_1|s_0, u_0) \]

\[ f_2(s_2) = \sum_{s_1, u_1} f_1(s_1) p(u_1) p(s_1|s_1, u_1) p(s_2|s_1, u_1) \]

\[ E_g \  \phi(D) = \begin{bmatrix} 1 + D + D^2 & 1 + D^2 \end{bmatrix} \]
$f_1(s_1)$

$S_1$

$S_2$

$11$ $11$

$01$ $01$

$10$ $10$

$00$ $00$
\[ f_2(s_2) = \sum_{s_1} f_1(s_1) \sum_{u_1} \phi(u_1) \phi(y_1|s_1,u_1) \phi(s_2|s_1,u_1) \]

\[ \uparrow(s_2, s_1) \]

is \( \{0,1,1\} \)-valued function.
\[
\begin{bmatrix}
    f_2 \circ s_2 \\
    f_2 \circ 10 \\
    f_2 \circ 01 \\
    f_2 \circ 11 \\
\end{bmatrix}
\begin{bmatrix}
    * & 0 & * & 0 & \\
    * & 0 & * & 0 & \\
    0 & * & 0 & * & \\
    0 & * & 0 & * & \\
\end{bmatrix}
\begin{bmatrix}
    \uparrow (s_2, s_1) \\
    \downarrow s_1 \\
\end{bmatrix}
\begin{bmatrix}
    f_1 \circ 00 \\
    f_1 \circ 10 \\
    f_1 \circ 01 \\
    f_1 \circ 11 \\
\end{bmatrix}
\]
\[ g_3(s_3) = \sum_{u_3} \sum_{s_4} \phi(y_3|s_3, u_3) \phi(u_3) \cdot \phi(s_4|s_3, u_3) g_4(s_4) \]

\[ = \sum_{s_4} g_4(s_4) \left\{ \sum_{u_3} \phi(u_3) \phi(y_3|s_3, u_3) \right\} \]

\[ \Delta(s_3, s_4) \]
\[ q_2(s_3) \quad \Delta(s_3, s_4) \quad q_4(s_4) \]

\[
\begin{bmatrix}
\end{bmatrix}
= \begin{bmatrix}
\end{bmatrix}
\begin{bmatrix}
\end{bmatrix}
\]

**Conclusion:** Both forward and backward recursions correspond to matrix multiplication.
\[ p(u_k | y) = p(u_2) \geq \sum_{s_2} p(y_2 | s_2, u_2) f_2(s_2) \]

\[ \geq \sum_{s_3} q_3(s_3) p(s_3 | s_2, u_2) \]

This concludes our discussion of the BCJR algorithm.
lec 29  LDPC codes

Recap

* completed discussion of the BCJR algorithm
Remark: The Viterbi algorithm can also be recovered using the CDL: one simply operates in the max-product semiring in place of the sum-product semiring. (In the algorithm the essential difference lies in replacing...
the summation operation by the "max-of" operation.
LDPC codes
(low-density parity-check codes)

Let \( C \) be a \([n, k, d]\) linear block code. Typically, when \( n \) is large, the codes of practical interest have

\[
\frac{k}{n} = \rho, \quad 0 < \rho < 1
\]

and \( \frac{\text{dmin}}{n} = \delta, \quad 0 < \delta < 1 \).
\[ H = \begin{bmatrix} \end{bmatrix} \]

\((n-k \times n)\)

Thus in a typical f.c. m×n, the # of entries is of order \(n^2\) \((\Theta(n^2))\) and since the entries
are typically equally likely to be either 0 or 1, the # of 1's would be on the order of $n^2$ as well.
However, in the case of an LDPC code, the number of 1's in $H$ is of order $n$.

A second difference in the case of LDPC codes is that the rows of $H$ need not necessarily be linearly independent.
However we still require that
the rows of $H$ generate the dual
code $\mathbb{R}^+$. This can be used to
show that once again this implies that
the null space $\ker H$ is precisely the
original code $C$.

A $(d_v, d_c)$-regular LDPC code
is one in which each row $J^t$ has $d_c$ 1's and each column $J^t$ contains $d_v$ 1's. 

Let $H = \begin{bmatrix} d_c \text{ 1's} \\ \vdots \end{bmatrix} \in \mathbb{R}^{m \times n}$ with $m \leq n$.

\[ \kappa_2(H) \leq m \quad (2) \]
i. \( \dim \left( \eta(H) \right) \geq n - m \)

ii. \( \dim (E) \geq n - m \)

iii. \( k \geq n - m \)

\[ \frac{k}{n} \geq 1 - \frac{m}{n} \quad \text{(1)} \]

\[ \frac{k}{n} \geq 1 - \frac{d \omega}{d\lambda} \quad \text{(3)} \]

follows from (1) and (2)
Tanner graph
(an example of a bipartite graph)

check nodes

degree on the right = 6

Variable nodes

degree on each node on the left = 3
This is an instance of a $(d_v = 3, d_c = 6)$ regular code.
1. \[ \frac{k}{n} \geq 1 - \frac{dV}{dc} = 1 - \frac{3}{6} = \frac{1}{2} \]

\[
\therefore \text{rate (R)} \geq \frac{1}{2}
\]

Note that the \((dV, dc)\) constraint implies that the total \#f is in the p.c. \(mx = n \cdot dV = m \cdot dc\)

and hence is \(f\) order \(n\) (since \((dV, dc)\) are fixed and
independent of \( n \)

The graphical, message-passing nature of the decoding algorithm in the case of an LDPC code implies that the complexity of the decoding algorithm is proportional to the \# of edges in the Tanner
The graph of the code and hence is linear (!!!) in the block length of the code.
Tanner graph of the $\sum_1^2 [7,4,2]$ code

Note: This code is not \((d_v, d_c)\) regular.
Recap

* introduced LDPC codes
  - $(d_v, d_c)$ regular code
  - rate
  - Tanner graph
  - decoding complexity
Tanner graph

Variable nodes

Check nodes
Computational Tree
A path in the graph (Tanner graph) is a directed sequence of directed edges

\[ e_1 \rightarrow e_2 \rightarrow \cdots \rightarrow e_k \quad \text{s.t.} \quad \forall \]

\[ e_i = (u_i, u_i'), \quad \text{then} \quad e_{i+1} = (u_{i+1}, u_{i+1}') \]
and \( u_{i+1} = u_{i+1} \).

The length of the path = \# of directed edges along the path.

Given two nodes in the graph, we will say that the 2 nodes are at a distance \( d \) if they are connected by a path of length \( d \), but not by a
path of length \( < d \).

\[ N^d_u = \text{nbhd of node } u \text{ to depth } d \]

= the induced subgraph consisting of all nodes reached and all edges traversed by paths of length at most \( d \) and starting from \( u \).
Note that

\[ u_1 \in N_{u_2} \iff u_2 \in N_{u_1} \]

If \( \overrightarrow{e} = (v, c) \) then the undirected neighbourhood to depth \( d \) of \( \overrightarrow{e} = N_d u \cup N_d c \)

\[ e \in N_{e'} \iff e' \in N_{e} \]
The directed nbhd to depth $d$ of edge $e = (v, c)$ denoted by $N^d e = \text{the induced subgraph containing all edges and nodes on paths } e_1, e_2, \ldots, e_d \text{ starting from } v$, but with $e_1 \neq e^2$. 


Channel Models

BSC

\[ X_t = 1 \]

\[ X_t = (-1) \]

\[ u_t \in \{0, 1\} \]

\[ J_t = x_t \cdot z_t \]

\[ z_t \in \{\pm 1\} \]

\[ p_{z_t}(1) = 1 - \epsilon \]

\[ p_{z_t}(-1) = \epsilon \]

\[ J_t \]
\[ p_{Y_t|x_t}(\frac{y}{x}) = p_{X_t}(\frac{y}{x}) \]

\[ = p_{Y_t|x_t}(-x'|-x) \]

Known as the channel symmetry condition (as applied to the BSC)
Binary-Input AWGN Channel

\[ x_t \in \{ \pm 1 \} \]

\[ j_t = x_t + n_t \]

\[ n_t \sim \mathcal{N}(0, \sigma^2) \]
Can write:

\[ J_t = x_t^2 \]

\[ z_t \sim N(1, \sigma^2) \]

\[ z_t \text{ independent of } x_t \]

\[ y_t | x_t \sim \mathcal{N} \left( \frac{y}{x} \right) \]

\[ = \frac{1}{2} \left( \frac{y}{x} - \frac{-1}{2\sigma^2} \right) \]

\[ = \frac{1}{2} \left( \frac{y}{x} - \frac{-1}{2\sigma^2} \right) = \frac{1}{2} \left( \frac{y}{x} \right) \]
(called the channel symmetry condition)

Message-Passing Terminology

channel input
Lec 31: Gallager Decoding Algorithm

Recap

* Terminology relating to LDPC codes:
  - edges, paths, nbhds
  - channel models
(called the channel symmetry condition)

Message-Passing Terminology

channel input

\in \Theta

\in M

\in M
\( \Theta \) = output alphabet of the channel

\( m \) = the common alphabet employed to pass messages from variable node to check nodes or vice versa.
\( \psi : \emptyset \rightarrow \mathcal{M} \) \{ the initial message map \}

\( \forall c \in \mathcal{M} \) \( d_{c-1} \)

\( \psi : \mathcal{M} \rightarrow \mathcal{M} \)

\( \forall v \in \mathcal{M} \) \( d_{v-1} \)

\( \psi : \emptyset \times \mathcal{M} \rightarrow \mathcal{M} \)

\( l = \# I \) \{ the iteration \}
Assumptions concerning message passing at a variable check node:

\[
\psi^{(0)}(b \cdot m) = \begin{cases} 
\psi^{(0)}(m), & b \in \{\pm 1\} \\
0, & m \notin \mathcal{M}
\end{cases}
\]

\[
\psi^{(2)}(b \cdot m_0, b \cdot m_1, \ldots, b \cdot m_d) = b \cdot \psi^{(0)}(m_0, m_1, \ldots, m_d) 
\]
\[ \chi_c^{(e)}(b_1, m_1, b_2, m_2, \ldots, b_{d_c - 1}, m_{d_c - 1}) \]

\[ = \left[ \prod_{j=1}^{d_c - 1} b_j \right]^{(e)} \chi_c^{(e)}(m_1, m_2, \ldots, m_{d_c - 1}) \] (3)

\[ b_j \in \{ \pm 1 \} \]

\[ m_j \in \mathbb{M} \cdot \]

Equations (1) - (3) are known as the
variable and check-node symmetry conditions.
channel

Input

Output

The drawing above represents a neural network architecture. The input layer is connected to multiple hidden layers, which in turn connect to the output layer. Each node in the network processes information and passes it to the next layer. The diagram is used to visualize the flow of data through the network.

As to the value of C, we will use the sign to indicate an indication.

The message to the user is as follows:
The message along an edge is said to be in error if its sign is not the true sign of the associated code symbol.

We will now proceed to show that the number of incorrect messages passed along the edges of the Tanner graph during each iteration is independent
Let the transmitted codeword (deferred to Lec 32) be

Gallager Decoding Algorithm A

\[ (\theta = \{ \pm 1 \}, \mathcal{M} = \{ \pm \bar{m} \}) \]

\[ \mathcal{U}_{\theta} (m) = m \]
\( \psi_v (m_0, m_1, \ldots, m_{d-1}) = \begin{cases} -m_0 & \text{if } m_j = -m_0 \text{ all } 1 \leq j \leq d-1 \\ m_0 & \text{else} \end{cases} \)

\( \psi_c (m_1, m_2, \ldots, m_{d-1}) = \prod_{j=1}^{d-1} m_j \)

Qn: How well does this algorithm perform?
we will evaluate performance by carrying out "density evolution" by which we mean that we will estimate the # of incorrect messages passed during each iteration iteratively!

We assume that the \( \underline{1} \) codeword was transmitted.
\( p^{(0)}_1 = \begin{cases} \text{prob that the message passed by a variable node during the } \ell \text{th iteration} = +1 \\ \text{similarly for } -1 \end{cases} \)

\( q^{(0)}_1 = \begin{cases} \text{prob that on the } \ell \text{th iteration the check node message } = 1 \\ \text{similarly for } -1 \end{cases} \)
\[
\begin{align*}
\phi_1^{(0)} &= 1 - \varepsilon \\
\phi_{-1}^{(0)} &= \varepsilon \\
\end{align*}
\]

Note that by symmetry the probability mass for messages along an edge only depends upon the number of the iteration (i.e., on \( \varepsilon \)) as well as the direction of the message (variable node to check node or vice versa).
This message is only a subset of \( J_9 \) - \( J_{16} \). 

These messages are independent!

This message is a subset of \( J_9 \) - \( J_{16} \). 

\( l = 2 \)

\( l = 1 \)
\[ p_0 = 1 - e^{-a} \]

\[ p_0 = 1 - e^{-a} \]

(\text{uses independence})

\[ p(k) = (1 - e^{-a})^k \cdot e^{-a} \]

...
Lec 32 \[ \text{BP Decoding of LDPC Codes} \]

(BP = Belief Propagation)

Recap

* Alphabets arising in message passing
* Independence assumptions
* Gallager decoding algorithm
\[ \prod_{j=1}^{l-1} \hat{H}_j \]
\[
\begin{align*}
\vdots
\quad \prod_{d_c = 1}^{d_c-1} f_{d_c} = \prod_{j=1}^{d_c-1} \prod_{i=1}^{d_c-1} f_{i,j} \\
\therefore \quad \prod_{d_c = 1}^{d_c-1} f_{d_c} = \prod_{j=1}^{d_c-1} \prod_{i=1}^{d_c-1} f_{i,j} \\

\{ h_1 (f_{j,1} = 1) = \phi_{1,1} \}^{(d-1)} \\
\{ h_2 (f_{j,1} = -1) = \phi_{1,-1} \}^{(d-1)} 
\end{align*}
\]
\[ \begin{align*}
\left[ \varphi_{1} - \varphi_{-1} \right] &= \left[ \begin{array}{c}
\varphi_{1} - \varphi_{-1} \\
\end{array} \right] \\
\Rightarrow 1 - 2 \varphi_{-1} &= \left[ \begin{array}{c}
1 - 2 \varphi_{-1} \\
\end{array} \right] \\
\Rightarrow \varphi_{-1} &= \frac{1}{2} \left\{ 1 - \left[ \begin{array}{c}
1 - 2 \varphi_{-1} \\
\end{array} \right] \right\} \\
\end{align*} \]

\[ \cdots \ (2) \]
\[
\varphi^{(e)} = \varphi^{(0)} \sum_{d_0-1} (1 - 2^{(e)} z_{-1}) \\
+ \varphi^{(0)} \left[ 2^{(e)} z_{-1} \right]^{d_0-1} - (3)
\]
From (2) and (3) we get:

\[
\alpha_{-1} = \frac{\phi_{-1}^{(0)}}{\phi_{-1}^{(0)}} \left\{ 1 - \left[ \frac{1}{2} \left( 1 + \left[ 1 - 2 \frac{\phi_{-1}^{(1)}}{d_{c-1}} \right]^2 \right) \right] \right\} \]

\[
+ \left[ 1 - \frac{\phi_{-1}^{(0)}}{\phi_{-1}^{(0)}} \right] \left[ \frac{1}{2} \left[ 1 - \left[ 1 - 2 \frac{\phi_{-1}^{(1)}}{d_{c-1}} d_{v-1} \right] \right] \right] \]

\[
\text{this is the desired expression for density evolution}
\]
(can be shown that)

(a) $\psi_1^{(e)}$ increases monotonically with increase in $\psi_1^{(0)}$

(b) when $\psi_1^{(0)}$ is below a certain threshold, $\psi_1^{(e)} \to 0$ as the # of iterations increases.
when \((d_u, d_c) = (3, 6)\)

then designed rate satisfies:

\[
\frac{k}{n} > 1 - \frac{3}{6} = \frac{1}{2}
\]

and the threshold exhibited by Gallager decoding algorithm A is

\(
\epsilon = 0.04
\)

(in comparison channel capacity)
dictates that we should be able to operate provided \( \varepsilon \leq 0.11 \)
We will now proceed to show that the number of incorrect messages passed along the edges of the Tanner graph during each iteration is independent of the transmitted codeword.

\[ J_t = x_t z_t \]

\[ X_t Y_v (z_t) \]
(because the parity check)

\[ x = \sum_{c=1}^{d-1} x \]

\[ (c \times 2^a \pmod{2}) \]

\[ x \]

\[ 1 \]

\[ x \]

\[ 1 \]

\[ x \]
Channel Symmetry Assumption

\[ p(j_t / x_t) = p(-j_t / -x_t) \] (4)

\[ x_t \xrightarrow{\text{chan.}} j_t \]

\[ j_t = x_t \cdot z_t \]

Define \[ j_t = x_t \cdot z_t \]

\[ x_t \in \{ \pm 1 \} \]

\[ x_t \in \{ \pm 1 \} \]

\[ \sum \text{we will now show that } x_t \text{ is} \]

\[ \sum \text{independent of } z_t. \]
\[ p_{2|\lambda}(z_t \mid \nu_t = 1) = p_{y|\chi}(z_t \mid x_t = 1) \]

\[ p_{2|\lambda}(z_t \mid \nu_t = -1) = p_{y|\chi}(-z_t \mid x_t = -1) \]

\[ = p_{y|\chi}(z_t \mid x_t = 1) \]

by (4)

\[ \therefore p_{2|\lambda}(z_t \mid \nu_t) = p_{z}(z_t) \]
Recap

* Gallager’s decoding algorithm
  - density evolution

* completed proof showing that assuming check node symmetry
  variable node channel output
one can assume for the purpose of estimating error probability that the all-1 code word was transmitted.
Objective fn.

\[ \beta_i(x_i) \]

\[ \alpha \beta(x_1 \mid x_A, x_B, x_C) \]

\[
\begin{bmatrix}
1 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 \\
\end{bmatrix}
\]
$p(x_4 | j_3, j_6, x_B)$

Current belief about the prob of $x_4$
Tanner graph

We assume that the nbhd of every edge is tree-like to depth $2\varepsilon$.
This is the joint tree that one would obtain if one posed the problem

\[ P(x_{21} | J_1 J_2 - J_{21} ) \]

\[ \chi_A \chi_B - \chi_I \]
of computing $p(x_{21} \mid z_1, z_2, z_{21})$ as an MRF problem.

Hence if we pass messages along the edges of the Tanner graph in exactly the same manner as in the case if messages passed along the edges of the tree associated with the example...
[7+2] block code, then when edge nbhds are tree-like, the messages passed along the edges of the Tanner graph can be interpreted as conditional beliefs.
BP-based message passing in more detail:

\[ x_t \in \{\pm 1\} \]

\[ u_t \]

\[ x_t = \langle 1 \rangle \]

\[ x_t \mid E_0 \]

\[ x_t \mid E_1 \]

\[ x_t \mid E_2 \]

\[ x_t \mid E_n \]
Under the ADL we have:

\[ p \left( x_t \mid u^t, E^t \right) \propto \prod_{j=0}^{d-1} p \left( x_t \mid E_{j \bar{t}} \right) \]

\[ \Rightarrow \frac{p \left( x_t = 1 \mid u^t, E^t \right)}{p \left( x_t = -1 \mid u^t, E^t \right)} = \prod_{j=0}^{d-1} \frac{p \left( x_t = 1 \mid E_{j \bar{t}} \right)}{p \left( x_t = -1 \mid E_{j \bar{t}} \right)} \]
\[
\mathbb{P}(u_t = 0 | U \cup E_i) = \prod_{j=0}^{d_v-1} \frac{\mathbb{P}(u_t = 0 | E_j)}{\mathbb{P}(u_t = 1 | U \cup E_i)}
\]

\[
\ell_j = \ln \left\{ \frac{\mathbb{P}(u_t = 0 | E_j)}{\mathbb{P}(u_t = 1 | U \cup E_i)} \right\}
\]

\[
E_d = \bigcup_{i=0}^{d_v-1} U \cup E_i
\]
Taking logs on both sides of \( \theta \) gives us:

\[
\ln \theta_v = \sum_{i=0}^{d_v-1} \ln \theta_i
\]
\[ \alpha \frac{p(X_{l_c} | U \cup E_1)}{\prod_{i=1}^{d_c} X_i} \]

\[ \prod_{i=l_c}^{L} p(X_{d_c-1} \mid E_{d_c-1}) \]
\[ p(x_{d_c} | u E_J^c) \propto \sum_{x \sim x_{d_c}} \prod_{j=1}^{d_c-1} p(x_j | E_j^c) \]

\[ \sum_{u_{d_c}} p(u_{d_c} | u E_J^c)(-1) \]

\[ u_{d_c} = 0 \]

\[ \sum_{E_1} \sum_{u_{d_c}} \prod_{j=1}^{d_c-1} p(u_j | E_j) \]

\[ u_{d_c} = 0 \]

\[ u_{d_c} = u_{d_c} \]

\[ \sum_{j=1}^{d_c-1} u_j = u_{d_c} \]
\[ = \sum_{u_1, \ldots, u_{l_c}} \prod_{j=1}^{l_c-1} \prod_{e_j} \prod \{ u_j \mid E_j \} \]

\[ \cdot \sum_{u_{l_c}} \prod_{j=1}^{l_c} \prod_{e_j} \prod \{ u_j \mid E_j \} \]

\[ \therefore \quad p(u_{l_c} = 0 \mid E_{l_c}) - p(u_{l_c} = 1 \mid E_{l_c}) \]

\[ = \prod_{j=1}^{l_c-1} \frac{p(u_{l_c} = 0 \mid E_j)}{p(u_{l_c} = 0 \mid E_j) + p(u_{l_c} = 1 \mid E_j)} \]

\[ = \prod_{j=1}^{l_c-1} \frac{p(u_j = 0 \mid E_j)}{p(u_j = 0 \mid E_j) + p(u_j = 1 \mid E_j)} \]
In terms of log-likelihood ratios, reduces to:

\[
\frac{\ell_{dc}}{\ell_{dc} + 1} = \prod_{j=1}^{c-1} \frac{\ell_{j}}{\ell_{j} + 1}
\]

\[
\therefore \quad \text{tanh} \left( \frac{\ell_{dc}}{2} \right) = \prod_{j=1}^{c-1} \text{tanh} \left( \frac{\ell_{j}}{2} \right)
\]
Recap

- Discussion of BP decoding
  - LDPC codes and relation to message passing along a junction tree
  - Messages (beliefs) expressed in terms of LLRs
Log-Likelihood Ratios (LLR) at input and output of a variable node.

Note: variable node symmetry condition is met.
\[
\tanh\left(\frac{d_c}{2}\right) = \prod_{j=1}^{l_c-1} \tanh\left(\frac{l_i}{2}\right)
\]

(2) \quad l_{c-1}

On the other hand,

\[
\hat{d}_c = 2 \tanh^{-1} \left\{ \prod_{j=1}^{l_c-1} \tanh\left(\frac{l_i}{2}\right) \right\}
\]
Can verify that if \( \ell_j = \ell_j \), \( b_j \in \{ \pm 1 \} \), then

\[
\ell_c = 2 \tan^{-1} \left\{ \frac{\prod_{j=1}^{d_c-1} \tanh \left( \frac{\ell_j \cdot b_j}{2} \right)}{} \right\}
\]

\[
= \left( \prod_{j=1}^{d_c-1} b_j \right)^{-1} \left( \prod_{j=1}^{d_c-1} \tanh \left( \frac{\ell_j}{2} \right) \right)
\]

and hence the check-node symmetry condition is also met.
With this we are done with the description of BP-based message-passing decoding of LDPC codes.

We now turn our attention to performance analysis. It is to be carried out using density evolution.
In carrying out performance analysis, we assume that 1 was transmitted.
Replace \( \tanh \left( \frac{\ell}{2} \right) \) by \((x, y)\)

Where:

\[ x = \text{sign} \left( \ell \right) \]

\[ y = -\ln \left| \tanh \left( \frac{\ell}{2} \right) \right| \]

Keeps track of the sign of \( \tanh \left( \frac{\ell}{2} \right) \)

Keeps track of the magnitude of \( \tanh \left( \frac{\ell}{2} \right) \)
\[ \operatorname{sgn}(l) = \begin{cases} 0 & l > 0 \\ 1 & l < 0 \end{cases} \]
Thus
\[
\tanh \left( \frac{b \cdot c}{2} \right) = \prod_{j=1}^{l_c-1} \tanh \left( \frac{b_j \cdot c}{2} \right)
\]
is equivalent to:
\[
(x_c, y_c) = \left( \sum_{i=1}^{l_c-1} x_{i \mod 2}, \sum_{i=1}^{l_c} y_i \right)
\]
\[ \tanh\left(\frac{\ell}{2}\right) \]

\[ -\ln \left(\left|\tanh\left(\frac{\ell}{2}\right)\right|\right) \]

\[ |\tanh\left(\frac{\ell}{2}\right)| \]

\[ \text{sgn}(\ell) \]
\[ y : \Theta \to \mathcal{M} \]

\[ y : \Pi_0 \to \Pi_1 \]

\[ y : \Theta \times \mathcal{M} \to \mathcal{M} \]

\[ y : \Pi_0 \times \Pi_1 \to \Pi_1 \mathcal{M} \]

Corresponding map in terms of density functions.
\[ \psi_c^{(e)} : \Phi_{c-1}^e \rightarrow \Phi \]

\[ \forall \gamma^{(e)}_c : \prod_{m} \rightarrow \prod_{m} \]

(at a check node)
Assume that \( l_0 \) was transmitted.

The \( l_i \)'s are random since they are a function of the particular channel realization. Consider 2 rounds.
1 message passing and assume the Tanner graph to be such that the neighborhood of every node to depth 2 is tree-like (as in figure on page following). Then the different $h_i$, $1 \leq i \leq d_v - 1$ are linearly
independent as they are functions $f$ disjoint subsets $F_i$ receive variables $\{ x_j \}_{j=1}^n$. 
Scenario where the nbd is tree-like.
\[ l_v = \sum_{i=0}^{d_v-1} l_i \]

\[ l_v = \sum_{i=0}^{d_v-1} l_i \]

\[ e^{-j\omega l_v} = \prod_{j=0}^{d_v-1} e^{-j\omega l_j} \]

\[ e^{-j\omega l_v} = \prod_{j=0}^{d_v-1} e^{-j\omega l_j} \]
and hence by symmetry,

\[
\prod_{\ell} \left( 1 - \mathbf{e}^\ell_{1} \right) \prod_{\ell} \left( 1 - \mathbf{e}^\ell_{1} \right) \prod_{\ell} \left( 1 - \mathbf{e}^\ell_{1} \right) \prod_{\ell} \left( 1 - \mathbf{e}^\ell_{1} \right) \prod_{\ell} \left( 1 - \mathbf{e}^\ell_{1} \right) \prod_{\ell} \left( 1 - \mathbf{e}^\ell_{1} \right) \prod_{\ell} \left( 1 - \mathbf{e}^\ell_{1} \right) \prod_{\ell} \left( 1 - \mathbf{e}^\ell_{1} \right)
\]

this is a description of density evolution at a variable node.
\[ y = -\ln \left| \tanh \left( \frac{\ell}{2} \right) \right| \]

\[
\begin{align*}
\phi (0,y) &= \frac{1}{\sinh(y)} \cdot \left( -\ln \tanh \left( \frac{y}{2} \right) \right) \\
\phi (1,y) &= \frac{1}{\sinh(y)} \cdot \left( \ln \tanh \left( \frac{y}{2} \right) \right)
\end{align*}
\]

"densities after the transform"
Define
\[ \phi_r(x, s) = \frac{1}{\pi} \sum_{j} (-1)^{j} \int e^{s \chi_j} f_j(x) \prod_{k \neq j} \left( 1 - e^{s \chi_k} \right) \, dx \]
\[ = \sum_{n=0}^{\infty} \sum_{j} (-1)^{j} \int e^{s \chi_j} f_j(x) \prod_{k \neq j} \left( 1 - e^{s \chi_k} \right) \, dx \]
\[ = \mathcal{L} \left( \phi \left( 0, \chi \right) \right) + (-1)^{j} \mathcal{L} \left( \phi \left( 1, \chi \right) \right) \]

Laplace transform
Since the $\xi_j$'s and hence the $\xi(x_j)$'s are statistically independent, it follows that:

$$\Phi_{d_c}(x,s) = E\left\{ \sum_{x \in d_c \cap \mathbb{Z}_{d_c}^*} (-1)^{x \cdot x'_{d_c} - s_{x'_{d_c}}} \right\}$$

$$= \prod_{j=1}^{d_c-1} \Phi_j(x,s)$$
Hence

\[ \left\{ \phi_j \right\}_{j=1}^{d_{\mathcal{C}}-1} \quad \left\{ \chi_j \right\}_{j=1}^{d_{\mathcal{C}}-1} \quad \left\{ \Psi_j \right\}_{j=1}^{d_{\mathcal{C}}-1} \]

\[ \phi_{d_{\mathcal{C}}} (\chi, s) \]

\[ \left( m, j \right) \]
The last computation use that:

\[ \Phi_c (\Lambda_S) \]

\[ = \int \left( \Phi (0, \mathbf{J}) \right) + \left( -1 \right) \int \left( \Phi (1, \mathbf{J}) \right) \]

\[ = \left. \Phi (0, \mathbf{J}) \right|_{x_1 \xi_1 \xi_2} \]

\[ \times d\xi_1 \xi_2 \]

\[ \times d\xi_1 \xi_2 \]
\[ I^{-1} \left\{ \phi_{dc} (0, s) + \phi_{dc} (1, s) \right\} \]
\[ = \frac{1}{2} \]

\[ I^{-1} \left\{ \phi_{dc} (1, s) \right\} = \]

\[ \sum_{x \in x_{dc}} \sum_{y \in y_{dc}} \]

\[ I^{-1} \left\{ \phi_{dc} (0, s) - \phi_{dc} (1, s) \right\} \]
\[ = \frac{1}{2} \]

Letting \( x = x_{dc} \), \( y = y_{dc} \), \( \lambda = \lambda_{dc} \).
\[ x = \text{sgn}(e) \]

\[ y = -\ln |\text{tanh}\left(\frac{e}{2}\right)| \]

Finally,

\[ \phi_L(e) = \phi(0, -\ln \text{tanh}\left(\frac{e}{2}\right)) \]

\[ = \phi(1, -\ln \text{tanh}\left(\frac{e}{2}\right)) \quad e \leq 0 \]

\[ = \frac{\sinh(e)}{-\sinh(e)} \]
SUMMART:

density evolution was accomplished by tracking densities across locations: 

A  B  C  D
Convergence & Concentration

Theorem - LDPC Codes

Recap

Completed discussion of density evolution w.r.t. B? decoding of LDPC Codes

"The Capacity of Low-Density Parity-Check Codes Under Message
Passing Decoding

T. J. Richardson & Urbanke

IEEE Trans. on. Inform. Theory

Feb. 2001
Theorem  For any $\varepsilon > 0$,

\[ \Pr \left\{ \left| Z - \mathbb{E}(Z) \right| > \frac{\eta \delta \varepsilon}{2} \right\} \leq 2 \varepsilon \]

(concentration around the mean)  

(a) 

\[ \Pr \left\{ \left| Z - \mathbb{E}(Z) \right| > \frac{\eta \delta \varepsilon}{2} \right\} \leq 2 \varepsilon \]

(b)  For any $\varepsilon > 0$ and $n > \frac{2\delta}{\varepsilon}$,

\[ \left| 1\mathbb{E}(Z) - n \eta \delta \varepsilon \right| < \frac{\eta \delta \varepsilon}{2} \]

(convergence to the cycle-free case)
c) For any $\epsilon > 0$ and $n > \frac{2\beta}{\epsilon}$, we have:

\[
\Pr_\pi \left\{ |z - \mu_{n\pi}^+| > n\epsilon \right\} \leq 2\epsilon - \beta \epsilon^2 n
\]

(Concentration around the cycle-free case)
where:

(i) the probabilities are computed over all choices of \((d_v, d_c)\)-regular codes and over all channel realizations.

(ii) \(z = \#\) of incorrect messages passed from the \(n_d\) variable nodes to the \(n_d\) check nodes in the \(l\)th iteration, \((n\) denotes the \# of check nodes and \(n_d = n_d c\)).
$i^{th}$ Iteration
p below is the probability of an incorrect message being passed during the \( l \)th iteration in the tree-like case.

b) For any \( \epsilon > 0 \) and \( n > \frac{2\Phi}{\epsilon} \)

\[
| \text{IE}(2) - n d(v) p | < \frac{n d(v) \epsilon}{2}
\]

\( p \) derived from density evolution
Also, in the theorem,

\[ \beta = \beta(d_v, d_c, \varepsilon) \quad \text{and} \quad \gamma = \gamma(d_v, d_c, \varepsilon) \]

are constants.
The ensemble \( \mathcal{J} (d_v, d_c) \) - regular LDPC codes.

- \( n \) variable nodes, each of degree \( d_v = d_v \).
- \( r \) check nodes, each of degree \( d_c = d_c \).
Each variable is associated with a "sockets". Each check node is linked to \( d_c \). "Sockets".

\[
(n_d, v)! \text{ possible Tanner graphs preserving the (} d_v, d_c \text{) - regular property.}
\]
Let $z_i$ be the number of incorrect messages passed along the $i$th edge $e_i$ during the $k$th iteration. $z_i \in \{0, 1\}$.

$$t = \sum_{i=1}^{n_d} z_i$$

$$\mathbb{E} \left[ \sum_{i=1}^{n_d} z_i \right] = \mathbb{E} \left[ \sum_{i=1}^{n_d} \mathbb{E} \left[ z_i \right] \right] = n_d \mathbb{E} \left[ z_i \right] \quad \text{(by symmetry)}$$

Focus on this.
\[ \mathbb{E} \{ z_i \} = \mathbb{E} \{ z_i \mid N_{e_1}^{2e} \text{ is } T-L \} \]

\[ \leq \frac{1}{n} \leq 1 + \mathbb{E} \{ z_i \mid N_{e_1}^{2e} \text{ is not } T-L \} \leq \frac{2}{n} \leq \Pr \{ N_{e_1}^{2e} \text{ is not } T-L \}. \]

When \( n \) is large, turns out that

\[ \Pr \{ N_{e}^{2e} \text{ is } T-L \} \geq 1 - \frac{\gamma}{n}, \]

\( \gamma \) is a constant.
\[ \frac{X}{n} - \frac{1}{n} \leq \frac{\sum z_i}{n} - \frac{1}{n} \leq \frac{\sum z_i}{n} \leq \frac{X}{n} + \frac{\varepsilon}{n} \]

\[ |\sum z_i| - n\varepsilon \leq n\varepsilon \left( \frac{X}{n} \right) \]

We assume \( n \) large enough so that

\[ \frac{X}{n} < \frac{\varepsilon}{2} \]

\[ |\sum z_i| - n\varepsilon \leq n\varepsilon \left( \frac{X}{n} \right) \leq \frac{n\varepsilon}{2} = \varepsilon \]
This proves (2). We will skip the lengthy proof of (1).

To prove (3) given (1) and (2), we argue as follows:

\[ \exists \varepsilon \left\{ z - \varepsilon \phi \geq n \varepsilon \right\} \]

\[ = \exists \varepsilon \left\{ z - \left[ z - \sum_{i=1}^{n} z \right] + \left[ \sum_{i=1}^{n} z - n \varepsilon \phi \right] \geq n \varepsilon \right\} \]

but for large enough n when

\[ \left| \sum_{i=1}^{n} z - n \varepsilon \phi \right| < \frac{n \varepsilon}{2} \]
\[ \leq \frac{1}{n} \sum |z - \mathbb{E}z| > \frac{\eta_1 \xi}{2} \]

- \beta \in \eta

\[ \leq 2c \quad \text{from part a) } \]
Lec 36

A construction for

Finite Fields

Recap

* reviewed discussion of
density evolution w.r.t
the BP decoding of LDPC
code 8

* Concentration & convergence
  theorem
Finite fields are the basis upon which the widely-used classes of BCH and Reed-Solomon codes are built.
Example of a finite field $\mathbb{F} = \mathbb{R}[i]$, $i^2 = -1$, $\sqrt{-1}$, imaginary

Collection of all polynomials in $i$ having real coefficients
\[ \mathbb{R}[x] / (x^2 + 1) \]

\[ \mathbb{R}[x] \] (modulo \( x^2 + 1 \))

\( \mathbb{R}[x] \) the ring of all polynomials in \( x \) over the real numbers

forms a field \( \mathbb{Z}_p \) if and only if \( p \) is prime.
In \( \mathbb{R}[x]/(x^2+1) \),

\[ x^2 = (x^2+1)-1 = -1 \]

\[ \mathbb{R}[i] \]

\[ i^2 + 1 = 0 \]

\[ = \{ a + bi \mid a, b \in \mathbb{R} \} \]

\[ i^2 = -1 \]
We will proceed to provide a construction for finite fields of size $q = p^m$ for every prime $p$ and every integer $m \geq 1$.

When $m = 1$, $\mathbb{Z}_p$

$\mathbb{Z}_p = \{ \text{set of integers modulo } p \}$

is a finite field of size $p$. 
We focus therefore on the case $m > 2$.

Let $f(x)$ be a (monic) irreducible polynomial of degree $m$ over $\mathbb{F}_p$.

(monic =) highest degree coefficient equals 1

$$\mathbb{F}_p \ni \sum_{i=0}^{m} f_i \cdot x^i$$

we say that $f(x)$ is monic if $f_m = 1$. 
A (monic) polynomial $f(x)$ over $\mathbb{F}_p$ of degree $d$ is said to be irreducible if it cannot be factored into the form:

$$f(x) = g(x) \cdot h(x)$$

where

$$0 < \deg(g(x)), \deg(h(x)) \leq \deg(f(x))$$
\[ p = 2 \quad F_2 = \{0, 1\} \]

| Degree \( d \) | List of irreducible poly. 
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( x, x + 1 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>( x^2 + x + 1 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>( x^3 + x + 1, x^3 + x^2 + 1 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>( x^4 + x + 1, x^4 + x^3 + 1, x^4 + x^3 + x^2 + x + 1 )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
It can be shown, that for every prime $p$, and every integer $m \geq 1$, irreducible polynomials of degree $m$ exist.

Next, consider the set

$$\mathbb{F}_p[x] / (f(x))$$

where $f(x)$ is irreducible of degree

$$= m \geq 2.$$
$F_p \{ x \} \div (f(x))$ is the collection of equivalence classes where we define

$$g_1 \sim g_2 \iff f(x) \equiv g_1(x) - g_2(x)$$

Ex. Verify that the reflexive, symmetric, and transitive properties hold!
Deductive approach to finite fields

Let \( \mathbb{F}_q \) be a finite field of size \( q \), where \( q \geq 2 \) is an integer.

\( (\mathbb{F}_q, +, \cdot) \)

\[ n \cdot e = 0 \]

\[ \therefore e = 1 \]
\[ \#_\text{2 contains 1, \therefore it contains} \]

\[ 1, 1+1, \underbrace{1+1+1, 1+1+1+1, \ldots}_{3} \]

This list must repeat entries.

\[ \therefore m = n \text{ for some } n \geq m. \]

\[ \underbrace{1+1+1+\ldots+1}_{m \text{ times}} \]

\[ \Rightarrow (n-m)! = 0 \]

Let \( p \) be the smallest integer s.t. \( p \cdot 1 = 0 \). Then, \( p \) must be prime.
else \( \phi = \phi_1 \phi_2 \Rightarrow \phi_1 \phi_2 = 0 \) in the field,

\( \Rightarrow \phi_1 = 0 \) or \( \phi_2 = 0 \).

This contradicts the minimality of \( \phi \), hence \( \phi \) is prime.

This prime number is called the characteristic of the finite field.
In the finite field, within this set, one carries out arithmetic modulo a prime $p$. 
lec 37

\{ Finite Fields \\
\{ A deductive approach \\

Recap

* Motivating a construction for finite fields

* \( \bigcup \mathbb{F}_p \left[ x \right] \mid (f(x)) \)

* \( \deg (f(x)) = m \).
If \( p \{ x \} \) is a collection of equivalence classes where we define

\[
g_1(x) \sim g_2(x) \quad \text{if} \quad f(x) \left| g_1(x) - g_2(x) \right|
\]

Exercise Verify that this is an equivalence relation. We will denote the equivalence of \( g_1(x) \)

by \( \left\lfloor g_1(x) \right\rfloor = \left\lfloor g_1 \right\rfloor \).
Thm: Let \( p \) be prime and \( f(x) \) be monic, irreducible of degree \( m \) over \( \mathbb{F}_p \). Set

\[ R = \mathbb{F}_p [x] / (f(x)) \]

Then under the operations:

\[ [a(x)] + [b(x)] = [a(x) + b(x)] \]

\[ [a(x)] \cdot [b(x)] = [a(x) \cdot b(x)] \]

\( (R, +, \cdot) \) is a field.
Pf. Can show that these operations are well defined, i.e., the end product of the 2 operations described above does not depend upon choice of the particular representative.

Step 1 T.s \((\mathbb{R}, +)\) is an Abelian group:

- Closure ✓  i.e. ✓
- Associative ✓  Inverse ✓  Commutative ✓
\[
\begin{bmatrix} a(x) \end{bmatrix} + \begin{bmatrix} b(x) \end{bmatrix} = \begin{bmatrix} a(x) + b(x) \end{bmatrix}.
\]

\text{i.e. } = \begin{bmatrix} 0 \end{bmatrix} = \left\{ \text{set of all multiples of } f(x) \right\}

\text{Inverse of } \begin{bmatrix} a(x) \end{bmatrix} = \begin{bmatrix} -a(x) \end{bmatrix}

\underline{Step 2} \text{ Under } (\mathbb{R}, \cdot) \text{ satisfies :}

- Closure \checkmark
- \text{i.e. } \checkmark
- Commutative
- Associative \checkmark
- Inverse \text{ (circular)}
\[ \begin{bmatrix} a(x) \\ b(x) \end{bmatrix} \cdot \begin{bmatrix} a(x) \\ b(x) \end{bmatrix} = \begin{bmatrix} a(x) \cdot b(x) \end{bmatrix} \]

\[ = \begin{bmatrix} 1 \end{bmatrix} , \text{ i.e.} \]

We will demonstrate through example, the presence of a multiplicative inverse.

\[ \text{Eq} \quad \text{F} = 2 \quad f(x) = x^4 + x + 1 \]

\[ \begin{bmatrix} a(x) \\ b(x) \end{bmatrix} = \begin{bmatrix} (x + 1) \end{bmatrix} \]

\[ \begin{bmatrix} a(x) \end{bmatrix}^{-1} = ? \]
\[
\begin{bmatrix}
    a(x) \\
    b(x)
\end{bmatrix}
= \begin{bmatrix}
    c(x) \\
    d(x)
\end{bmatrix}
= \begin{bmatrix}
    1
\end{bmatrix}.
\]

We make use of the extended Euclidean division algorithm (EDA).
\[ f(x) = \frac{x^4 + x + 1}{x + 1} \]

<table>
<thead>
<tr>
<th></th>
<th>(x + 1)</th>
<th>((x + 1)^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x^4 + x + 1)</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(x + 1)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Rem</td>
<td>1</td>
<td>(x^3 + x^2 + x)</td>
</tr>
</tbody>
</table>

Quotient
\[ \frac{x_2 + x}{x_2 + x + 1} \quad \frac{x_3}{x + x + 1} \quad \frac{x^3}{x^3 + x^2 + x + 1} \]
\[ l = 1 \left( x^4 + x + 1 \right) + (x+1) \left( x^3 + x^2 + x \right) \]

\[ \therefore [1] = \begin{bmatrix} (x+1) \left( x^3 + x^2 + x \right) \end{bmatrix} \]

\[ = \begin{bmatrix} (x+1) \end{bmatrix} \begin{bmatrix} (x^3 + x^2 + x) \end{bmatrix} \]

and hence \[ \begin{bmatrix} (x+1) \end{bmatrix}^{-1} = \begin{bmatrix} (x^3 + x^2 + x) \end{bmatrix} \].
hence in this way, we are always guaranteed to find an inverse.

\[
\text{hence } \mathbb{F}_p[x] / (f(x)) \text{ is a field.}
\]
(of the construction of a finite field of size \( p = 16 \) as well as of its representation in terms of an imaginary element \( \alpha \)).

\[
p = 2, \quad m = 4, \quad f(x) = x^4 + x + 1
\]

\[
\mathbb{F}_p[x] / (f(x)) = \mathbb{F}_2[x] / (x^4 + x + 1)
\]
we introduce the imaginary element \( \lambda \) which is a zero of \( x^4 + x + 1 \):

\[
\lambda^4 + \lambda + 1 = 0
\]

\[
\mathbb{F}_2[x] / (x^4 + x + 1) = \mathbb{F}_2[\lambda]
\]

\[
\mathbb{F}_2[\lambda] = \left\{ \sum_{i=0}^{3} a_i \lambda^i \mid a_i \in \{0, 1\} \right\}
\]
<table>
<thead>
<tr>
<th>( \lambda )</th>
<th>( \lambda^0 = 1 )</th>
<th>( \lambda^4 = \lambda + 1 )</th>
<th>( \lambda^{11} = \lambda^3 + \lambda^2 + \lambda )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lambda^0 )</td>
<td>( \lambda^1 = \lambda )</td>
<td>( \lambda^5 = \lambda^2 + \lambda )</td>
<td>( \lambda^{12} = \lambda^3 + \lambda^2 + \lambda^2 + \lambda + 1 )</td>
</tr>
<tr>
<td>( \lambda^1 )</td>
<td>( \lambda^2 = \lambda^3 + \lambda^2 )</td>
<td>( \lambda^6 = \lambda^3 + \lambda^2 )</td>
<td>( \lambda^{13} = \lambda^3 + \lambda^2 + \lambda^2 + \lambda + 1 )</td>
</tr>
<tr>
<td>( \lambda^2 )</td>
<td>( \lambda^3 = \lambda^4 + \lambda^3 )</td>
<td>( \lambda^7 = \lambda^3 + \lambda^3 )</td>
<td>( \lambda^{14} = \lambda^3 + \lambda^3 + \lambda = \lambda^3 + 1 )</td>
</tr>
<tr>
<td>( \lambda^3 )</td>
<td>( \lambda^4 = \lambda^3 + \lambda + 1 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \lambda^4 )</td>
<td>( \lambda^5 = \lambda^4 + \lambda^2 + \lambda )</td>
<td>( \lambda^8 = \lambda^4 + \lambda^2 + \lambda )</td>
<td>( \lambda^{15} = \lambda^4 + \lambda = \lambda^2 + \lambda + 1 )</td>
</tr>
<tr>
<td>( \lambda^5 )</td>
<td>( \lambda^6 = \lambda^3 + \lambda + 1 )</td>
<td>( \lambda^9 = \lambda^3 + \lambda )</td>
<td>( = 1 )</td>
</tr>
<tr>
<td>( \lambda^6 )</td>
<td>( \lambda^7 = \lambda^2 + \lambda )</td>
<td>( \lambda^{10} = \lambda^3 + \lambda^2 )</td>
<td></td>
</tr>
<tr>
<td>( \lambda^7 )</td>
<td>( \lambda^8 = \lambda^2 + \lambda + 1 )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
From the table, we observe that the finite field also has representation in terms of the element $\omega$ and the various powers of $\omega$:

$$\mathbb{F}_2 = \{0, \omega \} \cup \{ \omega^i \mid 0 \leq i \leq 14 \}$$
Recap a completed discussion of the general construction of finite fields

\[ \mathbb{F}_p[x] / (f(x)) \]
Deductive Approach

Let $\mathbb{F}_q$ denote a finite field of size $q$. Then:

- $(\mathbb{F}_q, +)$ is an Abelian group

- $(\mathbb{F}_q, \cdot)$ satisfies
  - Closure
  - Assoc.
  - $\cdot e = 1$
  - inverse
  - Comm.

- multiplication distributes over addition
$1 \in \mathbb{F}_2 \Rightarrow$

$1, 1+1, 1+1+1, \ldots, 1+1+1, \ldots \in \mathbb{F}_2$

$\Rightarrow m = n$ some $n > m$

$\therefore$

$\Delta \equiv 1+1+\ldots+1 \quad \text{m times} \Rightarrow n-m = 0$

**Defn.** The characteristic $\wp$ of a finite field $\mathbb{F}_2$ is the smallest integer $\wp$ such that

$0 = 1+1+\ldots+1 \quad \text{m times}$
\[ \phi = 1 + 1 + \ldots + 1 = 0 \quad \text{in } \mathbb{F}_2 \]

\[ \phi \text{ times} \]

**Theorem 1**  The char. \( \phi \) is a prime.

**Proof** Suppose \( \phi = a \cdot b \), \( 1 < a, b < \phi \)

Then \( a \cdot b = 0 \implies a = 0 \text{ or } b = 0 \)

which contradicts the minimality of \( \phi \).
$F_\phi$ contains the set \[ \{0, 1, 2, \ldots, \phi-1\} \]

the arithmetic used to operate on these elements is $(\text{mod } \phi)$ arithmetic since $\phi \equiv 0 \pmod{\phi}$. Hence $F_\phi$

$F_\phi$ contains a copy of $F_2$.
It can be shown that $E$, $F$ are fields and $E \supset F$, then $E$ is a vector space over $F$.

It follows that $F_\Sigma$ is a vector space over $F_p$. Let $m$ be the dimension of this vector space. Then

$$F_\Sigma = \left\{ \sum_{i=1}^{m} a_i \frac{x_i}{\gamma_i} \mid a_i \in F_p \right\}$$
where \( \{ x_1, \ldots, x_m \} \) is a basis of \( \mathbb{F}_q \mid \mathbb{F}_p \),

it follows that \( \mathbb{F}_q \) is of size \( q^m \).

**Thm.** Every \( \mathbb{F} \) (finite field) \( \mathbb{F}_q \) has

size \( q \) of the form \( q = p^m \), \( p \) prime,

(moreover \( p \) is the characteristic of \( \mathbb{F}_q \)).

\( q \in \{ 2, 2^2, 3, 3^2, 2^3, 5 \} \)
Multiplicative Structure of $\mathbb{F}_q$

Let $\beta \in \mathbb{F}_q^\times = \{ \alpha \in \mathbb{F}_q \mid \alpha \neq 0 \}$. Then $\mathbb{F}_q^\times$ contains

$1 = \beta^0, \beta, \beta^2, \ldots, \beta^a, \beta^b, \ldots$

by finiteness of $\mathbb{F}_q^\times$, $\beta = \beta^b$ for some integers $b > a \Rightarrow \beta^{b-a} = 1$.

This motivates:
Defn. The (multiplicative) order of a nonzero element $\beta \in F^\times$ is the smallest exponent $e$ s.t. $\beta^e = 1$.

Lemma 1. Let $\beta \in F^\times$ have order $e$. Then $\beta^e = 1$ iff $e | \ell$. 
If \( l = ul + o \), \( 0 \leq o \leq e-1 \)

quotient remainder

Then, \( l = ul + o = (\beta^c)^u \cdot \beta \)

\[ \beta = \beta \cdot \beta = \beta \cdot \beta = \beta = 1 \]

but this contradicts the minimality of \( e \) unless \( o = 0 \). Hence \( e \mid e \)

//
Lemma 2. Let \( \beta \in F_5^* \) have order \( e \).

Then \( \beta^e \) has order \( \frac{e}{(e, e)} \).

\[ e = 15, \quad e = 10, \quad \text{ord } (\beta^1) = (15, 10) = 5 \]

Pf. (Exercise, straightforward).

Lemma 3. Let \( \beta, \gamma \in F_5^* \) have orders \( a, b \) s.t. \( (a, b) = 1 \). Then \( (\beta \gamma) \) has order \( e = ab \).
Let \((\beta \gamma)\) have order \(\ell \).

Then \(\beta^\ell = \gamma^\ell\).

\[
\Rightarrow \quad \frac{a}{(a, e)} = \frac{b}{(b, e)} \quad \text{since } \gamma, \gamma^{-1}
\]

have the same order.

\[
\Rightarrow \quad a \cdot (b, e) = b \cdot (a, e)
\]

\[
\therefore \quad a \mid (a, e) \Rightarrow a \mid \ell
\]

Similarly \(b \mid (b, e) \Rightarrow b \mid \ell\).

\[
\therefore \quad a b \mid \ell.
\]
On the other hand,

\[(\beta x) = (\beta^a)^b \cdot (\gamma^b)^a = 1\]

hence \( x | ab \). i.e. \( x = ab \) \( \Box \)

Among all the elements in \( \mathbb{F}_q \), let \( \beta \) have maximal order \( = \alpha \).

Claim: Every element \( \theta \in \mathbb{F}_q \) has
Order dividing $x$.

Let $\theta$ have order $= \ell + r$.

(via Lemma 3)

Then we can write $x = \phi_1 \cdots \phi_s$

$\ell = \phi_1 \cdots \phi_s$

with $\phi_i \geq e_i$ for some $i$. WLOG assume $\phi_1 \geq 1$.

Then, $x = \phi_1 (\phi_1) \cdots (\phi_1)$

\[\ell = \phi_1 (\phi_1)\]
Now consider \( \phi_{e_1} \) and \( \beta \). \( \theta \) has order \( q_1 \), \( e_1 \in (\ell_{\ell_1} x_1) = \phi_{e_1} \).

has order
\[
\phi_{e_1} = e_1 = \pi_1
\]

\((\pi_1, \phi_{e_1}) = 1\), it follows that \( \phi_{e_1} \) \( \ell_1 \) \( \theta \) has order.
\[ = \phi^a \cdot x_1 \Rightarrow \phi^b \cdot x_1 = x \]

which contradicts the maximality of \( x \).

Hence, \( x \, | \, 2 \).


Lemma The maximal order of an element \( \beta \in \mathbb{F}_x = 2^{m-1} = \phi - 1 \).
Let \( \beta \) have maximal order \( n \).

To show that \( n = \frac{m}{p-1} \). Since \( \theta \in \mathbb{F}_m^* \)

\[ \Rightarrow \theta \text{ has order dividing } n, \]

\[ \theta^l = 1 \Rightarrow x^l - 1 \mid x^n - 1 = 0. \]

\[ x = \theta \]

It follows that every nonzero element in \( \mathbb{F}_2 \) is a zero of \( (x^l - 1) \).
\[ z \geq \phi^m - 1. \]

On the other hand, consider

\[ 1, \beta, \beta^2, \ldots, \beta^n, -\gamma, -\gamma^2, \ldots \]

Clearly of some \( a, b \),

\[ a \leq b \iff b - a = 1 \]

with \( b - a \leq \phi^m - 1 \).

Hence \( z \leq \phi^m - 1 \).
Corollary Every $\mathbb{F}_q$ contains an element $\alpha$ of order $q - 1$. In terms of $\alpha$, $\mathbb{F}_q$ has the representation:

$$\mathbb{F}_q = \{0, \alpha \} \cup \{ \alpha^i \mid 0 \leq i \leq q - 2 \}$$

**Defn.** An element $\alpha \in \mathbb{F}_q^*$ of order $q - 1$ is called a primitive element of $\mathbb{F}_q$. 
Consider \( F_2^4 = F_2[x] / (x^4 + x + 1) \).

\( x^4 + x + 1 \) is irreducible over \( F_2 \).

Write \( \alpha \) for the equivalence class \( \alpha = [x] \) in \( F_2[x] / (x^4 + x + 1) \).

(Alternately, we may regard \( \alpha \) as an imaginary element satisfying \( \alpha^4 + \alpha + 1 = 0 \).

\( \uparrow \) more practical! \( ) \)
\[
\begin{align*}
\lambda^0 &= 2 + 1 \\
\lambda^1 &= 2^3 + 2^2 \\
\lambda^2 &= 2^4 + 2^3 \\
\lambda^3 &= 2^5 + 2^4 \\
\lambda^4 &= 2^6 + 2^5 \\
\lambda^5 &= 2^7 + 2^6 \\
\lambda^6 &= 2^8 + 2^7 \\
\lambda^7 &= 2^9 + 2^8 \\
\lambda^8 &= 2^10 + 2^9 \\
\lambda^9 &= 2^{10} + 2^9 \\
\lambda^{10} &= 2^{11} + 2^{10} \\
\lambda^{11} &= 2^{12} + 2^{11} \\
\lambda^{12} &= 2^{13} + 2^{12} \\
\lambda^{13} &= 2^{14} + 2^{13} \\
\lambda^{14} &= 2^{15} + 2^{14} \\
\lambda^{15} &= 2^{16} + 2^{15} = 1!
\end{align*}
\]
Thus \( \alpha \) as defined above, is a p.e. (primitive element) of \( \mathbb{F}_{16} \).

### Preview of Minimal Polynomials (by example)

\[ \text{Eq } p = 2 \quad f(x) = x^4 + x + 1 \]

\[ \overline{\mathbb{F}_2} = \mathbb{F}_2[x] / (x^4 + x + 1) \]
\[ F_q = \left\{ \sum_{i=0}^{3} a_i d^i \mid a_i \in \{0,1,3\} \right\} \]

where \( d = [x] \) in \( \mathbb{F}_2[x]/(x^4 + x + 1) \)

and hence satisfies \( d^4 + d + 1 = 0 \)

---
\( m_\beta(x) \)

\[
(x + 2)(x + 2^2)(x + 2^3) = (x + x + 1)
\]

\[
(x + x + x + x + 1)
\]

\[
(x^2 + x + 1)
\]

\[
(x^3 + x + 1)
\]

\[
\frac{1}{4}
\]

\[
\frac{1}{2}
\]
\begin{align*}
\sum_{-2}^{2} (x - x) &= (x) (x + 1) \\
\sum_{2}^{2} (x + x + 1) &= (x + x + 1) (x + x + 1) (x + x^3 + x^2 + x + 1)
\end{align*}
Minimal Polynomials

Note: Every element \( \beta \in \mathbb{F}_2 \) is a zero of \( x - 1 \). Hence every element of \( \mathbb{F}_2 \) is a zero of \( x^2 - x \).

This motivates:
Define the minimal polynomial $m_\beta(x)$ of $\beta \in \mathbb{F}_q$ is the smallest degree monic polynomial of which $\beta$ is a zero.
Lemma \( m_\beta(x) \) is irreducible

If suppose not. Then

\[ m_\beta(x) = f(x)g(x) \text{ with} \]

\[ 0 < \deg(f), \deg(g) < \deg(m_\beta). \]

But \( m_\beta(\beta) = 0 \Rightarrow f(\beta)g(\beta) = 0 \)
Lemma

\[ f(\beta) = 0 \Rightarrow m(\beta) = 0 = f(\beta) \]

This contradicts the minimality of \( m(x) \).

\[ f(\beta) = 0 \Rightarrow g(\beta) = 0. \]

either

\[ f(\beta) = 0 \Rightarrow g(\beta) = 0. \]
\[ f(x) = a(x) m_\beta(x) + b(x) \quad \text{(2)} \]

**Remainder of quotient**
\[ \text{deg } \leq m_\beta(x) \]

\[ \therefore f(\beta) = 0 \Rightarrow a(\beta) m_\beta(\beta) + b(\beta) = 0 \]

\[ \Rightarrow b(\beta) = 0 \]

but this once again,

contradicts the minimality of \( m_\beta(x) \)

unless \( b(x) = 0 \). Hence \( m_\beta(x) \mid f(x) \)
Corollary \[ m_\beta(x) \mid x^2 - x \]

Proof. Every element in \( \mathbb{F}_2 \) is a zero of \( x^2 - x \). Hence \( \beta - \beta = 0 \)

Thus: \[ m_\beta(x) \mid x^2 - x \] \( \Box \)
Lec 39  Subfields of a finite field

Recap
* characteristic of a finite field
* multiplicative order of an element
* primitive elements
* minimal polynomials
Define \( F^*_L = F_L \setminus \{0\} \) (set of all non-zero elements).

**Lemma 1** Let \( \beta \in F^*_L \), \( q = p \). Then
\[
\deg (\beta(x))^m \leq m.
\]

**Proof** Consider
\[
1, \beta, \beta^2, \ldots, \beta^m
\]
(m + 1) elements.
These cannot all be linearly independent for this would mean that the $m+1$ elements $m \sum_{i=0}^{m} a_i \beta_i$, $a_i \in \mathbb{F}_p$ are all distinct, which is impossible since $|\mathbb{F}_p| = p$. Thus there exists a linear dependence expression of the form:
\[ \sum_{i=0}^{m} c_i \beta^i = 0 \] with at least one \( c_i \neq 0 \).

\( \Rightarrow \) \( \beta \) is a zero of \( \sum_{i=0}^{m} c_i x^i \).

Hence \( m \beta(x) \mid \sum_{i=0}^{m} c_i x^i \).

\( \Rightarrow \) \( \deg(m \beta(x)) \leq m \).
Lemma 2. If $\beta \in F_q^*$, $q = p^m$, is a primitive element, then

$$\deg (m_{\beta}(x)) = m.$$ 

Proof. Let $\deg (m_{\beta}(x)) = s$. From Lemma 1, $s \leq m$. Since $\beta$ is a p.e. of $F_q$, every element $\theta$ in $F_q$ can be expressed as a polynomial in $\beta$ and
hence has an expression of the form

\[
\theta = \sum_{i=0}^{8-1} \beta_i a_i, \quad a_i \in \mathbb{F}_p
\]

Hence \( p \nmid p^m \) by counting

\[
\therefore 8 \nmid m
\]

\[
\therefore 8 = m
\]
Defn. The minimal polynomial $m_\beta(x)$ of a primitive element $\beta$ in $\mathbb{F}_q$,
$q = \beta^m$, is called a primitive polynomial (note that $m_\beta(x)$ has degree $= m$).

$\mathbb{F}_q = \mathbb{F}_2 \left< x \right> / (x^4 + x + 1)$

$= \mathbb{F}_2 \left< x \right>$ where $\alpha = [x]$ and thus satisfies $\alpha^4 + \alpha + 1 = 0$. 
As seen before \( \lambda \) is a \( \phi \), 

\[
\lambda \text{ has order } \lambda \in \left( \frac{m}{\phi - 1} \right) = \frac{15}{(15, k)}
\]

and hence is primitive \( \text{iff } (15, k) = 1 \).

Given an integer \( n = \prod_{i=1}^{r} p_i^{e_i} \) the number of

integers \( \lambda \),

\[
0 \leq \lambda \leq n - 1 \quad \text{s.t. } (\lambda, n) = 1 \]

equals \( \phi(n) = \prod_{i=1}^{r} \phi(p_i^{e_i - 1}) \) (Euler's totient \( \phi \)).
where \{p_i\} are all primes.

\[ \therefore \phi(15) = \phi(5 \cdot 3) = (5-1)(3-1) = 8. \]

Hence \( F_{16}\) contains \( \phi \cdot e \):

- PRIMITIVE ELEMENT COMMON MIN. POLY.
- \( \{x, x^2, x^4, x^8\} \equiv x^4 + x + 1 \)
- \( \{x^2, x, x^4, x^8\} \equiv x^4 + x^3 + 1 \)
Hence \((x^4 + x + 1)\) and \((x^4 + x^3 + 1)\) are the only primitive polynomials associated to \((this)\) \(F_{16}\).
Subfields of a finite field

Goal: characterize all the subfields of $\mathbb{F}_p$.

Example: $p = 2$, $m = 12$. $\mathbb{F}_{12}$ has the following subfield structure:
The diagram illustrates a hierarchical relationship between different fields, indicated by the arrows. The fields are labeled as follows:

- $F_2$ (subfield of $F_{12}$)
- $F_{12}$
- $F_6$ (subfield of $F_{12}$)
- $F_3$ (subfield of $F_6$)
- $F_2$ (subfield of $F_3$)
- $F_2$ (subfield of $F_4$)
- $F_4$ (subfield of $F_{12}$)

The arrows signify the subfield relationship, with $F_2$ being a subfield of $F_3$, $F_3$ of $F_6$, and so forth.
Note that all the subfields are of the form $\mathbb{F}_2^k$ with $k \mid 12$. As it turns out, this is the case in general:
Theorem: $F \subseteq F'$ if and only if $\phi \models F \Rightarrow F' \Rightarrow \phi \models F'$. The proof is provided at the end of the lecture notes in the form of an appendix.
A Useful Lemma

Lemma In any field \( \mathbb{F} \) with characteristic \( p \),
\[
(x + y)^p = x^p + y^p.
\]

Proof: \( \sum_{i=0}^{p-1} \binom{p}{i} x^{p-i} y^i \) because:
\( (\phi) = \begin{cases} 1 & \hat{i} = 0 \text{ or } \hat{i} = \phi \\ 0 \pmod{p} & \text{else} \end{cases} \)

Since:

\[
\binom{\phi}{\hat{i}} = \frac{\phi!}{(\phi - \hat{i})! \cdot \hat{i}!} = \frac{\phi \cdot (\phi - 1) \cdot \ldots \cdot (\phi - \hat{i} + 1)}{1 \cdot 2 \cdot \ldots \cdot \hat{i}}
\]
Test for membership in a subfield

Thm Let $F \subseteq F_m$. Then

\[ \emptyset \subseteq F \quad \text{iff} \quad \emptyset = \emptyset. \]

(proof is in the Appendix)
Let $F_{16} = \mathbb{F}_2[x] / (x^4 + x + 1)$.

Consider the subfield $\mathbb{F}_2$. Test: $x = x$.

$\Rightarrow x = 0$ or else $x = 2^k$ with $k \leq 3k$

$(x^2) = x \Rightarrow x = 1 \Rightarrow k \in \{0, 5, 10\}$

$\therefore F_{16} = \{0, 1, 2^5, 2^{10}\}$
Similarly,

\[ F_2 = \text{Test: } x = x \]

\[ \Rightarrow x = 0 \text{ or else } x = x \text{ with } k \]

\[ \frac{2k}{\lambda} = \frac{k}{\lambda} = 1 \Rightarrow k = 0 \]

\[ \Rightarrow F_2 = \{0, 1\} \]
\[ F_2^n = \{ 0 \} \cup \{ x_i \mid 0 \leq i \leq 14 \} \]

\[ F_2^2 = \{ 0, 1, 2^5, 2^{10} \} \]

\[ F_2^2 = \{ 0, 1 \} \]
The finite fields of size $p^m$ exist for every prime $p$, $m \geq 1$.

Proof: When $m = 1$, the set of integers modulo $p$, $\mathbb{Z}_p$, is an example of a field of size $p$.

For $m \geq 2$, we will recursively
construct finite fields of characteristic $p$ of increasing size until we reach a field that contains all the zeros of $\mathbb{F}_p[x]$. Then this collection of $\mathbb{F}_p$ zeros can be shown to form the desired finite field.

(further details may be found in the Appendix)
**Thm** The polynomial $x^m - x$ over $\mathbb{F}_p$ has the factorization:

$$x^m - x = \prod_{d|m; \text{gcd}(d, m) = 1} \prod_{1 \leq i < m} f_i(x)$$

where $d | m$ and $\deg(f_i(x)) = d$
We illustrate by example. The proof may be found in the Appendix.

Eq. Let \( q = 2^4 \) so \( p = 2 \). Then

\[
\begin{align*}
2^4 & = (x)(x+1)(x^2 + x + 1) \\
& \Rightarrow (x+1)(x^3 + 1)(x^3 + x^2 + x + 1) \\
\end{align*}
\]

areducible factors

\[ \text{deg} = 4 \]
Thm. Any two finite fields $F_q$, $F_{q'}$ of the same size $q = q'$ are isomorphic.

**Proof (sketch)** Let $f$ be a $\phi$ in $F_q$ and $m_f(x)$ its min. poly.

Then $m_f(x) \mid x^q - x$. 

Over in \( F' \), every element is a zero \( \int x^m \). Hence some element \( \theta \) must be a zero \( \int m \beta(x) \). Then the map \( \phi : \beta \rightarrow \theta \):

\[
\sum a_i \beta \rightarrow \sum a_i \theta \\
\sum a_i \in F_p
\]

\( i = 0 \)
can be shown to be an isomorphism. We illustrate below with an example.
Let $q = 2^t$.

We note from the factorization:

\[ x^4 - x = (x)(x+1)(x^2 + x + 1) \]

\[ (x+x+1)(x^3 + x^2 + 1)(x^4 + x^3 + x^2 + x + 1) \]

that there are 3 different irreducible polynomials of degree 4 over $\mathbb{F}_2$. 

Let

\[ F_2^L = \frac{F_2 \{x\}}{\left(4 + x^3 + x^2 + x + 1\right)} \]

and

\[ F_2'^L = \frac{\sum F_2 \{x\}}{\left(x^4 + x + 1\right)} \]
Let \( \beta = \{x\} \) in \( \mathfrak{F}_3 \) so that

\[
m_\beta(x) = x^4 + x^3 + x^2 + x + 1
\]

and

\[
\alpha = \{x\} \in \mathfrak{F}_3 \text{ so that } m_\alpha(x) = x^4 + x + 1.
\]
Then in $\mathbb{F}_2^1$, the element $\theta = x^3$ has minimal polynomial $m_{\theta}(x) = x^4 + x^3 + x^2 + x + 1$.

Hence the map:

$$
\begin{align*}
\phi : \mathbb{F}_2^1 &\rightarrow \mathbb{F}_2^1 \\
\beta &\rightarrow \theta
\end{align*}
$$
is an isomorphism
The "add-1" table

Finite field computations are greatly simplified by the creation of an add-1 table. We present an example.
Eq. \quad q = 2 \quad p = 2

\[ F = \mathbb{F}_2 \left[ x \right] / (x^4 + x + 1) \]

with \( \lambda = [x] \) so that \( x^4 + x + 1 = 0 \).

Then

\[ F = \{ 0 \} \cup \{ x^i \mid 0 \leq i \leq 14 \} \]
### Polynomial Representation

<table>
<thead>
<tr>
<th>0</th>
<th>$x^4 = x + 1$</th>
<th>$x^{11} = x^3 + x^2 + x$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x^0 = 1$</td>
<td>$x^5 = x^2 + x$</td>
<td>$x^{12} = x^7 + x^3 + x^2 = x + 2 + x^{11}$</td>
</tr>
<tr>
<td>$x^1 = x$</td>
<td>$x^6 = x^3 + x^2$</td>
<td>$x^{13} = x^7 + x^3 + x^2 + x = x^2 + x^{11}$</td>
</tr>
<tr>
<td></td>
<td>$x^7 = x^2 + x^3$</td>
<td></td>
</tr>
<tr>
<td>$x^2 = x$</td>
<td>$x^8 = x^3 + x + 1$</td>
<td></td>
</tr>
<tr>
<td>$x^3 = x^3$</td>
<td>$x^9 = x^4 + x^2 + x$</td>
<td>$x^{15} = x^4 + x = x + x^{11}$</td>
</tr>
<tr>
<td></td>
<td>$x^{10} = x^3 + x$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$x^{11} = x^4 + x^2 = x^2 + x + 1$</td>
<td></td>
</tr>
</tbody>
</table>

The polynomial representation shows how polynomials of different powers are derived from the basic polynomials $x^0 = 1$ and $x^1 = x$. Each subsequent polynomial is formed by adding the previous polynomial to itself, and then adding the result to the previous polynomial again, and so on. This process continues, with each step building upon the previous, until the desired polynomial is reached.
With the aid of the polynomial representation given above, one can create the adf-1 table given below:
The add-1 table

<table>
<thead>
<tr>
<th></th>
<th>1+x</th>
<th></th>
<th>1+x</th>
<th></th>
<th>1+x</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2^5</td>
<td>2^10</td>
<td>2^11</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>2</td>
<td>2^6</td>
<td>2^13</td>
<td>2^13</td>
</tr>
<tr>
<td>2</td>
<td>2^4</td>
<td>2^7</td>
<td>2^9</td>
<td>2^14</td>
<td>2^3</td>
</tr>
<tr>
<td>2^2</td>
<td>2^8</td>
<td>2^8</td>
<td>2^2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2^3</td>
<td>2^14</td>
<td>2^9</td>
<td>2^7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2^4</td>
<td>2</td>
<td>2^10</td>
<td>2^5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The add-1 table is frequently used in conjunction with Haner's method to add a string of powers of $x$: 

$$
2^3 + x + 2^7 + 2^8 = x + 2 + x + 2
$$

$$
= x \left( 1 + x^2 \left( 1 + x^4 \left( 1 + x \right) \right) \right) = x^2
$$
CYCLOTOMIC COSETS

These cosets will be used to explain the structure of minimal polynomials as well as to construct cyclic codes.
Let $p$ be prime, $m \geq 1$.

If $x$ is a d.c. $F_p^m$, then all arithmetic in the exponent of $x$

is conducted (mod $p^m - 1$) since $x$ has order $\equiv p^m - 1$.

\[ \text{Fermat's Little Theorem: } x^{p-1} \equiv 1 \text{ mod } p, \text{ etc.} \]
Define $a$ and $b$ as $2^{p^m}$. This can be verified to be an equivalence relation. The resulting equivalence classes are called the $p$-adic orbits.
Proof of equivalence:

(i) \( a = \hat{\theta}^0 a \vdash a \wedge a \text{ REFLEXIVE} \)

(ii) \( a = \hat{\theta}^k b =) b = \hat{\theta}^{m-k} a \) (mol \( \hat{\theta}^{n-1} \) 

SYMMETRY

(iii) \( a = \hat{\theta}^k b, b = \hat{\theta}^l c =) a = \hat{\theta}^{k+l} c \)

TRANSITIVE
Example $q = 2$, $I = 2^+$

The 2-cyclotomic cosets (mod $2^{m-1}$) are shown alongside $\Rightarrow$

is an equivalence class

$0 \rightarrow 1, 2, 4, 8$

$3, 6, 12, 9$

$5, 10$

$7, 14, 13, 11$
The smallest element within each cyclotomic coset is called a coset leader. Thus here, \( \{0, 1\} \) are the coset leaders.
Thm. Let $\beta \in F_{p^m}$. Then

$$m_{\beta}(x) = \prod_{l=0}^{d-1} (x - \beta^l)$$

where $d$ is the smallest integer such that $\beta^d = \beta$, i.e., $F_{p^d}$ is the smallest subfield in $F_{p^m}$ which $\beta$ is an element.
If (see the Appendix)

**Defn.** The elements $\beta^e$, $1 \leq e \leq d-1$ are called the conjugates of $\beta$. 
Thus the theorem also asserts that all the conjugates of $f$ share the same minimal polynomial.
APPENDIX

- Proofs to complete the lecture notes
Lemma 1 Let $n > 1$, $x > 1$, then

\[ n^2 - 1 \mid n^8 - 1 \text{ iff } x \mid 8 \]

Proof Write: \[ s = ar + b \quad b < r \]

\[ (n^8 - 1) \pmod{n^2 - 1} \]

\[ = (n^{2r} + b) \pmod{n^2 - 1} \]
\[ = \left( \left( n^{\frac{b}{c}} \right)^2 \cdot n^b - 1 \right) \pmod{n^c - 1} \]

\[ = \left( n^{\frac{b^2}{c}} - 1 \right) = 0 \quad \text{iff} \quad c | b \]

\[ \text{i.e.,} \quad \text{iff} \quad x \mid 8 \]

\[ \| \]
Lemma 2: Let $S = \{2, 7, 1\}$. Then

$$(x^2 - 1) \mid (x^8 - 1) \iff n \equiv \pm 1 \pmod{8}.$$ 

Proof:

$$(x^5 - 1) \equiv (x^2 - 1) \pmod{x^3 - 1}$$

$$= (x^2 - 1) \equiv (x^2 - 1) \pmod{x^3 - 1}$$

(1) Ching $g = ax + b$ as before.
\[ = (x^b - 1) \ (mod \ x^n - 1) \]

\[ = 0 \ \text{iff} \ b = 0 \]

i.e. \( x \mid 8 \).

\[
\text{Corollary:} \quad \frac{1}{x^{f-1}} \ | \ x^{\frac{1}{f}} - 1 \ \text{iff} \ \frac{d}{m}.
\]
Thm \ F_d \subseteq F_{\phi^m} \iff d \mid m.

pf  a) Suppose \ d \mid m

\Rightarrow \quad \phi^d \mid \phi^m

= \quad (x^{\phi - 1} - 1)(x^{\phi - 1} - 1)

= \quad (x^{\phi - x})(x^{\phi - x})
But the elements of $F_2^n$ are precisely the 2 zeros of $f(x^p - x)$.

Thus we have the picture:
and Lemma 3 will show that this picture has a subfield interpretation:
Lemmas: The collection of \( p^d \) gens of \( F \) \( x \in F \) in \( F_m \) from a df of size \( p^d \).

Let

\[ S = \left\{ \theta \in F \mid \theta^d - \theta = 0 \right\}. \]

To show that \((S, +)\) is an Abelian group, it suffices to show that.
\[ a, b \in S \implies a - b \in S. \]

\[
\begin{align*}
(a - b)^2 &= (a + (-b))^2 \\
&= a + (-b) \\
&= a - b \\
&= a - b
\end{align*}
\]

\[ \therefore (a - b) \text{ is also a zero of } f \]
\[(x^p - x) : (a - b) \in S\]

Here it suffices to show that \(a, b \in S \implies ab \in S\) and \(a^{-1} \in S\). But

\[a^p = a, \quad b^p = b = (ab)^p = ab\]
\[ a^\ell = a \Rightarrow (a^{-1})^\ell = a^{-1} \]

\[ \therefore a^{-1} \in S \]

\[ \therefore S \text{ is a subfield of size } \overline{F}_{p^\ell}. \]

b) next, suppose \( \overline{F}_{p^\ell} \subseteq F_{p^m} . \)

Let \( \beta \) be a p.e. of \( F_{p^d} \).

Then \( \beta \) has order \( d^\ell - 1 \).
But since \( \beta \in F_{\phi^n} \), \( \beta^{\phi^n-1} = 1 \)

\[ \therefore (\phi-1) | (\phi^n-1) \Rightarrow d | m \]
Thm: Let \( F \subseteq F_m \). Then

\[ \phi \not\in F \quad \text{if and only if} \quad \phi \not\in F_m. \] 

\[ \phi \in F_m \quad \text{belongs to} \quad F \quad \text{iff} \quad \phi = 0. \]

(test for membership in the subfield)
\[ \text{Pf. Suppose } \theta^p = 0 \]

\[ \implies 0 \text{ is a zero of } x^p - x. \]

But the collection of zeros of \( x^p - x \) in \( \mathbb{F}_p \) forms a subfield \( \mathbb{F}_q \) of size \( p^q \). i.e. \( \theta \in \mathbb{F}_q. \)
Conversely, suppose \( \Theta \in \mathcal{F} \).

Thus \( h = 0 \), \( \Theta = \emptyset \).
Thm Finite fields of size $p^m$ exist for every prime $p$, $m \geq 1$.

**Pf** When $m = 1$, the set of integers modulo $p$, $\mathbb{Z}_p$, is an example of a field of size $p$.

For $m \geq 2$, we will recursively...
Construct finite fields of characteristic $= p$ of increasing size until we reach a field that contains all the zeroes $f^m - x$. Then this collection of $f^m$ zeroes can be shown to form the desired finite field.

We begin with $F_p = \mathbb{Z}_p$. Let us factorize $x^m - x$ over $F_p$ to get:
\[ \phi^m (x) = \prod_{i=1}^r e_i (x) \prod_{j=1}^s f_j (x) \]

where the \( \{ e_i (x) \} \) are irreducible polynomials of degree 1 (i.e., of the form \((x-a)\))

and the \( \{ f_j (x) \} \) are also irreducible polynomials of degree \( d_j \geq 2 \).
Then the field \( F_{p^{a_1}} = F_p[x] / (f_1(x)) \).

Then it follows that the number of zeros \( f(x^m - x) \) over \( F_{p^{a_1}} \) is greater than the number of zeros over \( F_p \) since the equivalence class \([x] \) in \( F_p[x] / (f_1(x)) \) is an additional zero.
Continuing in this fashion by picking an irreducible factor $f(x^m - x)$ of degree $\geq 2$ and enlarging the finite field at each step, we will eventually end up with a finite field $\mathbb{F}$ that contains all the zeros of $x^m - x$. This set of zeros can be shown (by arguing as in the proof of...
Lemma 3) to form the desired finite field.
Thm. The polynomial $x^{m} - x$ over $\mathbb{F}_p$ has the factorization:

\[ x^{m} - x = \prod_{1 \leq \ell \leq m} \prod_{d | m, \ell \text{ gcd}} f(x) \]

\[ \deg(f) = d \]
Let \( f(x) \) be irreducible \( \deg = d \) with \( d \mid m \). Then
\[
\text{GF}_p[x]/\langle d(x) \rangle = \text{GF}_p^d.
\]
Let \( \beta = [x] \) in \( \text{GF}_p[x]/\langle d(x) \rangle \). Then
\[
f(\beta) = 0 \quad \text{and since} \quad f(x) \quad \text{is irreducible}, \quad f(x)
\]
is the min. poly. \( m_{\beta}(x) \).
But $m_p(x) \mid x^p - x \mid x^m - x$ since $d \mid m$.

hence $f(x) \mid x^m - x$.

Next let $f(x)$ be irreducible if $d \mid f = d$ and let $f(x) \mid x^m - x$. T.S. $d \mid m$

Again, let $\beta = [x] \text{ in } \mathbb{F}_p[[x]]/(f(x)) \simeq \mathbb{F}_p$. Then $m_p(x) = f(x)$ in $\mathbb{F}_p$. 

\( f(x) = x^m \phi^m - x = \phi^m = \beta. \)

Let \( \lambda \) be a d.e. \( \int f(x) \, dx. \) Then

\[
\lambda = \sum_{i=0}^{n} a_i \beta_i, \quad a_i \in \mathbb{F}_q
\]
Thus \( (\beta^l - 1) \mid \beta^m - 1 \Rightarrow d \mid m \)
Thm. Let $\beta \in F^*_m$. Then

$$m_\beta(x) = \prod_{l=0}^{d-1} (x - \beta^l)$$

where $d$ is the smallest integer such that $\beta^d = \beta$, i.e., $F_{\beta^d}$ is the smallest subfield in $F_m$ in which $\beta$ is an element.
Let

\[ m_\beta(x) = \sum_{i=0}^{\ell} g_i \cdot x^i \quad (g_i \in \mathbb{F}_p) \]

Then

\[ m_\beta(\beta) = \sum_{i=0}^{\ell} g_i \cdot \beta^i = 0 \]

\[ \Rightarrow m_\beta(\beta^i) = \sum_{i=0}^{\ell} g_i \cdot \beta^i = 0 \]
\[ \sum_{i=0}^{d} \beta_i p_i = 0 \]

\[ \Rightarrow \beta_i p_i \text{ is also a zero of } \int m \beta(x) \]

On the other hand, define

\[ h(x) = \sum_{j=0}^{d-1} h_j x^j = \prod_{i=0}^{d-1} (x - \beta_i) \]

The coefficients of \( h(x) \) are clear.
symm. for $\sum \beta^k \beta^{-1}$.

\[ \sum_{k=0}^{d-1} \beta^k \]

\[ \sum_{k=0}^{d-1} \beta^k \sum_{l=0} \beta^l = \sum_{l=1}^{d-1} \beta^l \]

\[ \therefore \sum_{k=0}^{d-1} \beta^k \sum_{l=1}^{d-1} \beta^l = \sum_{l=1}^{d-1} \beta^l \]

\[ \therefore \beta^k \sum_{l=0}^{d-1} \beta^l = \beta. \]

Since $\beta^k \sum_{l=0}^{d-1} \beta^l = \beta$.

Hence $h(x) \in \Phi \beta^k \{x\}$. 
\[ m_\beta(x) = \prod_{\lambda=0}^{d-1} (x - \beta_\lambda) \]
Recap

* completed discussion on finite fields
* completed discussion on min. poly.

* subfield structure

\[(x+y)^p = x^p + y^p\]

* \(\alpha \in \mathbb{F}_p\) \implies \alpha^p = 0

* finite fields of every size \(p^m\) exist

\[x^p - x = \prod_{d|m} f(x)\]

\(d \mid m\)

\(\text{deg } f = d\), \(f\) irreducible
* many two ft of the same size

same isomorphic

* add-1 table

* $p$-cyclotomic cosets $(\text{mod } p^m-1)$

* $m_p(x) = \prod_{l=0}^{d-1} (x - p^{l \phi^e})$
Our interest is in cyclic codes $\mathcal{C}$ of block length $N$ over $\mathbb{F}_q$, $\mathbb{L} = \mathbb{F}_q$, $(\mathbb{L}, N) = 1$.

Goal: Find the smallest field $\mathbb{F}_m$ of order $\mathbb{L} = \mathbb{F}_q$ that also contains $\mathbb{F}_q$. 
an element \( \alpha \in F_m \) of order \( \Sigma \).

Consider

\[
1, 2, \ldots, \Sigma \ldots \Sigma \ldots \ (\text{modulo } N)
\]

Since \( \Sigma_H \) is finite,

some \( k \leq \ell \),

\[
\Sigma \equiv \Sigma, \quad \ell \geq k, \quad (\text{mod } N)
\]
\[ \ell - k \]

\[ \ell \equiv 1 \quad (\text{mod } N) \]  

\[ q^m \equiv 1 \quad (\text{mod } N) \]

Let \( m \) be the smallest power of \( q \) such that

\[ q^m \equiv 1 \quad (\text{mod } N) \]

\( m \) is then called the multiplicative order of \( q \) \( \pmod{N} \).
Note: this implies that
\[ q^m - 1 \equiv 0 \pmod{N} \]

\[ \Rightarrow N \mid (q^m - 1) \]

\[ \Rightarrow \sum_{m} F_m \text{ contains an element of order } \frac{q-1}{N} \text{, for example, } \lambda = \beta \]

where \( \beta \) is a q.e. f from \( \sum_{m} F_m \).
\[ q = 2 \quad N = 15 \implies m = 4 \]

\[ \chi = \beta, \quad \beta \text{ primitive in } \mathbb{F}_{2^4}. \]

\[ q = 1^e \quad N = (q-1) = m = 1 \]

\[ \chi = \beta, \quad \beta \neq 1 \text{ for } \mathbb{F}_e \]
Finite Field Transform
Define \((q, N) = 1\), \(m\) be the multiplicative order of \(q \mod N\).

Let \(a\) be an element of order \(N\) in \(\mathbb{F}_q\). Let \((a_t)_{t=0}^{N-1}\) be a vector of length \(N\) over \(\mathbb{F}_m\). Then

\[
\hat{\alpha} = \sum_{t=0}^{N-1} a_t x^t, \quad 0 \leq x \leq \frac{1}{N-1},
\]

\[
\alpha = \sum_{t=0}^{N-1} a_t x^t, \quad 0 \leq x \leq \frac{1}{N-1}.
\]
is called the finite field transform (FFT) of \((a_t)\).

**Properties**

1). **Linearity**:

\[
(a_t) \Leftrightarrow (\hat{a}_\lambda) \quad (b_t) \Leftrightarrow (\hat{b}_\lambda)
\]

\[
(a_t + \theta b_t) \Leftrightarrow (\hat{a}_\lambda + \theta \hat{b}_\lambda)
\]

(Pf. Exercise!) all \(\theta \in \mathbb{F}_2^m\).
2). Let \( b_t = a_t (t - t) \pmod{N} \) for some \( 0 \leq t \leq N-1 \). Then \((b_t)\) is called a cyclic shift of \((a_t)\).

In the sequel, we will always assume that subscripts \( t \) on \( a \) and \( b \)
are always computed \((\mod n)\) and will therefore simply write
\[
\left( a_{t-\tau} \right) \text{ in place of } \left( a_{t-\tau} \mod n \right).
\]

Now
\[
\sum_{t=0}^{N-1} a_{t-\tau} x^t = \sum_{t=0}^{N-1} a_{t-\tau} \left( x^{\tau} \right)^{t-\tau} \]
\[
= \mathbf{2} \times \mathbf{1} \times \left[ \sum_{s=0}^{N-1} a_s \mathbf{v}^s \right], \quad \text{Setting}\ t - \tau = \alpha \ (mod\ N)
\]

\[
= \mathbf{2} \times \mathbf{1} \times \hat{a}_x.
\]

Thus

\[
(a_k) \ L \geq \ (\hat{a}_x)
\]

\[
\Rightarrow \ (a_{t-\tau}) \ L \geq \ \mathbf{2} \times \mathbf{1} \times \ (\hat{a}_x)
\]
3) **INVERSION FORMULA:**

\[ a_t = \sum_{\lambda=0}^{N-1} \sum_{s=0}^{N-1} x_{s+t} (s-t) \]

**Proof:**

\[ \text{RHS} = (N)^{-1} \sum \sum a_s \]

\[ = (N)^{-1} \sum \sum_{\lambda=0}^{N-1} \sum_{s=0}^{N-1} x_{s+t} (s-t) \]

\[ = N \text{ (when } s = t \text{)} \]

\[ = \left[ \frac{N(s-t)}{(s-t)(s-t) - 1} \right]_{s \neq t} \]
\[ = 0 \quad \text{since} \quad \chi^N = 1 \]

\[ = a_t \]

4). **Cyclic Convolution**

\[(a_t) \iff (\hat{a}_\chi) \quad (b_t) \iff (\hat{b}_\chi)\]

\[\Rightarrow \quad (\hat{u}_\chi) = (\hat{a}_\chi \hat{b}_\chi) \quad \text{where} \quad u_t = \sum_{\tau=0}^{N-1} a_{t-\tau} b_{\tau}\]
Pf. ( Exercise ! )
5). **Conjugact**

Suppose \( a_t \in \overline{F}_\Sigma \), all \( t \), i.e., \( (a_t) \) lies in the "ground field."
Then

\[
\hat{a}_x^z = \left[\hat{a}_x^z\right]^2 + x
\]

(\text{where } x^z \text{ in the subscript is computed } \text{mod } n)\).

Conversely \( \hat{a}_x^z = \left[\hat{a}_x^z\right]^2 \mod e = \sum 2 \)

2) \( a_t \in \mathbb{F}^z \), \text{ all } t.
since arithmetic in the
subscript \( a \) or else \( \hat{a} \) is
always computed \((\text{mod} \, 8)\), it is often
convenient to think of both
\((a_t)\) as well as \((\hat{a}_t)\) as being
periodic sequences over \( \mathbb{F}_8 \)
period $N$, i.e.,

- $q_0, q_1, q_2, \ldots, q_{N-1}$, $q_0, q_1, \ldots$

one period

- $\hat{q}_0, \hat{q}_1, \hat{q}_2, \ldots, \hat{q}_{N-1}$, $\hat{q}_0, \hat{q}_1, \ldots$

one period
This is consistent with the defn:

\[ \hat{a}_\chi = \sum_{t=0}^{N-1} a_t x^t \]

\[ \hat{a}_\chi + N = \hat{a}_\chi \]

Since \( \chi \) has order \( N \)
\[
\text{Proof:} \quad a^q = \left[ a_x \right]^q \quad \text{when} \quad a_t \in \mathbb{F}_q \quad \text{for all} \quad t
\]

\[
a_x = \sum_{t=0}^{n-1} a_t x^t
\]

\[
= \sum_{t=0}^{n-1} \left( a_t x^t \right)^q \quad \text{since} \quad a_t^q = a_t
\]

\[
= \int_{t=0}^{n-1} \sum_{t=0}^{n-1} a_t x^t \quad \text{since} \quad q = e
\]
\[
\begin{align*}
\mathbf{a}^T \mathbf{a} & = \left( \sum_{i=1}^{n} \mathbf{a}_i \right)^2 \\
\text{Converse: Next suppose} & \quad \mathbf{a}_j \neq \mathbf{0} \quad \forall j,
\end{align*}
\]

\[
\mathbf{a}^T \mathbf{a} = \left( \sum_{i=1}^{n} \mathbf{a}_i \right)^2 = \sum_{i=1}^{n} \mathbf{a}_i^2 - \sum_{i \neq j} \mathbf{a}_i \mathbf{a}_j
\]

\[
\mathbf{a}_t^T \mathbf{a}_t = \left( \sum_{i=1}^{n} \mathbf{a}_i \right)^2 - \sum_{i \neq j} \mathbf{a}_i \mathbf{a}_j
\]
\[ \sum_{j=0}^{N-1} a_j x^j = a_t x^t \]

Set \( f = x^g \mod n \). Then

as \( x \) varies over \( \{0, 1, \ldots, n-1\} \), so does \( x^g \).
Thus \( [a_t]^q = a_t \) \( \forall t \)

\[ \Rightarrow a_t \in F_q \] \( \forall t \).
**Cyclic Codes**

Define a linear cyclic code \( C \) of block length \( N \) over \( \mathbb{F}_2 \) is a collection of \( N \)-tuples \( (c_t)_{t=0}^{N-1} \) such that:

1. **LINEARITY**
(1) \((c_1^{(1)}, c_2^{(2)}) \in \mathbb{C}\)

\[\Rightarrow (c_1^{(1)} + \theta c_2^{(2)}) \in \mathbb{C}, \quad \forall \theta \in \mathbb{F}_2\]

(thus \(\mathbb{F}\) is a vector space over \(\mathbb{F}_2\))

(ii) CYCLIC SHIFTS

\[\left(\begin{array}{c} c_2^{(1)} \\ c_1^{(1)} \\ \vdots \\ c_{N-1}^{(1)} \end{array}\right) = \left(\begin{array}{cccc} c_0 & c_1 & \cdots & c_{N-1} \end{array}\right) \in \mathbb{C}\]

\[\Rightarrow \left(\begin{array}{c} c_2^{(1)} \\ c_1^{(1)} \\ \vdots \\ c_{N-1}^{(1)} \end{array}\right) = \left(\begin{array}{cccc} c_{N-\tau} & c_{N-\tau+1} & \cdots & c_{N-H-1} \end{array}\right) \in \mathbb{C}\]
(i.e., \( C \) is closed under cyclic shifts).

Given a codeword \((c_t) \in F_L^N\) in \( C \), we define the transform \((\hat{c}_x)\) of \((c_t)\) via

\[
\hat{c}_x = \sum_{t=0}^{N-1} c_t \chi_x^t
\]
where $a$ has order $N$ in $\mathbb{F}_m$, in which $m$ is the multiplicative order of $f \pmod{N}$.

Note: Thus while the code symbols belong to the ground field $\mathbb{F}_2$, ...
their transforms lie in $\sum_{m}^{N}$ in general.

In this context, we will regard $\hat{x}$ as the transform coefficient of $x(t)$ at frequency $\omega$. 
EQUIVALENCE RELATION ON FREQUENCIES

Define

\[ \lambda_2 \sim \lambda_1 \quad 0 \leq \lambda_1, \lambda_2 \leq N-1 \]

if \( \lambda_2 \equiv \lambda_1 \pmod{N} \) for some \( i \geq 0 \).

This can be verified to be an equivalence relation.
The corresponding equivalence classes are called the 9-cyclotomic cosets \((\mod N)\).
Eq \[ q = 2 \quad N = 15 \]

there are 5 equivalence classes \( \Rightarrow \)

\[ \{ \lambda \} \]

\[ \lambda_1 = 2 \] \( \lambda_1 \) \( \lambda_2 \)

\[ \lambda_1 \oplus \lambda_2 \]

\[ \lambda_1 = 2 \lambda_2 \pmod{N} \]

\[ \begin{array}{cccc}
0 & 1 & 2 & 4 & 8 \\
3 & 6 & 12 & 9 \\
5 & 10 \\
7 & 14 & 13 & 11 \\
\end{array} \]

\( \) (coset leader)
The smallest element within each cyclotomic cosets is called the coset leader.

In general, if there \( \ell \) cosets with \( \eta_i \) elements in the \( i \)th coset, then
\[
\eta_1 + \eta_2 + \cdots + \eta_\ell = n, \\
\ell = 1
\]
Defn. A subset
\[ s = \{ 0, 1, 2, \ldots, N-1 \} \]
is said to be a \textit{closed set of frequencies} if
\[ \forall x \in s, \exists y \in \text{ (mod N) } \in s. \]
It is straightforward to show that every closed set is the union of cyclotomic cosets.
**NULL SPECTRUM**

Defn Let $C$ be a linear, cyclic code of block length $N$ over $\mathbb{F}_q$. Then the collection of frequencies

$$NS_c = \left\{ \lambda \bigg| 0 \leq \lambda \leq N - 1, \sum_{\ell} \lambda c_\ell = 0 \text{ for all } (c_\ell) \in \mathbb{F}_q \right\}$$

is called the null spectrum of $C$. 
$E_9 \quad q = 2 \quad N = 15$

Preview of transform domain view of cyclic codes

\[
\begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
\times
\begin{bmatrix}
0 & 1 & 2 & 4 & 8 \\
3 & 6 & 12 & 9 \\
7 & 14 & 13 & 11
\end{bmatrix}
\]

NSR
lec 4-1

Estimating the parameters of a cyclic code.
Recap

- Finite field transform
  - expanding $\mathbb{F}_2 \rightarrow \mathbb{F}_2^m$
  - definitions
  - properties

- Cyclic codes
  - closed set
  - cyclotomic cosets
  - null spectrum
Lemma 1. The null spectrum $NS_E$ of a cyclic code $E$ is a closed set.

If $\hat{\chi} \in NS_E \iff \chi = 0$ and $(c_t) \in E$

$\hat{\chi} = 0 \implies \sum_{\chi} \chi^2 = 0 \implies \chi = 0$

Since $(c_t) \in F_E^2$ (conjugacy)
\[ x \not\in (NS)^c \Rightarrow \therefore \text{Hence } (NS)^c \text{ is a closed set.} \]

Lemma 2 Let \( S = \{0, 1, \ldots, N-1\} \) be a closed set of frequencies.

Then

\[
\mathcal{C} = \left\{ (c_k) \in \mathbb{F}_2^N \mid \sum_{k=0}^{N-1} c_k \equiv 0 \mod{N}, \forall x \in S \right\}
\]

is a linear cyclic code.
If \( (a_t), (b_t) \in \mathcal{F} \)

1) \( \hat{a}_\lambda \sim 0 \) \( \hat{b}_\lambda \sim 0 \) for all \( \lambda \in \mathcal{S} \)

2) \( \hat{a}_\lambda + \theta \hat{b}_\lambda = 0 \) for all \( \lambda \in \mathcal{S} \), \( \theta \in \mathbb{F}_q \)

3) \( (a_t + \theta b_t)_{t=0}^{N-1} \in \mathcal{C} \).

This proves that \( \mathcal{C} \) is linear.
Next, if \((c_t) \in \mathbb{R}\) and \(a_t = c_{t-1}\) then

\[ a_t = \alpha \tau c_t \]

\[ a_t = 0 \quad \forall \tau \leq S \]

\[ (a_t)_{t=0}^{N-1} \in \mathcal{C} \]

\( \therefore \) \( \mathcal{P} \) is cyclic.
REMINDER: COMMON SETTING

\[ q = p^e \quad (q, N) = 1 \]

\( m \) is the order of \( q \) \( (\text{mod} N) \)

\( C \) is a cyclic code of block length \( N \) over \( \mathbb{F}_q \).
A basic sequence $(B_t)$ of frequency $\omega_0$, $0 \leq \omega_0 \leq N-1$ is a sequence satisfying:

$$B_t = \sum_{\nu=1}^{\nu} \sum_{\nu=0}^{\omega_0} \quad \text{else}$$
\[ l = 2 \quad N = 15 \quad m = 4 \]

\[ \lambda_0 = 3 \]

Note:

\[ \frac{\lambda_0}{3} = 1 \Rightarrow \frac{\lambda_0}{3} = \frac{3}{6} = \frac{12}{36} \]

\[ \frac{\lambda_0}{3} + \frac{\lambda_0}{12} = \frac{1}{2} \]

\[ \lambda_0 = 3 \]

as well!
Lemma 3  Let $C$ be a cyclic code having null spectrum $NS_C$. Then

$$C = \{ (a_t) \in F_q^n \mid \hat{a} \wedge = 0 \neq 0 \}$$

In particular, the basic sequence $(B_t)$ of every frequency $\lambda \in NS_C$ belongs to $C$.  

(proof is in the Appendix)
Thm (Null Spectrum Theorem)

There is a 1-1 correspondence between cyclic codes over \( \mathbb{F}_2 \) of block length \( N \) and closed sets of frequencies \( \mathcal{S} \subseteq \{0, 1, \ldots, N-1\} \).
The map taking a cyclic code to its null spectrum can be shown to be 1-1
and onto. This follows (after some thought) from Lemmas 1-3.
Thm. The dimension $K$ of a linear, cyclic block code of block length $N$ over $\mathbb{F}_2$ equals the size of its non-null spectrum, i.e.,

$$K = |NS^c| = N - |NS_e|$$
 Pf (presented in the Appendix).
\[ q = 2 \quad N = 15 \quad m = 4 \]

\[ NS \subseteq \{ 1, 2, 4, 8, 3, 6, 12, 9 \} \]

\[ \Rightarrow \dim (\mathcal{C}) = 15 - 8 = 7. \]
Thm (weight theorem) Let \((a_t) \in \Sigma^n\)

have Hamming weight \(w_H((a_t)) = \omega\).

Then \((\hat{a}_\tau)\) satisfies a linear recursion of degree \(= \omega\) i.e.,

\[
\hat{a}_\tau = \sum \begin{array}{c}
u_i \cdot \hat{a}_{\tau - i} \end{array} \Rightarrow \tau,
\]

where \(u_i \in \Sigma\).
Pf. (please see the Appendix)
Thm Let $C$ be a cyclic code over $\mathbb{F}_2$ of length $N$, $(N, 2) = 1$, whose null spectrum $NS_C$ contains the consecutive set of frequencies
\[
\left\{ m_0, m_0 + 1, m_0 + 2, \ldots, m_0 + k - 2 \right\}
\subset \left\{ 0, 1, 2, \ldots, N-1 \right\}
for some integers $m_0, d$ with $d \geq 1/2$.

Then,

\[ d_{\min}(R) \geq d \]

$d$ is called the designed distance of the code.

and hence $C$ is a

\[ \left[ N, N - \text{INS}_c, d \right] \text{ code} \]

over $F_q$. 

Pf. Since \( C \) is a linear code, it suffices to prove that the min Hamming weight of the code \( C \) is \( d \).

Suppose \( (c_t) \in C \) has Hamming weight \( w > 0 \).
Then \( \hat{\omega} \) satisfies a linear recursion of the form:

\[
\hat{\omega} = \sum_{\lambda} \omega_{\lambda} - \hat{\lambda}
\]

so in particular
\[ \hat{\nu} = \sum_{i=1}^{m_{0}+d-1} \nu_i \]

\[ x = \begin{bmatrix} 0 & 0 & \ast & \ast & \ast \\ 0 & m_0 & m_0+1 & \ldots & m_0+d-2 & m_0+d-1 & \ldots & (N-1) \end{bmatrix} \leq (NS)_R \]
If \( w < d \), then

\[
m_0 + d - 1 - i \geq m_0 + d - 1 - (d - 1) = m_0.
\]

i.e.

\[
\hat{C}_{m_0 + d - 1 - i} = 0
\]

one can then proceed to show inductively that

\[
\hat{C}_\lambda = 0 \quad \forall \lambda \implies (C_\lambda) = 0
\]
which contradicts our initial assumption that $w > 0$.

Hence $w > d$ and

\[ d_{\min}(C) > d. \]
Eq Binary, (primitive) BCH codes

Here \( q = 2 \), \( N = 2^m - 1 \),

\((=) \text{ the multiplicative order of } q \pmod{m} = m \quad \text{CHECK!} \)

\[ NS \geq \begin{cases} m_0, m_0 + 1 - m_0 + d - 2 \end{cases} \]

i.e. binary, primitive BCH codes have parameters
\[ \left\lfloor N = 2^m - 1, \; N - 1 \text{NS}_c \right\rfloor \geq d_{\text{min}} \geq d \right\rfloor 

A popular choice is \( m_0 = 1 \).

\[ \text{i.e.} \; \text{NS}_c \geq \left\lfloor 1, 2, \ldots, m_0 + d - 2 = d - 1 \right\rfloor 

= \left\lfloor 1, 2, \ldots, 2t \right\rfloor \quad \text{if} \; d = 2t + 1 \]

Note that in the \( \mathbb{Z} \)-cyclotomic cosets mod \( N \) (i.e., the
2- cyclotomic cosets \((\text{mod } 2^m - 1)\)

1) The size of a cyclotomic coset is at most \(m\) since

\[
\chi 2^m \quad \text{(mod } 2^m - 1) \]

\[
= \chi \quad \text{(mod } 2^m - 1) \]

\[
\leq \chi ! \quad \text{for} \quad \chi \rightarrow 2 \chi + 4 \chi \ldots -2 \chi \]
(ii) if \( k \in \{ 1, 2, \ldots, 2t \} \) is even, then \( k = 2x \), then \( x \notin \{ 1, 2, \ldots, 2t \} \).

and \( \{ k, x \} \) belong to the same
cyclic coset.

\( k \) it suffices to ensure that
\( \{ 1, 3, 5, \ldots, 2t - 1 \} \subseteq N_S e \)
\[ \Downarrow \]
set is of size \( t \).
Thus the null spectrum need be no larger than \( mt \).

\[
\dim(\mathcal{R}) = |\text{NS}_c^c| = N - |\text{NS}_c|
\]

\[
\geq 2 - 1 - mt.
\]

BCH code parameters:

\[
\left[ \begin{array}{c}
m \\ 2 - 1, 2 - 1 - mt, d_{\text{min}} \geq 2t + 1 \end{array} \right]
\]
$E = 2$  $N = 15$  $m = 4$

$\lambda_0 = 3$  \implies

Note

$3^2 = 1 \implies$

$\frac{3^n}{3} = \frac{3^{12}}{3} = \frac{3^9}{3}$

$= 1$

as well!
Lemma 3 Let $C$ be a cyclic code having null spectrum $NS_C$. Then $C$ contains the basic sequence of every frequency $\lambda \in NS_C$. 
Lec 42  Decoding Cyclic Codes

Recap

* The null spectrum theorem
* \( \dim(C) = N - 1 \frac{N}{2} \epsilon \)
* BCH codes
  - \( d_{\text{min}} \geq d \)
  - Proof via the
\[ \text{weight theorem} \]

- primitive binary \( \text{BCH codes} \)

- dimension estimate

\[ [2^m-1, \geq 2^m - mt, d_{\text{min}} \geq 2t+1] \]

(parameters of a \( t \)-error-correcting \( \text{BCH code} \)).
Thm (Bch codes) Let $C$ be a cyclic code over $\mathbb{F}_2$ of length $N$, $(N, 2) = 1$, whose null spectrum $NS_C$ contains the consecutive set of frequencies $\{m_0, m_0 + 1, m_0 + 2, \ldots, m_0 + k - 2\} \subseteq \{0, 1, 2, \ldots, N-1\}$.
for some integers $m_0, d$ with $d \geq 2$.

Then,

\[ d_{\min}(R) \geq d \]

and hence $C$ is a \([N, N-\text{IN}_{\text{err}}, d]\) code over $\mathbb{F}_q$.

$d$ is called the designed distance of the BCH code.
Eg (Reed-Solomon codes)

Consider the case \( q = e \)

\( N \mid q - 1 \). Then

\[ q = 1 \pmod{N} \Rightarrow \]

the multiplicative order of \( q \pmod{N} \)

\[ = 1. \]
\[ q = 2^4 \quad N = q - 1 = 15 \]

\[ m_0 = 2 \quad d = 5 \]

\[ \mathcal{N} \subseteq \left\{ m_0, m_0 + 1, \ldots, m_0 + d - 2 \right\} \]

\[ \mathcal{N} \subseteq \left\{ 2, 3, \ldots, 5 \right\} \]

\[ d_{\min} \geq 5 \]
\( q \) cyclotomic cosets \((\text{mod } N)\)

\[ a \sim b \iff a = \gamma^i b \pmod{N} = b \pmod{N} \]

Since \( \gamma^q = 1 \pmod{N} \)!

\[ \therefore a \sim b \iff a = b. \]

Thus each equivalence class contains
only a single element:

\[ \dim (C) = N - (d-1) \]

\[ k = N - d + 1 \geq N - d_{\min} + 1 \]

hence the code is MDS and \( d_{\min} = d \).
General parameters of a \( k - s \) code:

\[
\begin{bmatrix}
N, & N - d + 1, & d
\end{bmatrix}
\]
DECODING BCH CODES
BCH code Setting

\[ q = \frac{e}{\phi(q, N)} = 1 \]

\[ m \text{ is the order } \sum_{\text{mod } N} \]

\[ \mathcal{C} \subseteq \text{cyclic code } \subseteq F_q^2 \]

\[ \lambda \in F_m \] has order \( N \)

\[ N_{\mathcal{C}} \supseteq \{ m_0, m_0 + 1, \ldots, m_0 + d - 2 \} \]
Channel Model:

\[ c_t, e_t, n_t \in \mathbb{F}_q \quad \text{all} \]
\[ 0 \leq t \leq n-1 \]
\[\hat{x} = \sum_{t=0}^{n-1} \Delta z x_t\]

\[= \sum_{t} c_t x_t + \sum_{t} z_t x_t\]

\[= \hat{c} + \hat{z}\]

\[\Rightarrow \hat{r} = c x + m_0 \leq x \leq m_0 + \ell - 2\]
**SYNDROME** $S(2)$

\[ S(2) = \sum_{\lambda=0}^{d-2} \sum_{\lambda+m_0}^2 \]

Set $\pi \equiv d-1$ and define:

\[ S_{\pi}(2) = \sum_{\lambda=0}^{\infty} \sum_{\lambda+m_0}^2 \]
\[
\sum_{i=0}^{n-1} \sum_{t=0}^{t_{\text{mote}}} \lambda (x + \lambda_0) t \geq 2
\]

where

\[
w = \# \text{ of exons in locations } \{t_i\}
\]
\[ \hat{\omega}(2) = \frac{w(2)}{\hat{\sigma}(2)} \quad \text{(say \( F \))} \]

\[ \hat{\sigma}(2) = \prod_{i=1}^{t} (1 - 2x_i^2) \quad \hat{v}_0 = 1 \]

\{ \text{ERROR} \} \quad \{ \text{LOCATOR} \} \quad \{ \text{POLYNOMIAL} \}
\[\omega(z) = \sum \varepsilon_z z \prod_{j=1}^{\text{not } \omega} (1 - z^j \cdot z)\]

**ERROR EVALUATOR POLYNOMIAL**

**Note:**

\[\deg \left( \omega(z) \right) = \omega \leq \left\lfloor \frac{d-1}{2} \right\rfloor = \left\lfloor \frac{n}{2} \right\rfloor\]
\[ \deg \left( w(t) \right) = n - 1 \leq \left\lfloor \frac{n}{2} \right\rfloor - 1 \]

Replacing \( S_0 (z) \) by \( S(z) \)

\[ S(z) = \sum_{\lambda \in \chi + \mathbb{N}_0} z^\lambda \]

\[ \chi = 0 \]

\[ = S_0 (z) \left( \text{mod} \, z^2 \right) \]
\[ s(2) = \frac{w(2)}{\sigma(2)} \pmod{2^n} \]

\[ \therefore s(2) = \frac{w(2)}{\sigma(2)} + \frac{\xi(2)}{\sigma(2)} z^n \]

Power series in $z$ with no neg. exponents
\[ \sigma(2) s(2) - \omega(2) = A(2) \sigma(2) z^2 \]

\[ \text{polynomial } B(2) \]

\[ \omega(2) = \sigma(2) s(2) + B(2) z^2 \]

\[ \deg g \leq \left\lfloor \frac{2}{2^2} \right\rfloor - 1 \]

\[ \deg g \leq \left\lfloor \frac{2}{2} \right\rfloor \]
this is reminiscent of \[ \gcd \]
computation via the Euclidean division algorithm.
DECODING EXAMPLE

$q = 2 \quad N = 15 \quad m = 4 \quad d = 5$

\[ \left\{ m_0, m_0 + 1, \ldots, m_0 + e - 2 \right\} \]

= \{ 1, 2, 3, 4, 5, 6 \}

(Triple-canon connecting BCH code)

Consequent choice of null spectrum:
null spectrum

\[ \pi(t) = \begin{cases} 1 & t \in \{0, 3, 4, 5, 6, 7, 8, 12, 13\} \\ 0 & \text{else} \end{cases} \]
STEP 1: Compute \( \hat{\chi} \)

\[ m_{\text{m}} \leq \chi \leq m_{\text{m}} + d - 2 \]

\[ x_t^{14} \]

\[ \therefore \hat{\chi} = \sum_{t=0}^{d} \]

\[ = 1 + 2 + 3 + 5 + 6 + 7 + 8 + 12 + 13 \]

\[ + 2^2 + 2 \]

\[ \therefore \]

\[ \therefore \]
\[ = 2^0 \left( 1 + 2^3 \right) \left( 1 + 2 \right) \left( 1 + 2 \right) \left( 1 + 2 \right) \left( 1 + 2 \right) \]

\[ = 2^6 \]

\[ \hat{x}_2 = \left[ \hat{x}_{11} \right]^2 = 2^{12} \]
\[ \hat{\mathbf{r}}_4 = [\hat{x}_1]^4 = 2^9 \]

\[ \hat{\mathbf{r}}_3 = 2^{13} \quad (\text{direct computation}) \]

\[ \hat{\mathbf{r}}_5 = 2^5 \quad (\text{direct}) \]

\[ \hat{\mathbf{r}}_6 = [\hat{x}_3]^2 = 2 \]
**STEP 2:**

\[ S(z) = \sum \lambda_i z^i \]

\[ \lambda = 0, \lambda + m \]

\[ S(z) = 1 - 2z + \lambda z^2 + 2z^3 + 3z^4 + \cdots \]

\[ = 2z^2 + 2z^2 + 2z^2 + 2z^2 + \cdots + 12z^6 + 2z^2 + 2 \]

\[ = (1, 5, 9, 13, 12, 6) \]

**SHORTHANDED NOTATION**
<table>
<thead>
<tr>
<th>$x^2$</th>
<th>$s(x)$</th>
<th>Quotient</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>12</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>deg 2</td>
<td>14</td>
<td>2</td>
</tr>
<tr>
<td>$x^2$</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>= $K \omega(2)$</td>
<td>= $K \omega(2)$</td>
<td></td>
</tr>
</tbody>
</table>
\[ \begin{array}{cccccccccc}
1 & 1 & 3 & 1 & 2 & 6 & 1 & 2 & 13 & 4 \\
2 & 2 & 2 & 2 & 2 & 2 & 1 & 0 & 0 & 0
\end{array} \]

Quotient

\[
\begin{array}{cccccccc}
1 & 2 & 9 & 13 & 2^2 & 2 & 2^10 & O \\
2 & 9 & 13 & 2^2 & 2 & 2^10 & 0 \\
2 & 9 & 3 & 2^7 & 2 & 2^11 & 0 & 2^4 \\
\text{Rem} = 1^8 & 1^2 & 1^6 & 0 & 2^4
\end{array}
\]
\[ \begin{array}{c|cccc} \hline 14 & 3 & 2 & 0 \\ \hline \hline & 2 & 12 & 6 & 0 \\ \hline & 2 & 12 & 0 & 0 \\ \hline & 0 & 0 & 0 & 0 \\ \hline \end{array} \]

Quotient: \[ 2^9 \]

Remainder: \[ 0 \]
\[ 1 + (2^2 + 2^{13})(2^3 + 2^2) \]

\[ = 1 + 2^2 2^{17} + 2 (2 + 2^6) + 1 \]

\[ = 2^2 2^{17} + 2 2^{11} \]
\[(2^4 + x^{13}) + (2^9 x^2) (2^2 x^2 + 2^6 x)\]

\[= 2^3 x + 2^2 x^5 + 2^4 x^{13}\]

\[\therefore \prod (1 - 2^\lambda x) = k \left\{ \begin{array}{l}
\lambda = 1 \\
\therefore k = 2^1 \end{array} \right. \]

\[\therefore k = 2^2 \]

\[\therefore \sigma(x) = 2^3 x^3 + 2^2 x^2 + 2^6 x + 1\]
Chien search

2 = 1 \implies 2^3 + 2^7 + 2^6 + 1

= 1 + 2^3 (1 + 2^3 (1 + 2) 2^4

= 2^11 \neq 0
\[ 2 = 2^2 \implies 2^3 2^6 + 2^7 2^9 + 2^6 2^2 + 1 \]

\[ = 1 + 2^6 \left( 1 + 2 \right) \left( 1 + 2^2 \right) \]

\[ = 1 + 2^6 \cdot 2^8 \]

\[ = 0 \]

\[ \therefore 2 = 2^2 \text{ is a zero} \]
Similarly it turns out that

\[
\prod_{i=1}^w \left(1 - \frac{t^i}{2}\right) \quad \text{has}
\]

\[
2 - 2 \quad \text{as the}
\]

3 zeros.

\text{. Exon locations are the}

reciprocals:

\[
\begin{pmatrix}
-2 & -11 & -14 & 13 \\
2 & 2 & 2 & =
\end{pmatrix}
\]
\[ t_1 = 1 \]
\[ t_2 = 4 \]
\[ t_3 = 13 \]
Basics of Block Codes
1. What is the smallest possible minimum distance of a block code of length \( n \) that can correct 2 errors and detect 5 errors? If used only for error-detection, what is the maximum number of errors that the code can detect?

2. A ternary code \( C \) is a code whose symbol alphabet is the set \( \{0, 1, 2\} \), i.e., \( C \) is a subset of \( \{0, 1, 2\}^n \). Even in \( \{0, 1, 2\}^n \), the definitions of Hamming weight and Hamming distance remain as in the binary case. In the binary case, a code is a \((t_d, t_c)\) code iff

\[
d_{\min} \geq t_d + t_c + 1.
\]  

(1)

Is this also true in the ternary case? (The definition of a \((t_d, t_c)\) code remains as in the binary case.) Justify your answer.

Mathematical Preliminaries
3. Prove that if \( G \) is an Abelian group under the operation + and \( H \) is a finite subset of \( G \), then \( H \) is a subgroup of \( G \) if and only if

\[
a + b \in H \quad \text{whenever} \quad a \in H \quad \text{and} \quad b \in H.
\]

Hint: Associativity and commutativity carry over to any subset. If \( a \in H \) consider the list \( \{la \mid 0 \leq l, l \text{ an integer} \} \) (\( la \) is the sum of \( l \) copies of \( a \)). There are only a finite number of distinct elements in this list as \( H \) is finite. Thus \( l_1a = l_2a \) for some distinct integers \( l_1, l_2 \). From this you should be able to conclude the existence of the identity element and of the inverse.

4. Consider the group

\[
G = \mathbb{Z}_2^4 = \{ \text{all binary 4 tuples} \}
\]
Let $H$ be the subgroup

$$H = \{[0 \ 0 \ 0 \ 0], \ [1 \ 1 \ 0 \ 0], \ [1 \ 1 \ 1 \ 1], \ [0 \ 0 \ 1 \ 1]\}.$$  

Define two elements $a, b \in G$ of $G$ to be equivalent, written, $a \equiv b$ if $a + b \in H$. For example,

$$[1 \ 0 \ 0 \ 0] \equiv [0 \ 1 \ 0 \ 0],$$

since

$$[1 \ 0 \ 0 \ 0] + [0 \ 1 \ 0 \ 0] = [1 \ 1 \ 0 \ 0] \in H.$$  

This equivalence relation allows $G$ to be partitioned into 4 subsets of size 4 called equivalence classes where all the 4 elements within a subset are equivalent. Identify the four equivalence classes and the elements that they contain.

5. Provide as best an algebraic characterization as you can (i.e., specify if it is a group, a ring, a field, a vector space, etc) of the set

$$S = \left\{ \sum_{i=2}^{2} a_i z^i \mid a_i \in \mathbb{F}_2 \right\}$$

endowed with the addition operation

$$\sum_{i=0}^{2} a_i z^i + \sum_{i=0}^{2} b_i z^i = \sum_{i=0}^{2} c_i z^i$$

where $c_i = a_i + b_i \pmod{2}$ and with multiplication given by

$$\sum_{i=0}^{2} a_i z^i \sum_{i=0}^{2} b_i z^i = \sum_{i,j=0}^{2} a_i b_i z^{i \oplus j}$$

where $i \oplus j := i + j \pmod{3}$.

6. If $G$ is a non-Abelian group having subgroup $H$, and if for $g_1, g_2 \in G$, we define $g_1 \sim g_2$ iff $g_2^{-1} g_1 \in H$, does this represent an equivalence relation. Justify your answer.

7. Identify coset representatives for the cosets of the subgroup

$$H = \{a_0 + a_2 x^2 \mid a_0, a_2 \in \mathbb{F}_2\}$$

of the group

$$G = \left\{ \sum_{i=0}^{3} a_i x^i \mid a_i \in \mathbb{F}_2 \right\}.$$
**Linear Codes**

8. Consider decoding of the length \( n = 7 \) repetition code using two methods:

   (a) using bounded distance decoding (BDD) with \( d_{\text{min}} = 7 \)
   (b) maximum likelihood decoding (MLD) assuming that the channel is a binary symmetric channel having crossover probability \( \epsilon < \frac{1}{2} \).

Is there a difference between the two methods when applied to this code? Explain your answer.

9. Write down a parity-check (p.c.) matrix for the binary linear code \( C \) whose generator matrix is given by
   \[
   G = \begin{bmatrix}
   1 & 1 & 1 & 0 & 0 \\
   1 & 1 & 0 & 1 & 0 \\
   1 & 0 & 0 & 0 & 1 \\
   \end{bmatrix}.
   \]

10. Identify a \([4,2]\) linear code \( C \) such that \( C^\perp \) is \( C \) itself, i.e., such that the code and its dual are one and the same. Such codes are called *self-dual codes*.

11. For the purposes of this problem, let us define a linear \([n,k]\) block code \( C \) to be *systematic* if there exists a generator matrix for the code such that when the code is encoded using that generator matrix, the first \( k \) code symbols \((c_0, c_1, \cdots, c_{k-1})\) are precisely the \( k \) message symbols, \((m_0, m_1, \cdots, m_{k-1})\).

   Under this definition, is the linear block code \( C \) having generator matrix
   \[
   G = \begin{bmatrix}
   1 & 1 & 0 & 1 & 1 & 1 & 1 \\
   0 & 1 & 1 & 0 & 1 & 1 & 0 \\
   1 & 0 & 1 & 0 & 0 & 1 & 1 \\
   1 & 0 & 1 & 1 & 1 & 0 & 1 \\
   \end{bmatrix}
   \]
   a systematic code?

   Justify your answer in a few words. Show all your working.

12. Consider the linear block code \( C \) having generator matrix given by
   \[
   G = \begin{bmatrix}
   1 & 0 & 0 & 1 & 1 & 1 \\
   0 & 1 & 1 & 0 & 1 & 1 \\
   0 & 1 & 0 & 0 & 0 & 0 \\
   \end{bmatrix}.
   \]

   (a) What is the minimum distance of \( C \)? Explain how you obtained your answer.

   (b) What is the minimum distance of the dual code \( C^\perp \)? Again, explain how you obtained your answer.
13. Determine the minimum distance $d_{\text{min}}$ of the $[7,3]$ linear block code $C$ having parity-check matrix

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

Explain your reasoning and show all your working.

14. Show that the repetition code and the parity-check code are the only possible MDS codes of length $n = 7$. (Hint: Start by attempting to construct an $[n,k]$ MDS code by attempting to build up a parity-check matrix $H$ for the code, one column at a time. Keep in mind that the parity-check matrix has $n - k$ rows and it is required that any $n - k$ columns of $H$ be linearly independent. ) Note: The same proof carries over to any length $n$. However, you are only required to do the case $n = 7$.

15. Use cosets of the linear block code $C$ having parameters $[5,2]$ and generator matrix:

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

to partition the set $F_2^5$, i.e., identify all cosets of $C$.

16. A linear block code $C$ is used to accomplish error-correction over a Binary Symmetric Channel (BSC) with cross-over probability $\epsilon$. The standard array is used to carry out maximum-likelihood decoding (MLD) of the code. Then the probability of codeword error $P_{\text{we}}$ can be determined

(a) just from knowing the weight distribution of the code
(b) just from knowing the list of all coset leaders,
(c) only if both the weight distribution of the code and the list of coset leaders is known
(d) only if the entire standard array is provided.

Identify the most appropriate answer(s).

17. Use the Hamming bound to determine an upper limit to the size of a binary block code of length $n = 15$ and minimum distance $d_{\text{min}} = 7$.

18. Consider the linear block code of length 5 and dimension 2 with the following generator matrix

$$G = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$
(a) Choose as coset leaders the zero vector, all 5-tuples with weight 1 and \{00011, 10001\}. Construct the standard array together with syndromes for complete decoding. (The first row in this table should list the codewords and the first column to the left should contain all the coset leaders. The last column should list the corresponding syndromes.)

(b) Given that the received vector

\[ r = [1 \, 1 \, 1 \, 0 \, 1]^t, \]

what is the decoded codeword? What is the residual error? How many message bits are in error? You may assume that the transmitted codeword is the all-zero codeword.

Repeat for the case

\[ r = [1 \, 1 \, 0 \, 0 \, 0]^t. \]

Again, you may assume that the transmitted codeword is the all-zero codeword.

(c) When the code is used only for the purposes for correcting error, what is the probability \( P_{we} \) of codeword error when the crossover probability of the BSC is \( \epsilon = 10^{-4} \)?

(d) On the same channel as in the previous problem, what is the probability of undetected error if the code is used solely for the purposes of error detection?

(e) With systematic encoding, the codeword corresponding to the message vector \([m_0, m_1]\), is given by \([m_0, m_1, p_0, p_1, p_2]\). On the same channel as in the previous problem, what is the probability that the first message symbol will be in error if the code is used solely for the purpose of error correction?

(f) On the same channel as in the previous problem, what is the probability that of the two message bits \([m_0, m_1]\), only the second message bit \(m_1\) is decoded incorrectly?

19. Let \( C \) be an \([n, k, d]\) linear code having \((k \times n)\) generator matrix \(G\). Prove that any collection of \(n-d+1\) columns selected from the \(n\) columns of \(G\) is a linearly independent set.

20. The covering radius of a linear \([n, k]\) code \(C\) is the smallest integer \(\rho\) such that for any \(x \in \mathbb{F}_2^n\), there exists a codeword \(c \in C\) such that

\[ d_H(c, x) \leq \rho. \]

(a) How would you determine \(\rho\) from a standard array decoding table of the code?

(b) How would you determine \(\rho\) from inspection of the parity-check matrix \(H\) of the code?

(c) What is the covering radius of the \([7, 4, 3]\) Hamming code?

21. Derive the analogue of the Hamming bound as it applied to ternary codes, i.e., to codes having the ternary alphabet \(\{0, 1, 2\}\).
Convolutional Codes

22. In the field of formal power series $\mathbb{F}_2[[D]]$, find the first 7 terms in the power-series expansion of

(a) $\frac{1}{1 + D^2}$
(b) $\frac{D}{1 + D + D^2}$
(c) $\frac{D^2}{1 + D^2 + D^3}$

23. Determine whether the convolutional codes encoded using the $G(D)$ below are catastrophic. If so, find an infinite weight input sequence that generates a codeword of finite weight.

(a) $G(D) = [1 + D + D^3, 1 + D + D^2, 1 + D^2 + D^3]$.
(b) $G(D) = [1 + D^3, 1 + D + D^2 + D^4, 1 + D^2 + D^3 + D^4]$.

**Hint:** The irreducible polynomials of degree $\leq 3$ over $GF(2)$ are listed below:

- degree 1: $D, 1 + D$
- degree 2: $1 + D + D^2$
- degree 3: $1 + D + D^3, 1 + D^2 + D^3$.

24. Consider the rate 1/2 convolutional code with

$G(D) = [1 + D + D^2, 1 + D^2]$.

(a) Draw a complete trellis diagram up to node level 6 (beginning at node level 0). Label all branches with code symbols.
(b) Use the trellis to determine the free distance $d_{\text{free}}$ of the code.
(c) If the received sequence (across a BSC) is

$r = (01 \ 00 \ 01 \ 00 \ 00 \ 00 \ldots.)$

find (the information sequences associated to) all survivors at node level 6.
(d) If the received sequence (across an AWGN channel) is

$r = (4 \ -1 \ -3 \ 2 \ 6 \ -5 \ 2 \ 4 \ 5 \ 3 \ 5 \ 5 \ldots.)$

find (the information sequences associated to) all survivors at node level 6.

25. Consider a rate 1/3 convolutional code with

$G(D) = [1 + D \ 1 + D^2 \ 1 + D + D^2]$.

(a) Draw the state diagram for the encoder.
(b) Compute the generating function \( A_F(L = 1, D, I) \)
(c) Use this generating function to determine the free distance \( d_{\text{free}} \) of the code.

26. Will the choice of generator matrix,
\[
G(D) = \begin{bmatrix}
1 + D + D^2 + D^3, & 1 + D^2 + D^3 + D^5, & 1 + D^4
\end{bmatrix},
\]
cause the associated convolutional code \( C \) to exhibit catastrophic error propagation? Explain fully your answer.

27. This question pertain to convolutional codes of rate \( \frac{1}{n} \), with \( m \) memory elements in the encoder that are required to encode a given set of \( N \) message symbols \( \{u_i\}_{i=0}^{N-1} \) that are i.i.d. and equally likely to be 0 or 1. Under the conventional encoding of message symbols using a terminated convolutional code, the convolutional encoder is forced to begin and end at the all-zero state. In encoding using a “tail-biting” however, the only restriction that is placed is that the encoder is required to begin and end at the same state, but this state could be any of the possible encoder states. Derive an expression for the exact rate of the convolutional code when operated in tail-biting fashion. [Hint: How many code symbols does the tail-biting convolutional encoder need to transmit?]

**The Generalized Distributive Law**

28. Consider the “min-star” semi-ring \((\mathbb{R}, \infty), \min^*, +)\) in which the \( \min^* \) operation is given by:
\[
\min^*(x, y) := -\ln(e^{-x} + e^{-y}).
\]

(a) Identify the identity element under the min-star operation
(b) Verify that the distributive law holds.

29. Consider the single parity-check code of length 3 having parity check matrix \( H = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \). Thus \( \mathbf{v}^T = [v_1, v_2, v_3] \) is a codeword if and only if \( Hv = 0 \). In a certain instance, when communicating over a binary symmetric channel (BSC) having crossover probability \( \epsilon < 0.5 \), the received vector was found to be
\[
\mathbf{y}^T = [0, 1, 0].
\]

Use the GDL to carry out ML code-symbol decoding of this code. Show all intermediate steps

(a) the formulation as an MPF problem
(b) the junction tree
(c) the message-passing schedule and the messages passed
30. Consider the rate $\frac{1}{2}$ convolutional code having polynomial generator matrix

$$G(D) = [1, \ 1 + D].$$

On a certain transmission, two message symbols $u_0$ and $u_1$ were encoded using the code and then transmitted across a binary symmetric channel (BSC) having crossover probability $\epsilon < \frac{1}{2}$. The corresponding received symbols were

$$y^T = [01 \ 01].$$

Use the GDL and the junction tree shown below to carry out maximum a posteriori (MAP) decoding of ONLY the message bit $u_1$. (Note that the encoder is NOT returned to the all-zero state, i.e., there are no tail bits inserted into the message stream).

Show ALL intermediate steps and all your working clearly.

31. Write down the distributive law as it applies to the semi-rings numbered 5,6,8,9 in Table I of the “GDL paper” (the paper by Aji and McEliece).

32. Read Example 2.2 of the “GDL paper” and determine the savings in computation between using the brute force approach to computing the 8 transform coefficients $F(x_1, x_2, x_3)$ and the approach that makes intelligent use of the distributive law.

33. Consider the rate $1/2$ convolutional code with

$$G(D) = [1 + D + D^2 \ 1 + D^2].$$

If the received sequence (across a discrete memoryless AWGN channel) is

$$r = (4 \ -1 \ -3\ 2 \ 6 \ -5),$$
use the GDL algorithm to implement minimum probability of bit error decoding of the message bits $u_0, u_1, u_2$. Show all your working including the graphs that you use and the message passing schedule.

34. Consider decoding the $[7, 4, 2]$ linear block code for the case when the received vector across a binary symmetric channel with crossover probability $\epsilon << 1$, is the vector $y = [1 0 0 0 0 0 0]^T$.

Use the GDL to make decisions based on maximizing the a-posteriori probabilities $p(u_i/y)$ of the code symbols $u_i, i = 1, 2, 3, 4, 5, 6, 7$.

35. Verify that the $\text{min}^*$ sum semi-ring is in fact a semi-ring, starting from the definition of the $\text{min}^*$ operation:

$$\text{min}^*(x, y) = \min\{x, y\} - \ln(1 + e^{-|y-x|}) = -\ln(e^{-x} + e^{-y}).$$

Identify the underlying set and the identity element under the $\text{min}^*$ operation.

36. Set up a schedule for computing the objective function at vertex $W$ for Example 2.4 of the GDL paper. Draw the corresponding message trellis.

37. Write down the distributive law as it applies to the semi-rings numbered 9,10 in Table I of the “GDL paper” (by Aji and McEliece).

38. Set up an efficient message-passing schedule for computing the objective function at vertex $W$ for Example 2.4 of the GDL paper, i.e., identify the sequence in which you would pass messages.

39. In the computation:

$$\beta(x_3) = \sum_{x_1, x_2, x_4, x_5} f(x_1)g(x_2)h(x_1, x_2, x_3)p(x_3, x_4)q(x_3, x_5),$$

all the variables $x_i, i = 1, 2, 3, 4, 5$ take on values from an alphabet $\mathcal{A}$ of size $|\mathcal{A}| = q$. If you were to reorganize this expression to minimize the number of operations (additions and multiplications), how would you do it and how many operations would you end up needing?

40. Consider the problem of computing

$$F(x_1) = \sum_{x_2=0}^9 \sum_{x_3=0}^9 \sum_{x_4=0}^9 f(x_2, x_3, x_4)g(x_3, x_4)h(x_1, x_2, x_4).$$

The functions $f(\cdot), g(\cdot), h(\cdot)$, are all real-valued functions.
(a) It is desired to pose this problem as a marginalize a product function problem. Identify the corresponding universal set, the corresponding local domains and the local and global kernels.

(b) Organize if possible these local domains into a junction tree. Make clear all your working.

41. Consider maximum-likelihood code-symbol decoding of the binary block code having parity check matrix $H$ given by

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

Thus all codewords $c = [c_1, c_2, c_3, c_4, c_5, c_6, c_7]^T$, satisfy $Hc = 0$. The decoding problem when posed as an marginalize-a-product function problem, leads to the graph shown in Fig. 2. Use the distributive law in conjunction with this graph, to efficiently compute the probabilities $\Pr(c_4 = 0)$ and $\Pr(c_4 = 1)$ where the received vector $r$ is given by $r = [1111111]^T$. You may assume that the channel is a binary symmetric channel (BSC) having crossover probability $0 < \epsilon << 1$.

**A suggested approximation**  In your computations, you will run into expressions of the form

$$a_i\epsilon^i + a_{i+1}\epsilon^{i+1} + a_{i+2}\epsilon^{i+2} + \cdots + a_{i+k}\epsilon^{i+k},$$

where $i \geq 0, k \geq 0$ and the $a_i$ are integers $\leq 10$. It is suggested that whenever you encounter such an expression, you make the approximation

$$a_i\epsilon^i + a_{i+1}\epsilon^{i+1} + a_{i+2}\epsilon^{i+2} + \cdots + a_{i+k}\epsilon^{i+k} \approx a_i\epsilon^i.$$

42. Consider the joint probability function

$$p(\{u_i\}_{i=0}^3, \{s_i\}_{i=0}^4, \{y_i\}_{i=0}^3) = p(s_0) \prod_{i=0}^3 p(u_i)p(s_{i+1}/s_i, u_i)p(y_i/s_i, u_i)$$

associated with a convolutional code. As in class the $\{u_i \in \{0, 1\}\}$ represent the binary message symbols, the $\{s_i\}$ is the state sequence and $\{y_i\}$ are the received symbols.

Consider the problem of maximum-likelihood code-symbol decoding of this code, i.e., of computing $p(u_k/\{y_i\}_{i=0}^3)$, $0 \leq k \leq 3$.

(a) Present this as an MPF problem,

(b) organize the local domains into a junction tree

(c) show that message passing can be organized into a forward wave and a backward wave and that the forward wave is in essence, a sequence of matrix multiplications

Note: It is NOT necessary to do anything beyond what is asked above!
Figure 2: Junction Tree associated to the [7, 4, 2] code.

**LDPC Codes**

43. In the density evolution analysis of \((d_v, d_c)\)-regular LDPC codes, where the goal is to determine the evolution of the density of number of incorrect messages passed between variable nodes and check nodes, it is customary to assume that the all-1 codeword is transmitted. What are the assumptions on the channel and the processing carried out at the variable and check node under which this assumption is valid? Explain your answer while making clear any notation that you introduce.

44. Derive from first principles, the transformation of densities that takes place during an iteration at a check node.

45. Consider density evolution associated to Gallager Decoding Algorithm A applied to an LDPC code \(C\). Thus the channel is a BSC with cross-over probability \(\epsilon << 1\) and all messages passed are either 1 or \(-1\). You may assume that the neighborhood of every node in the Tanner graph of \(C\) is tree-like to depth 8. What is the probability \(p_{i}^{(1)}\) that at the end of iteration 1, the message passed from a variable node to check node will be in error. Notation is as in class. Following an initial round of message passing, from the variable nodes to check nodes, based only on channel inputs, each subsequent iteration is composed of two rounds of message passing: from check node to variable node followed by from variable node back to check node. Show all your
working clearly. You may use the fact that $\epsilon << 1$ to simplify calculations. Hence $a\epsilon^2$ for integer constants $a < 100$ (say) may safely be ignored in comparison with $\epsilon$, etc.

46. Consider the variation of belief propagation decoding of binary LPDC codes in which, in place of beliefs, the messages passed correspond to log-likelihood ratios (as discussed in class).

(a) Identify (it is not necessary to derive them) the variable and check node maps

$$\psi_v^{(0)}(l_0), \ \psi_v^{(1)}(l_0, l_1, \ldots, l_{d_v-1}), \ \psi_c^{(l)}(l_1, l_2, \ldots, l_{d_c-1}).$$

(b) Do these maps satisfy the variable-node and check-node symmetry conditions which (along with the channel symmetry condition) permit us to conclude that the number of incorrect messages passed is the same regardless of the transmitted codeword? Make clear your reasoning.

47. Is the computational tree associated with variable node 11 (at the top of the graph and incorrectly labelled as node 10 :-) ) in the Tanner graph in Fig. 3 of a certain LDPC code a junction tree? If so, identify the associated MPF problem along with the local domains and the local kernels. What is the objective function being computed if messages are passed as indicated by the arrows?

![Figure 3: Computational tree associated to node 10 in Tanner Graph.](image)

48. (a) In the context of the performance analysis of LDPC codes, state (in terms of the notation introduced in class), the variable and check-node symmetry assumptions that go into showing that the probability of passing an incorrect message is independent of the transmitted codeword.
(b) Show clearly that the check-node symmetry condition holds when LDPC codes are decoding using belief propagation with log-likelihood ratios (LLR) in place of beliefs.

**Finite Fields & Cyclic Codes**

49. Use the Euclidean division algorithm (EDA) to determine the \( \text{gcd} \) of 6711 and 831. Express the \( \text{gcd} \) as a linear combination \( u \ast 6711 + v \ast 831 \) of 6711 and 831.

50. Find the inverse of 7 modulo 13 using the EDA.

51. Identify all primitive elements of the finite fields of size 7 and 13 (the finite field of size 13 is the set of all integers modulo 13).

52. Over GF(2), compute if possible, the inverse of \((1 + x)\) modulo \((1 + x + x^2 + x^3 + x^4)\).

53. Let \( \alpha \) be a primitive element of GF(64). Identify all the elements in all the subfields of GF(64) in terms of \( \alpha \).

54. Use the irreducible polynomial (irreducible over GF(5)) \( x^2 + x + 2 \) to construct a finite field of 25 elements. If \( \alpha \) denotes a root of \( x^2 + x + 2 \), then \( \alpha \) is known to be primitive in GF(25). Set up an add-1 table for GF(25). Identify the 5-cyclotomic cosets modulo 24. Find the minimal polynomials of all elements in the field. Compute the product of all the minimal polynomials (each distinct polynomial is taken just once) including the minimal polynomial \( x \) of the zero element. Which powers of \( \alpha \) constitute the subfield GF(5) of GF(25) ?

55. Identify the 3-cyclotomic cosets modulo 26 as well as the 2-cyclotomic cosets modulo 19.

56. Let \( \alpha \) be a primitive element of GF(2^6). Identify all the correct answers below with a √

   - \( \alpha + \alpha^4 \in GF(4) \)
   - \( \alpha + \alpha^8 \in GF(8) \)
   - none of the above

57. The polynomials over GF(2) given below are all irreducible. Identify with a √, all those having the property that all of their zeros are contained in GF(256).

   - \( x^2 + x + 1 \)
   - \( x^3 + x^2 + 1 \)
   - \( x^4 + x^3 + 1 \)
• $x^5 + x^2 + 1$
• $x^6 + x + 1$
• $x^8 + x^6 + x^5 + x^4 + 1$

58. Identify the smallest finite field of characteristic 2 that contains a primitive 17-th root of unity.

59. In the notation used in class with regard to finite field Fourier transforms, let $q = 2$, $N = 15$ and $\alpha$ be a primitive element of $F_{16}$ satisfying $\alpha^4 + \alpha + 1 = 0$. Let 

$$(s(t), \; t = 0, 1, 2, \ldots, 14) \; = \; 00011000101101.$$ 

Compute the Fourier transform $\hat{s}(\lambda)$ of $s(t)$. Compute also the Fourier transform of $s(t) + s(t+2)$.

60. In the notation used in class with regard to finite field Fourier transforms, let $q = 2$, $N = 15$ and $\alpha$ be a primitive element of $F_{16}$ satisfying $\alpha^4 + \alpha + 1 = 0$. Determine the basic sequence $b(t)$ of “frequency” $\lambda = 6$. Determine the Fourier transform of the sequence $c(t)$ given by $c(t) = b(2t)$, $0 \leq t \leq 14$.

61. Consider the binary (i.e., $q = 2$) cyclic code of length $N = 15$ consisting of all binary codewords $(c(t), \; 0 \leq t \leq 14)$ satisfying 

$$\mathcal{C} = \{c(t) \mid \hat{c}(\lambda) = 0, \; \lambda = 0, 7, 14, 13, 11\}.$$ 

Transforms are computed using a primitive element of $GF(16)$ satisfying $\alpha^4 + \alpha + 1 = 0$. Find a codeword $B(t) \in \mathcal{C}$ such that every codeword $c(t)$ in $\mathcal{C}$ can be expressed as a linear combination of cyclic shifts of $B(t)$, i.e., can be expressed in the form

$$c(t) = \sum_{\tau=0}^{14} u(\tau) B(t-\tau)$$

where $u(\tau) \in \{0, 1\}, \forall \tau$.

62. Determine the number of binary sequences $\{a_t\}$ of period $N = 15$ that satisfy the condition

$$\hat{a}_\lambda \in \{0, 1\}, \text{ all } \lambda, \; 0 \leq \lambda \leq 14.$$ 

63. Let $q = 2$ and $N = 23$. What is the order $m$ of $q \pmod{N}$ ? Let $\alpha$ be a primitive $N$-th root of unity lying in $GF(2^m)$. Determine the dimension of the binary $q = 2$ cyclic code of length $N = 23$ all of whose codewords $c(t)$ satisfy

$$\hat{c}(\lambda) = 0, \lambda = 1.$$ 

64. Why are there no interesting linear, cyclic binary codes of length $N = 19$ ?
65. Design a single-error correcting, double-error detecting binary linear, cyclic code of length 21. Naturally you would like to have dimension $k$ as large as possible.

66. Identify the null spectrum of a Reed-Solomon (RS) code over $GF(9)$ code of length $N = 8$ and designed distance $d_{\text{min}} = 6$.

67. How many distinct binary cyclic codes of length 41 are there? (Include in your count, the cyclic code corresponding to the set of all binary 41-tuples as well as the cyclic code consisting of just the all-zero codeword).

68. Consider the binary cyclic code $C$ of length $N = 15$ with null spectrum $\{1, 2, 3, 4, 5, 6, 8, 9, 10, 12\}$. You may assume that transforms are computed using primitive element $\alpha \in GF(16)$ satisfying $\alpha^4 + \alpha + 1 = 0$.

Does the all-one codeword $(1, 1, \ldots, 1, 1)$ belong to the cyclic code $C$? Explain your answer.

69. Use the irreducible polynomial (irreducible over $F_3$) $x^2 + x + 2$ to construct a finite field of 9 elements. If $\alpha$ denotes a root of $x^2 + x + 2$, then $\alpha$ is known to be primitive in $F_9$.

(a) Set up an add-1 table for $F_9$.
(b) Identify the 3-cyclotomic cosets modulo 8.
(c) Find the minimal polynomials of all elements in the field.

70. Use the Möbius inversion formulae to determine the number of irreducible polynomials of degree 12 over the binary field $F_2$.

71. If $\alpha, \beta$ in $F_{16}$ have orders $a, b$, then is it always true that $\alpha \beta$ has order $= \text{lcm}(a, b)$? Justify your answer.

72. (a) How many binary cyclic codes of length 23 are there?
(b) Design a double-error-correcting cyclic code of length 23 and identify its dimension.