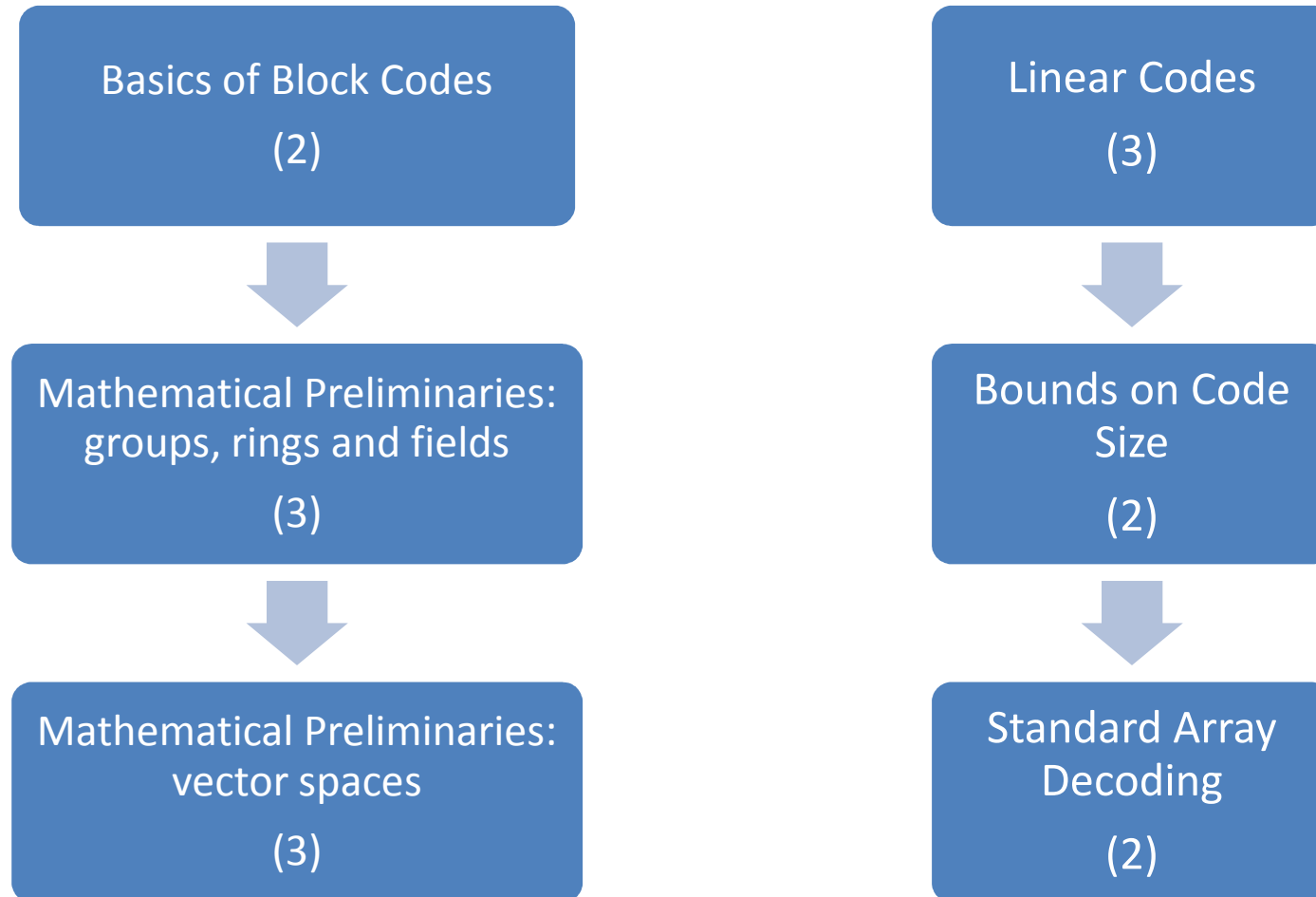


# ERROR - CORRECTING CODES

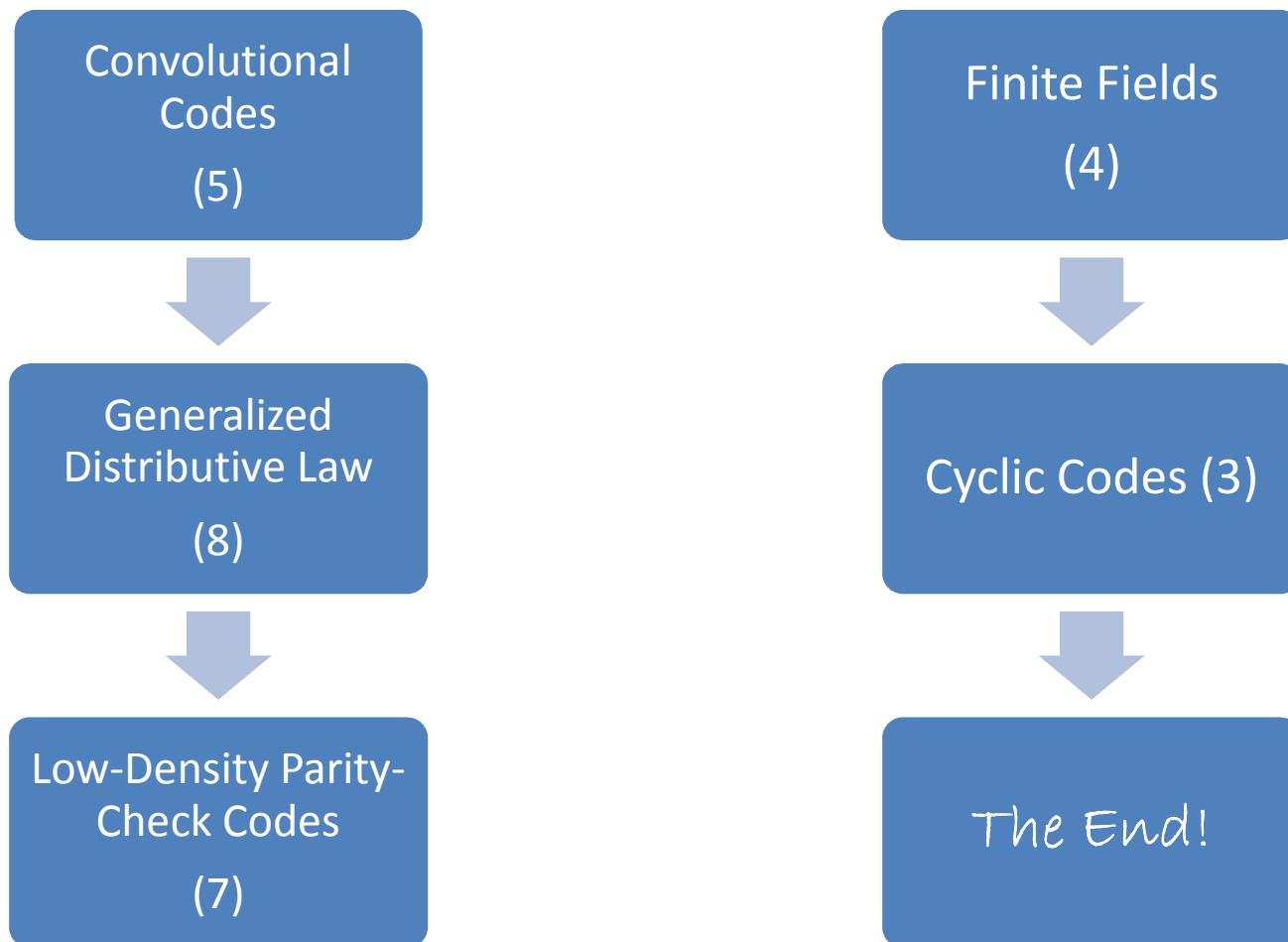
P. VIJAY KUMAR

# Course Outline:

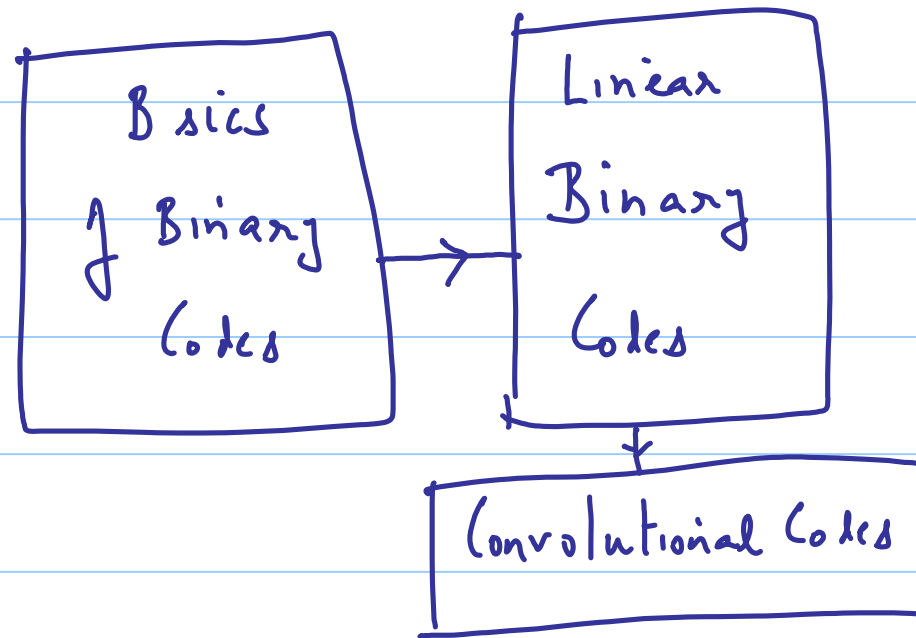
## Lectures 1-15



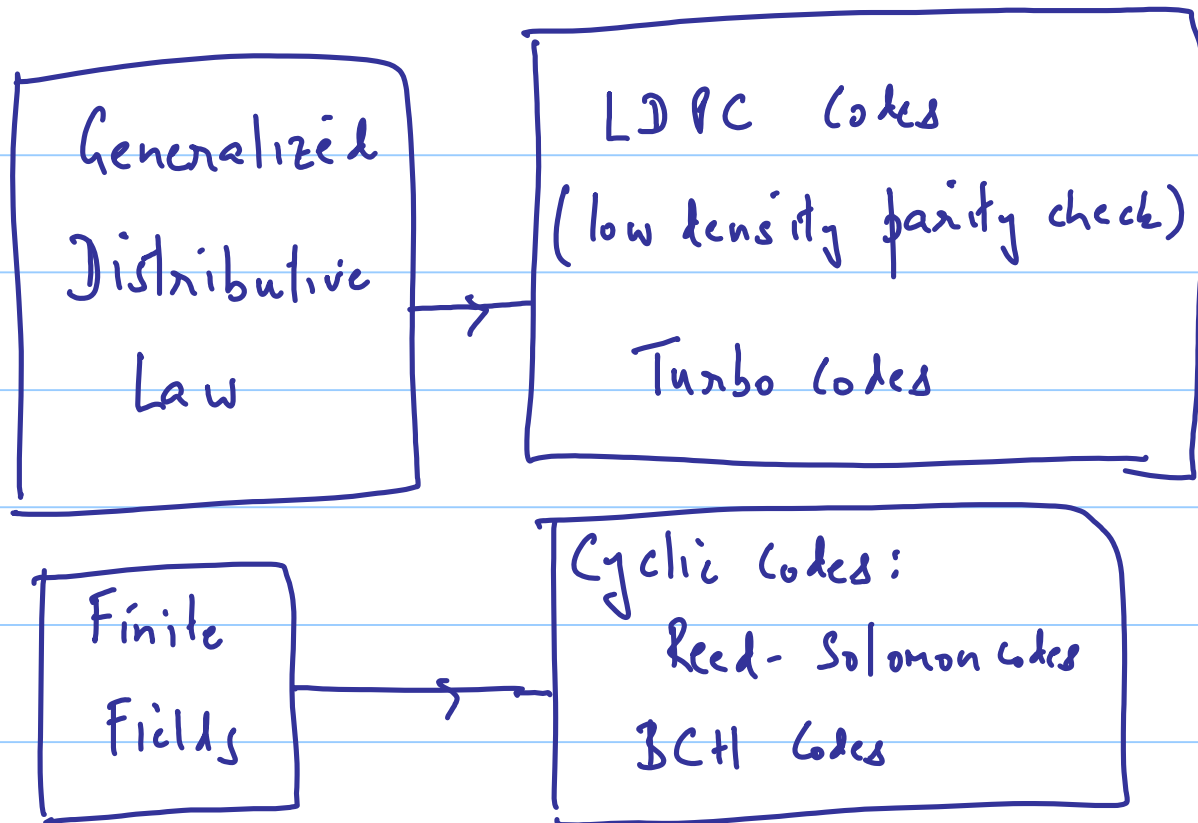
# Course Outline: Lectures 16-42



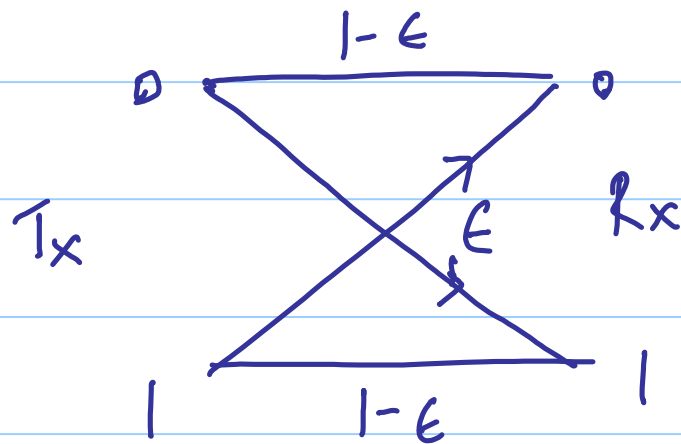
# Lec 1: Course Overview } Basics



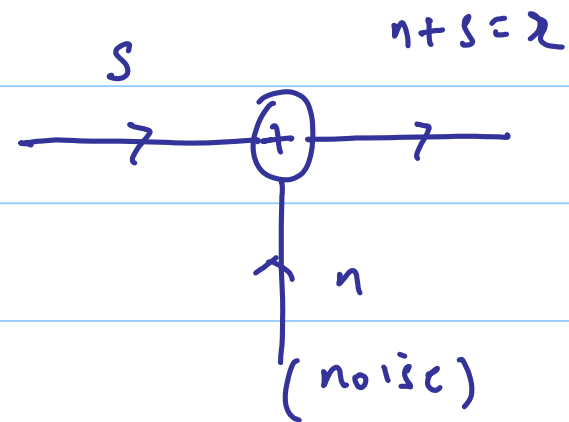




## Channel Model



Binary Symmetric  
Channel (BSC)



Additive White  
Gaussian Noise  
Channel

$\mathbb{F}_2 = \{0, 1\}$  arithmetic is modulo 2

+	0	1
0	0	1
1	1	0

ADDITION  
(XOR)

.	0	1
0	0	0
1	0	1

MULTIPLICATION  
(AND)

$$\mathbb{F}_2^n = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \mid x_i \in \mathbb{F}_2 \right\}$$

Ex  $n=2$   $\mathbb{F}_2^2 = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\}$

In general  $|\mathbb{F}_2^n| = 2^n$ .

## Hamming Weight

Definition The Hamming weight  $w_H(\underline{x})$

of a vector  $\underline{x} \in \mathbb{F}_2^n$  is the number of non zero components in  $\underline{x}$ .

Ex  $n=3$   $\underline{x} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$ ,  $w_H(\underline{x}) = 2$ .

## Properties

(i)  $w_H(\underline{x}) \geq 0$  with equality holding  
iff (if and only if)

$$\underline{x} = \underline{0}$$

(ii)  $w_H(\underline{x} + \underline{y}) \leq w_H(\underline{x}) + w_H(\underline{y})$

Ex.  $\underline{n} = 5$

$$\underline{x} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \quad \underline{y} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \underline{z} = \underline{x} + \underline{y}$$

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

$$w_H(\underline{z}) = 4 \leq w_H(\underline{x}) + w_H(\underline{y})$$
$$= 4 + 2 = 6.$$

Define

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

(Schor  
or componentwise  
product)

$$W_H(\underline{x} + \underline{y}) = W_H(\underline{x}) + W_H(\underline{y})$$

Ex

$$\underline{x} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \quad \underline{y} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \underline{x} \odot \underline{y} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$- 2 W_H(\underline{x} \odot \underline{y})$

$$W_H(\underline{x} + \underline{y}) = 4 + 2 - 2 = 4$$

## Hamming Distance

Definition The Hamming distance

$d_H(\underline{x}, \underline{y})$  between two vectors  
 $\underline{x}, \underline{y} \in \mathbb{F}_2^n$  is defined by:

$$d_H(\underline{x}, \underline{y}) = w_H(\underline{x} + \underline{y})$$

$$\text{Ex } \underline{x} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \quad \underline{y} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \underline{x} + \underline{y} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$d_H(\underline{x}, \underline{y}) = w_H(\underline{x} + \underline{y}) = 4$$



## Properties

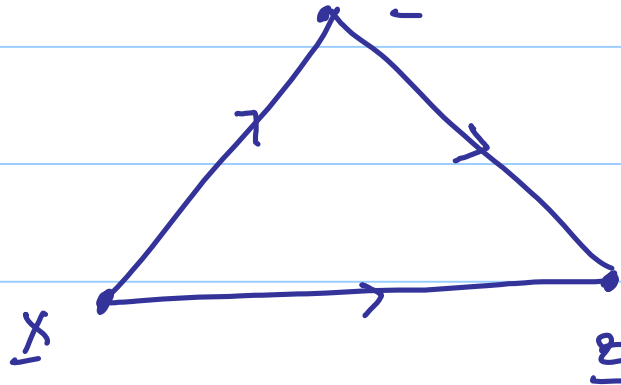
(i)  $d_H(\underline{x}, \underline{y}) \geq 0$  with equality holding if  $\underline{x} = \underline{y}$

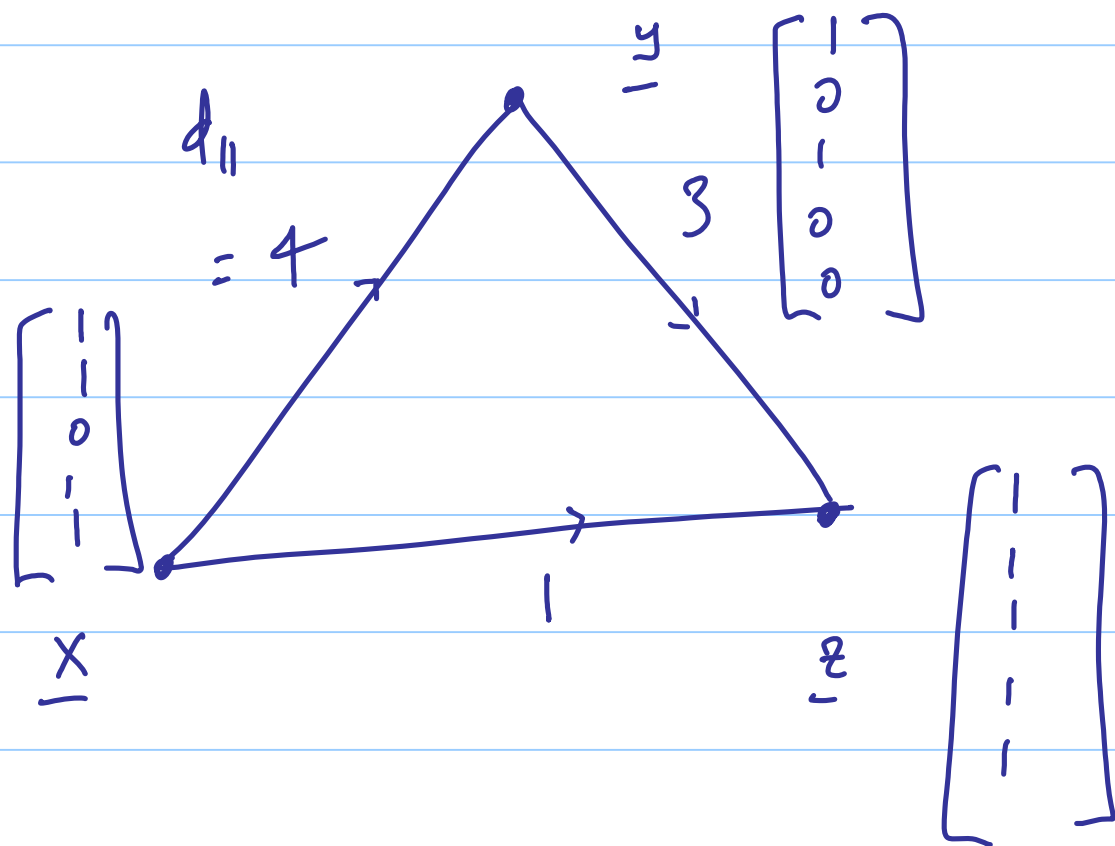
$$(ii) d_H(\underline{x}, \underline{y}) = d_H(\underline{y}, \underline{x})$$

(iii) Triangle Inequality

$$d_H(\underline{x}, \underline{z}) \leq d_H(\underline{x}, \underline{y})$$

$$+ d_H(\underline{y}, \underline{z})$$

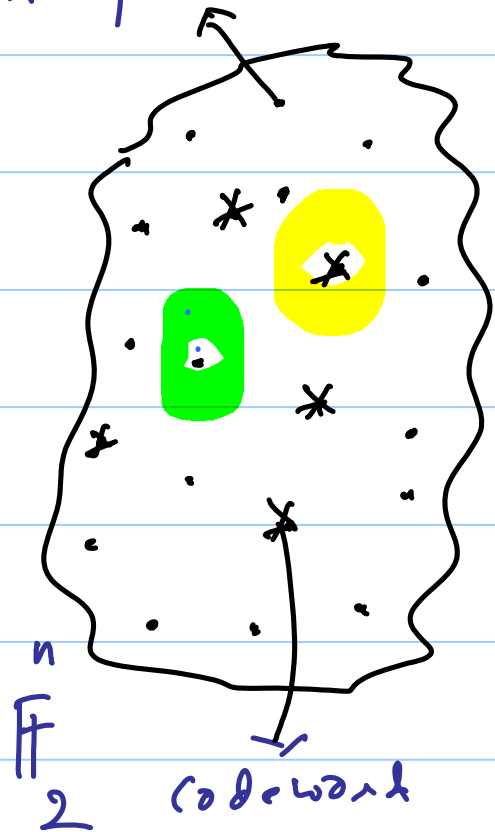




## Binary (Block) codes $n$ -tuple

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

The elements of the code are called codewords.



## Parameters of a code $\mathcal{C}$

- (1) size of a code  $\mathcal{C} = |\mathcal{C}| = \left. \begin{array}{l} \# \text{ of} \\ \text{code words} \\ \text{in the code} \end{array} \right\}$
- (2) rate  $R$  of  $\mathcal{C} = \frac{\log_2 |\mathcal{C}|}{n}$

where  $n$  is the block length of the code

(meaning that  $\mathcal{C} \subseteq \mathbb{F}^n$ )

2

③ block length  $n$  of the code itself

④ the minimum distance  $\Delta$  definition

$$d_{\min}(\mathcal{C}) = \Delta$$

$$\min \left\{ d_H(\underline{x}, \underline{y}) \mid \begin{array}{l} \underline{x}, \underline{y} \in \mathcal{C} \\ \underline{x} \neq \underline{y} \end{array} \right\}$$

# REFERENCES – ERROR-CORRECTING CODES

P. VIJAY KUMAR

1. Shu Lin, Daniel J. Costello, *Error Control Coding - Second Edition*, (often used as a course textbook) Pearson Education Inc. Pearson Prentice Hall, 2004.
2. W. Cary Huffman, and Vera Pless, *Fundamentals of Error-Correcting Codes*, (block coding emphasis) Cambridge University Press, 2010.
3. Ron M. Roth, *Introduction to Coding Theory*, (block coding emphasis) Cambridge University Press, 2006.
4. Tom Richardson and Ruediger Urbanke, *Modern Coding Theory*, (emphasis on LDPC and related codes) Cambridge University Press, 2008.
5. F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error - Correcting Codes*, (older but classical book on block coding) North-Holland Mathematical Library, 1977.
6. Stephen B. Wicker, *Error Control Systems for Digital Communication and Storage*, (block coding emphasis) Prentice Hall Englewood Cliffs, NJ 07632, 1995.
7. Keith Chugg, Achilleas Anastasopoulos and Xiaopeng Chen, *Iterative Detection: Adaptivity, Complexity Reduction, and Applications*, (iterative decoding emphasis) Kluwer Academic Publishers, 2001.
8. Rolf Johannesson and Kamil Sh. Zigangirov, *Fundamentals of Convolutional Coding*, (convolutional coding only) IEEE Press, 1999.
9. Andrew J. Viterbi and Jim K. Omura, *Principles of Digital Communication and Coding*, (nice discussion on convolutional coding) McGraw-Hill Book Company, 2009.
10. William E. Ryan and Shu Lin, *Channel Codes: Classical and Modern*, (recent book treating both block and LDPC codes) Cambridge University Press, 2009.
11. Chris Heegard and Stephen B. Wicker, *Turbo Coding*, (early book on turbo codes) Kluwer Academic Publishers, 2010.
12. T. R. N. Rao and Eiji Fujiwara, *Error-Control Coding for Computer Systems*, Prentice Hall series in computer engineering, Jan 1989.
13. Hideki Imai, *Essentials of Error-Control Coding Techniques*, Academic Press, 1990.
14. Ezio Biglieri, Dariush Divsalar, Peter J. McLane and Marvin K. Simon, *Introduction to Trellis-Coded Modulation With Applications*, (trellis-coded modulation) Macmillan Publishing Company, 1991.

15. V.S.Pless and W.C.Huffman, *Handbook of Coding Theory : Volumes I & II*,(two volume handbook on coding theory: block and convolutional codes, allied topics) North Holland, 1998.
16. Richard E. Blahut, *Algebraic Codes for Data Transmission*,Cambridge University Press, 2011.
17. Richard E. Blahut, *Theory and Practice of Error Control Codes*,Addison-Wesley Publishing Company, 1983.
18. George C. Clark Jr. and J. Bibb Cain, *Error-Correction Coding for Digital Communications*,(an older book, but makes for easy reading) Plenum Press, New York, June 1981.
19. Irving S. Reed and Xuemin Chen, *Error-Control Coding for Data Networks* , Kluwer Academic Publishers, 1999.
20. R. J. McEliece, *The Theory of Information and Coding*,Cambridge University Press, 2004.
21. Elwyn R. Berlekamp, *Algebraic Coding Theory Revised 1984 Edition*,Aegean Park Press, 1984.
22. Stephen B. Wicker and Vijay K. Bhargava, *Reed-Solomon Codes and Their Applications*,IEEE Press, 1999.
23. Scott A. Vanstone and Paul C. van Oorschot, *An Introduction to Error Correcting Codes with Applications*,Kluwer Academic Publishers, 1989.
24. J. van Lint and G. van der Geer, *Introduction to Coding Theory and Algebraic Geometry*,Birkhauser Verlag, 1989.
25. V.D.Goppa, *Mathematics and Its Applications V.D.Goppa Geometry and Codes*,Kluwer Academic Publishers, 1988.
26. Richard B. Wells, *Applied Coding and Information Theory for Engineers*,Prentice Hall Information and System Sciences Series, 1998.
27. Peter Sweeney, *Error Control Coding: From Theory to Practice* , John Wiley and Sons, Ltd, 2002.
28. Shu Lin,Tadao Kasami,Toru Fujiwara and Marc Fossorier, *Trellises and Trellis-Based Decoding Algorithms for Linear Block Codes*,Kluwer Academic Publishers, 1998.
29. H. Stichtenoth , *Algebraic Function Fields and Codes*,Springer-Verlag Berlin Heidelberg, 2010.

### **Book Chapter**

30. P. V. Kumar, M. Win, H-F. Lu, C. Georghiades, “Error-Control Coding Techniques and Applications,” invited chapter in the handbook, *Optical Fiber Telecommunications IV*, edited by Ivan P. Kaminow and Tingye Li, Spring 2002.

## Journal Articles

31. Srinivas M. Aji and Robert J. McEliece, “Generalized Distributive Law” *IEEE Trans. Inform. Theory*, March 2000.
32. T. J. Richardson, A. Shokrollahi, and R. Urbanke, Design of capacity-approaching low-density parity-check codes, *IEEE Trans. Inform. Theory*, vol. 47, Feb. 2001.
33. A. J. Viterbi, “Convolutional codes and their performance in communication systems,” *IEEE Transactions on Communications Technology*, , October 1971.



## Lecture 2:: Example Codes and their parameters.

Eg 1 The repetition code. (all of our example codes will have block length  $n = 7$ )

$$\mathcal{C} = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

Parameters:

(i) size = 2

(ii) rate =  $\frac{\log_2 |\mathcal{C}|}{n} = \frac{1}{7}$

(iii)  $d_{\min}(\mathcal{C}) = 7$

Fig 2. Single Parity Check Code (SPC Code)

$$C = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_7 \end{bmatrix} \mid \sum_{i=1}^7 x_i = 0 \right\} \quad \text{(modulo 2 sum = 0)}$$

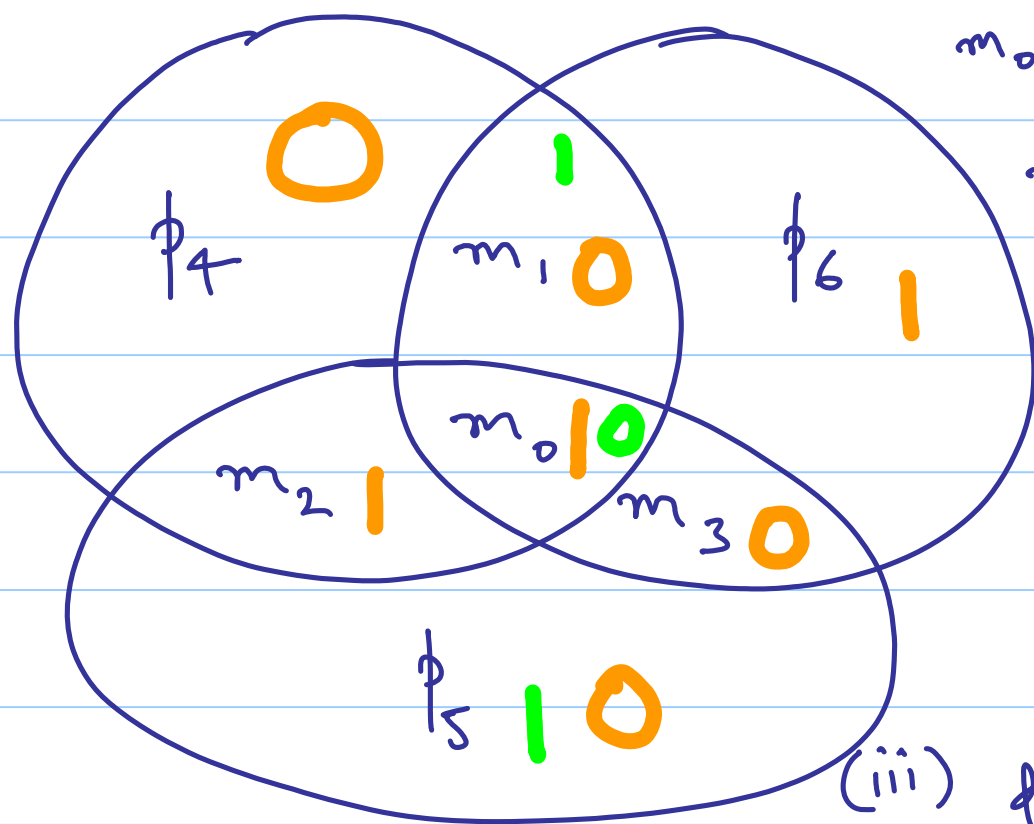
Code Parameters

- (i) size = 2<sup>6</sup>
- (ii) rate =  $\frac{6}{7}$

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

Fig 3  $H = \left\{ \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right\}$

Fig 3 The Hamming code of block length  $n=7$ .



$$m_0 + m_1 + m_2 + p_4 = 0$$

$$m_0 + m_2 + m_3 + p_5 = 0$$

$$m_0 + m_1 + m_3 + p_6 = 0$$

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

code parameters:

can show that

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

$d_{\min} = 3$  (Exercise)

(ii) rate  $= \frac{4}{7}$

Definition A  $(t_c, t_d)$  code is a code in which

any combination of  $\leq t_c$  errors can be deleted 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  $t_d \geq t_c$ ).

Theorem 1 A binary block code  $C$  is a

$(t_c, t_d)$  code iff :

$$d_{\min}(C) \geq t_c + t_d + 1$$

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

Adopt the following decoding algorithm:

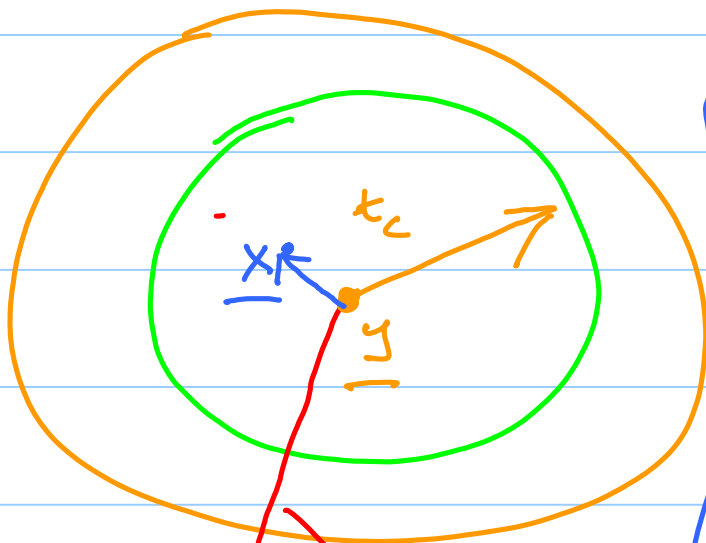
Let  $\underline{z}$  be the received vector.

Let for any vector  $\underline{a} \in \mathbb{F}_2^n$

define

$$B(\underline{a}, r) = \left\{ \underline{z} \in \mathbb{F}_2^n \mid d_{\mathbb{F}_2}(\underline{a}, \underline{z}) \leq r \right\}$$

$\mathbb{F}_2^n$



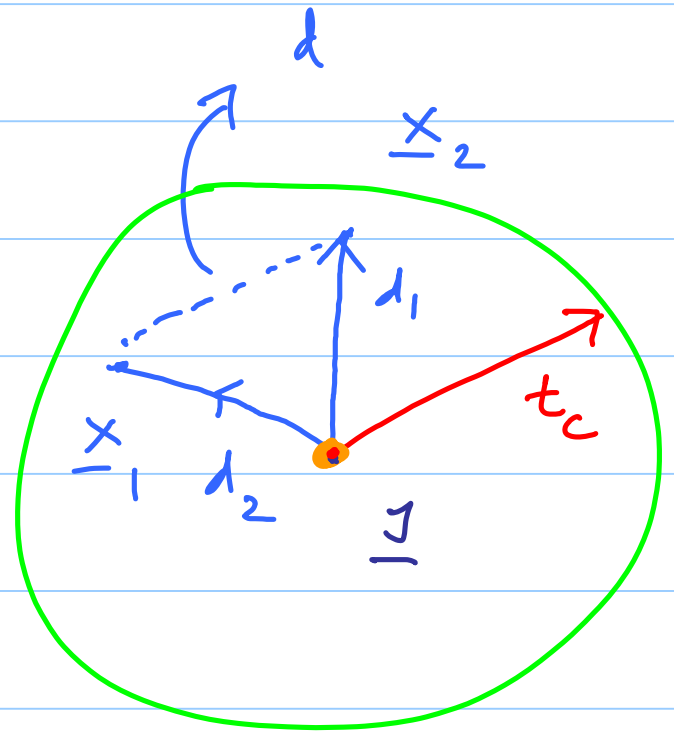
$\underline{x} \leq t_d$   
 $\underline{x} > t_c$

— if  $\mathcal{B}(\underline{y}, t_c)$  contains a codeword  $\underline{x}$ ,  
then we will declare  $\underline{x}$  to be the  
transmitted codeword.

— if not, we will declare that an uncorrectable  
# of errors have occurred.

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

If  $d_H(\underline{y}, \underline{x}_1) \leq t_c$



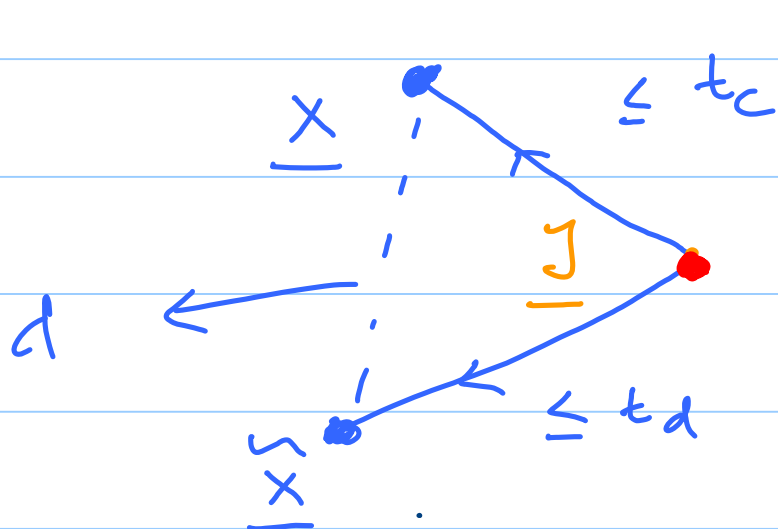
and  $d_H(\underline{y}, \underline{x}_2) \leq t_c$

$\Rightarrow$  (by the  $\Delta$  inequality) that

$$d_H(\underline{x}_1, \underline{x}_2) \leq 2t_c \leq t_c + t_d$$

$$< t_c + t_d + 1$$

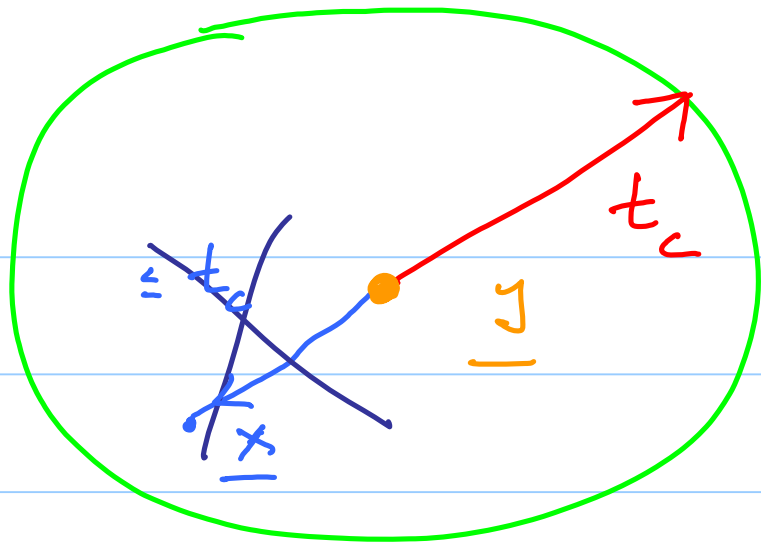
(contradiction)



$$d_H(\underline{x}, \underline{\tilde{x}}) \leq t_c + t_d$$

$$< t_c + t_d + 1$$

$$= d_{\min}(K).$$



suppose there is no  
code word to be found  
in  $B(\underline{y}, t_c)$ .

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  $\underline{y}$   
but this is impossible by our initial assumption  
that the ball  $B(\underline{y}, t_c)$  was empty.



If (of the only if part)

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

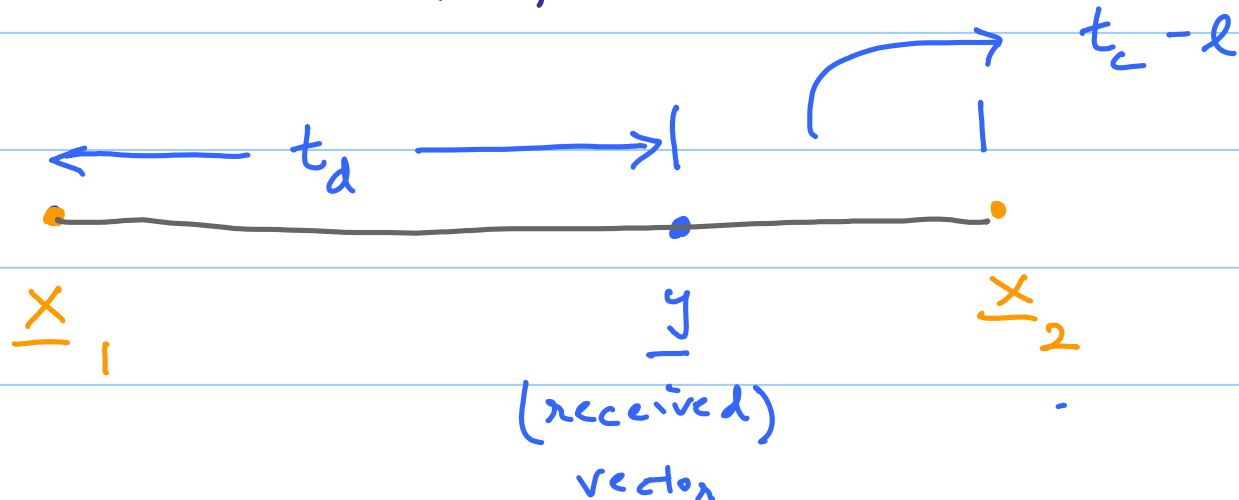
Suppose not  $\Rightarrow d_{\min} < t_c + t_d + 1$

$$d_{\min} = t_c + t_d - \ell \quad (\text{say})$$

$\ell \geq 0$

$\Rightarrow \exists$  a pair  $(\underline{x}_1, \underline{x}_2)$  in  $\mathcal{C}$   
(there exists)

such that  $d_H(\underline{x}_1, \underline{x}_2) = t_c + t_d - \ell$ .



The situation above presents the receiver with a dilemma that cannot be resolved for the case when  $\underline{y}$  is the received vector.

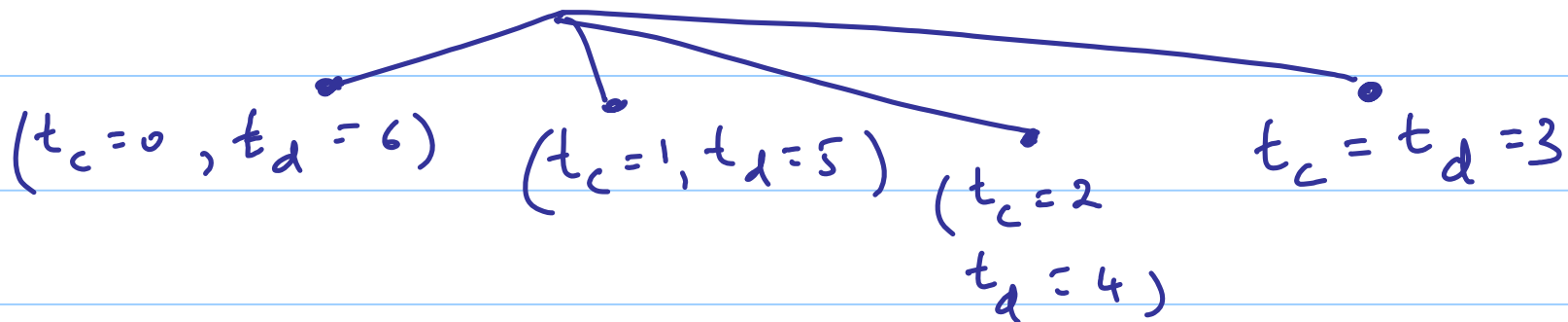
Thus when  $d_{\min} < t_c + t_d + 1$ ,

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

Example a). Repetition code

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

$$\therefore t_c + t_d + 1 \leq 7$$



Eg b). the single parity-check code.

$$d_{\min} = 2$$

$$t_c + t_d + 1 \leq d_{\min}$$

$$t_c = 0, t_d = 1$$

Eg c) Hamming code.

$$d_{\min} = 3$$

$$t_c + t_d + 1 \leq d_{\min}$$

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

$$(t_c = 1, t_d = 1)$$

# Lec 3

## Mathematical Preliminaries

### Groups

Defn 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. INVERSE

$$a \cdot a^{-1} = a^{-1} \cdot a = e.$$

pronounced "a inverse"

Furthermore in the case of Abelian groups, we also have that

$$(v) \quad a \cdot b = b \cdot a \quad \forall \quad a, b \text{ in } G \quad \text{COMMUTATIVE PROPERTY}$$

Note: Abelian groups are also called commutative groups.

Eg.  $(\mathbb{F}_2^n, +)$  mod 2 addition (componentwise)

$n = 3$   $G = \left\{ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}$

$$\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

can verify that the axioms are satisfied:

$\underline{0} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$  is the identity element.

$$a = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \quad a^{-1} = ? \quad a^{-1} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = a \quad !!$$

this is an example of an Abelian group.

Ex Consider  $(G, \cdot) = (\mathbb{Z}_n, +)$  ↗ modulo-n addition.

$$\mathbb{Z}_n = \left\{ 0, 1, 2, \dots, n-1 \right\}$$

$$a + b = \text{Rem} \left( \frac{a+b}{n} \right)$$

identity = 0

$$a^{-1} = (n-a)$$

Abelian group.

## Derived Properties

(i) the identity element is unique

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

$$\Rightarrow e_1 + e_2 = e_1$$

"

$$e_2 \Rightarrow e_1 = e_2$$

(ii) every element has a unique inverse.

Suppose both  $c$  and  $b$  are

inverses of  $a \Rightarrow c \underset{1}{a} b = c(ab) = c(e) = c$

$$(c \underset{1}{a}) b$$

"

$$(c) b = b$$

$$\therefore b = c$$

$$(iii) \quad (a b)^{-1} = b^{-1} a^{-1} \quad (\text{Exercise})$$

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

(v) cancellation law holds:

$$\text{i.e., if } ca = cb \Rightarrow a = b$$

$$\underline{\text{pf}} \quad c^{-1}(ca) = c^{-1}(cb)$$

$$\Rightarrow (c^{-1}c)a = (c^{-1}c)b$$

$$\Rightarrow ea = eb$$

$$\Rightarrow a = b$$

$$(vi) \quad a^m = \underbrace{a \cdot a \cdot a \cdots a}_{m \text{ times}}$$

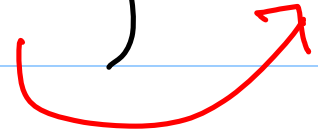


# Lec 4: $\begin{cases} \text{Subgroups and} \\ \text{Equivalence Relations} \end{cases}$

---

(a further example of a group)

Ex  $(G, \cdot) = (\mathbb{Z}_n^*, \cdot)$  multiplication



nonzero elements in

the set of integers modulo  $n$

$$\mathbb{Z}_n^* = \{1, 2, 3, \dots, n-1\} = \mathbb{Z}_n \setminus \{0\}$$

Ex  $n = 6$   $(\mathbb{Z}_6^*, \cdot)$   $\mathbb{Z}_6^* = \{1, 2, 3, 4, 5\}$

1. CLOSURE
2. ASSOCIATIVE
3. IDENTITY ELEMENT
4. INVERSE

5. COMMUTATIVE (for Abelian groups only)

$$2 \cdot 3 = 0 \pmod{6}$$

violates  
closure

$\therefore$  not a  
group!

$$\underline{\mathbb{F}_p} \left( \mathbb{Z}_p^*, \cdot \right)$$

$$p = \text{prime}$$

$$\left\{ \begin{array}{l} \text{CL} \quad \checkmark \\ \text{ASSOC} \quad \checkmark \\ \text{I.E.} \quad \checkmark \quad (=1) \\ \text{INVERSE} \end{array} \right.$$

$$\text{COMMUT.} \quad \checkmark$$

$$\underline{\mathbb{F}_p} \quad p=5 \quad \left( \mathbb{Z}_5^*, \cdot \right)$$

$$a \cdot b = 0 \pmod{p}$$

$$\Rightarrow a \cdot b \triangleq \text{Rem} \left\{ \frac{a \times b}{p} \right\}$$

$$\Rightarrow p \mid a \cdot b \Rightarrow p \mid a \text{ or } p \mid b$$

$$\Rightarrow \begin{array}{l} a = 0 \pmod{p} \text{ or } \text{else} \\ b = 0 \pmod{p} \end{array}$$

do inverses exist in  $\mathbb{Z}_p^*$ ?

$$\frac{p=5}{\hline}$$

$$(2)^{-1} = ?$$

$$\mathbb{Z}_p^* = \{1, 2, 3, 4\}$$

$$\left\{ \begin{array}{l} 2 \cdot 1 = 2 \\ 2 \cdot 2 = 4 \\ 2 \cdot 3 = 6 = 1 \pmod{5} \\ 2 \cdot 4 = 8 = 3 \pmod{5} \end{array} \right. \Rightarrow 3 = 2^{-1} !!$$

note: all entries on the R+1s above are forced to be distinct since

by cancellation:

$$2 \cdot a = 2 \cdot b \Rightarrow a = b$$

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

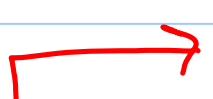
a similar proof can be used to show  
that every element of  $\mathbb{Z}_p^*$  has an  
inverse and hence  $(\mathbb{Z}_p^*, \cdot)$  is  
a group under multiplication.

---

# SUBGROUPS

Defn A subgroup  $(H, \cdot)$  of a group  $(G, \cdot)$  is a subset  $H$  of  $G$  such that  
(s.t.)  $(H, \cdot)$  is a group by itself.

Ex 1  $H = G$  clear ✓

Ex 2  $H = \{e\}$   identity element in  $G$ .

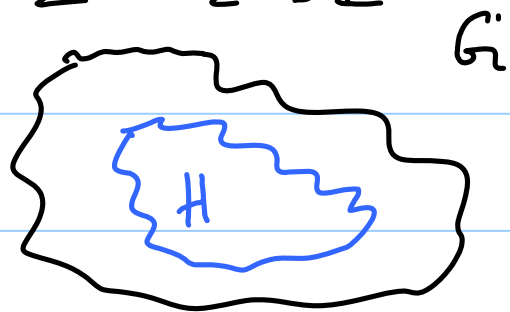
easy to check that  $(\{e\}, \cdot)$  is  
a subgroup.

(called trivial examples).

## Testing for a subgroup

Brute force:

- CL
- ASSOC
- I. E
- INVERSE



Better:

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

$\downarrow$   $(G, \cdot)$  we will  
simply say that

$H$  is a subgroup of  $G$ .

Lemma  $H \subseteq G$  is a subgroup iff

$$a, b \in H \Rightarrow a \cdot b^{-1} \in H$$

Pf.

CL

Assoc ✓

I.E. ✓

INV ✓

$$\begin{aligned} \text{Setting } a = b &\Rightarrow a b^{-1} = e \\ &\Rightarrow e \in H \end{aligned}$$

$$\text{Setting } a = e, \Rightarrow b^{-1} \in H$$

CLOSURE: given  $a, b \in H$ ,  
we know that  $b^{-1} \in H$

$$\Rightarrow a (b^{-1})^{-1} \in H$$



$$\Rightarrow ab \in H \quad //$$

there is a further simplification in the case of groups containing a finite # of elements :

Lemma If  $H$  is a finite subset of  $G$ ,  
then  $H$  is a subgroup of  $G$  'iff :

$$a, b \in H \Rightarrow a \cdot b \in H$$

Pf (left as an exercise!)

Hint: given  $a \in H$ , consider

$a, a^2, a^3 \dots$

this is an infinite sequence, yet  $H$

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  $A$  provided:

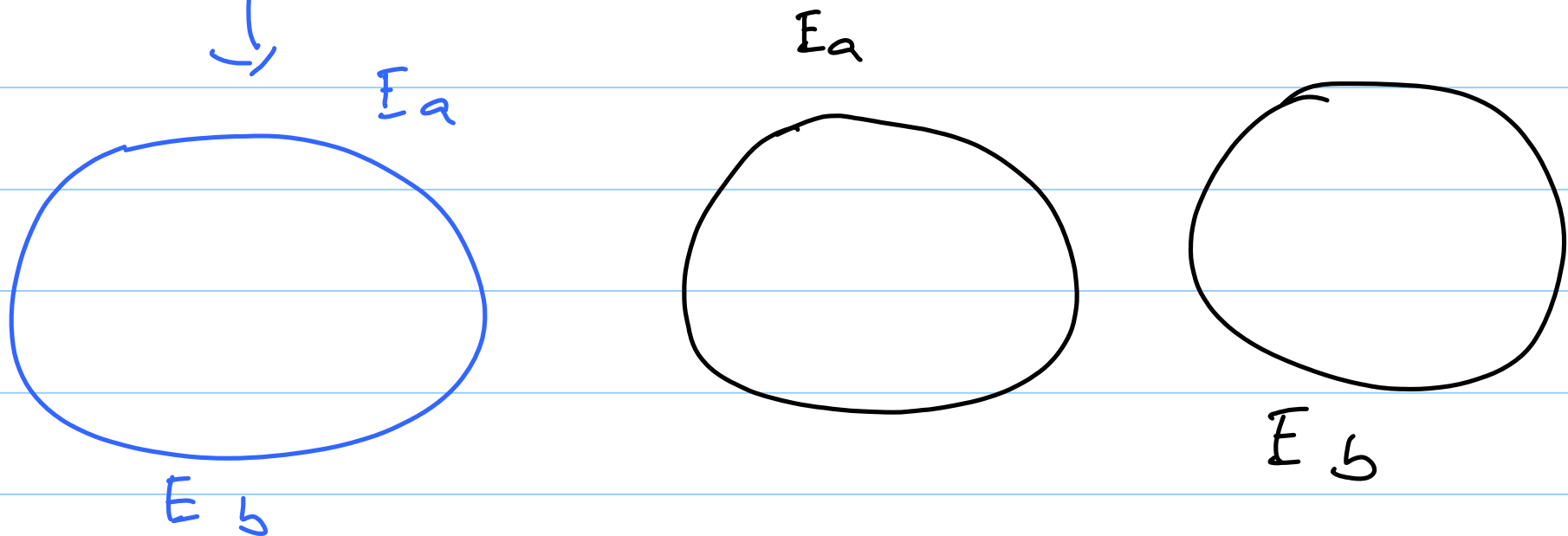
(i)  $a \sim a$  REFLEXIVE

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

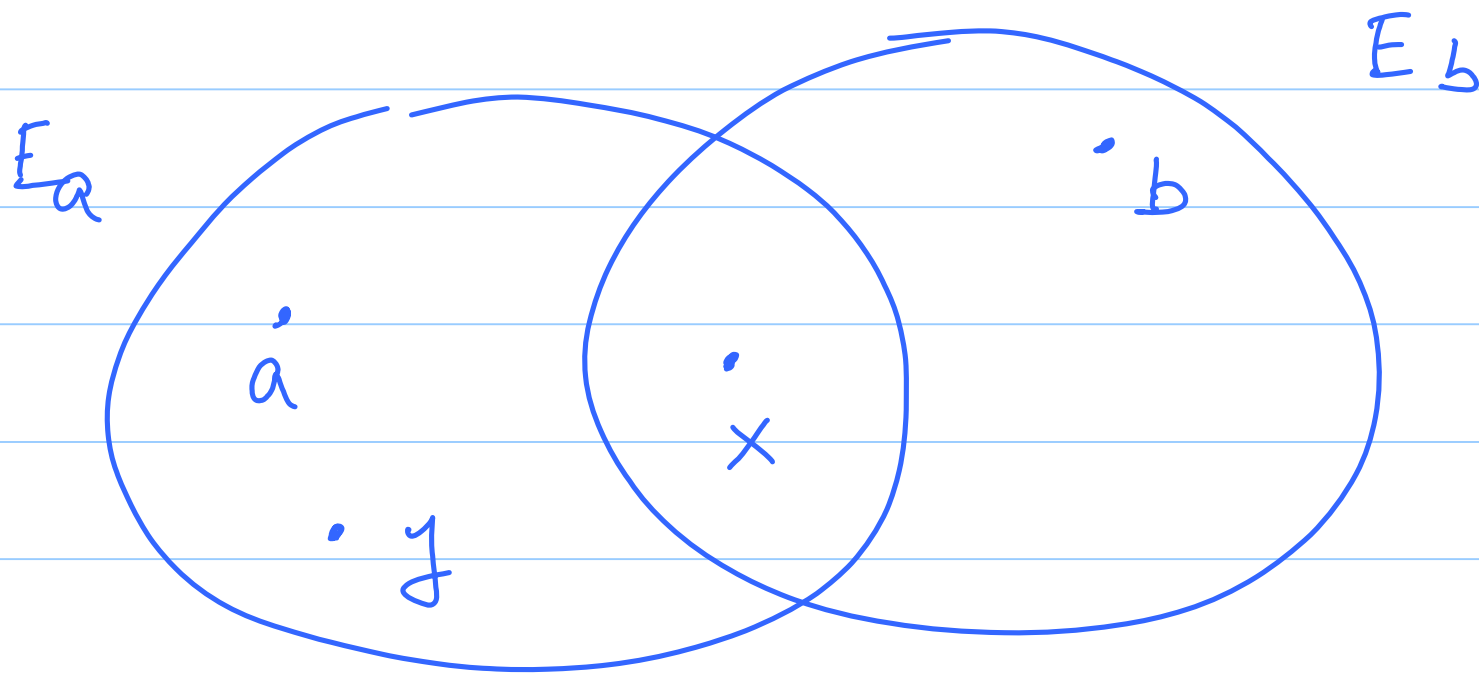
(iii)  $a \sim b, b \sim c \Rightarrow a \sim c$   
TRANSITIVE

CLAIM: If  $R$  is an equivalence relation then if  $a, b \in A$ , 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:)



Suppose  $j \in E_a \setminus E_b$

Suppose  $x \in E_a \cap E_b$

$$\Rightarrow y \sim x, \quad x \sim b \Rightarrow y \sim b$$

$$\Rightarrow y \in E_b$$

a contradiction!

---

cosets of a subgroup.

---

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

let us define

$$a \sim b \text{ if } a \cdot b^{-1} \in H$$

$$\text{i.e., } R = \left\{ (a, b) \in G \times G \mid a \cdot b^{-1} \in H \right\}$$

CLAIM: This is an equivalence relation. ✓✓

Pr. { REFLEXIVE ✓  $a \sim a$ ?  $\Rightarrow a a^{-1} \in H$ ?  
 SYMM ✓ but  $a a^{-1} = e, e \in H$   
 (since  $H$  is a subgroup  
 TRANS. ✓  $\therefore a \sim a$  )

SYMM:  $a \sim b \Rightarrow b \sim a$ ?

(=)  $a b^{-1} \in H \Rightarrow b a^{-1} \in H$



but since  $(a b^{-1})^{-1} = b a^{-1}$ , and

$H$  is a subgroup  $\Rightarrow b a^{-1} \in H$

$\therefore$  yes!

TRANS       $a \sim b, \quad b \sim c \Rightarrow a \sim c?$

$$\begin{array}{ccc} \Downarrow & & \Downarrow \\ a b^{-1} \in H & & b c^{-1} \in H \end{array}$$

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

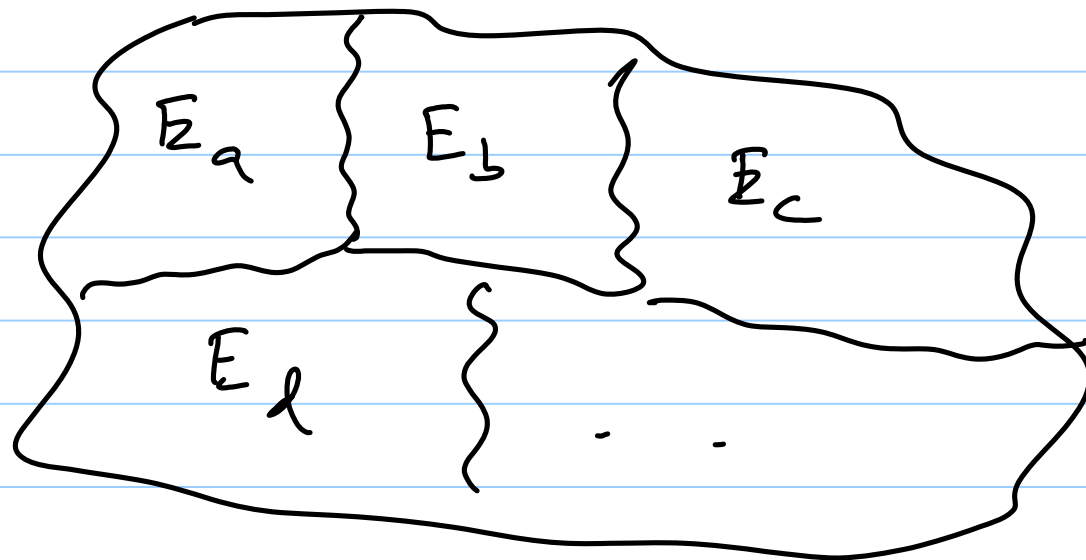
$$\Rightarrow a \underbrace{b^{-1} b} c^{-1} \in H$$

$$= a e c^{-1} \in H$$

$$\Rightarrow a c^{-1} \in H \Rightarrow a \sim c$$

## Lec 5: Cosets, Rings & Fields

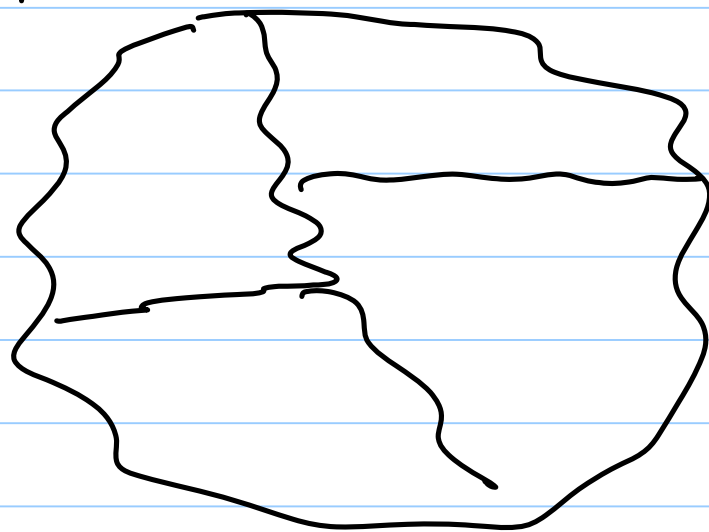
We saw last time that an equivalence relation partitions the set.



$\{ (G, \cdot) \text{ group}$

$\{ (H, \cdot) \text{ subgroup}$

$\{ a, b \in G \quad a \sim b \text{ if } ab^{-1} \in H$   
saw last time that this is an  
equivalence relation.  $G$



$$a b^{-1} \in H \Rightarrow a b^{-1} = h \in H$$

$$\Rightarrow a = h b, \quad h \in H$$

$\therefore a \sim b$  iff  $a \in H b$  where

$$H b = \{ h \cdot b \mid h \in H \}$$

$$\therefore \boxed{E_b = H b}$$

Subsets of  $G$   
 of this form  
 are called  
 cosets of  $H$  in  $G$ .

Eg  $(G, \cdot) = (\mathbb{Z}_6, +)$

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

(check that this is a subgroup  
- Exercise!)

$$a \sim b \quad \text{iff} \quad ab^{-1} \in H$$

the equivalence classes are all of  
the form:

$$H \cdot b \Leftrightarrow H + b$$

$$H = \{0, 2, 4\}$$

$$b = 0 \Rightarrow H + b = H \text{ itself}$$

$$b = 1 \Rightarrow H + 1$$

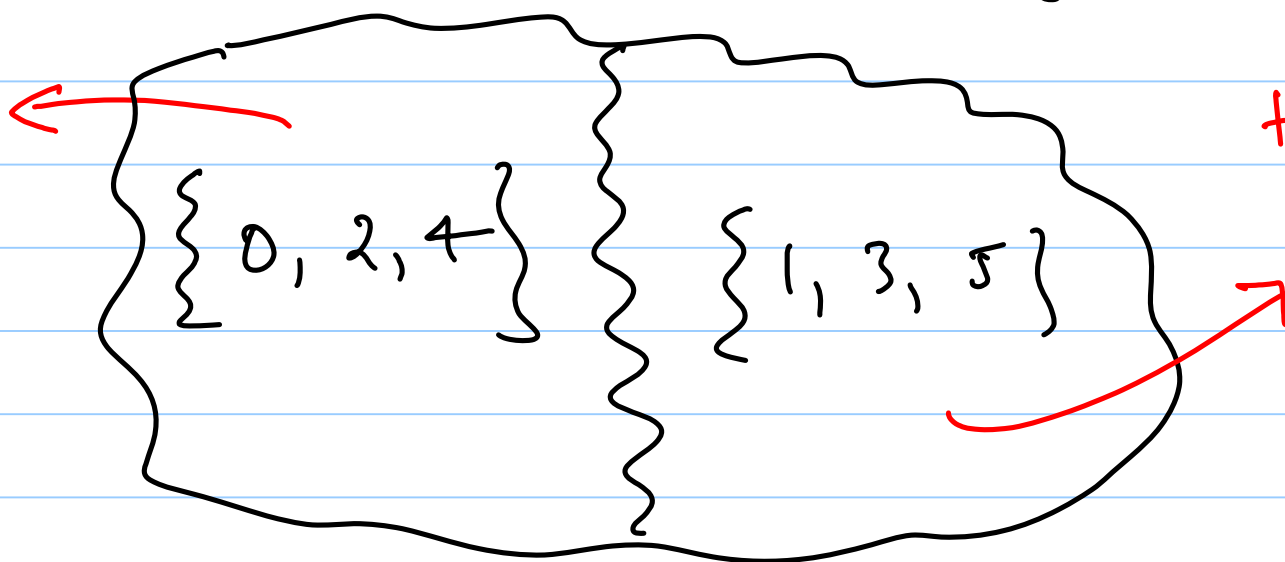
$$= \{ h + 1 \mid h \in H \}$$

$$= \{ 1, 3, 5 \}$$

$\mathbb{Z}_6$

coset:

$H + 0$



$H + 1$

Fig 2 (of partitioning into equivalence classes)

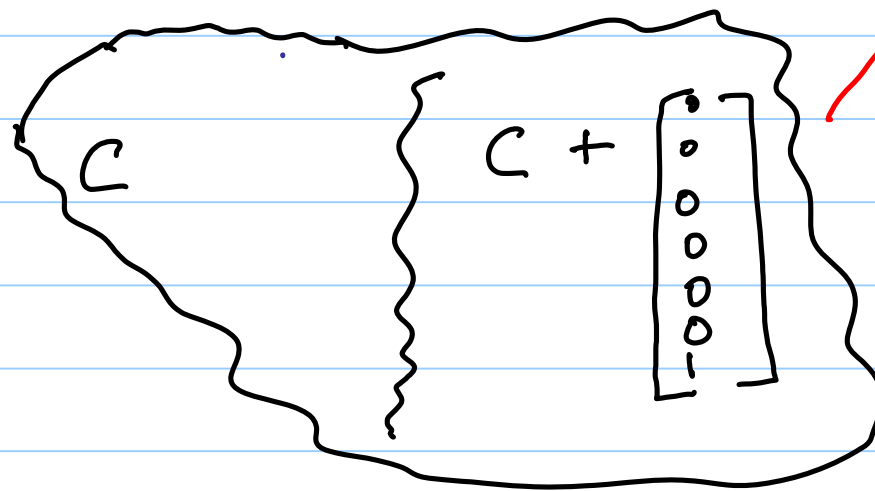
$$(G, \cdot) = (\mathbb{F}_2^7, +)$$

$$(H, \cdot) = (C, +)$$

where  $C$  is the even parity code (SPC code).

coset  $\subset$

even parity



coset  $\subset C + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$   
odd parity

verify that this is the  
partitioning that results.

Claim: Let  $(G, \cdot)$  be a group

and  $(H, \cdot)$  be a subgroup

Let  $a \sim b$  iff  $ab^{-1} \in H$

Then there is a 1-1 correspondence  
between the elements of

different equivalence classes.

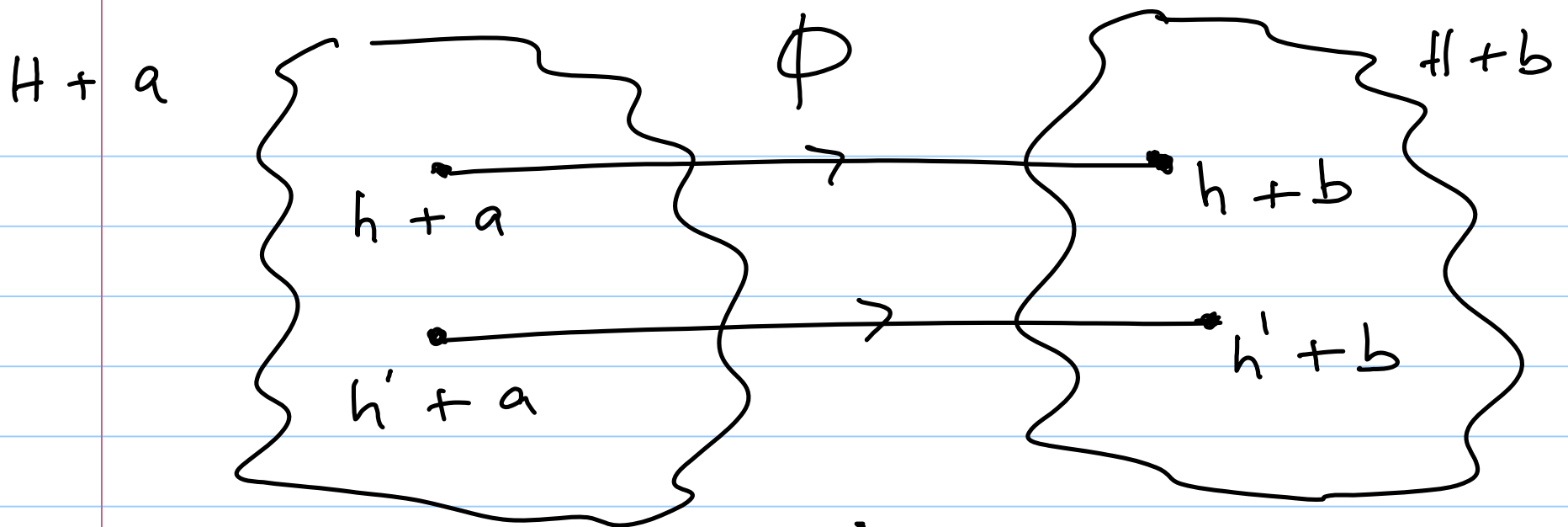
(i.e., there is a 1-1 correspondence between the  $\left. \begin{array}{l} \text{cosets of} \\ H \text{ in} \\ G. \end{array} \right\}$ )



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:

$$\phi(h+a) = h+b \quad \text{all } h \in H$$

clearly  $\phi$  is onto

To show 1-1, assume to the contrary that

$$\phi(h_1 + a) = \phi(h_2 + a)$$

$$\text{i.e., } h_1 + b = h_2 + b$$

$\Rightarrow$  by cancellation

$$\text{that } h_1 = h_2$$

$$\therefore h_1 + a = h_2 + a$$

$\therefore$  the map is 1-1.

---

# Rings

Defn 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

$$(ii) \quad a, b \in \mathbb{R} \Rightarrow a \cdot b \in \mathbb{R}$$

(CLOSURE UNDER MULTIPLICATION)

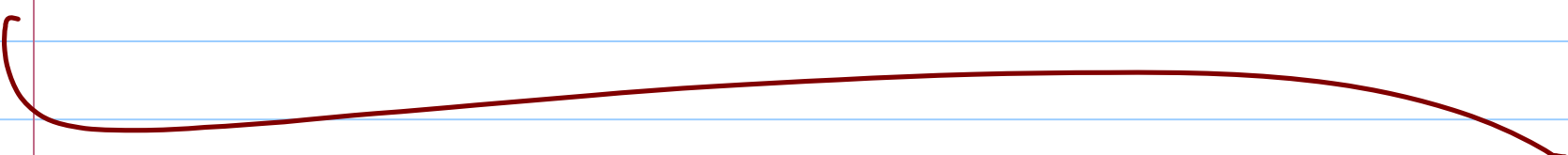
$$(iii) \quad a \cdot (b + c) = a \cdot b + a \cdot c$$

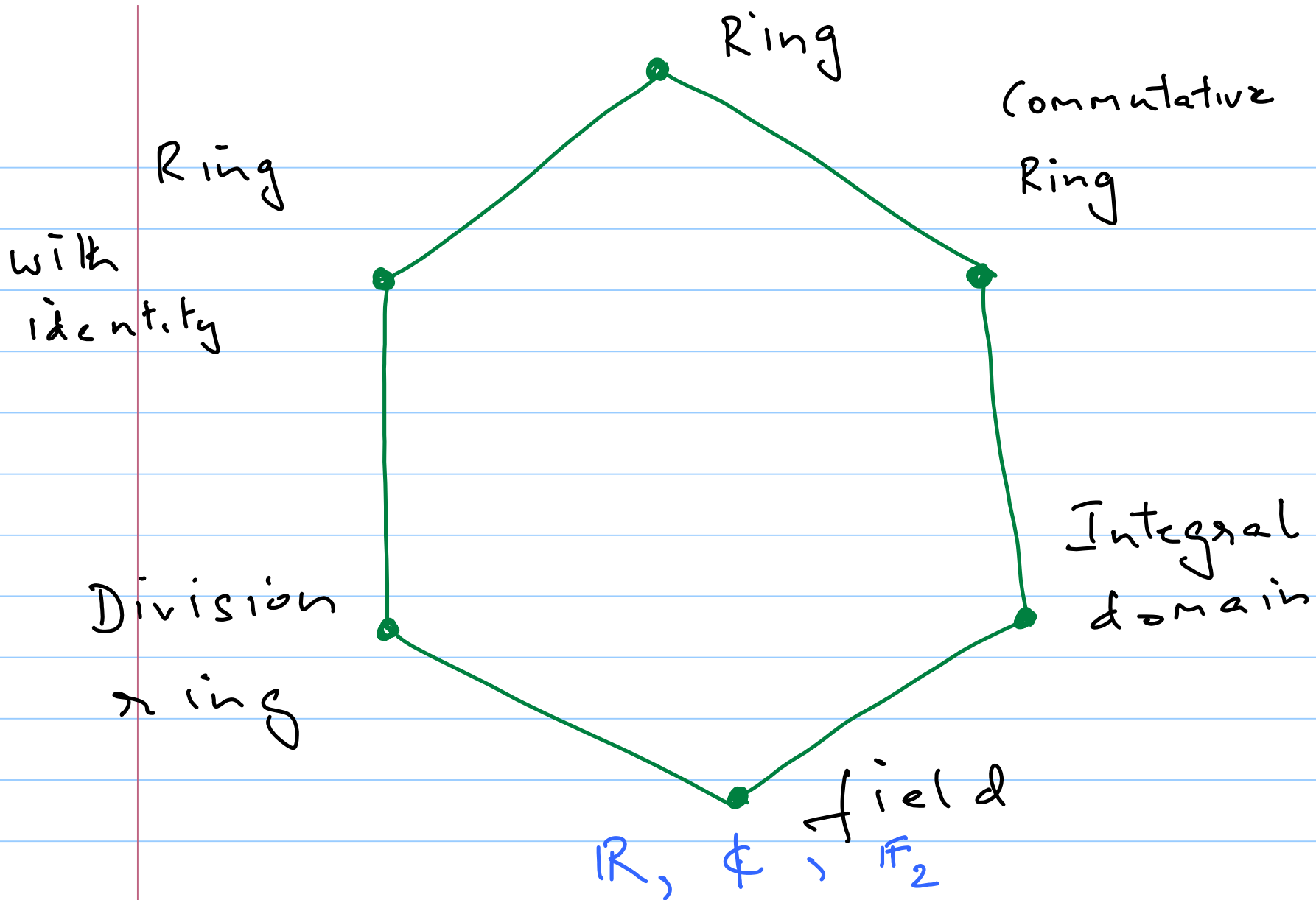
$$(a + b) \cdot c = a \cdot c + b \cdot c$$

(DISTRIBUTIVE LAWS)

$$(iv) \quad a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

ASSOCIATIVITY UNDER  
MULTPLH.





## Definitions

- A commutative ring is a ring in which multiplication is commutative, i.e.,  
$$a \cdot b = b \cdot a \quad \text{all } a, b \in R$$
- A ring with identity is a ring with multiplicative identity (which we will call 1), i.e.,  
(the identity under addition is written as 0)  
$$a \cdot 1 = a = 1 \cdot a.$$

— An integral domain is a commutative ring that has no zero divisors, i.e.,  
 $a \cdot b = 0$  for  $a, b \in R$

$$\text{iff } \begin{cases} a = 0 \text{ or} \\ b = 0 \end{cases}$$

— A division ring is a ring with identity in which every non zero element in  $R$  has a multiplicative inverse, i.e.,

$$a \in R, \quad a \neq 0$$

$$\Rightarrow \exists a^{-1} \in R \text{ s.t.}$$

$$a \cdot a^{-1} = a^{-1} \cdot a = 1.$$



(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

—  $\begin{cases} \text{non zero elements have} \\ \text{inverses} \end{cases}$

$\Rightarrow$  in a field  $F$ ,

$(F, +)$  is an Abelian group

$(F^*, \cdot)$   $\begin{cases} \text{is also an Abelian} \\ \text{group} \end{cases}$

# Examples of rings

— the real numbers  $\mathbb{R}$   
the complex numbers  $\mathbb{C}$   
the finite field  $\{0, 1\}$  } fields

$0 + 1 = 1$   
 $0 + 0 = 0$   
 $1 + 0 = 1$   
etc

→ { same operations  
as seen  
earlier

## 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 ?

## Lecture 6: Vector Spaces, Linear Independence and Basis

- 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  $F_2^n$  are concerned,

Ring

Ring

Commutative  
Ring

$\mathbb{Z}_6$

with  
identity  
 $\mathbb{Z}$   
 $\mathbb{R}^{m \times n}$   
 $\mathbb{F}[x]$

Division  
ring

Hamilton's  
quaternions }

Field  
 $\mathbb{R}, \mathbb{C}, \mathbb{F}_2$

Integral  
domain  
 $\mathbb{Z}, \mathbb{Z}^2$   
 $\mathbb{F}[x]$

$$\mathbb{F}_2 = \{0, 1\} \quad \text{field } (\mathbb{F}_2, +, \cdot)$$

+	0	1
0	0	1
1	1	0

·	0	1
0	0	0
1	0	1

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  $\left\{ \begin{array}{l} \text{integral} \\ \text{domain} \end{array} \right.$   
 $(\mathbb{Z}, +, \cdot)$

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

4)  $\mathbb{R}^{m \times n} = \left\{ \begin{array}{l} \text{the set of all } (m \times n) \text{ matrices} \\ \text{over the reals} \end{array} \right.$

$$(\mathbb{R}^{m \times n}, +, \cdot)$$

5)  $\mathbb{F}[x]$  — { the set of all polynomials  
in the indeterminate  $x$   
over  $\mathbb{F}$  }

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

(degree)

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

6)  $(\mathbb{Z}_6, +, \cdot)$  addition and  
multiplication  
are carried out  
modulo 6



$2 \cdot 3 = 0 \quad \therefore$  not an  
integral domain

---

# Vector Spaces

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

$+$   $\Rightarrow$  vector addition

$\cdot$   $\Rightarrow$  scalar multiplication

s.t.

(i)  $(V, +)$  is an Abelian group

$$(ii) \quad c \underline{v} \in V, \quad c \in \mathbb{F} \quad \left. \begin{array}{l} \text{CLOSURE UNDER} \\ \text{SCALAR} \\ \text{MULTPLH.} \end{array} \right\}$$

$$\underline{v} \in V$$

$$(iii) \quad c_1 (c_2 \underline{v}) = (c_1 c_2) \underline{v} \quad \text{ASSOCIATIVITY}$$

$$(iv) \quad \left. \begin{array}{l} (c_1 + c_2) \underline{v} = c_1 \underline{v} + c_2 \underline{v} \\ c(\underline{v}_1 + \underline{v}_2) = c \underline{v}_1 + c \underline{v}_2 \end{array} \right\} \quad \text{DISTRIBUTIVE LAWS}$$

$$(v) \quad 1 \underline{v} = \underline{v}$$

$\downarrow$   
 $\left\{ \begin{array}{l} \text{multiplicative} \\ \text{identity in } \mathbb{F} \end{array} \right.$

---

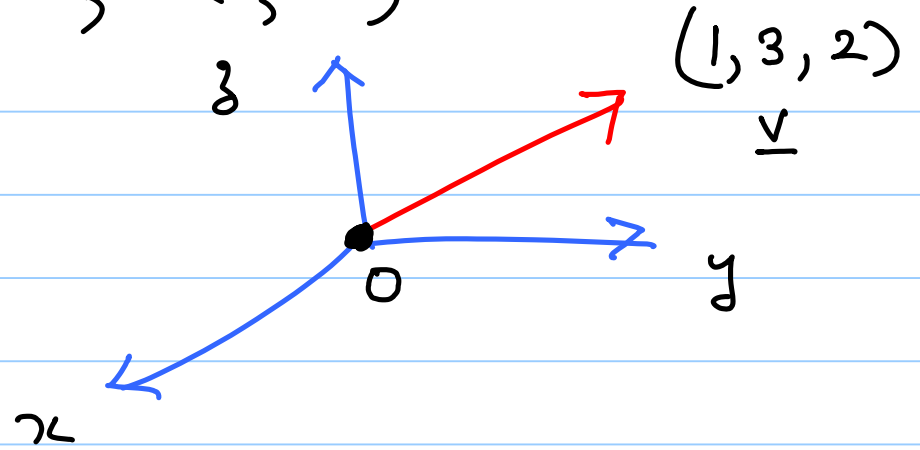
Examples

$$(i) \mathbb{R}^n \Rightarrow (\mathbb{R}^n, +, \mathbb{R}, \cdot)$$

Ex

$n = 3$

$$\underline{v} = \begin{bmatrix} 1 \\ 3 \\ 2 \end{bmatrix}$$



$$(ii) (\mathbb{F}_2^n, +, \mathbb{F}_2, \cdot)$$

$$(iii) (\mathbb{F}[x]_{m \times n}, +, \mathbb{F}, \cdot)$$

$$(iv) (\mathbb{R}^{m \times n}, +, \mathbb{R}, \cdot)$$

## Derived Properties

$$(i) \quad \underline{0} \underline{v} = \underline{0}$$

additive identity in the group  $(V, +)$

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

Pf  $(1 + 0) \underline{v} = 1 \underline{v} + 0 \underline{v} = \underline{v} + 0 \underline{v}$

$\parallel$

$1 \underline{v}$

$\parallel$

$\underline{v}$

adding  $-\underline{v}$  to both  $\underline{v}$ ,  $\underline{v} + 0 \underline{v}$   
we see that  
 $0 \underline{v} = \underline{0}$

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

Pf.  $c(\underline{0} + \underline{v}) = c \underline{0} + c \underline{v}$

$\parallel$

$$c(v)$$

$$\therefore c(v) = c_0 + c_v$$

$$\therefore c(v) + -\left(c(v)\right) = c_0 + c_v + (-c(v))$$

$\Uparrow$   
 (additive  
 inverse of  
 $c_v$ )

$$\therefore \underline{0} = c_0 \quad |||$$

(iii)  $c_v = 0$  iff either  $c = 0$  or  $v = \underline{0}$

Pf if  $c = 0$  done

if  $c \neq 0$ , consider

$$c^{-1}(c_v) = c^{-1}(\underline{0}) = \underline{0}$$

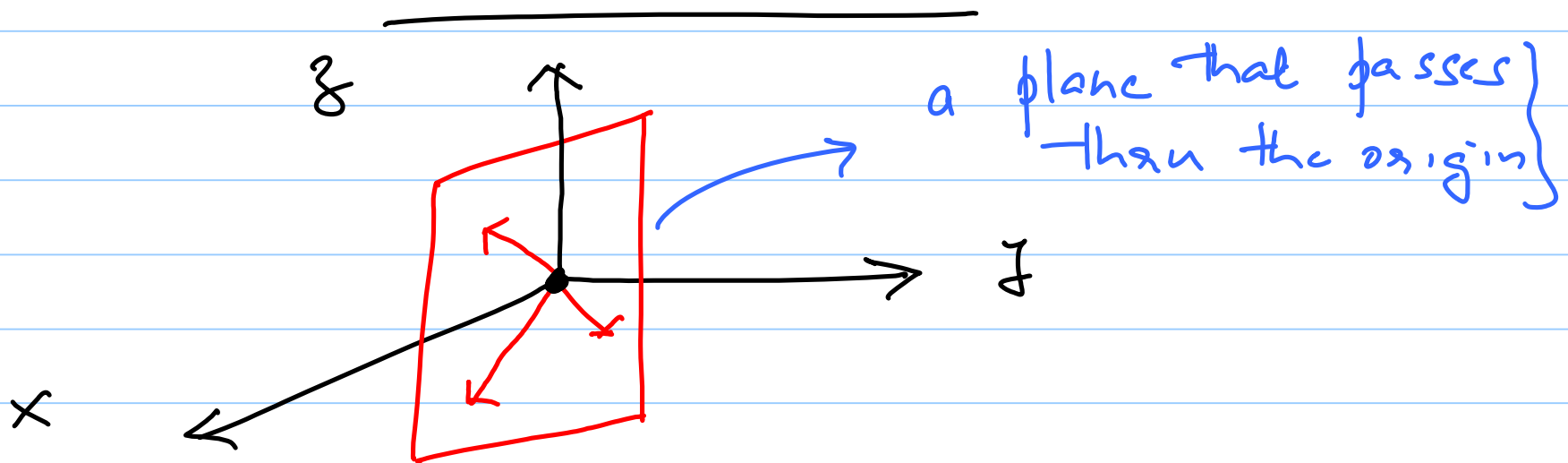
$$\begin{matrix} \parallel \\ (L^{-1} L) \underline{v} \end{matrix}$$

$$\begin{matrix} \parallel \\ \parallel \underline{v} \\ \parallel \\ \underline{v} \end{matrix}$$

$$\therefore \underline{v} = \underline{0}$$

$$(iv) \quad (-1) \underline{v} = -\underline{v}$$

(proof left as an exercise!)



# Subspaces

Defn A subspace of a vector space  $(V, +, \cdot, \mathbb{F})$  is a subset  $W$  of  $V$  such that  $(W, +, \cdot, \mathbb{F})$  is also a vector space.

Eg In  $\mathbb{R}^3$ , possible subspaces are:

- $\rightarrow \mathbb{R}^3$
- $\rightarrow \{ \underline{0} \}$
- $\rightarrow$  any line thru  $\{ \underline{0} \}$
- $\rightarrow$  any plane thru  $\{ \underline{0} \}$



## Test for the presence of a subspace?

---

$$(V, +, \mathbb{F}, \cdot) \quad W \subseteq V$$

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

To test whether or not " $W$  is a subspace of  $V$ " it is sufficient to check that

$$\underline{x} + c \underline{y} \in W$$

how does this

$\left\{ \begin{array}{l} \text{whenever} \\ \underline{x}, \underline{y} \in W \\ \text{and } c \in \mathbb{F} \end{array} \right.$

# Lec 7: Linear Codes, $\left\{ \begin{array}{l} \text{linear} \\ \text{independence} \end{array} \right.$

## Summary

- $\left\{ \begin{array}{l} \text{example rings and their} \\ \text{classification} \end{array} \right.$
- vector spaces
  - 5 examples
  - derived properties
- subspaces
  - test for a subspace

## Test for the presence of a subspace?

---

$$(V, +, \mathbb{F}, \cdot) \quad W \subseteq V$$

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

To test whether or not " $W$  is a subspace of  $V$ " it is sufficient to check that

$$\underline{x} + c \underline{y} \in W$$

how does this

{ whenever  
 $\underline{x}, \underline{y} \in W$   
and  $c \in \mathbb{F}$

$$\boxed{\underline{x} + c \underline{y} \in W}$$

— setting  $c = 1 \Rightarrow \begin{cases} \text{closure under} \\ \text{vector addition} \end{cases}$

— setting  $\underline{y} = \underline{x}$  and  $c = -1$

$$\begin{aligned} \underline{x} + (-1) \underline{x} &\in W \\ \Rightarrow \underline{0} &\in W \end{aligned} \quad \begin{array}{l} \text{identity} \\ \text{element} \end{array}$$

— setting  $\underline{x} = \underline{0}$ ,  $c = -1$ ,

ensures that

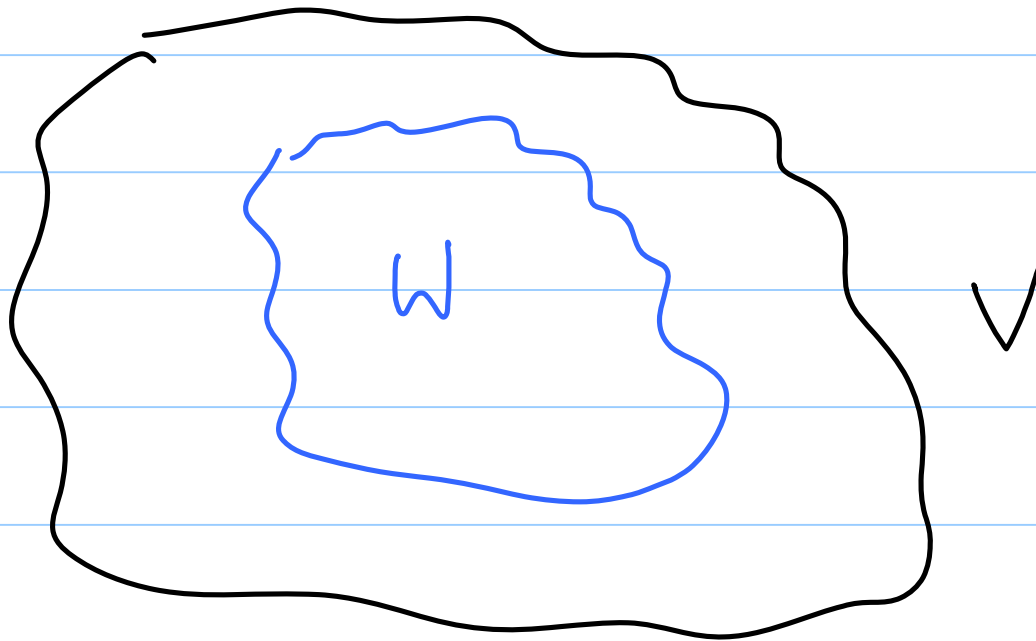
$$(-1) \underline{y} \in W$$

$\begin{cases} \text{inverse is} \\ \text{present} \end{cases}$

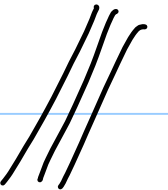
$$\Rightarrow -\underline{y} \in W$$

$$\underline{x} = \underline{0} \Rightarrow \underline{y} \in W$$

{ closure under  
scalar  
multiplication



the remaining 5 axioms follow  
simply from the fact that  
 $W \leq V$ .



$$\underline{\text{Eg}} \quad (V, +, \mathbb{F}, \cdot) = (\mathbb{F}_2^7, +, \mathbb{F}_2, \cdot)$$

$$(W, +, \mathbb{F}, \cdot) = (\mathcal{C}, +, \mathbb{F}_2, \cdot)$$

even parity or  
spec code.

note that  $\underline{x} \in \mathcal{C}$  iff  $\boxed{\underline{1}^t \underline{x} = 0}$

( $\Leftrightarrow$ )  $\sum_{i=1}^n x_i = 0$

if  $\underline{x}, \underline{y} \in \mathcal{C}$

$$\Rightarrow \left. \begin{array}{l} \underline{1}^t \underline{x} = 0 \\ \underline{1}^t \underline{y} = 0 \end{array} \right\} \Rightarrow \underline{1}^t (\underline{x} + \underline{y})$$

$$= \begin{pmatrix} 1 & x \\ 1 & 1 \end{pmatrix} + c \begin{pmatrix} 1 & y \\ 1 & 1 \end{pmatrix}$$

$$= 0 + c \cdot 0 = 0$$

///

Thus  $\mathcal{R}$  is a subspace  
of  $\mathbb{F}_2$ .

Note: that for the particular case when  $F = F_2$  the only non zero scalar = 1

∴  
the test

$$\underline{x} + c \underline{y} \in W$$

reduces to

$$\underline{x} + \underline{y} \in W$$



It follows from this that there  
is no distinction between

subspaces  $(\mathcal{R}, +, \mathbb{F}_2, \cdot)$  of  
 $(\mathbb{F}_2^n, +, \mathbb{F}_2, \cdot)$

and of subgroups

$(\mathcal{R}, +)$  of  
 $(\mathbb{F}_2^n, +)$

Defn A linear code of block

length  $n$  is any subspace  
of  $\mathbb{F}_2^n$  (i.e., of  $(\mathbb{F}_2^n, +, \cdot)$ )

It follows from this that every  
such linear code is also

a subgroup. For this reason,

linear codes are also called  
group codes.

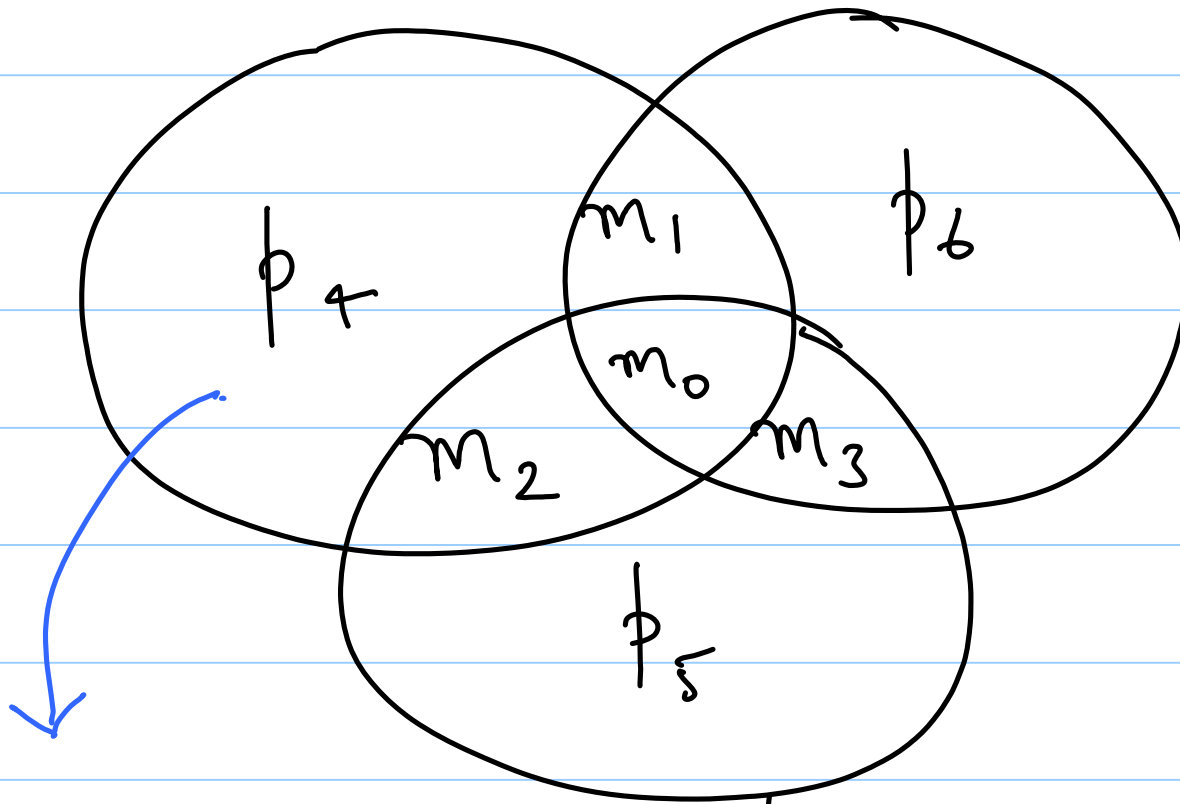
Eg 1 { We have already seen that  
the spc code is a linear code.

Eg 2 the repetition code:

$$C = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

clearly this is a linear code.

Fig 3 Hamming code



$$m_0 + m_1 + m_2 + p_4 = 0$$

$$\begin{array}{c}
 \begin{matrix} & 0 & 1 & 2 & 3 & 4 & 5 & 6 \end{matrix} \\
 \left[ \begin{array}{cccccc}
 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
 1 & 0 & 1 & 1 & 0 & 1 & 0 \\
 1 & 1 & 0 & 1 & 0 & 0 & 1
 \end{array} \right]
 \begin{array}{c}
 m_0 \\
 m_1 \\
 m_2 \\
 m_3 \\
 p_4 \\
 p_5 \\
 p_6
 \end{array}
 =
 \begin{array}{c}
 \left[ \begin{array}{c}
 0 \\
 0 \\
 0
 \end{array} \right]
 \end{array}
 \end{array}$$

$\underbrace{\hspace{10em}}_H \qquad \qquad \qquad \underbrace{\hspace{10em}}_C \qquad \qquad \qquad \underbrace{\hspace{10em}}_{\underline{0}}$

Thus the Hamming code is precisely  
 the set of all code words  $\subseteq$   
 satisfying

$$H \subseteq \underline{0}$$

$$\left. \begin{array}{l} \underline{x} \in \mathcal{C} \Rightarrow H \underline{x} = \underline{0} \\ \underline{y} \in \mathcal{C} \Rightarrow H \underline{y} = \underline{0} \end{array} \right\}$$

$$\therefore H(\underline{x} + c \underline{y}) = H \underline{x} + c H \underline{y} \\ = \underline{0}$$

$\therefore \left\{ \begin{array}{l} \text{the Hamming code is a} \\ \text{linear code.} \end{array} \right\}$



note: Given a  $m \times n$   $H$ , the  
collection of all vectors  $\underline{x}$  s.t.

$H \underline{x} = \underline{0}$  is called the nullspace

of  $H$ :

$$\text{null space of } H = \left\{ \underline{x} \mid H \underline{x} = \underline{0} \right\}.$$

It follows from this that the null space of any binary  $m \times n$  is a linear code.



# LINEAR INDEPENDENCE

Defn. The vectors  $\{\underline{x}_1, \underline{x}_2, \dots, \underline{x}_n, \dots\}$  are said to be linearly independent

if

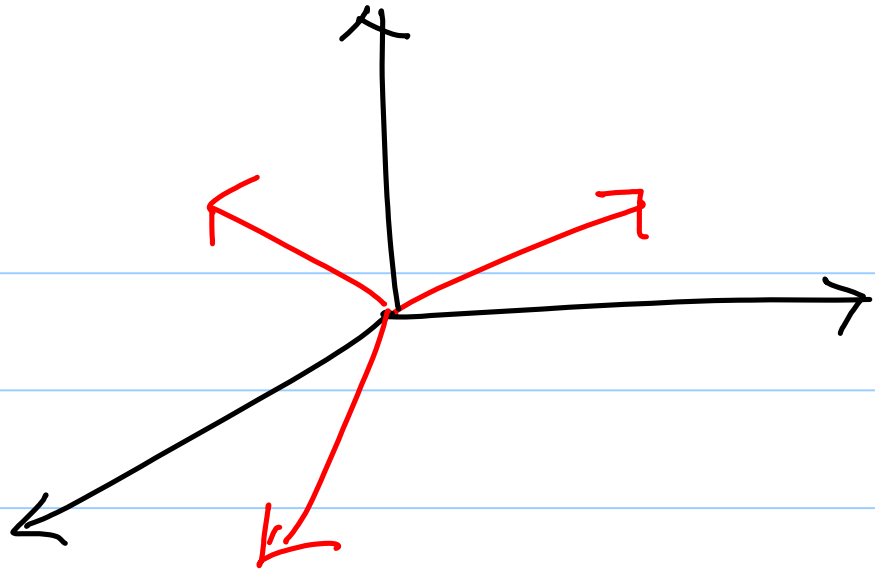
$$\sum_{j=1}^n c_j \underline{x}_j = \underline{0}$$

is possible iff

$$c_j = 0 \text{ all } j=1, \dots, n.$$

Ex. 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.

Ex 2

$A =$

row

- reduced  
echelon

matrix

$$\begin{bmatrix} 1 & 0 & 2 & 1 & 3 \\ 0 & 0 & 6 & 1 & 7 \\ 0 & 0 & 0 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{matrix}$$

— { the nonzero rows are linearly  
independent  
3

— the 5 columns containing the  
pivots are linearly independent.

Eg. vector space:

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

$$\left\{ \begin{array}{l} \alpha_1 = x + a \quad a \in \mathbb{F} \\ \alpha_2 = x^2 + bx + c \quad b, c \in \mathbb{F} \\ \alpha_3 = x^5 \end{array} \right.$$

can be verified that these 3

polynomials are linearly independent.

# Spanning

Defn A set  $\{ \underline{d}_1, \underline{d}_2, \dots, \underline{d}_n, \dots \}$

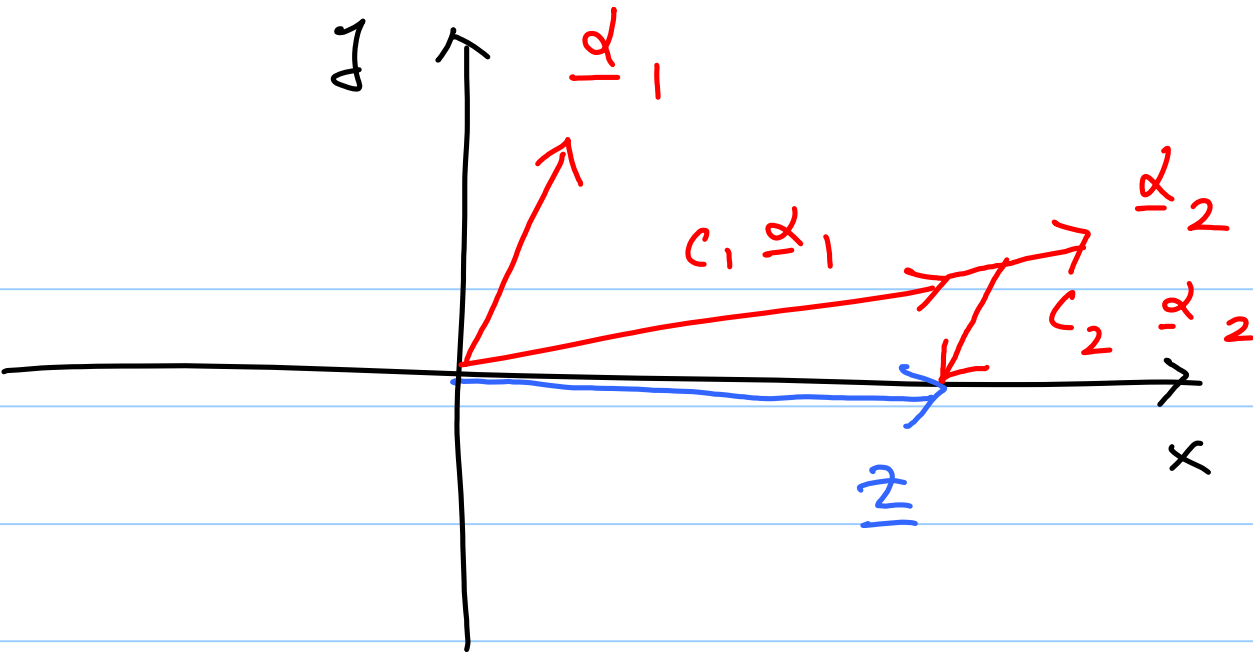
is said to span a vector space  $V$

if every vector  $\underline{z} \in V$  can be

expressed in the form:

$$\underline{z} = \sum_{j=1}^n c_{ij} \underline{d}_{ij} \quad , \quad c_{ij} \in \mathbb{F}$$

$n \geq 1$  is  
an integer



$$z = \sum_{i=1}^n c_i \alpha_i$$

a linear combination of  
 $\{\alpha_i\}$ .

# lec 8: Spanning & Basis

## Recap

- test for a subspace
- linear code definition
  - examples
- linear independence
  - examples
- spanning

Defn The space spanned by a set

$\{ \underline{x}_1, \underline{x}_2, \dots, \underline{x}_n, \dots \}$  is the set of

all (finite) linear combinations of

the form:

$$\sum_{j=1}^n c_j \underline{x}_j$$

$n \geq 1$   
is an  
integer,

$c_j \in F$   $\left\{ \begin{array}{l} \text{underlying field} \\ \text{of the vector space.} \end{array} \right.$



(the word space is used since this collection actually forms a vector space)

notation:

$$W = \langle \underline{\alpha}_1 \quad \underline{\alpha}_2 \quad \underline{\alpha}_3 \quad \dots \rangle$$

space spanned by  
the  $\{\underline{\alpha}_i\}$ .

Qn: What is the space spanned

$\downarrow$

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right\}$$

Setting:  $(\mathbb{F}_2^7, +, \mathbb{F}_2 \cdot)$

Ans { the above collection of  
6 vectors form a basis  
for the single parity check  
code.

# Basis

Defn. A basis for a vector space

$(V, +, \mathbb{F}, \cdot)$  is a collection

$$\left\{ \underline{\alpha}_1, \underline{\alpha}_2, \dots \right\}$$

of vectors such that:

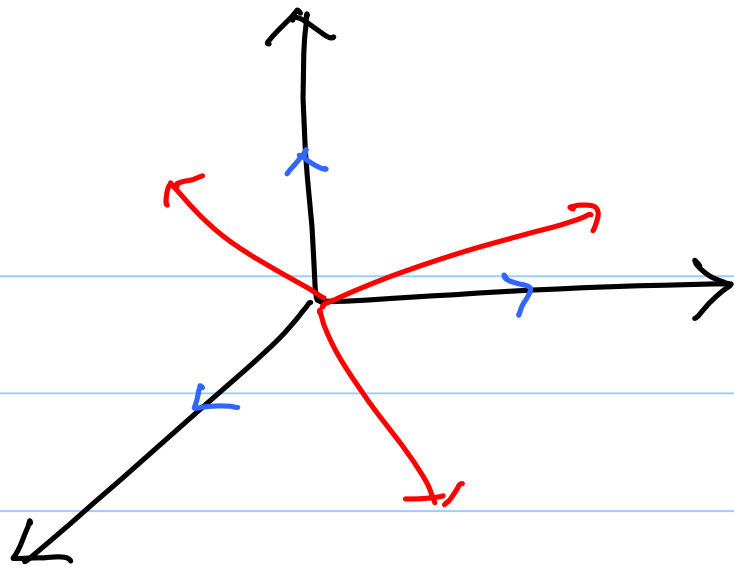
(i)  $\left\{ \begin{array}{l} \text{the set is a } \underline{\text{linearly independent}} \\ \text{set} \end{array} \right.$

(ii)  $\left\{ \begin{array}{l} \text{the set} \\ v \end{array} \right\}$  spans the vector space

Eg  $(\mathbb{R}^3, +, \mathbb{R}, \cdot)$  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 this  
example, a given vector space  
can have multiple bases.

Eg. vector space

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

$$\text{Basis} \Rightarrow \left\{ 1, x, x^2, x^3, \dots \right\}$$

Defn A finite-dimensional vector space  
is any vector space possessing a basis  
consisting of a finite # of elements.

Thm Let  $(V, +, \mathbb{F}, \cdot)$  be a

finite-dimensional vector space.  
(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.

Pf ( of theorem ). Let

$$\left\{ \alpha_1, \alpha_2, \dots, \alpha_m \right\} \begin{matrix} A \\ H \\ D \end{matrix} \left\{ \beta_1, \beta_2, \dots, \beta_n \right\}$$

be two bases for the vector space  $V$ .

Since  $\{\alpha_i\}_{i=1}^m$  form a basis and

the  $\{\beta_j\}_{j=1}^n$  are a linearly independent

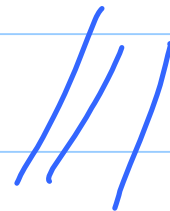
set, it follows that  $n \leq m$

Since  $\{\alpha_i\}_{i=1}^m$  form a basis and

the  $\{\beta_j\}_{j=1}^n$  span the vector space

$V$ , it follows that  $n \geq m$

$$\therefore n = m$$



## DIMENSION

Defn The dimension  $k$  of a f.d.

vector space  $\{V, +, \mathbb{F}, \cdot\}$

is simply the number of elements in  
any basis for  $V$ .

## DIMENSION OF A LINEAR CODE

Defn The dimension of a linear code  $\mathcal{C}$  of block length  $n$  is simply its dimension as a subspace of the vector space  $(\mathbb{F}_2^n, +, \mathbb{F}_2, \cdot)$

Ex

The repetition code:

$$\left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

$[7, 1, 7]$

code.

basis  $\Rightarrow$

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

$\therefore$  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  $k$ .

## Notation

— 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:

$[7, 6, 2]$  code

basis:

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$

# Lec 9: The Dual Code

Recap :

- spanning
- basis
  - examples
  - { any two bases  
for a given vector  
space contain the  
same # of elements

— dimension

—  $\downarrow$  a vector space

— of a linear code

— examples

---

Note (a) a given vector space can have  
more than one basis

Eg  $\mathbb{R}^3$  — the standard basis

$$\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$

— { also, any three vectors  
not on a plane also form  
a basis

⑥. Given a basis  $\{ \underline{x}_1, \underline{x}_2, \dots, \underline{x}_n, \dots \}$

for a vector space  $V$ , every vector

has a unique expansion as a

linear combination of elements of

the basis.

Pf Suppose

$$\underline{x} = \sum_{j=1}^n c_j \underline{\alpha}_j = \sum_{k=1}^s a_k \underline{\alpha}_k$$

$$\Rightarrow \sum_{j=1}^n c_j \underline{\alpha}_j - \sum_{k=1}^s a_k \underline{\alpha}_k = \underline{0}$$

But by the linear independence  
of the  $\{\underline{\alpha}_i\}$  it follows that  
this can happen iff

$n = 3$ ,  $\{\underline{x}_j\}$  and  $\{\underline{x}_k\}$  are the same  
and the coefficients are the same.  $///$

Eg  $\mathbb{R}^3$   $\underline{x} = \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}$

standard basis:

$$\left\{ \underline{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \underline{x}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \underline{x}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$



$$\underline{X} = 1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + 4 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

this coefficient set is unique.

Basis for the 3 example codes

Ex 1 (the repetition code)

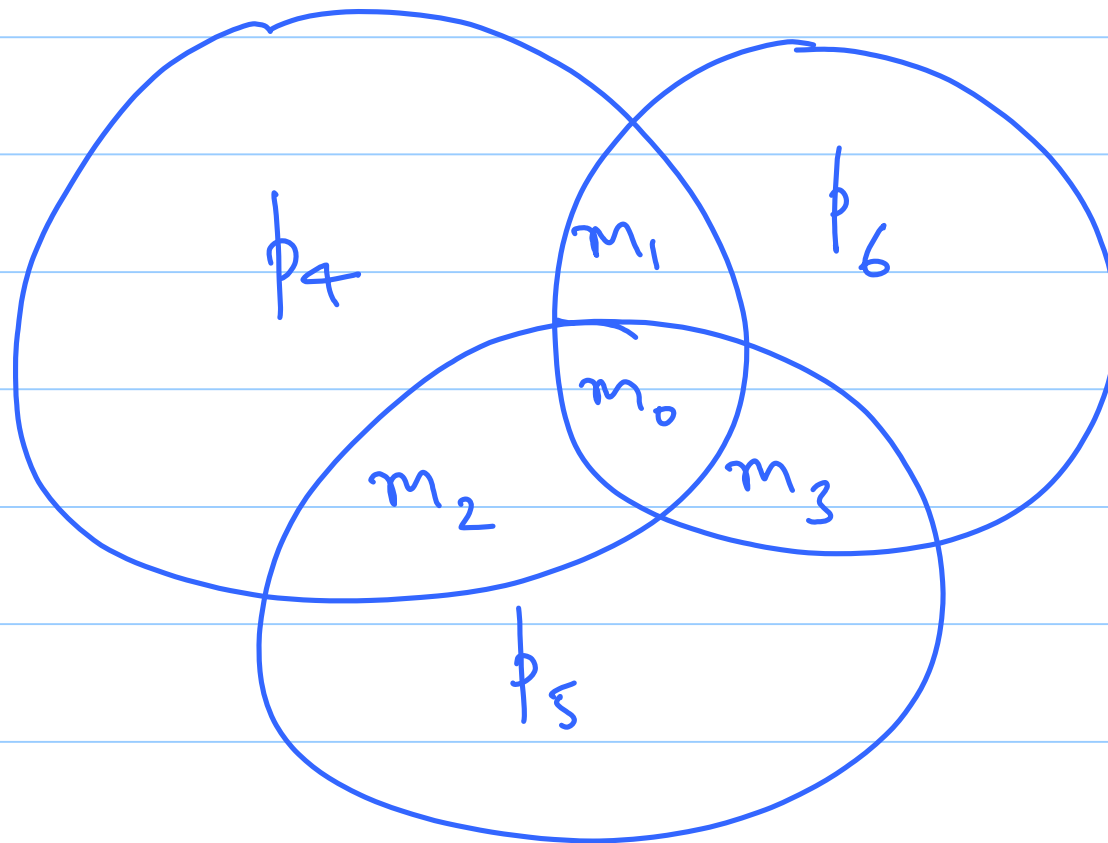
$$\text{basis} = \left\{ \begin{bmatrix} 1 \\ \vdots \end{bmatrix} \right\}$$

Ex 2 Single parity-check code

basis = { the collection of  
six vectors below:

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right\}$$

Fig 3      The Hamming Code.



$$[m_0 \ m_1 \ m_2 \ m_3 \ p_4 \ p_5 \ p_6]$$

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

the rows of  $G$  form a basis for the Hamming code.

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

(this is a generator  $G$  for the Hamming code)

# The Dual Code

Defn Let  $\mathcal{C}$  be an  $[n, k]$  code.

the dual  $\mathcal{C}^\perp$  of  $\mathcal{C}$  is defined by:

$$\mathcal{C}^\perp = \left\{ \underline{y} \in \mathbb{F}_2^n \mid \underline{x}^t \underline{y} = 0 \text{ for all } \underline{x} \in \mathcal{C} \right\}$$

Ex If  $\mathcal{C}$  is the repetition code

$$\mathcal{C} = \left\{ \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right\}$$

clearly, the dual of this code is  
the  $\text{spc}$  code.

$$\text{i.e., } (\mathcal{C}_{\text{rep}})^\perp = \mathcal{C}_{\text{spc}}$$

Qn: What is  $R_{\text{spec}}^\perp$ ? i.e.,

what is  $\left(\left(R_{\text{rep}}\right)^\perp\right)^\perp$ ?

Ans  $(R^\perp)^\perp = R$  always!

---



# GENERATOR MATRIX

Defn. Let  $\mathcal{C}$  be an  $[n, k]$  code.

Then any  $(k \times n)$  matrix whose rows form a basis for  $\mathcal{C}$  is called a generator matrix for  $\mathcal{C}$ .

Note: A code can in general, have more than one generator matrix.

Fig 1. The Hamming code  
(provided earlier)

Fig 2 (rep code)

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

Eg

SpC

Code

$[7, 6]$

$G =$

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

$(6 \times 7)$

$(k \times n)$

Lemma The nullspace of a generator  $m \times n$  for the code  $\mathcal{C}$  is precisely the dual code

Pf Let  $\underline{x}$  be in the nullspace of  $A$ .

(note: the nullspace of an  $(m \times n)$   $m \times n$

$A$  is precisely the set

$$\mathcal{N}(A) = \left\{ \underline{x} \mid A \underline{x} = \underline{0} \right\}$$

null space.

$$\therefore \begin{bmatrix} \vdots \\ \log t_1 \\ \vdots \\ \log t_2 \\ \vdots \\ \log t_k \end{bmatrix} \begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

clearly from the definition of the dual code it follows that if  $\underline{y} \in \mathbb{R}^t$ , then  $\underline{y} \in \mathcal{C}^\perp$ .

On the other hand if  $\underline{x} \in \mathcal{C}^\perp$ , then

$$\underline{x}^t \underline{g}_i = \underline{0}$$

$$\text{If } \underline{c} \in \mathcal{R}, \quad \underline{c} = \sum_{i=1}^k m_i \underline{g}_i$$

$$\therefore \underline{x}^t \underline{c} = \underline{x}^t \left( \sum_i m_i \underline{g}_i \right)$$

$$= \sum_i m_i \underbrace{\left( \underline{x}^t \underline{g}_i \right)}_{=0}$$

$$= 0$$

$$\Rightarrow \underline{x} \in \mathcal{R}^t.$$



# PARITY-CHECK MATRIX

H

Defn A parity-check  $m \times n$  for a linear code  $\mathcal{C}$  is any generator  $m \times n$  for the dual code  $\mathcal{C}^\perp$

Eg Let  $\mathcal{C}$  be the SPC code.

1. The dual code  $\mathcal{C}^\perp$  is the repetition code having generator  $m \times n$

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

Note that  $\underline{c} \in \mathcal{C}$  iff  $H\underline{c} = \underline{0}$

( $\Rightarrow$ )

$$\sum_{i=1}^n c_i = 0$$

this is precisely  
the parity condition  
satisfied by  
codewords in  $\mathcal{C}$ ,  
hence the name.



$$\text{Then } (\mathcal{R}^\perp)^\perp = \mathcal{R}.$$

(Sketch) Let  $H$  be a given  $m \times n$  in  $\mathcal{R}^\perp$ .

$$\eta(H) = (\mathcal{R}^\perp)^\perp \stackrel{?}{=} \mathcal{R}.$$

# Lec 10 { Systematic Generator Matrix

Recap :

- (i) uniqueness of representation w.r.t a basis
- (ii) examples for the 3 codes
- (iii) Dual code definition
  - example
- (iv) the generator matrix

$$(v) \quad \eta(C) = \mathbb{R}^\perp$$

(vi) the parity-check  $n \times$

---

Definition The row space of an  $(n \times n)$  matrix  $A$  is the set of all linear combinations of the rows of  $A$ .

Ex.

$G =$

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

row space of  $G$  is the spc code.

Thm (fundamental theorem of linear algebra)

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

then

$$\text{rank}(A) + \dim(\text{null space}(A)) = n$$

①

Also for any given matrix  $A$ ,

$$\text{rank}(A) = \dim(\text{rowspace of } A)$$

②

Then it follows from ① and ②  
that if  $A$  is  $(m \times n)$ , then

$$\begin{aligned} \dim(\text{rowspace}(A)) &= n \\ + \dim(\text{nullspace}(A)) & \end{aligned} \quad \dots \text{---} \textcircled{3}$$

$$\underline{\text{Thm}} \quad (C^\perp)^\perp = C$$

Pf. Let  $H$  be a generator  $n \times$   
for the dual code.

It follows from an earlier  
lemma that the nullspace  
of  $H$  is  $(C^\perp)^\perp$ .

consider the equation:

$$\begin{bmatrix} h_1^t \\ h_2^t \\ \vdots \\ h_{n-k}^t \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = 0$$

$H$

$((n-k) \times n)$

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  $\underline{x}$ , i.e.,

$$\mathcal{C} \subseteq n(H) = (C^\perp)^\perp$$

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

$$\therefore \dim(n(H)) = n - (n - k) = k$$

but  $\dim(\mathcal{C}) = k$

It follows that

$$\mathcal{N}(H) = \mathcal{C} = (\mathcal{C}^\perp)^\perp$$

Corollary Every code  $\mathcal{C}$  is precisely  
the nullspace of its parity-check matrix

Goal: { finding means to identify  
a p.c. mx for a given code  
 $\mathcal{C}$ .

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

$$\mathcal{C} = \eta(H),$$

then  $H$  is a valid p.c. mx  
for  $\mathcal{C}$ .

Pf.

$$\begin{bmatrix} h_1^t \\ h_2^t \\ \vdots \\ h_{n-k}^t \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = 0 \quad \dots \textcircled{4}$$

(the proof follows from noting that the rows  $h_i^t$  of  $H$  belong to the dual code from  $\textcircled{4}$  and form a basis for the code  $\mathcal{C}^\perp$  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  $n \times k$

for the  $[n, k]$  code  $\mathcal{C}$ , then  $H$

is a valid p.c.  $n \times n$  for the code  $\mathcal{C}$ .

Pf.

$$\begin{bmatrix} h_1^t \\ \vdots \\ h_{n-k}^t \end{bmatrix} \begin{bmatrix} g_1 & \dots & g_k \end{bmatrix} = [0]$$

$H$   $G^T$

It follows that

$$h_i^t g_j = 0 \quad \forall i, j$$

hence  $\mathcal{R}$  is contained in  $\mathcal{N}(H)$

But since  $\dim(\mathcal{R}) = \dim(\mathcal{N}(H))$

$$\Rightarrow R = \eta(H)$$

and hence by Lemma 1,  $H$  is  
a valid p.c.  $m \times n$  for  $R$ .

---

Ex If  $G = \begin{bmatrix} I_k & P \end{bmatrix}$

$\Downarrow$

$(k \times k)$

Identity  $m \times$

Then  $H = \begin{bmatrix} P^T & \vdots & I_{n-k} \end{bmatrix}$

is a valid p.c.  $n \times k$  of  $R$ .

This is because:

$$\begin{aligned} H G^T &= \begin{bmatrix} P^T & I_{n-k} \end{bmatrix} \begin{bmatrix} I_k & P \end{bmatrix}^T \\ &= \begin{bmatrix} P^T & I_{n-k} \end{bmatrix} \begin{bmatrix} I_k \\ -P^T \end{bmatrix} = P^T + P^T \\ &\quad \begin{matrix} (n-k \times k) \end{matrix} \quad \quad \quad = \begin{bmatrix} 0 \end{bmatrix} \end{aligned}$$

Also it is clear that  $\text{rank}(H)$   
 $= (n-k).$

---



Fig repetition code.

$$G = \left[ \begin{array}{c|ccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right]$$

$\mathbb{I}_k$                        $P$

$$\therefore H = \left[ \begin{array}{c|c} P^T & \mathbb{I}_{n-k} \end{array} \right] = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and this is clearly a valid pc

$m \times$  is the replication code.

---

# SYSTEMATIC GENERATOR MATRIX

A generator  $m \times n$   $G$  is said to be a systematic gen  $m \times n$  for an  $[n, k]$  code if it can be expressed in the form:

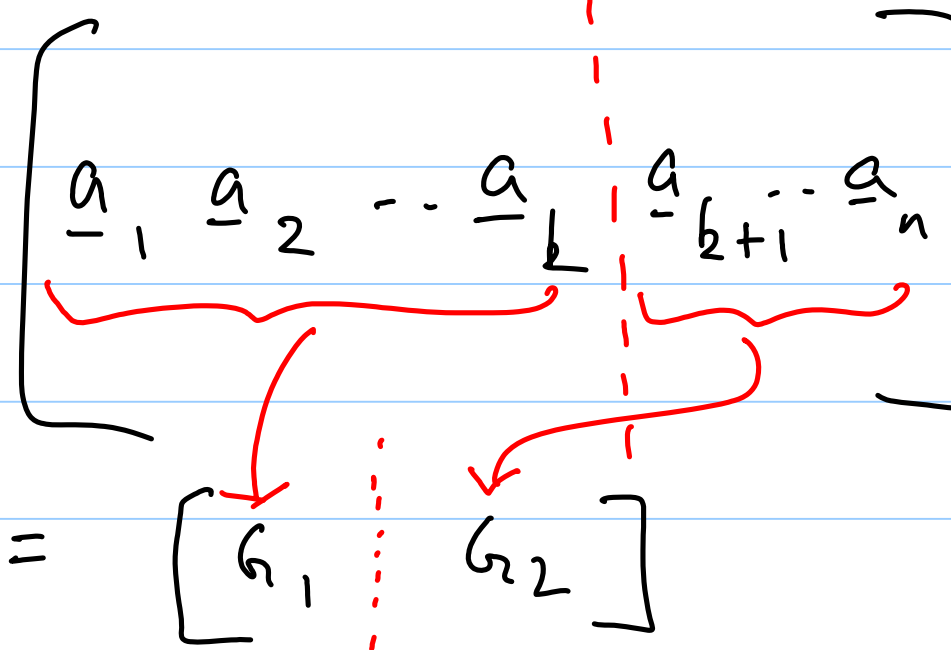
$$G = \left[ I_k \mid P \right]$$

Qn Does every code possess a systematic generator matrix?

Ans No. Let  $G$  be any

generator matrix for the code  $C$ .

$$G = \begin{bmatrix} \underline{a_1} & \underline{a_2} & \dots & \underline{a_k} & \underline{a_{k+1}} & \dots & \underline{a_n} \end{bmatrix}$$



$$= \begin{bmatrix} G_1 & \vdots & G_2 \end{bmatrix}$$

then since any other  $n \times k$   $\in \mathcal{R}$   
can be obtained through taking  
linear combinations of  $G$  above,

it follows that  $\mathcal{R}$  has a  
systematic  $n \times k$  iff

$$\text{rank}(G_1) = k.$$

---

$\mathbb{F}_q$   $R$  is the spc code.

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

$G_1$   
 $(k \times k)$   
 $(6 \times 6)$

$G_2$

Since  $G_1$  above has rank = 6,  
it follows that the Spc code  
does possess a systematic  $gen\ m \times$ .

---

A second reason for being interested  
in a systematic  $gen\ m \times$ :

Consider the map between message  
vector  $\underline{m}$  and code words  $\underline{c}$  given  
by:

$$\underline{m} \longrightarrow c$$

$$\boxed{\underline{m}^t G = c^t}$$

$$\underline{m}^t \begin{bmatrix} I_k & P \end{bmatrix} = \underbrace{\begin{bmatrix} \underline{m}^t \end{bmatrix}}_{\text{the message symbols are explicitly present in the code symbols}} \begin{bmatrix} \underline{m}^t P \end{bmatrix}$$

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.

# Lec 11 { Minimum Distance of a Linear Code

## Recap

— defined row space

— FTLA

—  $(R^\perp)^\perp = R$

— { 2 Lemmas helpful in  
finding a p.c.  $m \times n$  t1

— systematic generator  $m \times$   
— { 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  $\mathcal{C}$  is equivalent to a second linear code  $\mathcal{C}'$  which possesses a systematic generator matrix,

---

## Minimum distance of a linear block code

Thm The min. dist.  $d_{\min}$  of a linear block code  $\mathcal{C}$  is equal to the minimum Hamming weight  $w_{\min}$  of a nonzero codeword.

Pf. Let  $\underline{c}$  have  $W_H(\underline{c}) \leq W_{\min}$ .

Then  $d_H(\underline{c}, \underline{0}) = W_{\min}$

$$\therefore \boxed{d_{\min} \leq W_{\min}} \dots \textcircled{1}$$

On the other hand, let  $\underline{c}_1, \underline{c}_2 \in \mathcal{C}$  be such that

$$d_H(\underline{c}_1, \underline{c}_2) = d_{\min}$$

$$\Rightarrow W_H (\underbrace{c_1 + c_2}) = \phi_{\min}$$

$\in \mathcal{C}$  since  $\mathcal{C}$  is linear!

$$\Rightarrow \boxed{W_{\min} \leq \phi_{\min}} \dots \textcircled{2}$$

$\therefore$  from  $\textcircled{1}$  and  $\textcircled{2}$ ,

$$\boxed{\phi_{\min} = W_{\min}}$$


---

Ex  $\mathcal{C}$  repetition code

$$\mathcal{C} = \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right\}$$

i.  $w_{\min} = 7 = d_{\min}$



Ex  $\mathcal{R} = \text{spec } \mathbb{C}[x]$ .

$$\mathcal{R} = \left\{ \underline{c} \mid \sum_{i=1}^n c_i = 0 \right\}$$

$\therefore$   $w_{\min} = 2 = d_{\min}$

---

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

$\underline{I}$ 
 $\underline{P}$

$G$

$$\therefore A = \begin{bmatrix} P^T & \vdots & I_{n-k} \end{bmatrix}$$

$$\begin{array}{ccccccc}
 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
 \left[ \begin{array}{ccccccc}
 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
 1 & 0 & 1 & 1 & 0 & 1 & 0 \\
 1 & 1 & 0 & 1 & 0 & 0 & 1
 \end{array} \right] & \begin{array}{c} \left[ \begin{array}{c} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{array} \right] \end{array} & = & \begin{array}{c} \left[ \begin{array}{c} 0 \\ 0 \\ 0 \end{array} \right] \end{array}
 \end{array}$$

$H$

$h_1 \quad h_2 \quad h_3 \quad h_7$

$$H \subseteq \underline{= 0} \Leftrightarrow \sum_{i=1}^7 c_i h_i = 0$$

$d_{\min} = 3$  { for the Hamming code  
by inspection of the p.c.  
an x.

Defn Given an  $(n-k \times n)$  p.c.  $m \times H$

of an  $[n, k]$  linear block code  $\mathcal{C}$ ,

we define the parameter  $s$  to

be the largest integer s.t. any  
 $s$  columns of  $H$  are linearly  
independent.

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

Pf Follows from the observation  
that the existence of a codeword  
 $\leq$  of Hamming weight  $w$

$\Rightarrow$  the existence of a linear dependence relation amongst  $w$  corresponding columns of the p.c.  $n \times t$ .

Ex  $S = 2$  in the case of the Hamming code, hence  $d_{\min} = 3$ .

# General Hamming code.

Defn. Let  $r \geq 2$  be an integer,  
set  $n = 2^r - 1$ . Then a Hamming  
code of length  $n$  is any code  
possessing a p.c.  $n \times$  size  
 $(r \times (2^r - 1))$  all of whose  
columns are non zero & distinct.

Clearly it follows that the  
general Hamming code has  
parameters

$$[n = 2^r - 1, k = 2^r - 1 - r, 3]$$



Thm (Singleton bound)

$$d_{\min} \leq n - k + 1$$

for any  $[n, k]$  code  $\mathcal{C}$ .

Pf.

$H$

$\Rightarrow$

$$\text{rank}(H) \leq n - k$$

$$(n - k \times n)$$

$$\Rightarrow s \leq (n - k)$$

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.

$$[n, 1, n]$$

$n$   $k$   $d_{\min}$

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

Eg 2 (general) spc code :

$$\left[ \underset{n}{n}, \underset{k}{(n-1)}, \underset{d_{\min}}{2} \right]$$

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

---

Unfortunately these are the only possible families of binary MDS codes.  
(these two classes of codes are sometimes called

trivial codes)

---

## Bounds on the Size of a Code

---

### Hamming Bound

Thm (Hamming Bound) The size  $M$  of an  $(n, M, d)$  code  $\mathcal{C}$  is upper bounded by:

$$|R| \leq \frac{2^n}{\sum_{i=0}^t \binom{n}{i}}$$

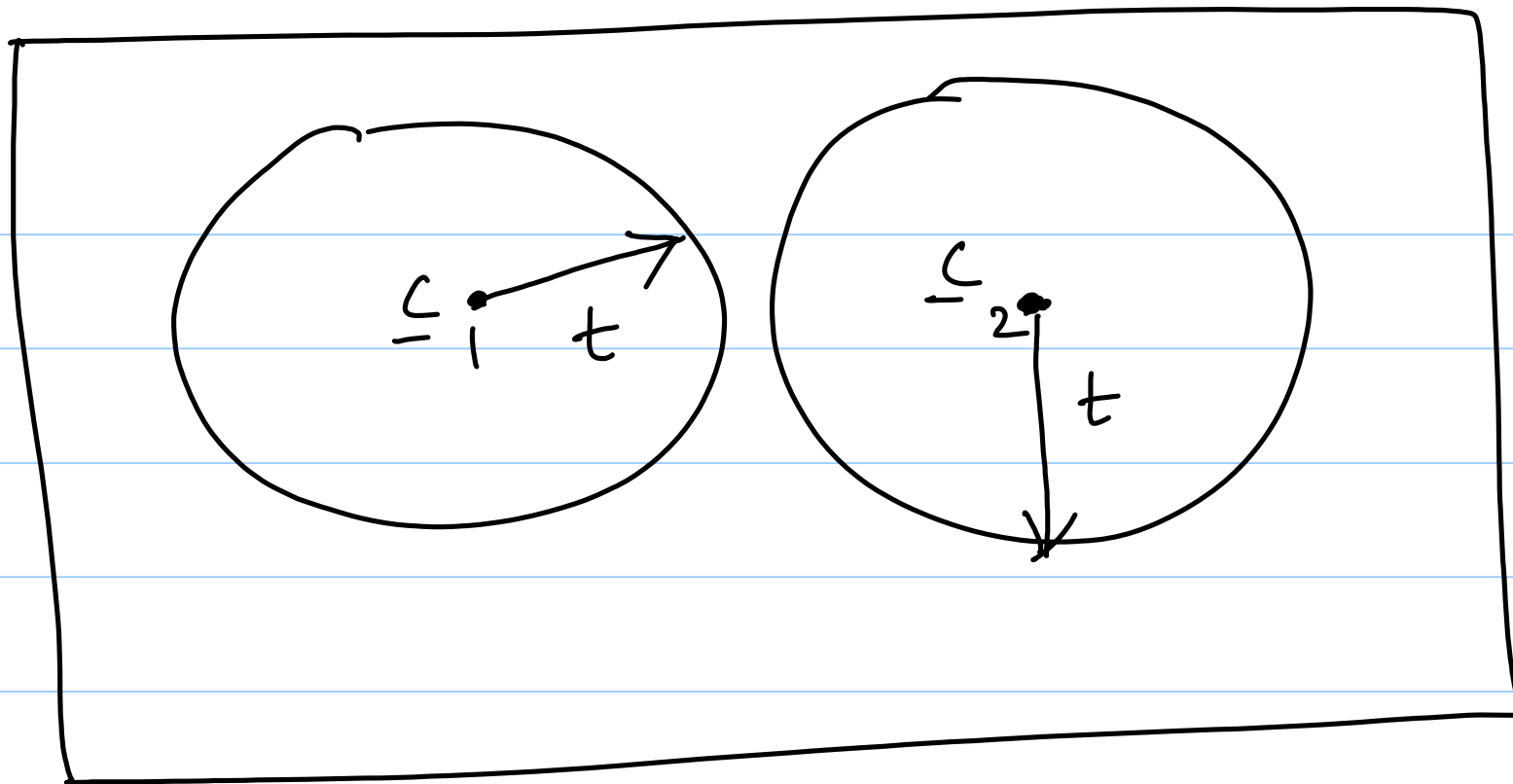
where  $t = \left\lfloor \frac{q_{\min} - 1}{2} \right\rfloor$ .

Pf.

$n$

$\mathbb{F}_2$

2



If  $c_1, c_2$  are code words, then

# Lec 12 : Bounds on the size of a code

## Recap

—  $d_{\min} = w_{\min}$

—  $d_{\min} = s + 1$

— examples

— General Hamming code

— Singleton bound & MDS codes

— Proof of the Hamming Bound

# Hamming Bound

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

$$t \triangleq \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$$

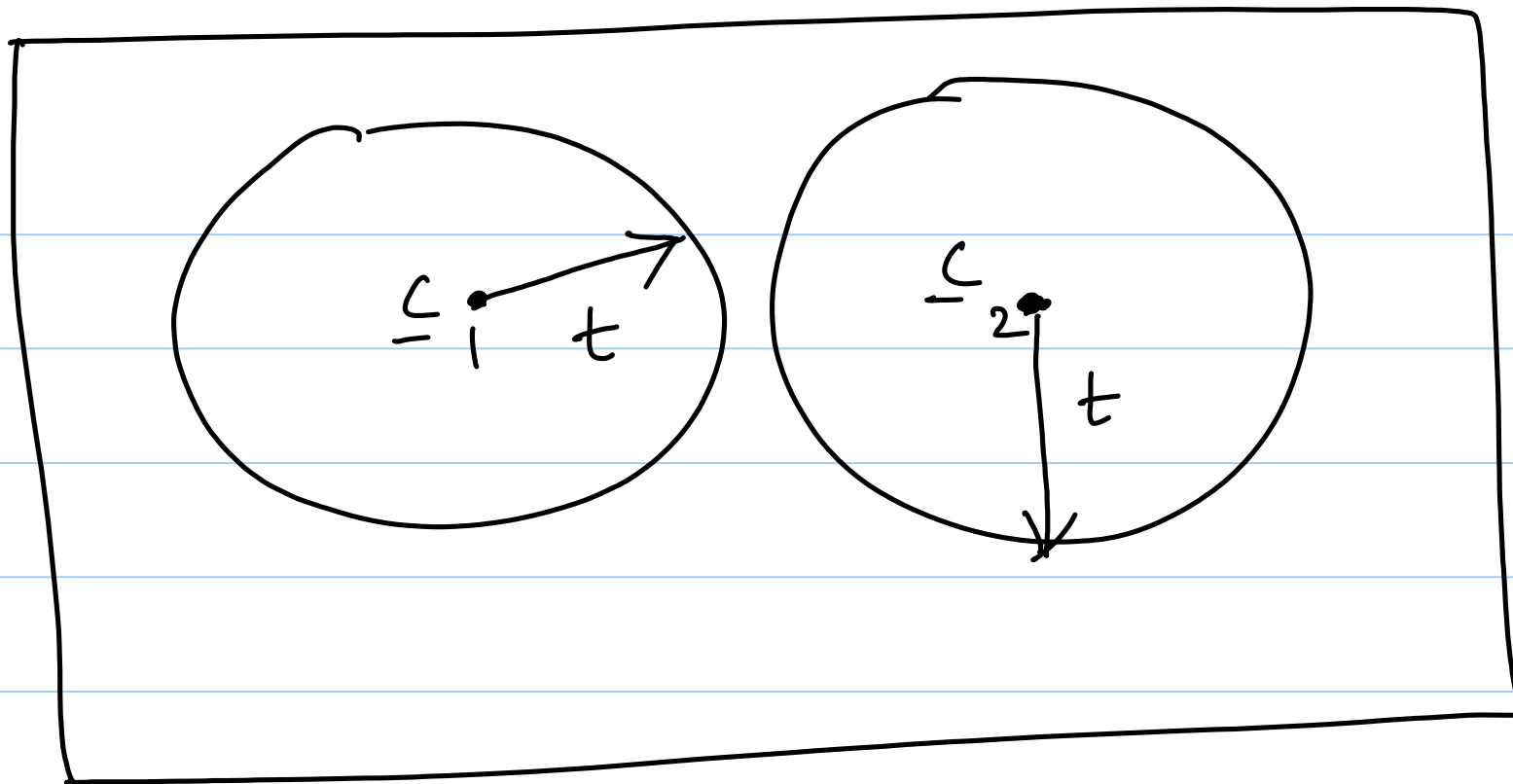


Pf.

$n$

$\mathbb{F}_2$

2



$$\mathbb{F}_2^n \supseteq \left\{ B(c, t) \mid c \in \mathbb{R} \right\}$$

But these balls are disjoint:

$$\mathcal{B}(\underline{c}, t) \cap \mathcal{B}(\underline{c}', t) = \emptyset$$

It follows that

$$2^n \geq M |\mathcal{B}(\underline{c}, t)|$$

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

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

Defn. A perfect code is a code that satisfies the Hamming bound with equality, i.e.,

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

Eg 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

$$\binom{n}{i} = \binom{n}{n-i}, \text{ it follows}$$

that

$$\sum_{i=0}^{\lfloor \frac{n-1}{2} \rfloor} \binom{n}{i} = 2^{n-1}.$$

$$\therefore M \leq \frac{2^n}{2^{n-1}} = 2$$

$\therefore$  all such repetition codes are perfect!

---

Exercise Verify that the single pc code is not perfect in general.

Ex 2 { The general Hamming code  
is always perfect!

Pf. parameters:

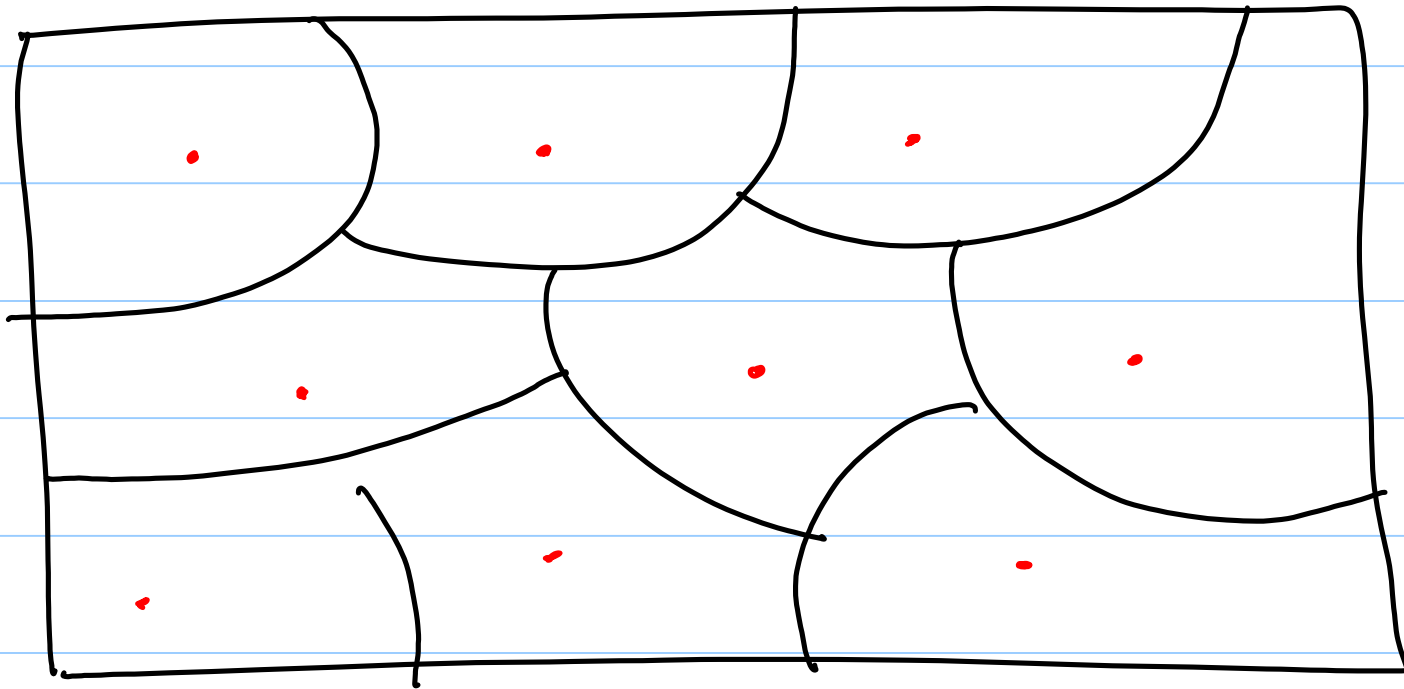
$$\left[ n = 2^{\pi} - 1, \quad k = 2^{\pi} - 1 - \pi, \quad d = 3 \right]^{t=1}$$

$\pi$                        $\pi$   
 $2^{\pi} - 1$

$$M \leq \frac{2}{\sum_{i=0}^1 \binom{n}{i}} = \frac{2}{1 + (2^{\pi} - 1)}$$

$$= \frac{2^2 - 1 - \pi}{2} \quad \therefore \text{perfect.}$$


---



The Hamming bound is also called the sphere-packing bound.



Golay's observation:

If a linear code is perfect.

$$2^k = M = 2^n$$

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

$$\Leftrightarrow \sum_{i=0}^t \binom{n}{i} = 2^{n-k} \quad \dots \textcircled{1}$$

# Ex 3 · The Golay Code

$$n = 23, \quad d = 7, \quad t = 3$$

$$\sum_{i=0}^3 \binom{23}{i} = \binom{23}{0} + \binom{23}{1} + \binom{23}{2}$$

$$= 1 + 23 + \frac{23 \cdot 22}{2}$$

$$+ \frac{23 \cdot 22 \cdot 21}{1 \cdot 2 \cdot 3}$$

$$+ \binom{23}{3}$$

$$\begin{array}{r} 23 \\ 22 \\ \hline 161 \end{array}$$

}

$$= 1 + 23 + 253 + (23)(77) \frac{1771}{1} \}$$

$$= \begin{array}{r} 1771 \\ 253 \\ 23 \\ 1 \\ \hline \end{array}$$

$$\frac{2048}{\hline} = 2 \quad \begin{array}{l} 11 \\ 11 \\ 11 \end{array}$$

This numerical calculation led

Golet to construct a perfect code:

$$[23, 12, 7]$$

— now called the Golay code!

---

Gilbert - Varshamov (lower) bound  
(GV)

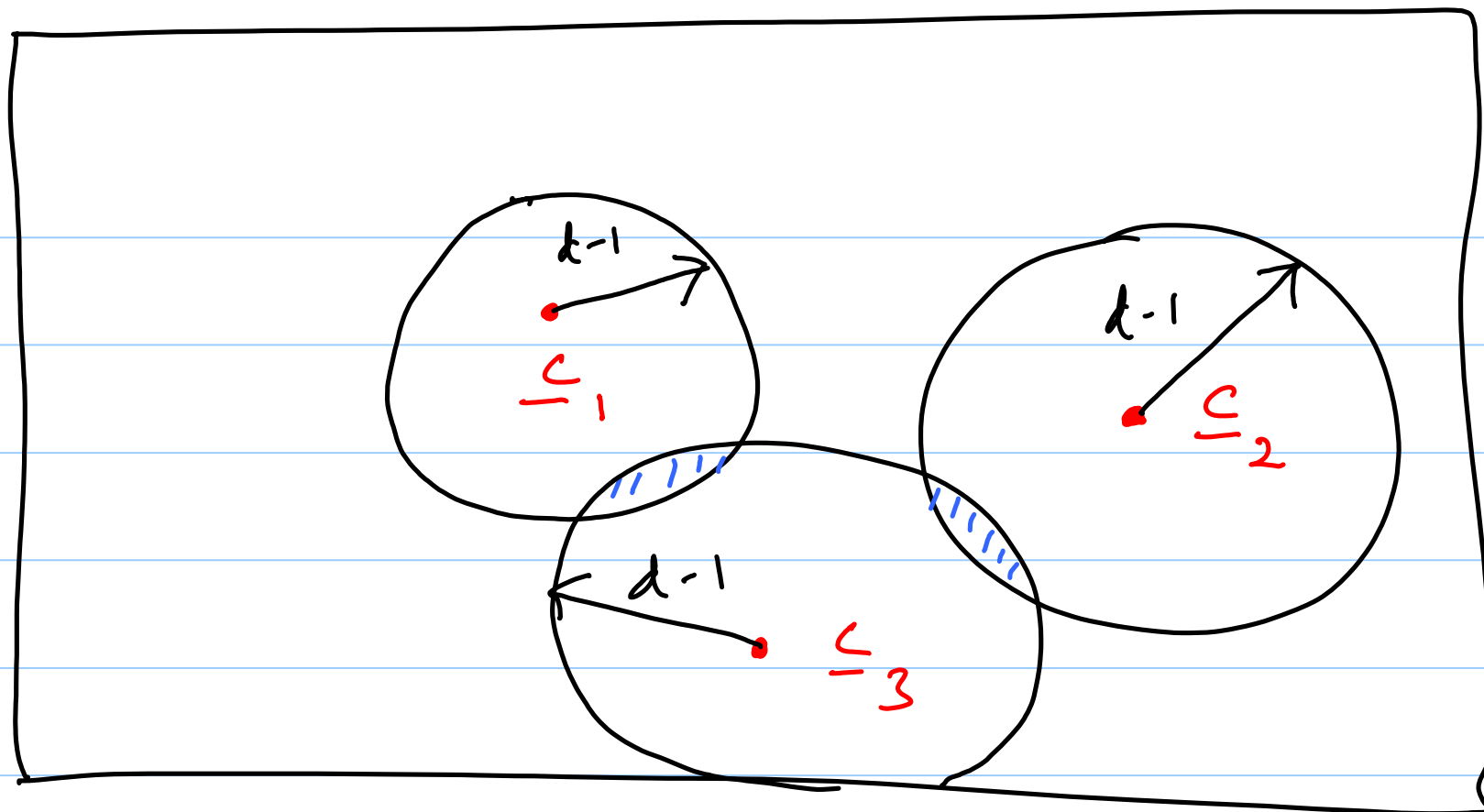
---

Thm (GV bound) The maximum possible size  $M$  of a code of length  $n$  and minimum distance  $d$

satisfies :

$$M \geq \frac{2^n}{\sum_{i=0}^{d-1} \binom{n}{i}}$$

Pf. (via a greedy algorithm for Gdc construction).


 $\mathbb{F}_2^n$ 

As long as

$$M \left[ \sum_{n=0}^{d-1} \binom{n}{n} \right] < 2^n$$

--- (2)

we can always enlarge the code to size  $(M+1)$  while maintaining a min. distance of  $d$ .

At the stopping point we will have

$$M \geq \frac{2^n}{\sum_{i=0}^{d-1} \binom{n}{i}}$$

and this is the G-V bound.

# An approach to attaining reliable communication

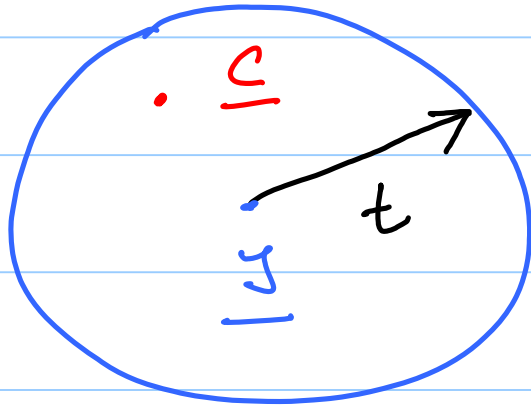
features:

- (i) long codes are employed
- (ii) bounded - distance decoding (BDD).



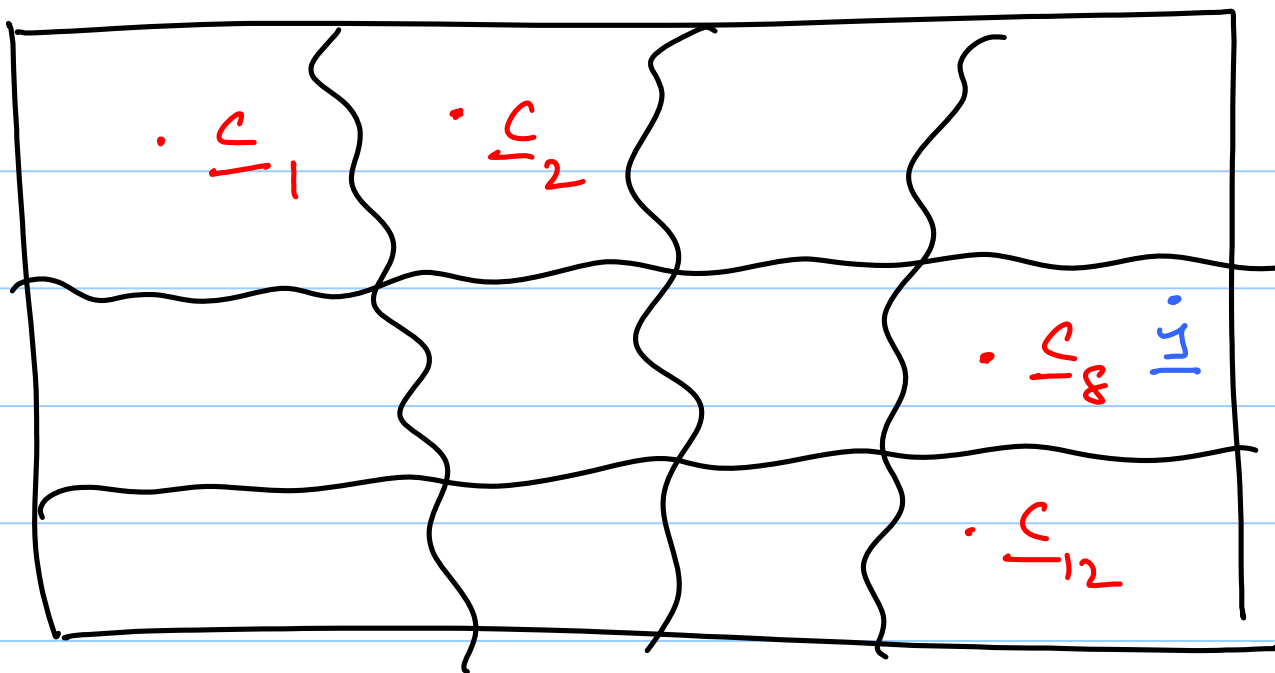
Defn A bounded-distance decoder is one which when given a received

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor$$



y examines the ball  $\mathcal{B}(\underline{y}, t)$  and declares  $\underline{z}$  (the decoded codeword)

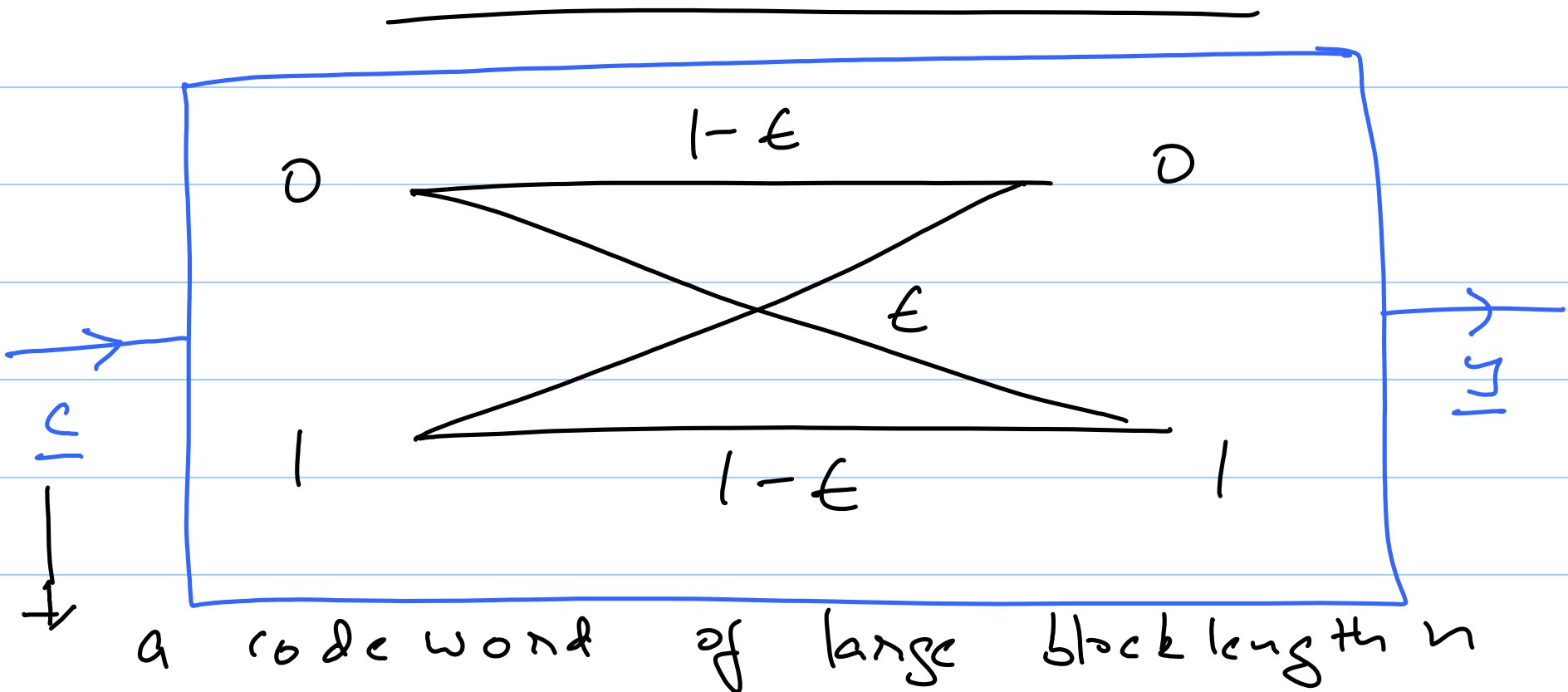
to equal  $\underline{c}$  if  $\underline{c}$  is the only codeword in the ball. (Else, gives up).



$\sim$   
 $\mathbb{F}_2$

note: by reliable communication, we  
communication with negligible probability

of error (virtually error free).



the probability that  $k$  code symbols are corrupted is given by:

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

(comes from the binomial distribution)

when  $n$  is large this distribution tends to become Gaussian with parameters:

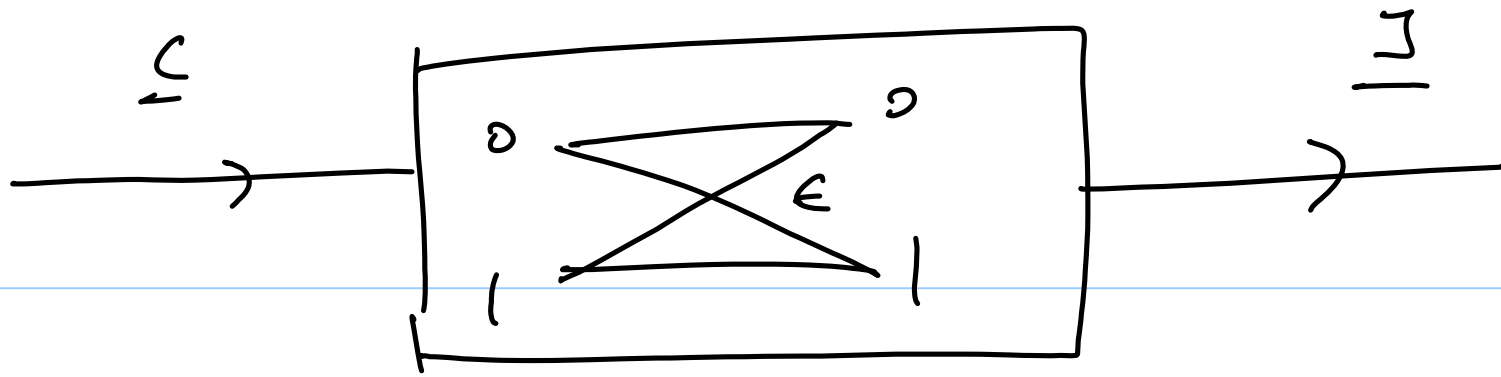
$$\text{mean} = n \epsilon, \quad \text{standard deviation} \\ = \sqrt{n \epsilon (1 - \epsilon)}$$

---

# Lec 13 : Asymptotic Bounds

## Recap

- Hamming bound
- perfect codes
- Gilbert - Varshamov bound
- an approach to achieving reliable communication.



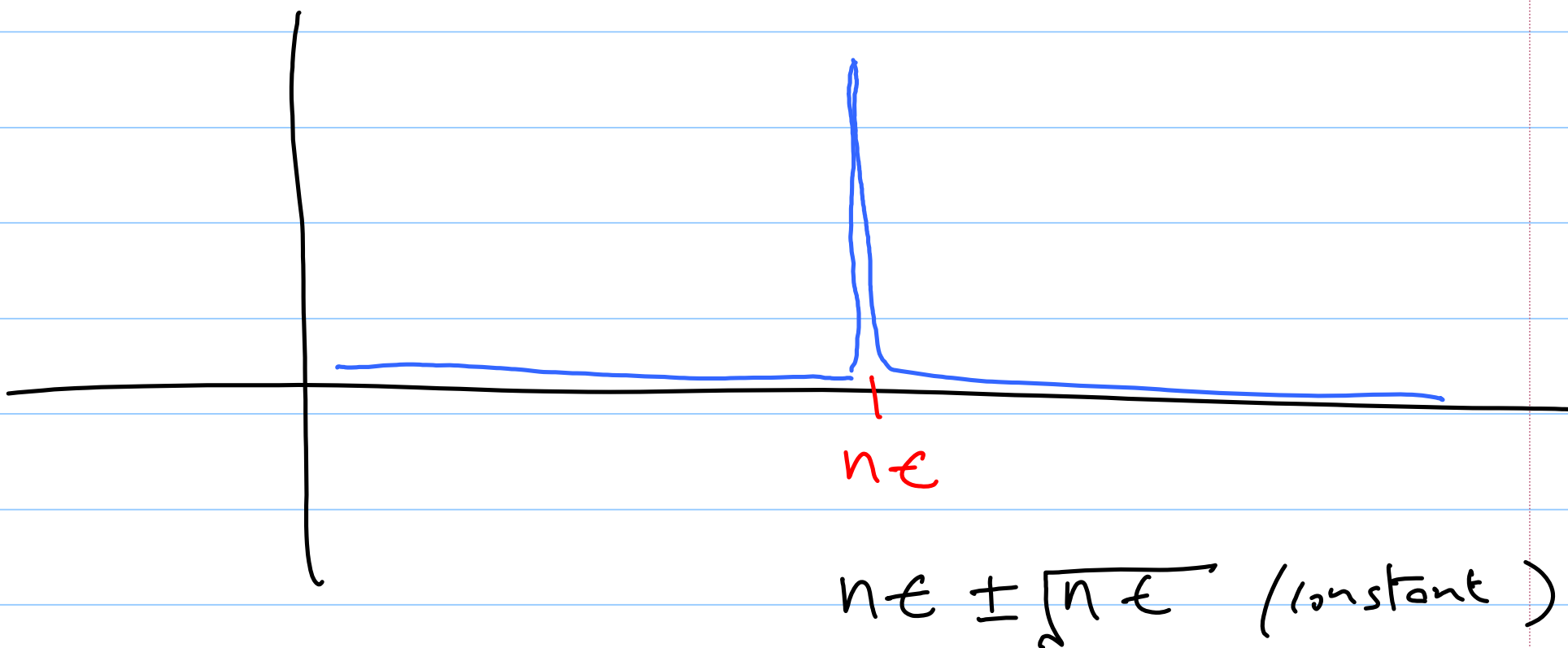
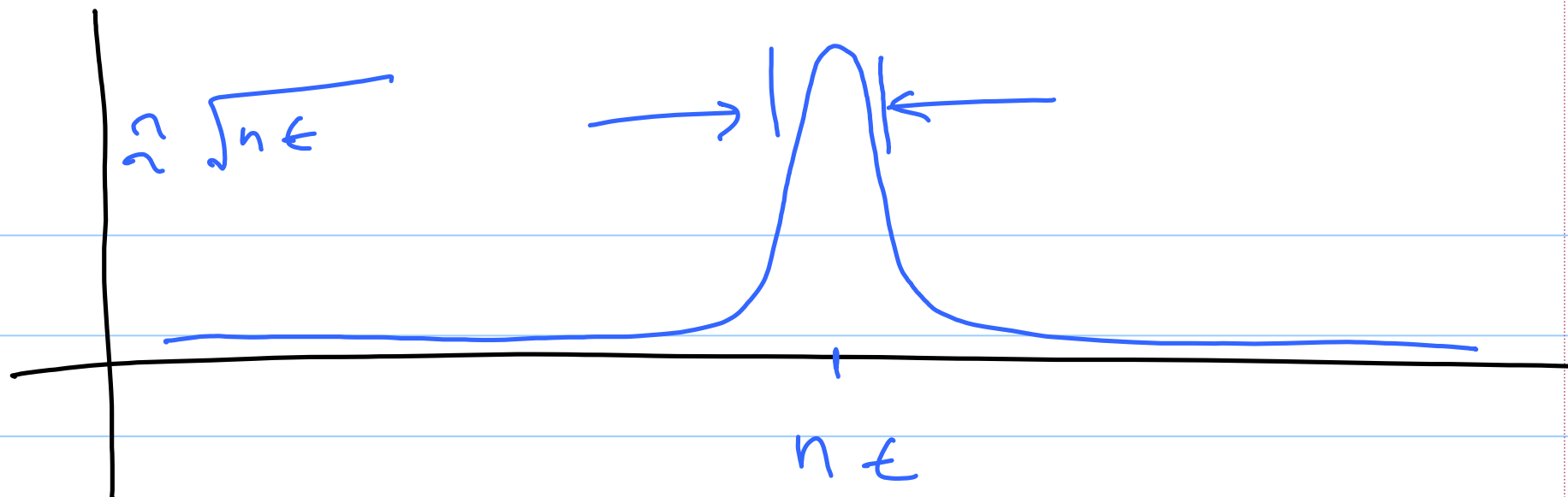
$$P_n(k \text{ errors}) = \binom{n}{k} \epsilon^k (1-\epsilon)^{n-k}$$

→ Gaussian distribution

$$n \left( n\epsilon, \quad n\epsilon(1-\epsilon) \right)$$

variance.

$$\sigma = \text{std dev } n = \sqrt{n\epsilon(1-\epsilon)}$$





Conclusion: long codes make the  
error pattern more predictable and  
hence more correctable.

Since there are  $n \epsilon$  errors, we will  
use a code  $d$

$$d = 2n\epsilon$$

Defn. Given  $0 < \delta < 1$ , let

$d = \lceil n\delta \rceil$  and let  $M(n, \delta)$

be the largest possible size of  
a block code of length  $n$  and

minimum distance  $d$ .

Set:

$$R(\delta) = \limsup_{n \rightarrow \infty} \left[ \frac{\log(M(n, \delta))}{n} \right]$$

(rate)

$\delta =$  fractional minimum distance

$Q_n$  : How does  $K(\delta)$  vary with  $\delta$ ?  
rate

Hamming bound:

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

Gilbert - Varshamov bound:

$$M \geq \frac{2^n}{\sum_{i=0}^{d-1} \binom{n}{i}}$$

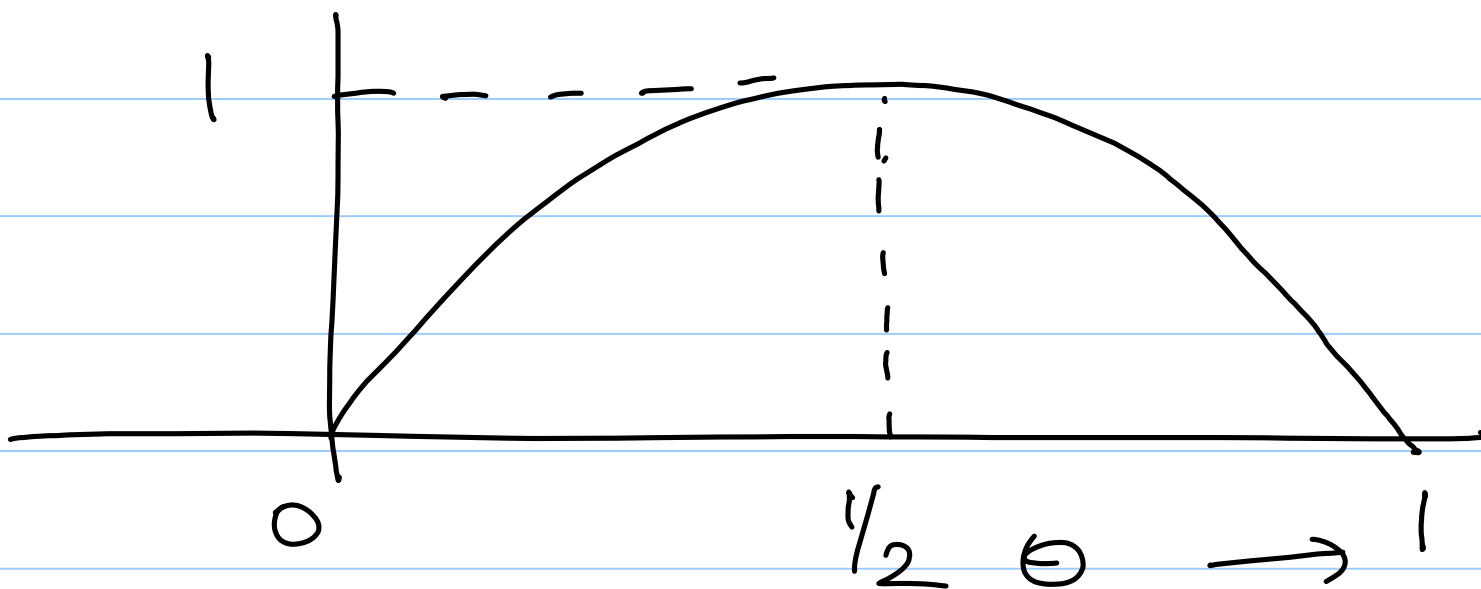
It can be shown that the  
Hamming & GV bounds imply that

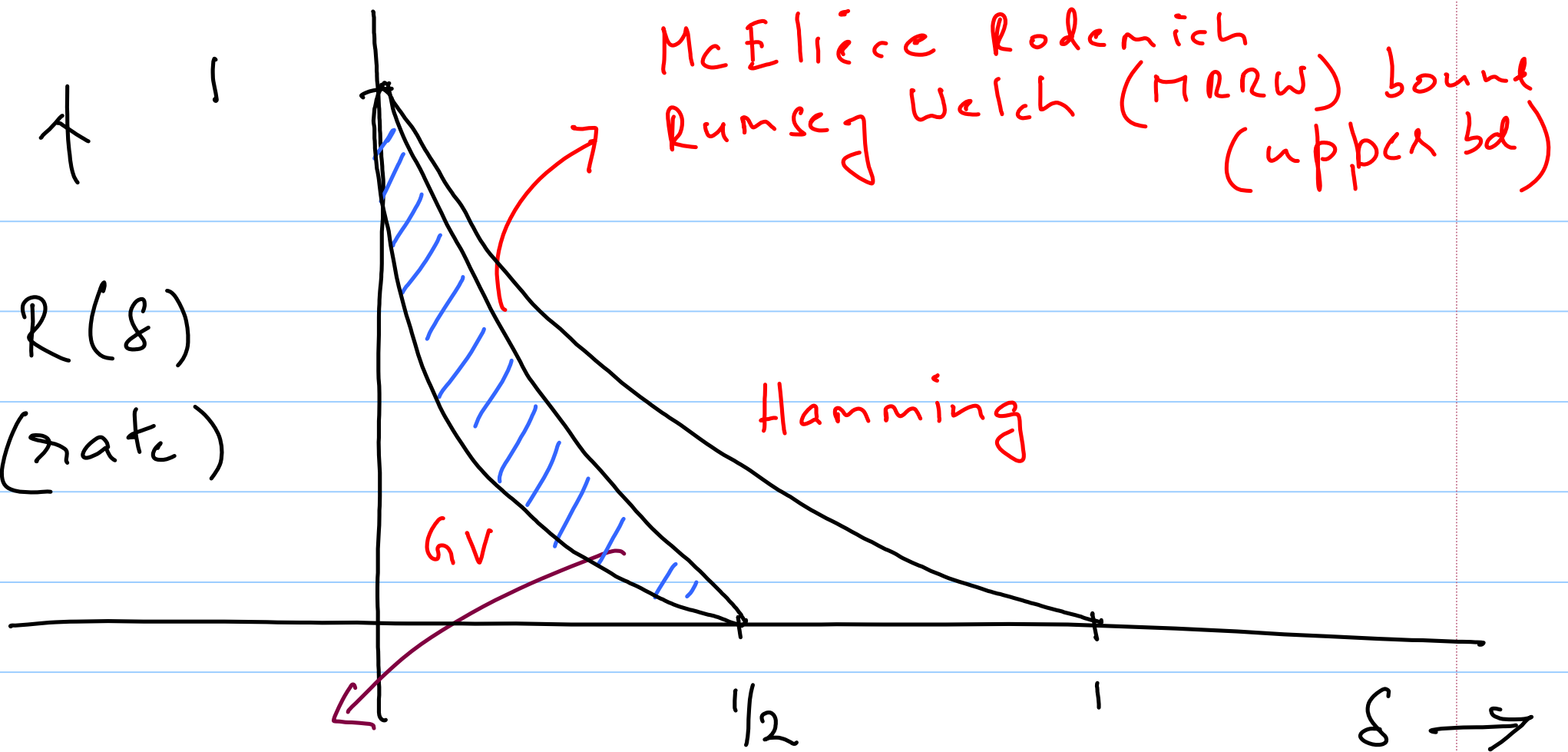
$$1 - H_2(\delta) \stackrel{\text{GV}}{\leq} R(\delta) \stackrel{\text{Hamming}}{\leq} 1 - h_2\left(\frac{\delta}{2}\right)$$

For  $0 \leq \theta \leq 1$ ,

$$H_2(\theta) \triangleq \theta \log \frac{1}{\theta} + (1-\theta) \log \frac{1}{(1-\theta)}$$

binary entropy fn.





{ region where the best codes  
 { lie

$$1 - H_2(\delta) \leq R(\delta) \leq 1 - H_2\left(\frac{\delta}{2}\right)$$

(our BDD philosophy causes us to set

$$\delta = \frac{d}{n} = \frac{2n\epsilon}{n} = 2\epsilon$$

$$1 - H_2(2\epsilon) \stackrel{\text{GV}}{\leq} R(\delta) \stackrel{\text{Hamming}}{\leq} 1 - H_2(\epsilon) \dots \textcircled{1}$$

On the other hand Shannon tells us that

$$R_{\max} \stackrel{\Delta}{=} C = 1 - H_2(\epsilon) \quad \textcircled{2}$$

Conclusion: Our combination of long block length and  $\mathbb{ZDD}$ , causes us to require the use of long block codes that achieve the Hamming bound to achieve channel capacity.



# Minimum Probability Error Decoder

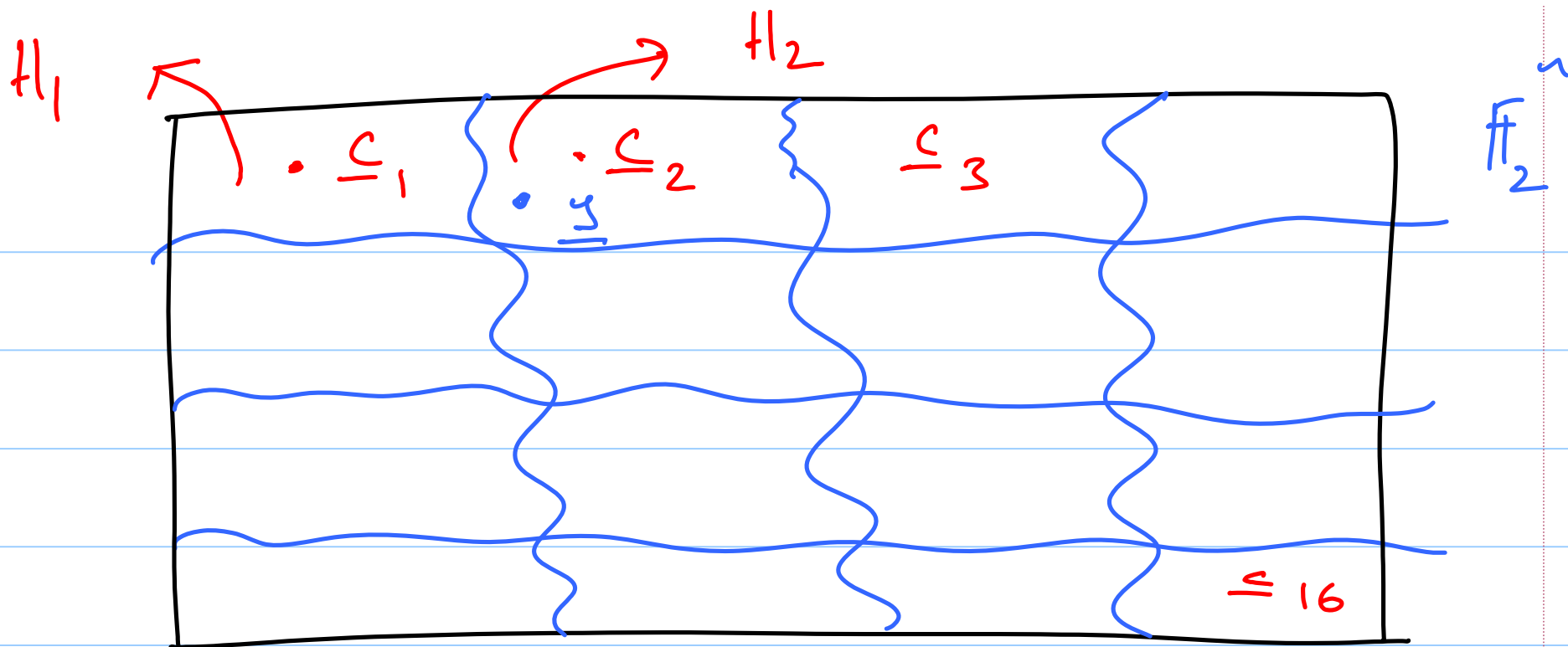
$\Sigma = \left\{ \begin{array}{l} \text{event that a codeword is erroneously} \\ \text{decoded} \end{array} \right.$

$\Sigma^c =$  event that is correctly decoded.

$$\begin{aligned} P_n(\Sigma^c) &= \sum_{i=1}^M P_n(\underline{c}_i) P_n(\Sigma^c | \underline{c}_i) \\ &= \sum_{i=1}^M P_n(\underline{c}_i) P_n(\underline{y} = \underline{c}_i | \underline{c}_i) \end{aligned}$$

$M = |\mathcal{C}|$

$\left. \begin{array}{l} \text{received} \\ \text{vector} \end{array} \right\}$



$$= \sum_{i=1}^M P_n(c_i) \sum_{\underline{y} \in F_2} P_n(\underline{I} | c_i) \underbrace{I_{H_i}(\underline{y})}_{\text{red bracket}}$$

$$I_{H_i}(\underline{z}) = \begin{cases} 1 & \underline{z} \in H_i \\ 0 & \text{else} \end{cases}$$

$$= \sum_{\underline{z} \in \mathbb{F}_2^n} \sum_{i=1}^M \underbrace{P_n(\underline{c}_i) P_n(\underline{z} | \underline{c}_i)} \bar{I}_{H_i}(\underline{z})$$

to maximize the probability of correct decisions, the decoder assigns

$$\begin{aligned} \underline{z} \text{ to } H_i &\Leftrightarrow P_n(\underline{c}_i) P_n(\underline{z} | \underline{c}_i) \\ &\geq P_n(\underline{c}_j) P_n(\underline{z} | \underline{c}_j) \\ &\quad j \neq i \\ &\quad 1 \leq i, j \leq M \end{aligned}$$

Typically, all codewords are equally likely i.e.,  $P_n(\underline{c}_i) = \frac{1}{M}$  all  $i$ .

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

$\underline{y} \in A_i$  iff

$$P_n(\underline{y} | \underline{c}_i) \geq P_n(\underline{y} | \underline{c}_j)$$

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 codeword error probability.

note: in case of ties, one flips a coin.

---

Lemma Over a BSC, MLD reduces  
to minimum distance decoding (MDD)

Pf. Let  $d_H(\underline{y}, \underline{c}_n) = d$

$$p_2(\underline{y} | \underline{c}_n) = \epsilon^d (1-\epsilon)^{n-d}$$

$$= (1-\epsilon)^n \left( \frac{\epsilon}{1-\epsilon} \right)^d$$

$$\text{If } \epsilon < 1, \quad \frac{\epsilon}{1-\epsilon} < 1$$

$\Rightarrow P_{\pi}(\underline{y} | \underline{c}_i)$  is maximized  
by minimizing  
 $d_H(\underline{y}, \underline{c}_i)$

---

# Lec 14 Standard Array Decoding

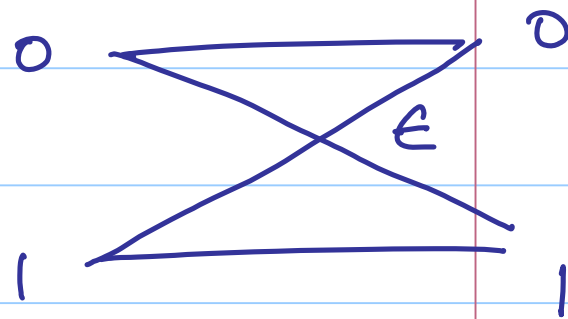
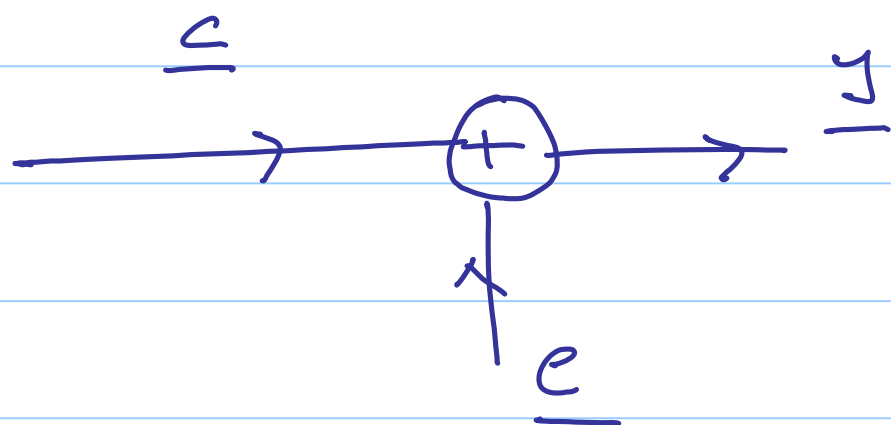
## Recap

- asymptotic bounds
- min prob of error decoder  $\rightarrow$  MLD
- maximum likelihood decoder
- { minimum Hamming distance  
decoder MDD



MDD

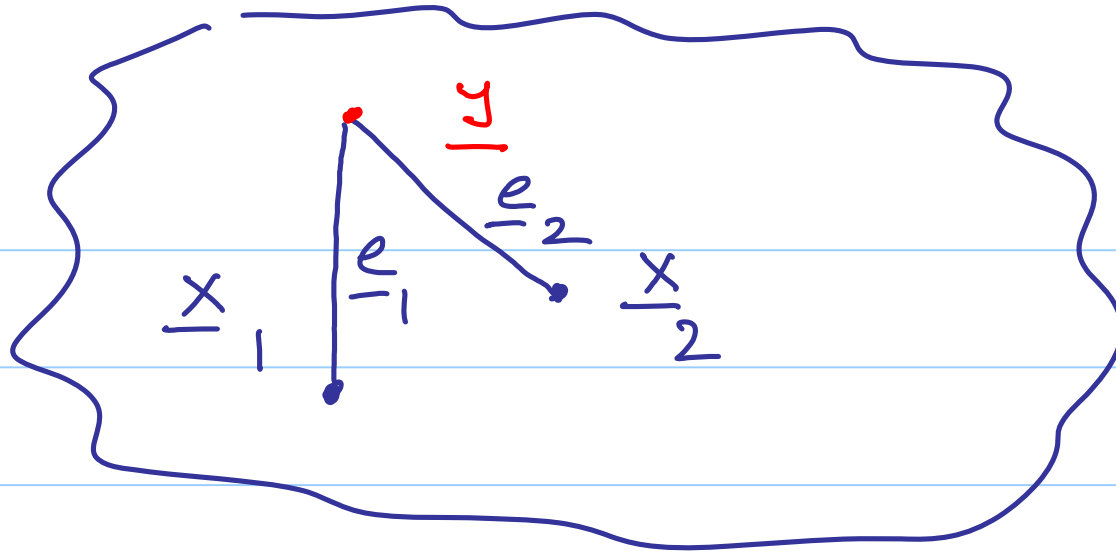
Channel Model:



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

$d_H(\underline{y}, \hat{x})$  is a minimum  
with  $\hat{x} \in \mathcal{C}$  (code)

$\Rightarrow$



$F_2^n$

$$\underline{y} + \underline{x}_1 = \underline{e}_1$$

$$\underline{y} + \underline{x}_2 = \underline{e}_2$$

Thus the decoding (MDD) algorithm  
can equivalently be phrased as follows:

Step 1 form the set

$$\underline{y} + \mathcal{C} \triangleq \left\{ \underline{y} + \underline{c} \mid \underline{c} \in \mathcal{C} \right\}$$

Step 2 Let  $\underline{\hat{c}}$  be the element in

$\underline{y} + \mathcal{C}$  having least Hamming weight.

Step 3 the decoded codeword  $\underline{\hat{c}}$  is then given by

$$\underline{\hat{c}} = \underline{y} + \underline{\hat{e}}$$

Note that (i)  $\underline{y} + \mathcal{R}$  is a coset of

the subgroup  $\mathcal{R}$  of  $\mathbb{F}_2^n$ .

(ii) and hence the decoder action is

only a fn of the coset of  $\mathcal{R}$  to which  $\underline{y}$  belongs and not to  $\underline{y}$  itself.

Ex  $\mathcal{C} = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$

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

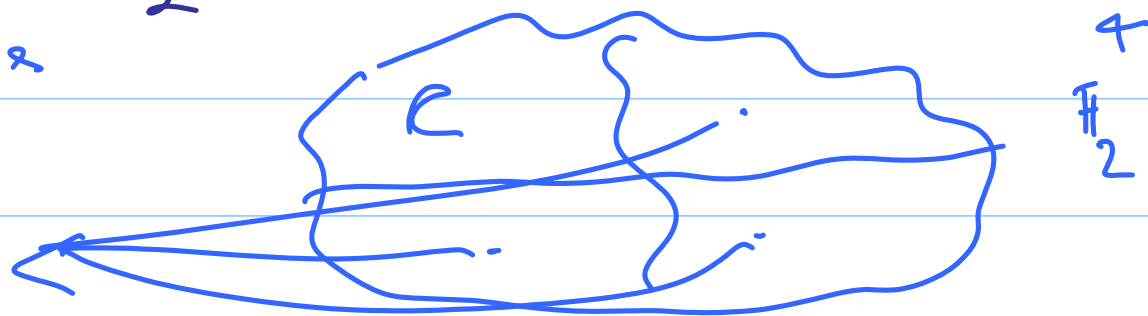
$$G = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

$I_2 \quad P$

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

3 other  
cosets of

$\mathcal{C}$



{ coset  
leaders  
↑

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

$$n - k = 2$$

s

C

0001 + C

0010 + C

0011 + C

0000	1010	0101	1111	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$
0001	1011	0100	1110	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$
0010	1000	0111	1101	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
0011	1001	0110	1100	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

Ex (of decoding)  $\underline{y} = 0111$

MDD algorithm  $\Rightarrow \underline{y} + \underline{e}$

$$= 0111 + 0010 = 0101 \checkmark$$

Defn. The syndrome  $\underline{s}$  associated to a received vector  $\underline{y}$  is given by:

$$\underline{s} = H \underline{y}$$

$(n-k \times n)$

Ex  $\underline{y} = 0111$        $H = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$

$$\underline{s} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Lemma There is a 1-1 correspondence  
between the cosets  $\{\underline{y} + \mathcal{R}\}$  of the code  
and syndromes  $\underline{s} \in \mathbb{F}_2^{n-k}$

Pf.  $\phi: \underline{y} + \mathcal{R} \rightarrow H \underline{y} = \underline{s}$

Is  $\phi$  well-defined?

Suppose  $\underline{y}' \in \underline{y} + \mathcal{R}$

$$\Rightarrow \underline{y}' = \underline{y} + \underline{c}, \quad \underline{c} \in \mathcal{R}$$

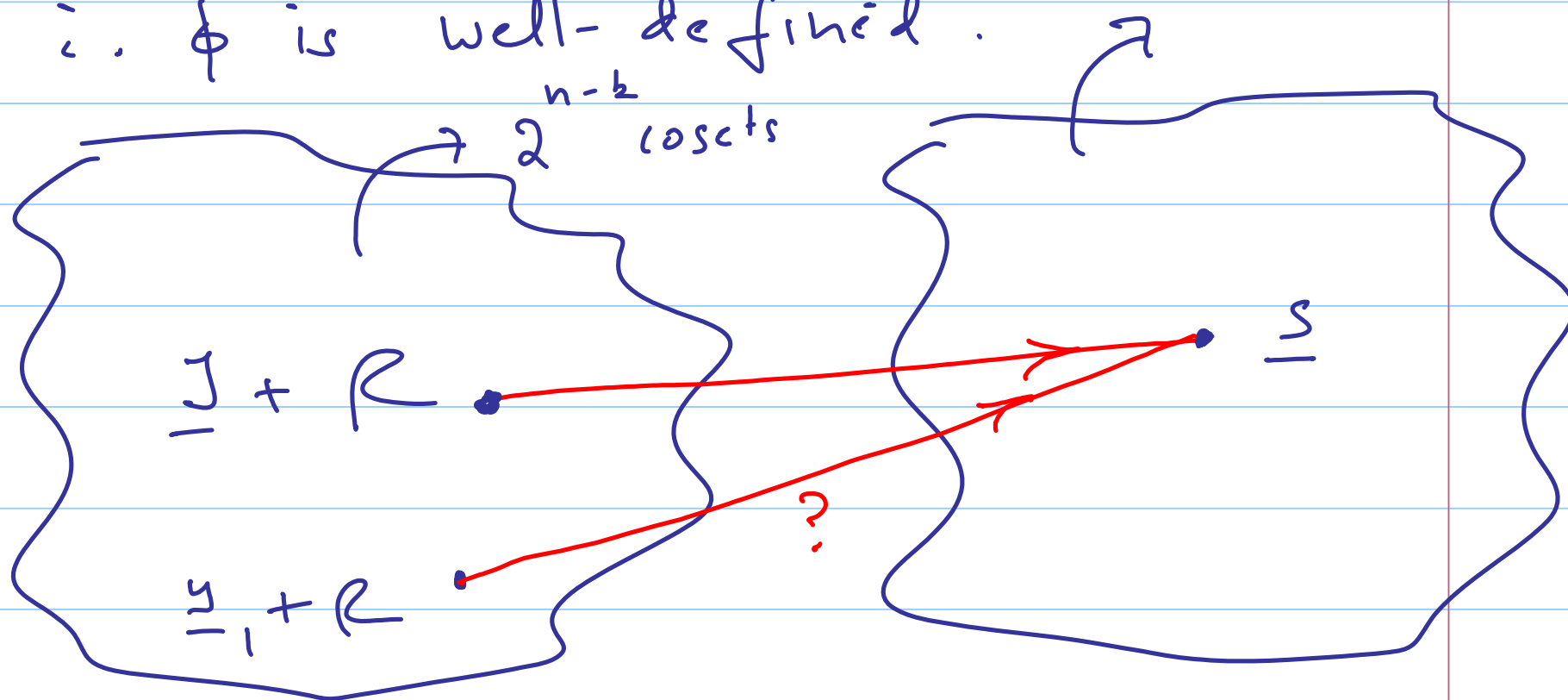


$$\therefore H_{\underline{y}'} = H_{\underline{y}} + H_{\underline{c}} \stackrel{||}{=} 0$$

$$\therefore \boxed{H_{\underline{y}'} = H_{\underline{y}}}$$

$$\text{size} = 2^{n-k}$$

$\therefore \phi$  is well-defined.



Suppose  $\phi(\underline{y} + \mathcal{R}) = \phi(\underline{y}_1 + \mathcal{R})$

$$\Leftrightarrow H\underline{y} = H\underline{y}_1$$

$$\Leftrightarrow H(\underline{y} + \underline{y}_1) = \underline{0}$$

$$\Leftrightarrow \underline{y} + \underline{y}_1 \in \mathcal{R}$$

$$\Leftrightarrow \underline{y} + \underline{y}_1 = \underline{c}, \quad \underline{c} \in \mathcal{R}$$

$$\Leftrightarrow \underline{y}_1 = \underline{y} + \underline{c}, \quad \underline{c} \in \mathcal{R}$$

$\Leftrightarrow \underline{y}_1, \underline{y}$  define the same

cost.

//

---

This leads to the following simple implementation of the MD algorithm (syndrome decoding):

Step 1      compute syndrome

$$\underline{S} = H \underline{J}$$

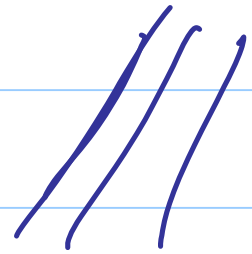
Step 2

Use table lookup to determine  
 $\hat{e}$  (the coset leader  
associated to syndrome  
 $\underline{s}$ )

Step 3

decode to:

$$\underline{\hat{c}} = \underline{y} + \hat{e}$$



# Performance Analysis via the Standard Array

0000	1010	0101	1111	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$
0001	1011	0100	1110	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$
0010	1000	0111	1101	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
0011	1001	0110	1100	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

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

associated to the code.



Lec 15

# Performance Analysis of the SAD

## Recap

- introduced the standard array
- defined the syndrome
- laid down the steps involved in carrying out SAD

# Performance Analysis via the Standard

residual  
error vector  $\underline{e}$  Array

$\underline{e}$	0000	1010	0101	1111	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	
	0001	1011	0100	1110	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	
lost leader	0010	1000	0111	1101	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\underline{s}$
leader	0011	1001	0110	1100	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$	

$\underline{e} = \text{true error pattern}$

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



Lemma 1 The received vector and the error pattern  $\underline{e}$  belong to the same coset of the code and hence share the same syndrome.

pf.  $\underline{y} = \underline{c} + \underline{e}, \quad \underline{c} \in \mathcal{C}$

$\therefore \underline{y}, \underline{e}$  belong to the same coset of  $\mathcal{C}$ .

$$H(\underline{y}) = H(\underline{c} + \underline{e}) = H\underline{e}$$

and hence  $\underline{y}, \underline{e}$  share the same

Syndrome.

Suppose  $\underline{e}$  to be the true error pattern.

the decoder computes  $H\underline{y} = H\underline{c} = \underline{s}$

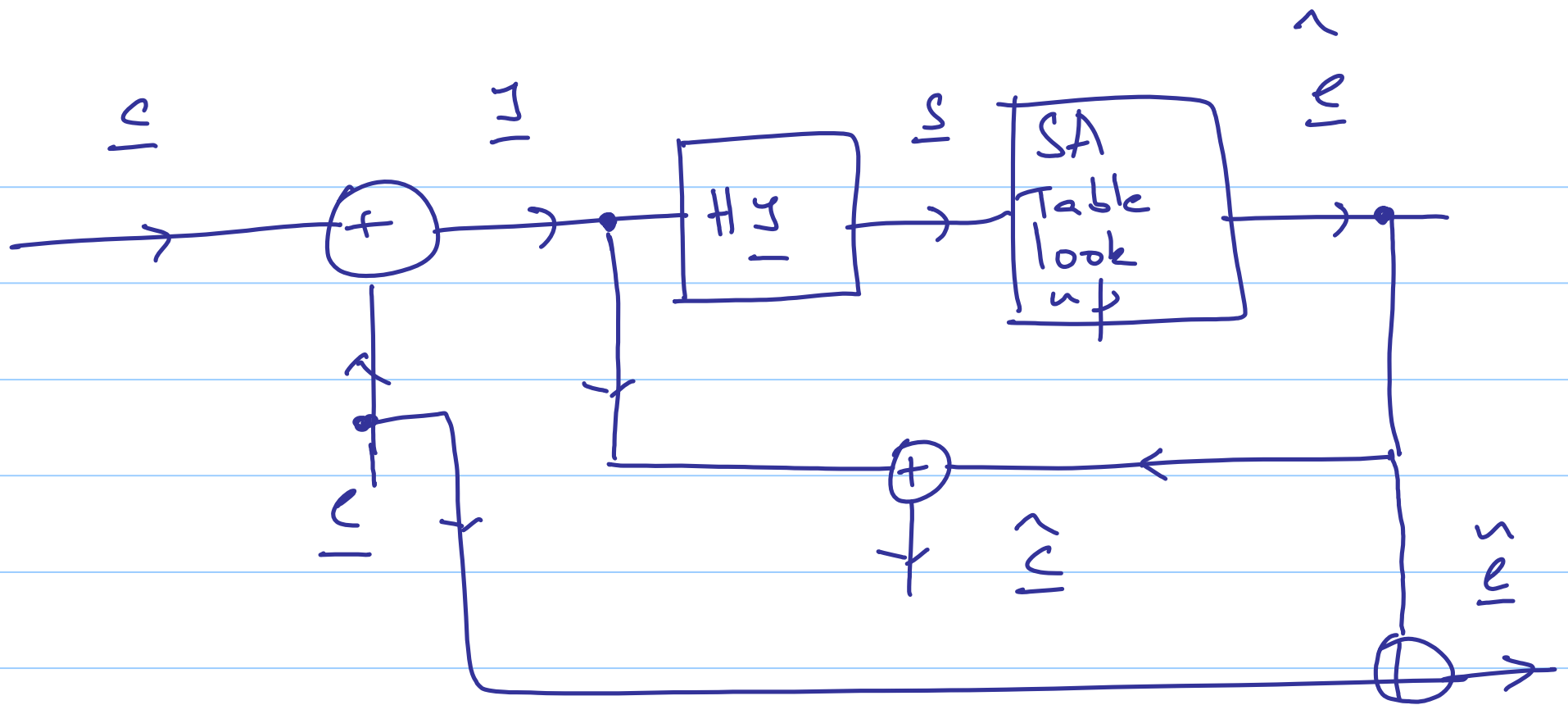
let  $\hat{\underline{e}}$  be the wset leader associated to syndrome  $\underline{s}$ .

Decoder computes:

$$\underline{y} + \underline{\hat{e}} = \underline{c} + \underbrace{\underline{e} + \underline{\hat{e}}}_{\substack{\sim \\ \underline{e} \\ \Downarrow}}$$

residual error pattern

It follows then that in the standard array, the residual error vector  $\underline{\hat{e}}$  is the vector in the table at the head of the column to which  $\underline{e}$  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!!

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

Hence the prob.

of codeword error is given by:

$$P_{cwe} = 1 - \left\{ (1-\epsilon)^4 + 2(1-\epsilon)^3\epsilon + (1-\epsilon)^2\epsilon^2 \right\}$$

$\underline{m}^t$   
 $\mathbb{C}$

00	10	01	11
<u>0000</u>	<u>1010</u>	<u>0101</u>	<u>1111</u>
0001	1011	0100	1110
0010	1000	0111	1101
0011	<u>1001</u>	0110	1100

$$\underline{m}^t G = \underline{c}^t$$

residual  
 message  
 - error patterns!

Fig 1

$$\underline{e} = 1001 \quad \underline{\hat{e}} = 0011$$

$$\therefore \underline{e} = 1001 + 0011 = 1010$$

$\Rightarrow m_1$  is decoded erroneously

$m_2$  is decoded correctly

$$\hat{\underline{c}} = \underline{c} + \underline{e} + \hat{\underline{e}} = \underline{c} + \hat{\underline{e}}$$

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

It follows that the probability that  $m_1$  is erroneously decoded while

$m_2$  is correctly decoded is given by:

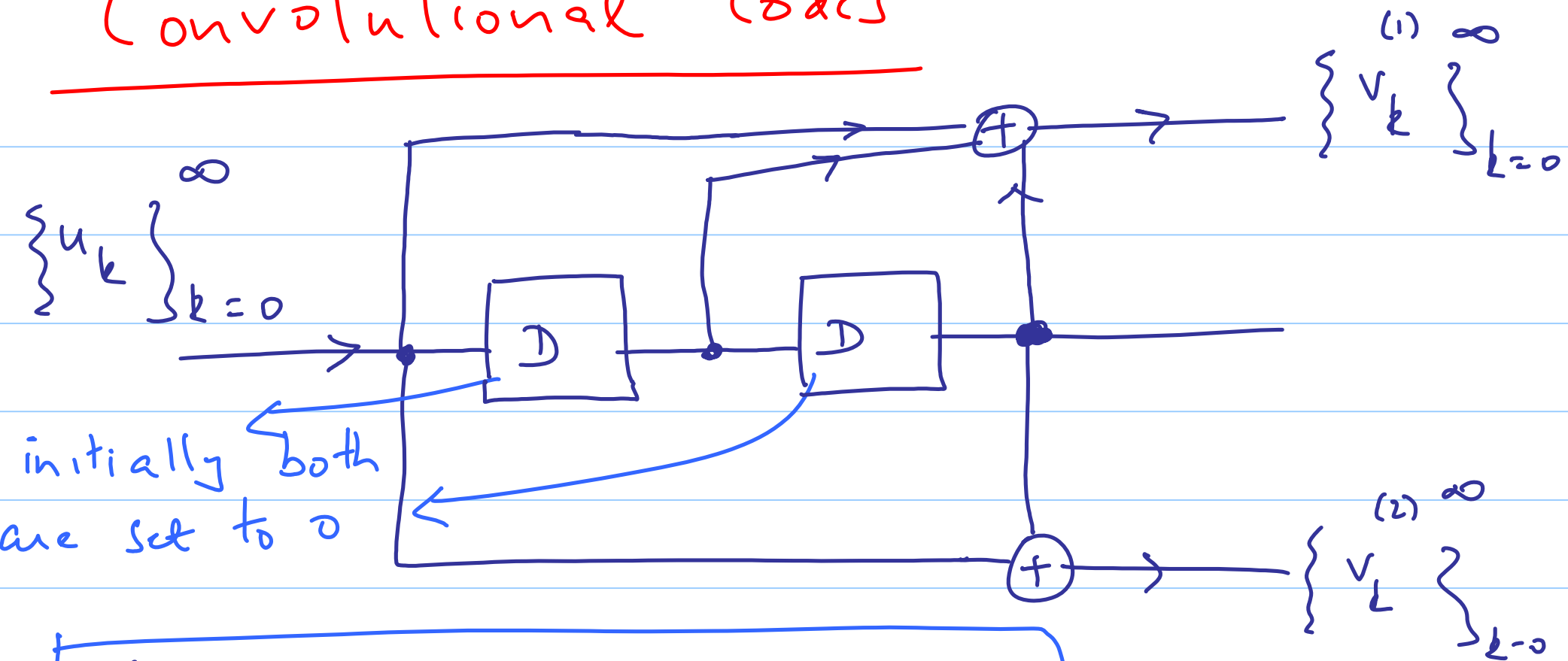
$$= (1-\epsilon)^3 \epsilon + 2(1-\epsilon)^2 \epsilon^2 + (1-\epsilon) \epsilon^3$$

Note: The residual error pattern  $\tilde{e}$  is independent of the transmitted codeword and this is what enables this analysis to be carried out.

---



# Convolutional Codes

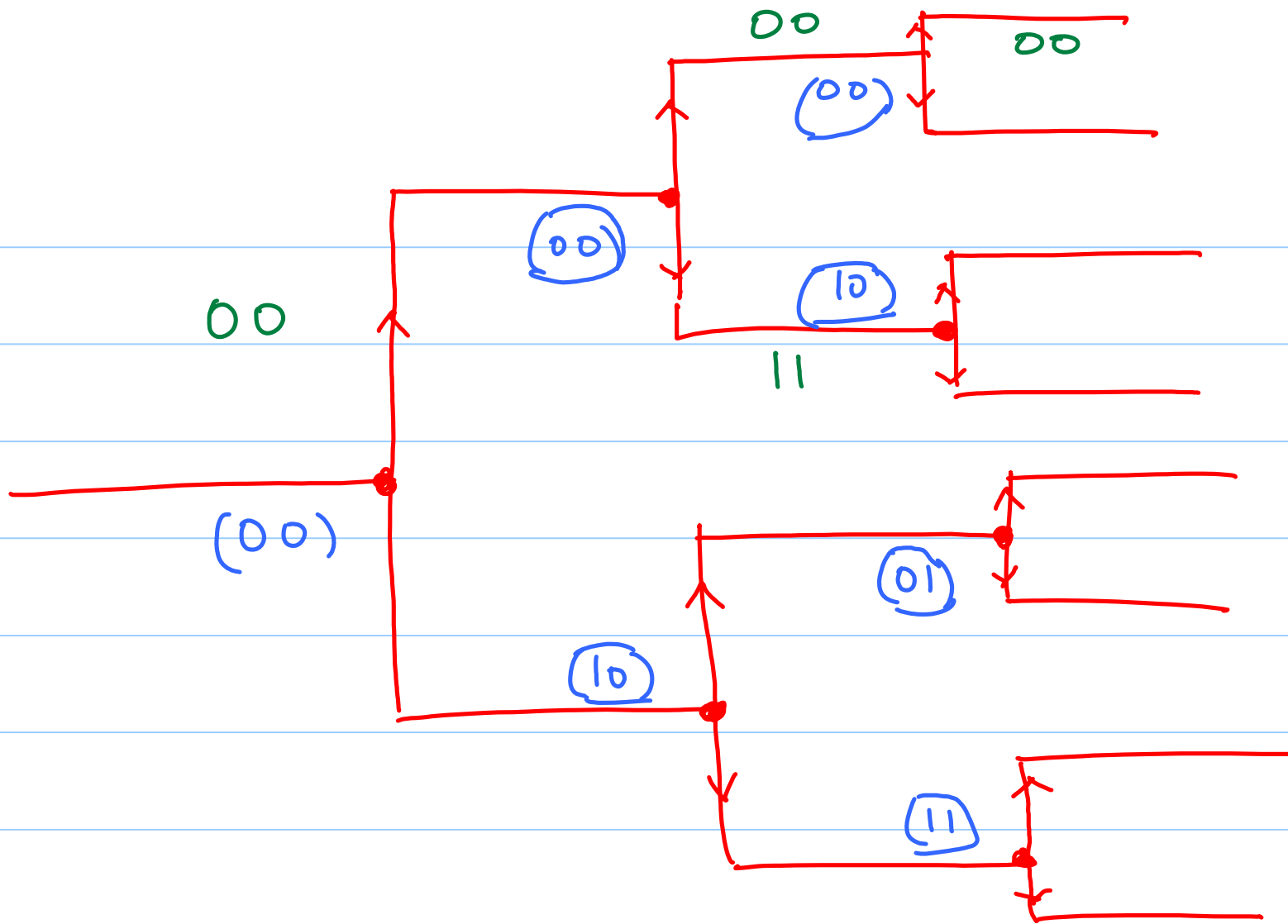


(1)

$$v_k = u_k + u_{k-1} + u_{k-2}$$

(2)

$$v_k = u_k + u_{k-2}$$



$$\uparrow u_k = 0$$

$$\downarrow u_k = 1$$

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

$u_k$	STATE	$V_k^{(1)}$	$V_k^{(2)}$
0	00	0	0
1	00	1	1
0	01	1	1
1	01	0	0
0	10	1	0
1	10	0	1
0	11	0	1
1	11	1	0

$(V_k^{(1)}, V_k^{(2)})$

Convolutional codes belong to the class

↓ free codes that are:

- (i) finite memory
- (ii) linear
- (iii) time invariant

$$\begin{bmatrix} u_0 & u_1 & u_2 \end{bmatrix} \begin{bmatrix} 11 & 10 & 11 \\ & 11 & 10 & 11 \\ & & 11 & 10 & 11 \\ & & & 11 & 10 & \ddots \\ & & & & 11 & \ddots \\ & & & & & \ddots \end{bmatrix} = \begin{bmatrix} v_0^{(1)} & v_0^{(2)} & v_1^{(1)} & v_1^{(2)} \\ & & & \\ & & & \ddots \end{bmatrix}$$

{ Semi-infinite  
 generator mx.

Field of formal power series  $\mathbb{F}((x))$   
over the scalar field  $\mathbb{F}$

$$\mathbb{F}((x)) = \left\{ \sum_{k=-d}^{\infty} a_k x^k \mid \begin{array}{l} a_k \in \mathbb{F} \\ d \geq 0 \end{array} \right\}$$

It is clear that all field axioms  
are satisfied with the possible exception  
of the multiplicative inverse.

Ex  $(1 + D^3)^{-1} = [1(1 + D^2)]^{-1}$

# Lec 16 State and Trellis

## Recap

- complete performance analysis of the SAD
- convolutional codes
  - encoder
  - $\left\{ \begin{array}{l} \text{semi-infinite generator} \\ \text{matrix} \end{array} \right.$



— formal power series

—

$$\mathbb{F}((D)) = \left\{ \sum_{k=-\infty}^{\infty} a_k D^k \mid a_k \in \mathbb{F} \right\}$$

$$(\mathbb{F}((D)), +, \cdot)$$

$(\mathbb{F}((D)), +)$  Abelian group

$$1 + D + D^3 + D^7 + D^{19} + \dots$$

$$+ (D^4 + D^5 + D^6 + D^7 + D^8)$$

$$= \underline{1 + D + D^3 + D^4 + D^5 + D^6 + D^7 + D^8 + D^{19} + \dots}$$

$(F(D), \cdot) \Rightarrow$  closure  
associative  
i.e. = 1  
commutative  
inverse ?

Computing inverses in  $\mathbb{F}((D))$ :

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

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

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

ASIDE:  $\frac{1}{1 + g(D)}$  polynomial in  $D$

that is divisible by  $D$ ,

then

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

Pf.

$$(1+g(D)) \left( 1 + g(D) + [g(D)]^2 + [g(D)]^3 + \dots \right) = 1$$

//

$$\therefore \frac{D^{-1}}{1+D^2} = D^{-1} (1 + D^2 + D^4 + D^6 + \dots)$$

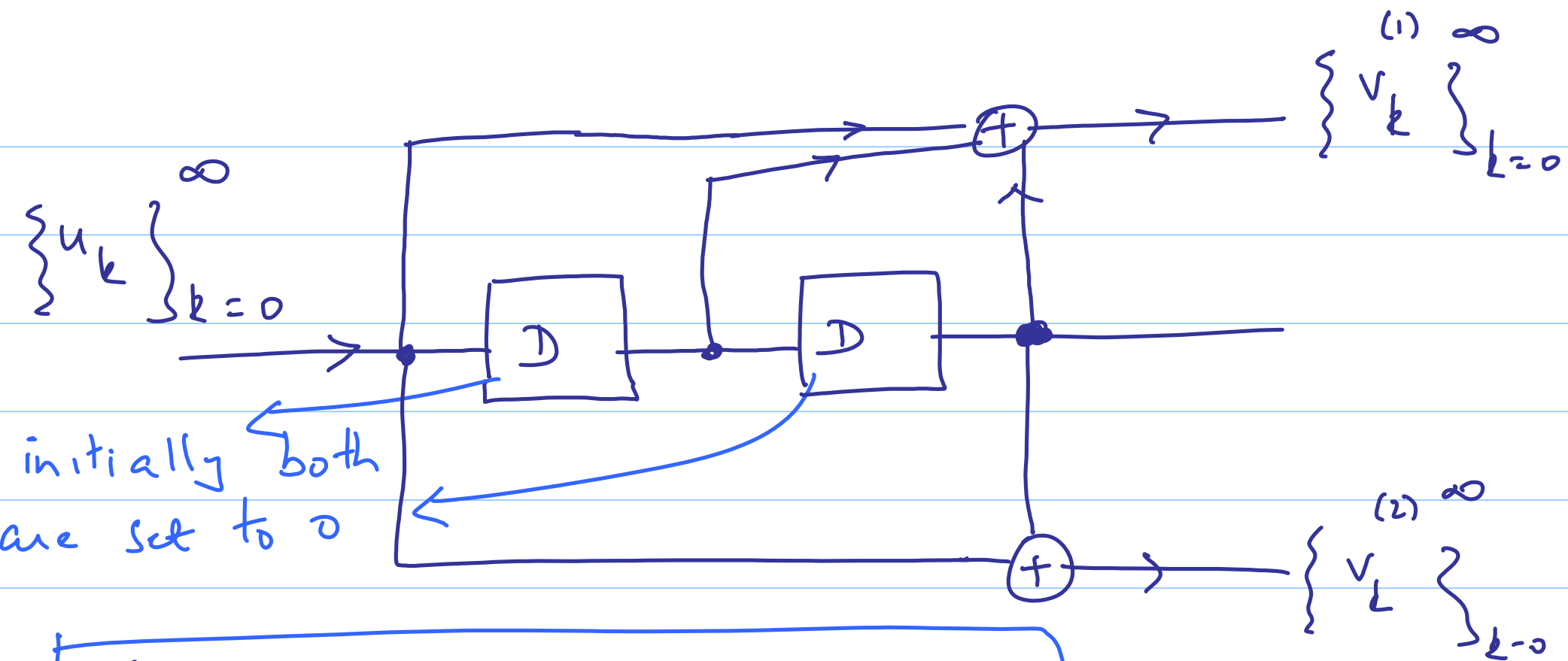
$$= D^{-1} + D + D^3 + D^5 + \dots$$


---

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

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

(1 × 2)



(1)

$$v_k = u_k + u_{k-1} + u_{k-2} + \dots + 1$$

(2)

$$v_k = u_k + u_{k-2}$$

Define:

$$U(D) \triangleq \sum_{k=0}^{\infty} u_k D^k \quad \left. \begin{array}{l} \text{input power} \\ \text{series} \end{array} \right\}$$

$$V^{(1)}(D) \triangleq \sum_{k=0}^{\infty} v_k^{(1)} D^k \quad \dots \quad \textcircled{2}$$

$$V^{(2)}(D) \triangleq \sum_{k=0}^{\infty} v_k^{(2)} D^k$$

converting the time-domain input-output relation given in ① to the D-transform



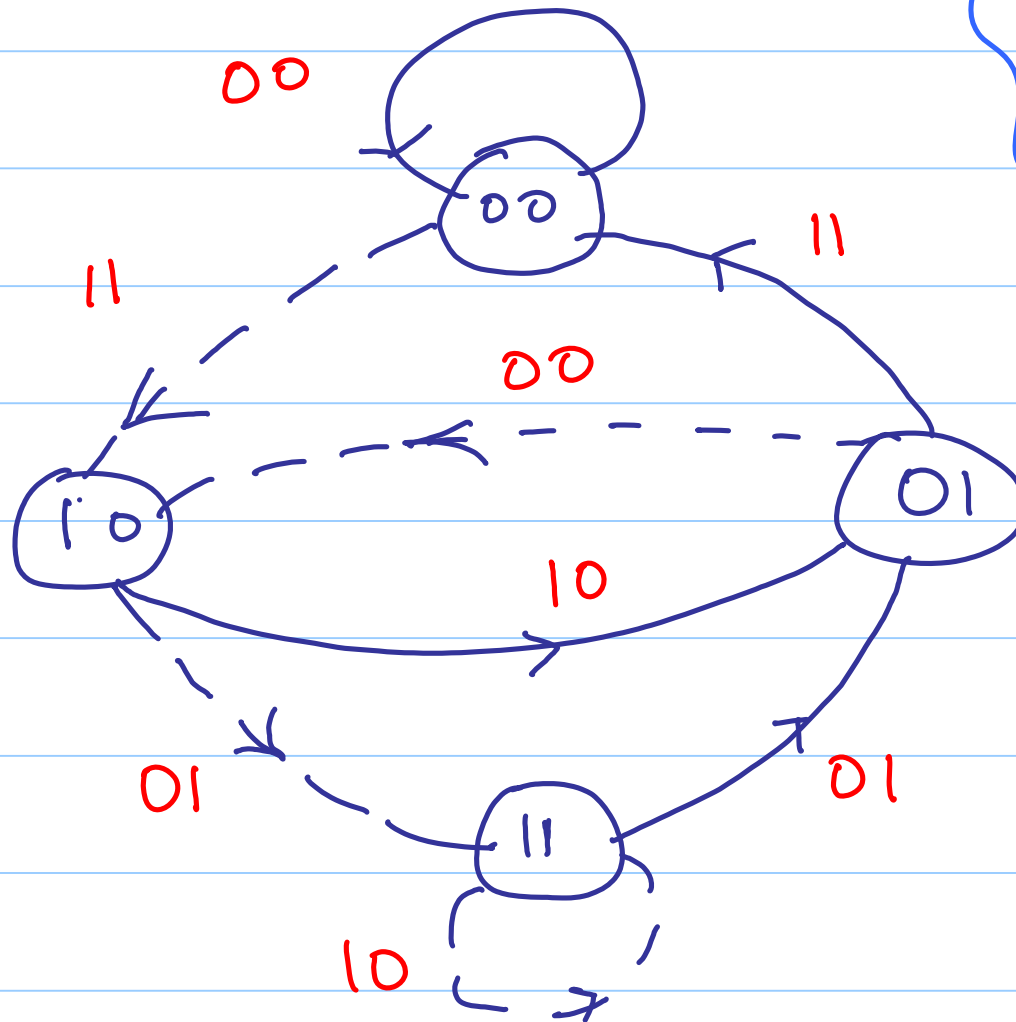
domain, we get:

$$\begin{bmatrix} \overset{(1)}{V}(\mathcal{D}) & \overset{(2)}{V}(\mathcal{D}) \end{bmatrix}$$

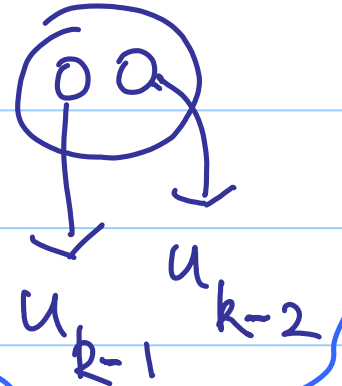
$$= U(\mathcal{D}) \begin{bmatrix} 1 + \mathcal{D} + \mathcal{D}^2 & 1 + \mathcal{D}^2 \end{bmatrix}$$

$$\begin{matrix} G(\mathcal{D}) \\ (1 \times 2) \end{matrix}$$

# Finite - State Machine Description



past 2 symbols

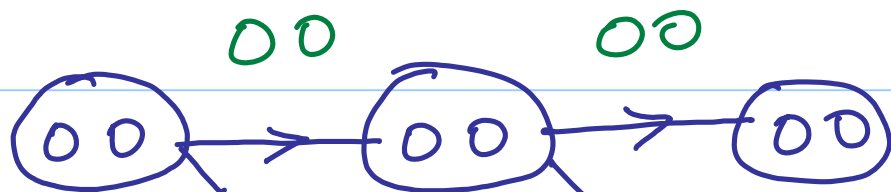


input = 0

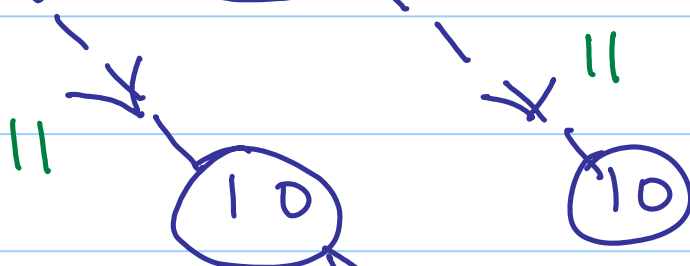
input = 1

In	State	$V^{(1)}$ OUTPUT	$V^{(2)}$
0	00	0	0
1	00	1	1
0	10	1	0
1	10	0	1
0	01	1	1
1	01	0	0
0	11	0	1
1	11	1	0

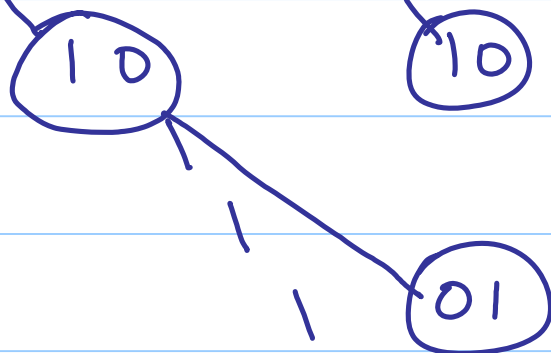
00



10

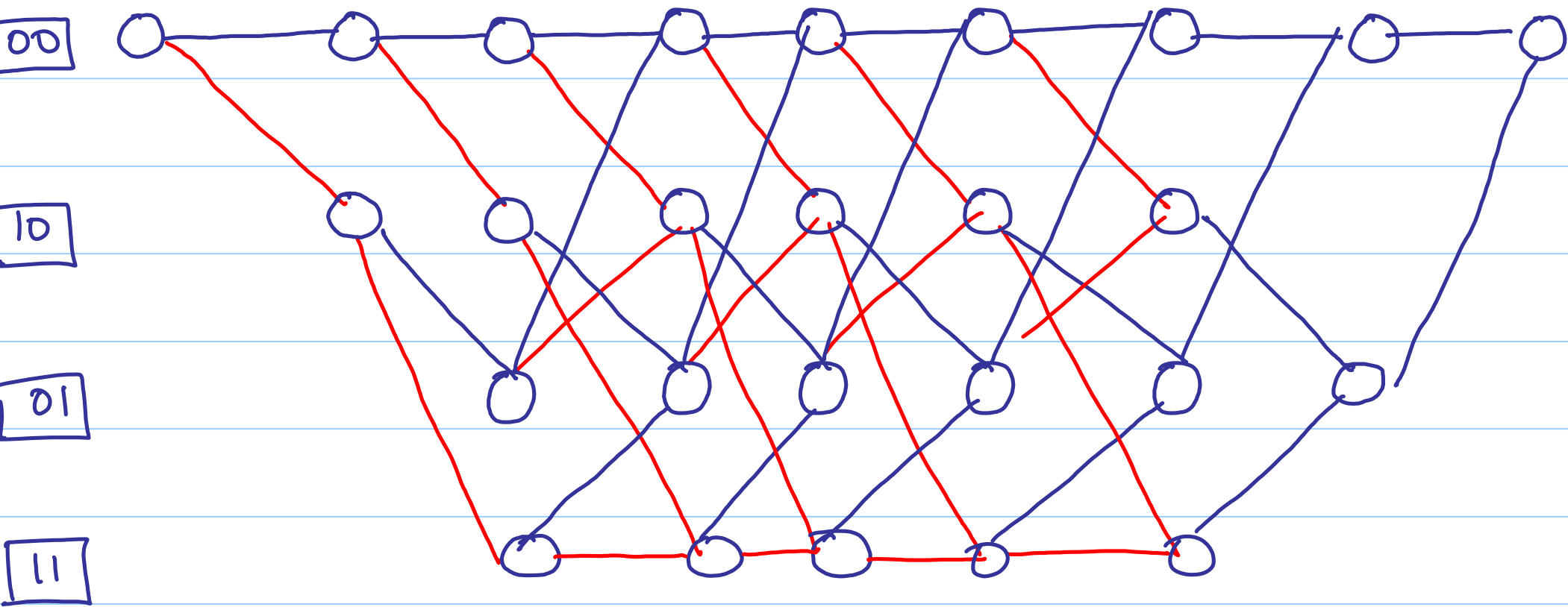


01



11





— 0

— 1

Input

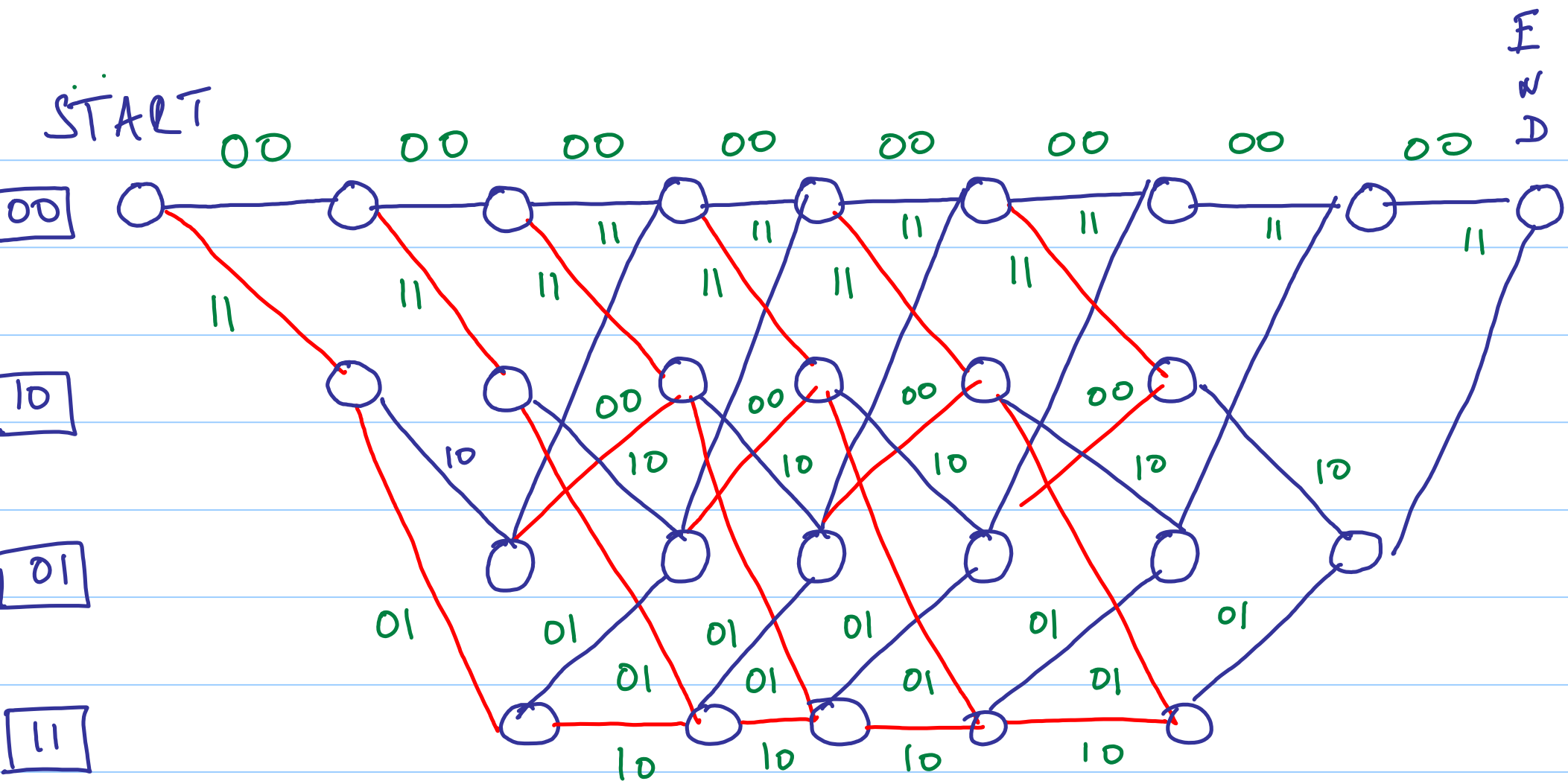


# Lec 17: The Viterbi Decoder

## Recap :

- x Completed our discussion on formal power series and polynomials leading to the polynomial generator matrix of the convn. code
- x state diagram of the encoder viewed as a FSM
- x trellis diagram

START

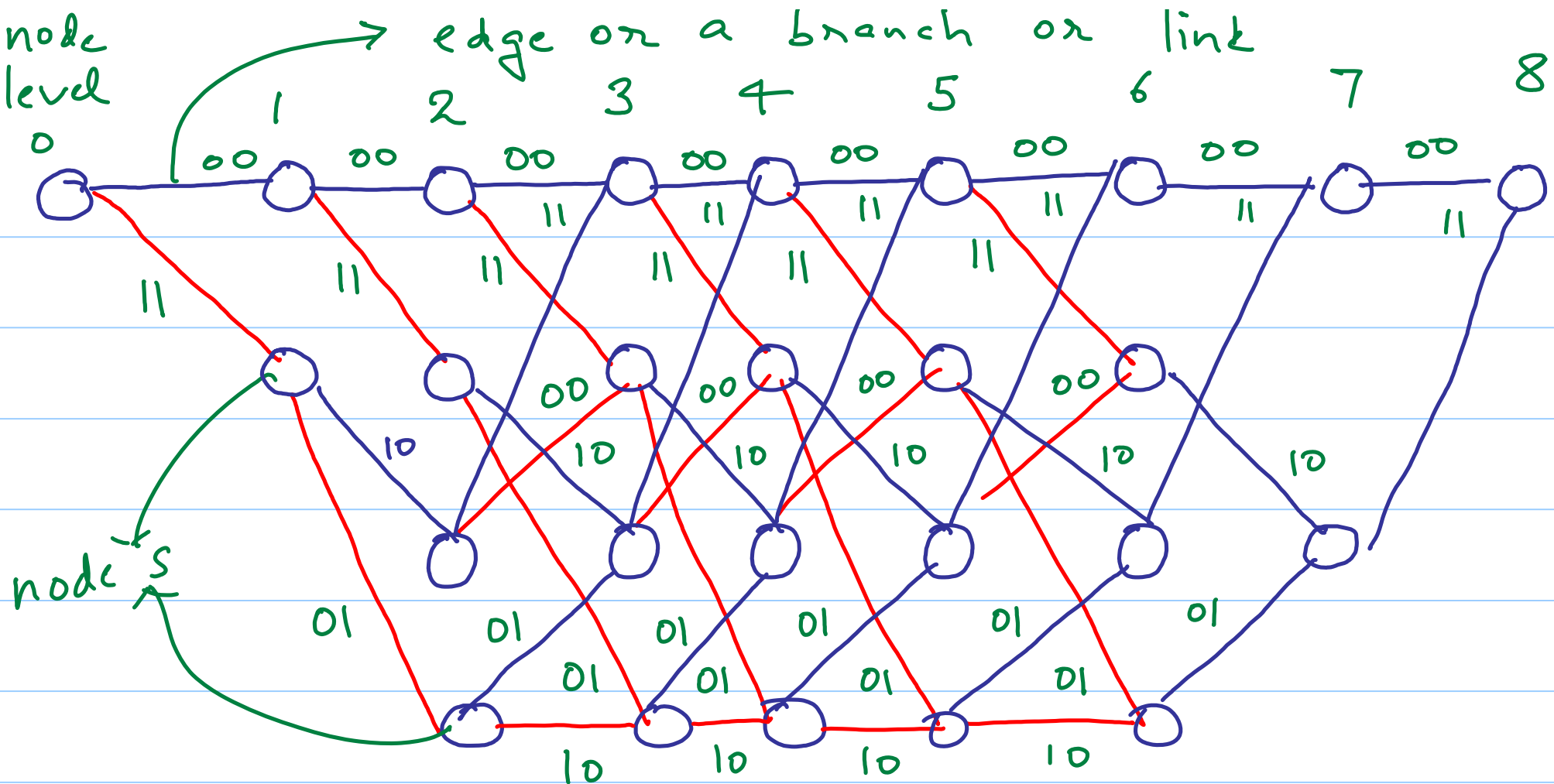


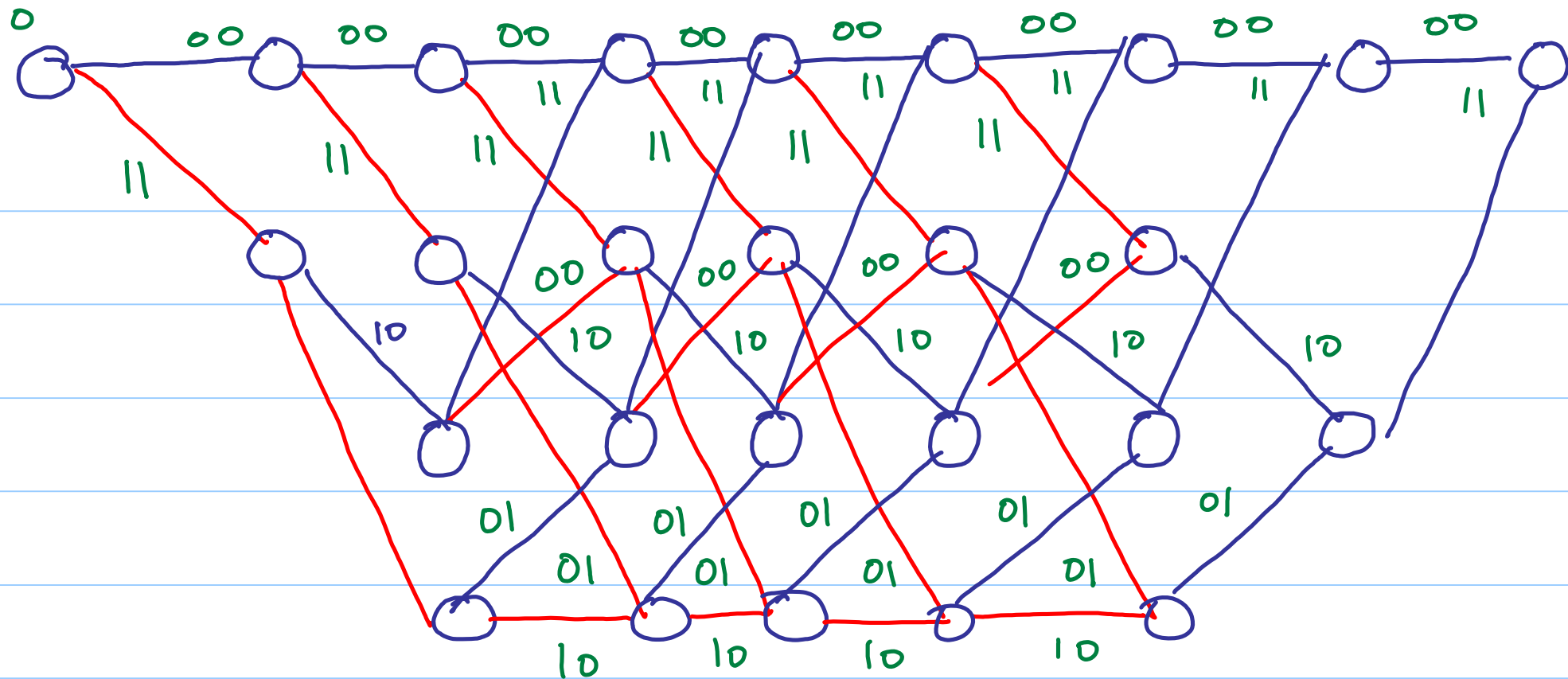
— 0

— 1

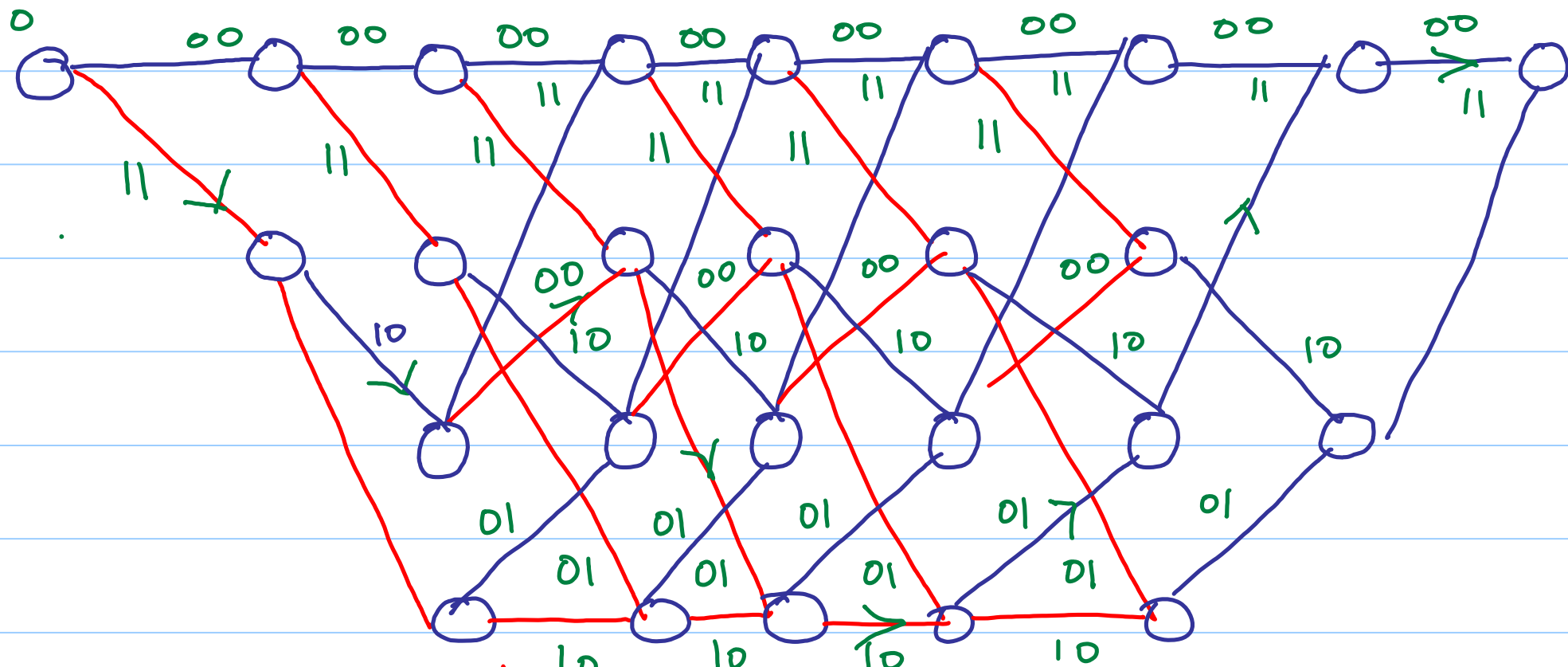
Input





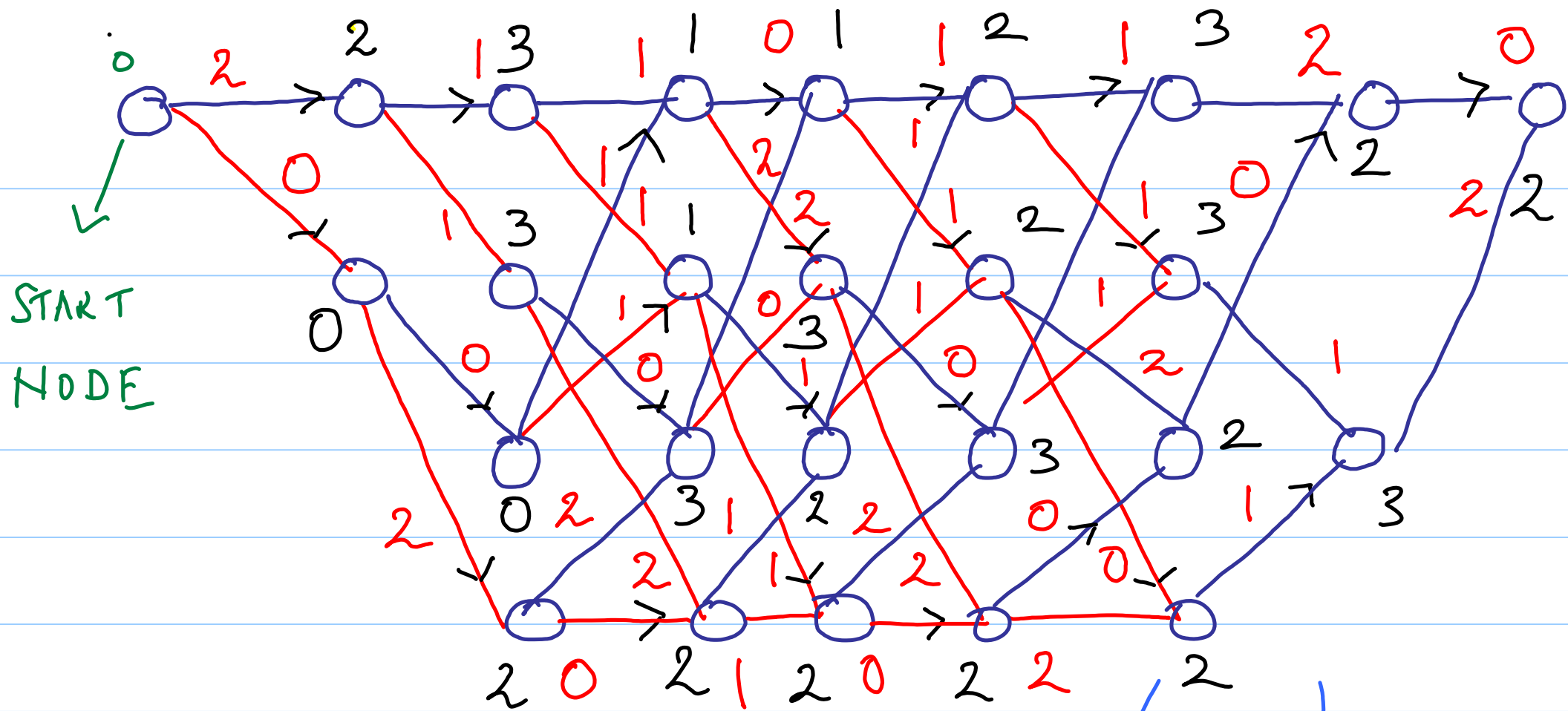


$\{u_k\}$  1 0 1 1 1 0 0 0

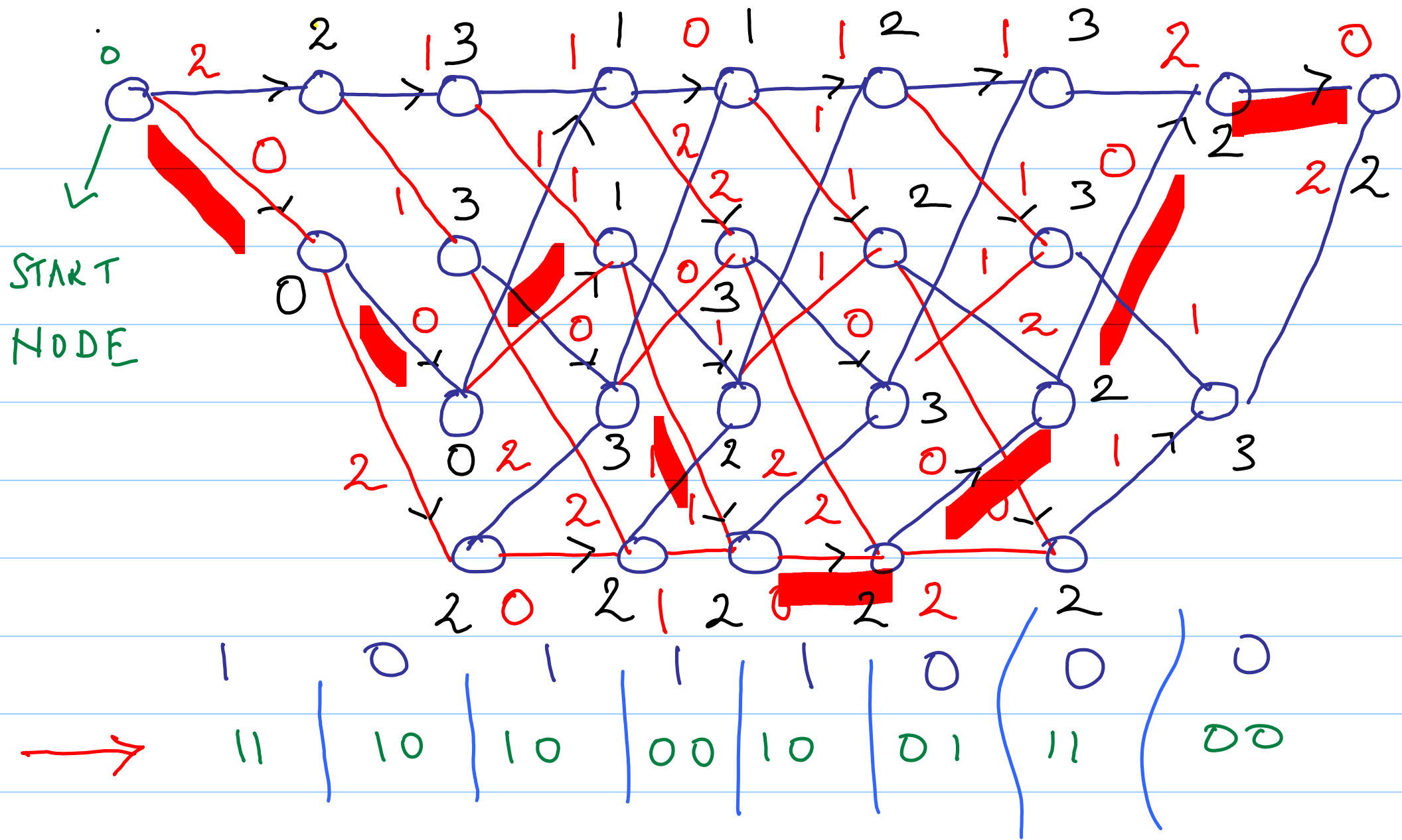


$\begin{Bmatrix} v_k^{(1)} \\ v_k^{(2)} \end{Bmatrix}$   
 $\begin{Bmatrix} \pi_k^{(1)} \\ \pi_k^{(2)} \end{Bmatrix}$

11	10	00	01	10	01	11	00
11	10	10	00	10	01	11	00



→ 11 | 10 | 10 | 00 | 10 | 01 | 11 | 00



If  $\left\{ v_k^{(1)} v_k^{(2)} \right\}_{k=0}^{M-1}, \quad M \leq N$

is the output of the convolutional code encoder, then the associated

path segment metric is given by:

$$d \left( \left\{ v_k^{(1)} v_k^{(2)} \right\}_{k=0}^{M-1}, \left\{ r_k^{(1)} r_k^{(2)} \right\}_{k=0}^{M-1} \right)$$



Similarly, an edge (or branch) in the trellis associated to 6 de

symbols

$$\begin{pmatrix} v_k^{(1)} & v_k^{(2)} \end{pmatrix} = d_{H1} \left( \begin{pmatrix} v_k^{(1)} & v_k^{(2)} \end{pmatrix}, \begin{pmatrix} x_k^{(1)} & x_k^{(2)} \end{pmatrix} \right).$$

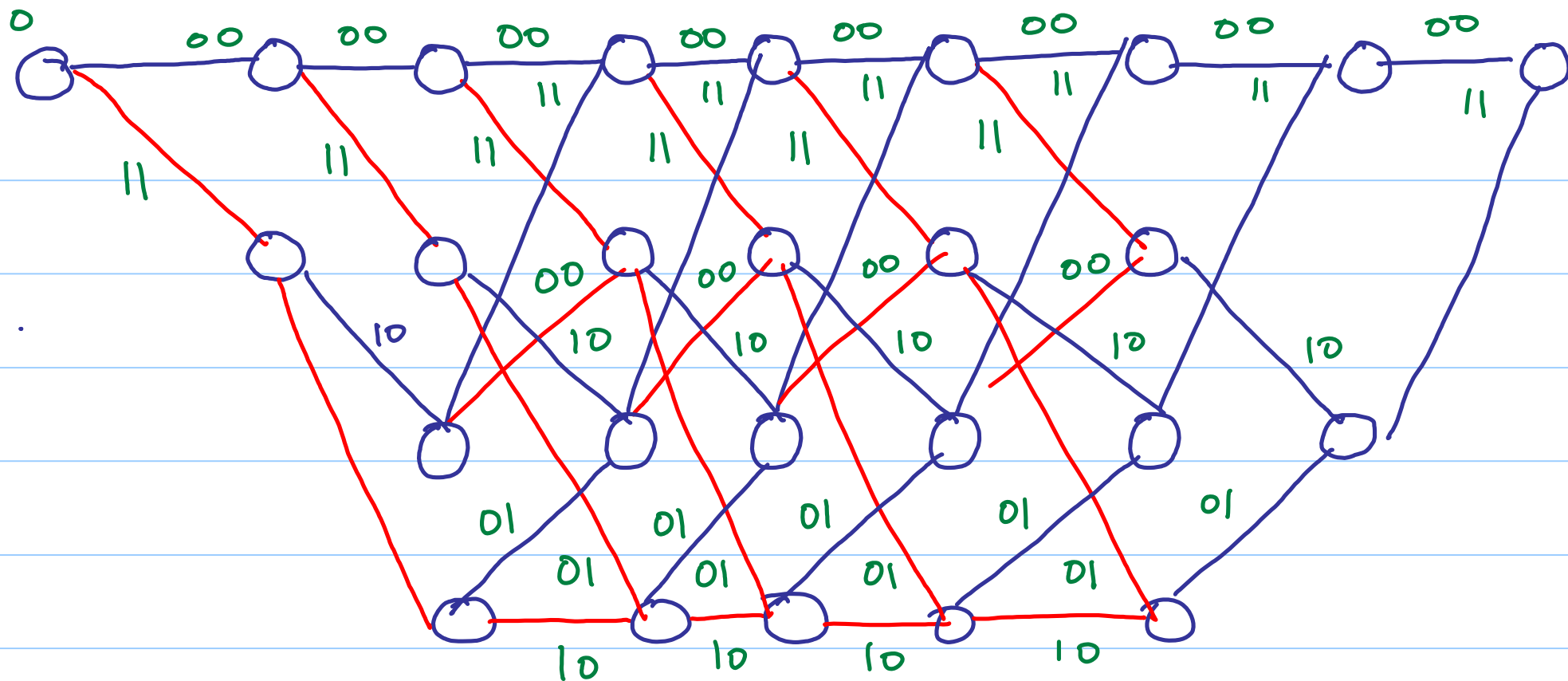
— in the Viterbi decoding process, with each node at each node level, we associate a survivor, where by

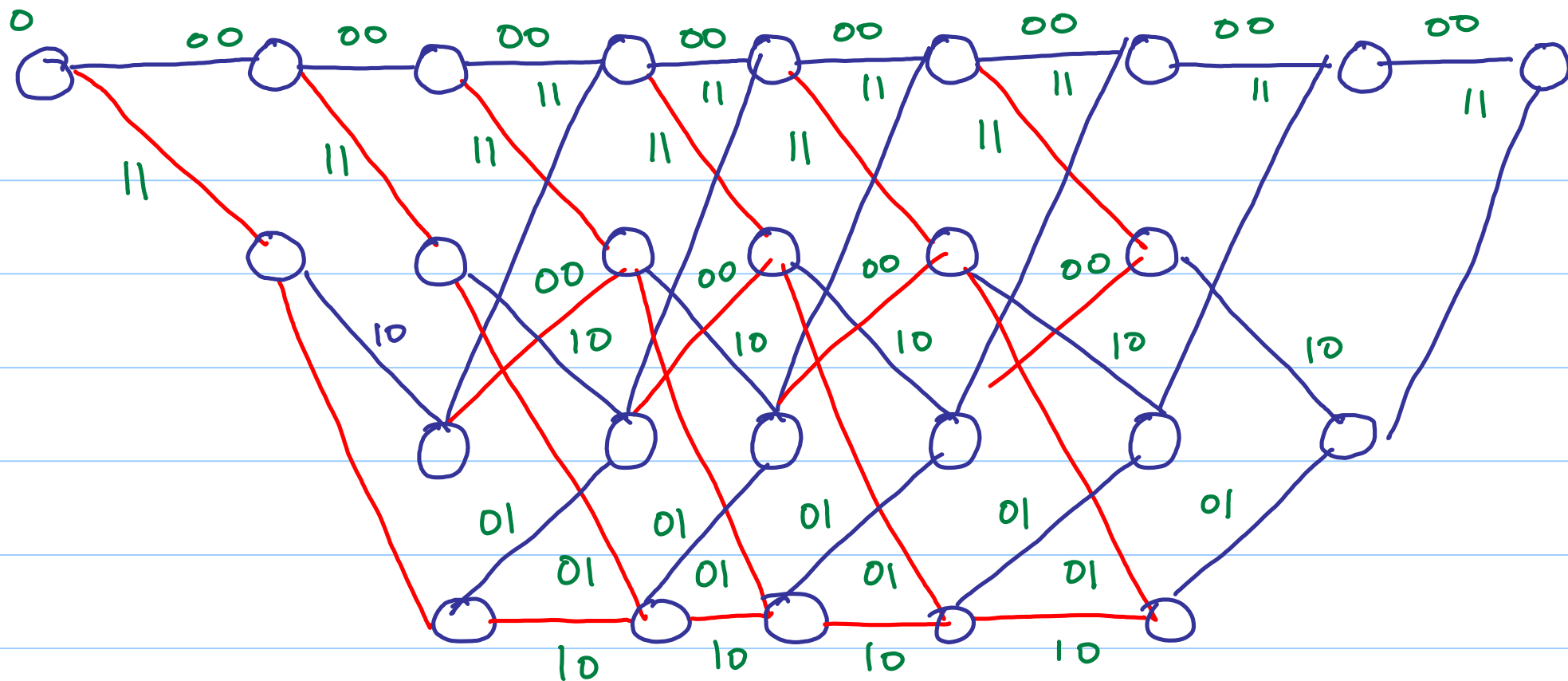


survivor at a node,

we mean, the path from the start node  
to that node having least path

metric





# Lec 18 Catastrophic Error Propagation

## 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  $1/n$ , and has memory  $\nu$

(# of shift registers on the input line), then the loss in rate

per message sequence of length  
N is given by:

$$\frac{1}{n} - \underbrace{\frac{1}{n} \left[ \frac{N-v}{N} \right]}_{\text{actual rate}}$$

loss in rate

Eg (of catastrophic error propagation)  
(LEP)

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

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

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

$$\Leftrightarrow \left\{ u_k \right\}_{k=0}^{\infty} = 1 \ 1 \ 1 \ 1 \ 1 \ \dots$$

Then

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

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

$$\boxed{\begin{bmatrix} V^{(1)}(D) & V^{(2)}(D) \end{bmatrix} = \begin{bmatrix} 1 & 1+D \end{bmatrix}}$$

$$\begin{aligned} \{V_k^{(1)}\} &\Leftrightarrow \textcircled{1} 0 0 0 \dots 0 \dots \dots \\ \{V_k^{(2)}\} &\Leftrightarrow \textcircled{1} 1 0 0 0 \dots 0 \dots \dots \end{aligned}$$

channel coupled



ASIDE

$$\frac{1+D^2}{1+D} = D+1$$

$$1+D$$

$$D+1$$

$$D+1$$

$$\overline{D^2+1}$$

$$D^2+D$$

$$D+1$$

$$D+1$$

...

An encoder  $m \times n$   $G(D)$  for a convolutional code is said to have catastrophic

error propagation if there is some input  $\mathbb{D}$  with  $\infty$  Hamming weight that generates an output

$$\left[ v^{(1)}(\mathbb{D}) \quad v^{(2)}(\mathbb{D}) \quad \dots \quad v^{(n)}(\mathbb{D}) \right]$$

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  $\infty$  # of message  
symbol errors.

---

$$G_1(D) = \begin{bmatrix} 1 + D + D^2 & 1 + D^2 \end{bmatrix} \quad \begin{array}{l} \text{no} \\ \text{CEP} \end{array}$$

$$G_2(D) = \begin{bmatrix} 1 + D & 1 + D^2 \end{bmatrix} \quad \begin{array}{l} \text{has} \\ \text{CEP.} \end{array}$$

Thm A necessary and sufficient condition on the generator  $m \times q$  of a rate  $\frac{1}{n}$  convolutional code to avoid

CEP is that:

$$\gcd(g_1(D), g_2(D), \dots, g_n(D)) = D^l, l \geq 0$$

$$\text{where } h(D) = [g_1(D) \ g_2(D) \ \dots \ g_n(D)]$$

is the PAM of the code.

---

GCD - quick review

---

$$\gcd(27, 63) = ?$$

$$63 \div 27 \Rightarrow$$

$$\begin{array}{r} 2 \\ 27 \overline{) 63} \\ \underline{54} \\ 9 \end{array}$$

Remainder

$$27 \div 9 \Rightarrow$$

$$\begin{array}{r} 3 \\ 9 \overline{) 27} \\ \underline{27} \\ 0 \end{array}$$

Remainder	63	27	Quotient
63	1	0	
27	0	1	2
9	1	-2	3
0	-3	7	

gcd

$$9 = 63 \cdot 1 + (-2) \cdot 27 \Leftarrow$$

Eg (polynomial case)

Remainder	$D^2 + D + 1$	$D^2 + 1$	Quotient
$D^2 + D + 1$	1	0	
$D^2 + 1$	0	1	1
$D$	1	1	$D$
gcd $\nearrow$ <span style="border: 1px solid black; padding: 2px;">1</span> 0	$D$	$D + 1$	$D$

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

there is no CEP (by the theorem).

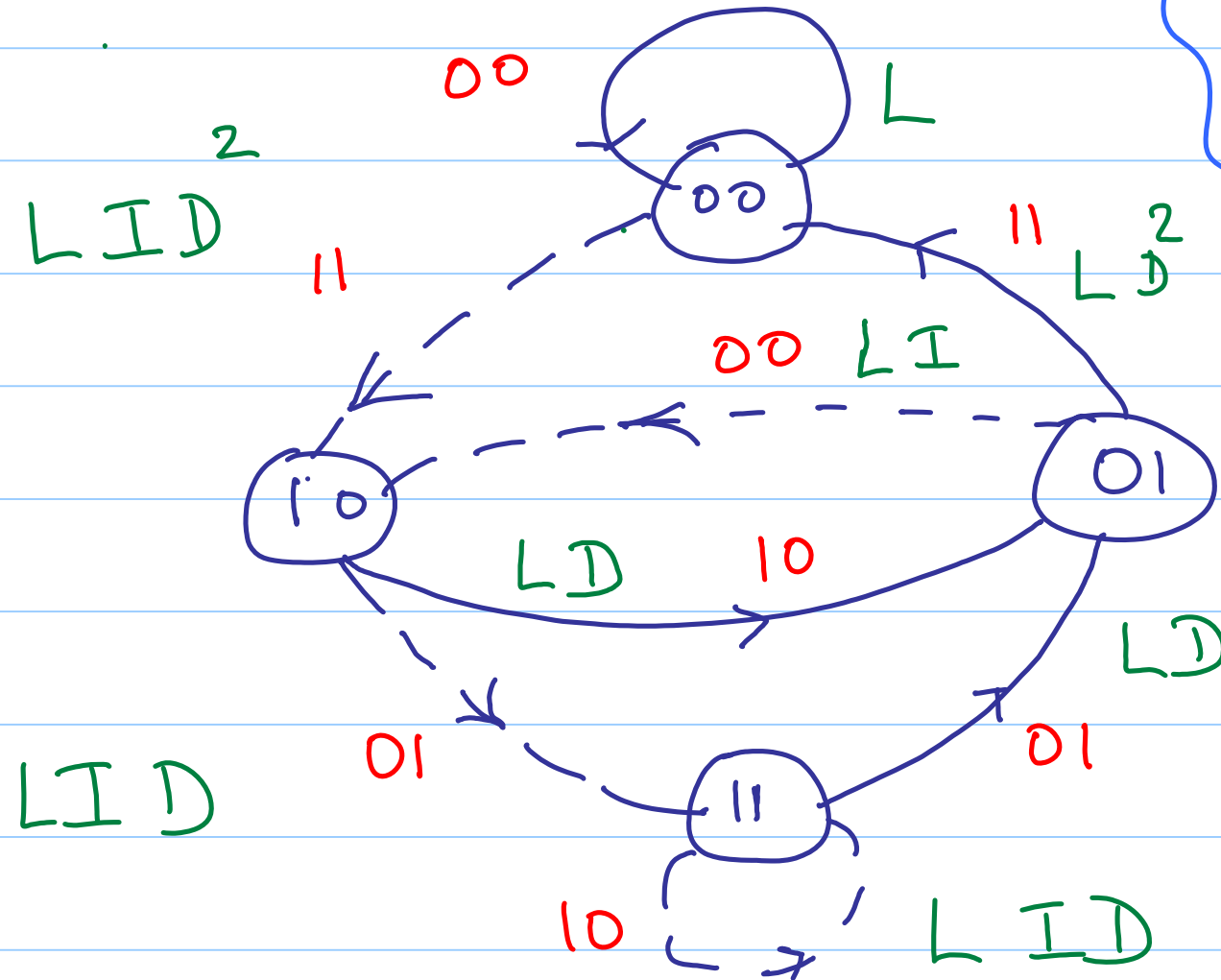
Ex  $G(D) = \begin{bmatrix} 1+D & 1+D^2 \end{bmatrix}$

Remainder	$D^2 + 1$	$D + 1$	Quotient
$D^2 + 1$	1	0	
$D + 1$	0	1	$D$
<u><math>D + 1</math></u> 0	1	$D$	1

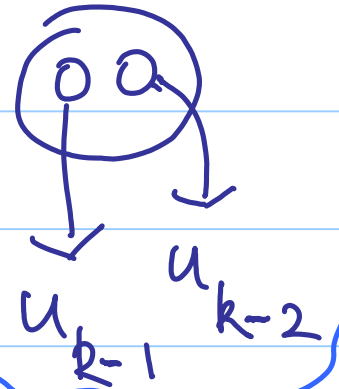
Note:  $\gcd \neq D^l, l \geq 0$  and hence,  
 this code has CIP.



# Finite - State Machine Description



past 2 symbols



input = 0

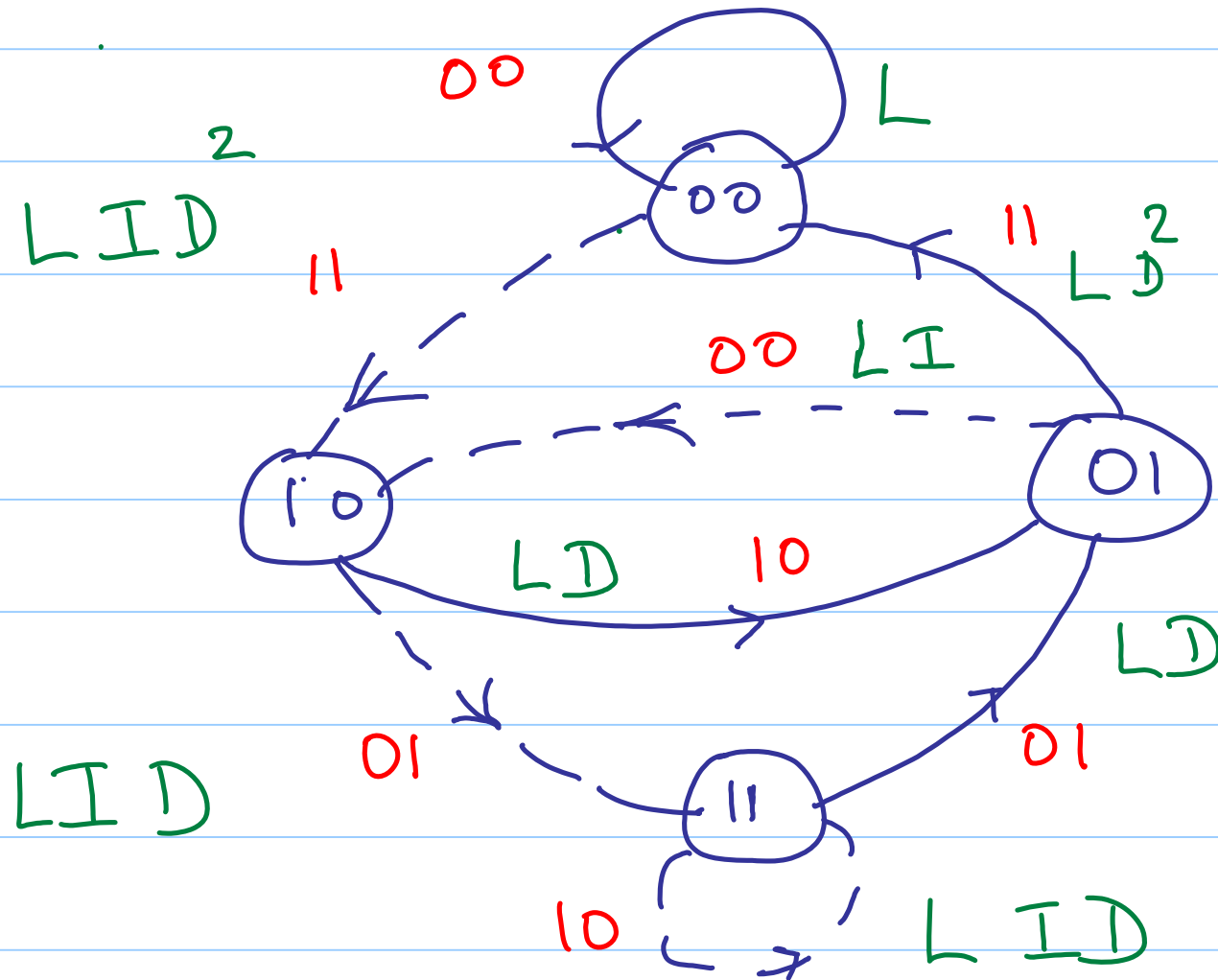
input = 1

# Lec 19 Path Enumeration

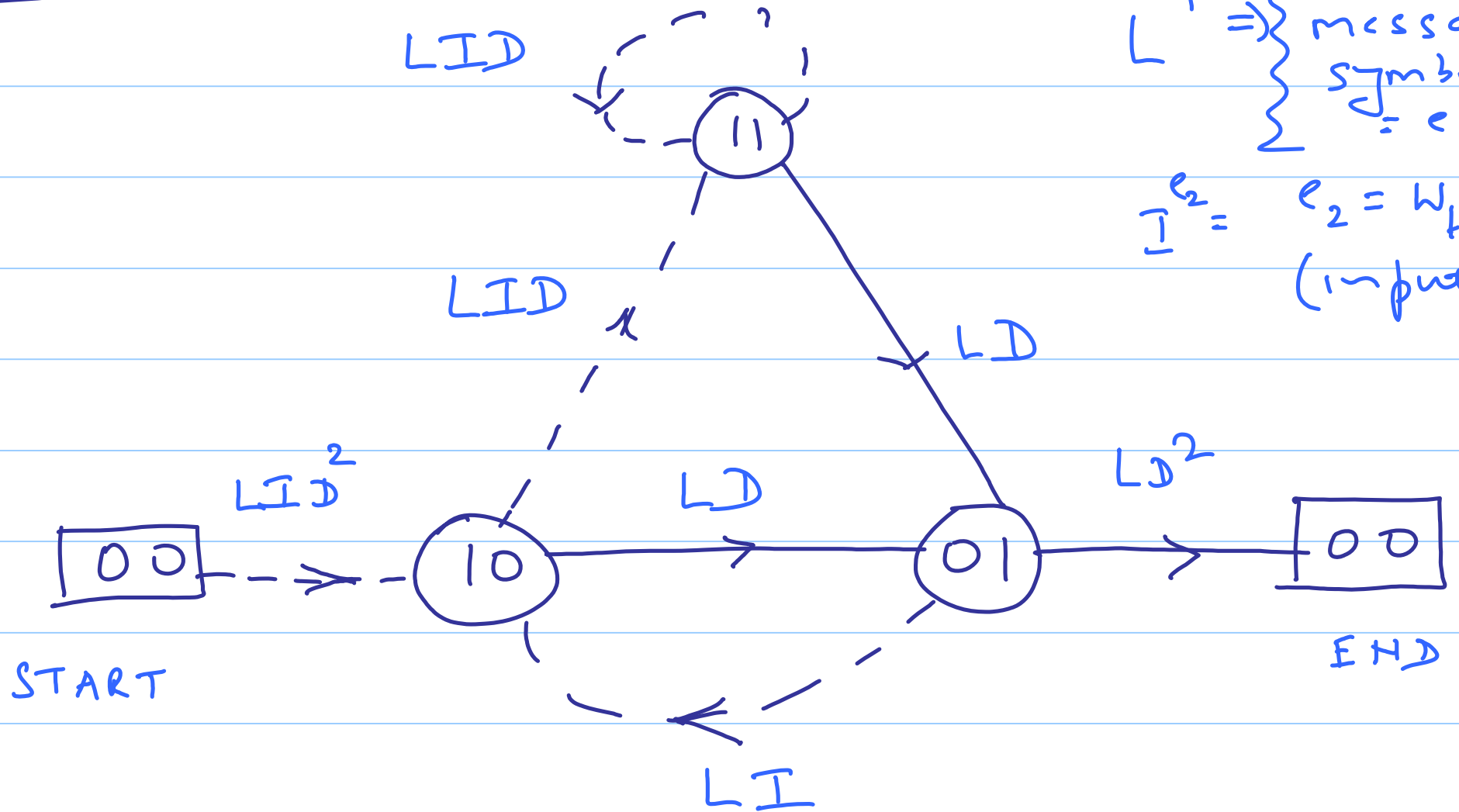
## Recap:

- { CEF in rate  $\frac{1}{n}$   
convolutional codes
  - gcd computation  
(polynomials)
- path enumeration

# Finite - State Machine Description



# Modified State Diagram



$$D^{e_3} \quad e_3 = W_H \text{ (output)}$$

Our interest is in computing for  
the END state the power series

$$A_{\text{END}}(L, D, I) = \sum_{ijk} a_{ijk} L^i I^j D^k$$

# of paths in the modified state  
diagram of path length  $i$ , whose  
associated input (message) sequence  
has Hamming weight  $j$  and  
whose associated output sequence

has Hamming weight  $k$ .

---

Power series such as  $A_{END}(L, I, D)$   
are also called generating functions

---

$$\begin{bmatrix} A_{10} \\ A_{11} \\ A_{01} \end{bmatrix} = \begin{bmatrix} 0 & 0 & LI \\ LID & LID & 0 \\ LD & LD & 0 \end{bmatrix} \begin{bmatrix} A_{10} \\ A_{11} \\ A_{01} \end{bmatrix} + \begin{bmatrix} LID^2 \\ 0 \\ 0 \end{bmatrix}$$

$A_{10} \equiv$  abbreviation for

$$A_{10}(L, I, D)$$

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

and  $A_{EHD}(L, I, D) = LD^2 A_{01}(L, I, D)$

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

$$\underline{x} = A \underline{x} + \underline{b}$$

$$\Rightarrow [I - A] \underline{x} = \underline{b}$$



Solving using Ramer's rule yields :

$$A_{01}(L, I, D) = \frac{L^2 I D^3}{1 - LID - L^2 ID}$$

$$\therefore A_{END} = LD^2 A_{01} = \frac{L^3 I D^5}{1 - LID(1+L)}$$

Note: Any expression of the form  
 $\frac{1}{1 - g(L, I, D)}$  can be expanded as:

$$\frac{1}{1 - g(L, I, D)}$$

{ with no  
constant  
term

$$\frac{1}{1 - g(L, I, D)} = 1 + g(L, I, D) + g^2(L, I, D) + g^3(L, I, D) + \dots$$

$$A_{END} = L D^2 A_0 = \frac{L^3 I D^5}{1 - L I D (1+L)}$$

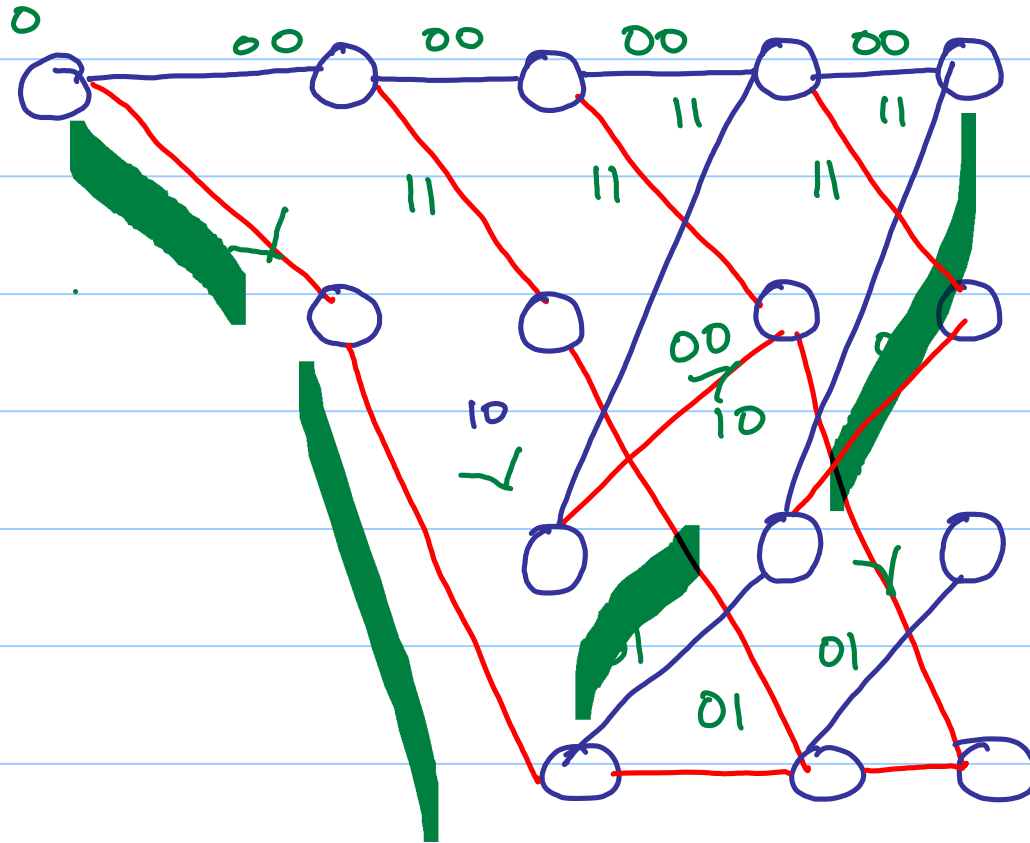
$$= L^3 I D^5 \left\{ 1 + L I D (1+L) + L^2 I^2 D^2 (1+L)^2 + \dots \right\}$$

How many paths have length 4?

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

$$(L^3 I D^5) L I D = L^4 I^2 D^6$$

$\Rightarrow$



$L^4 I^2 D^6$

# General rate $(k/n)$ convolutional code

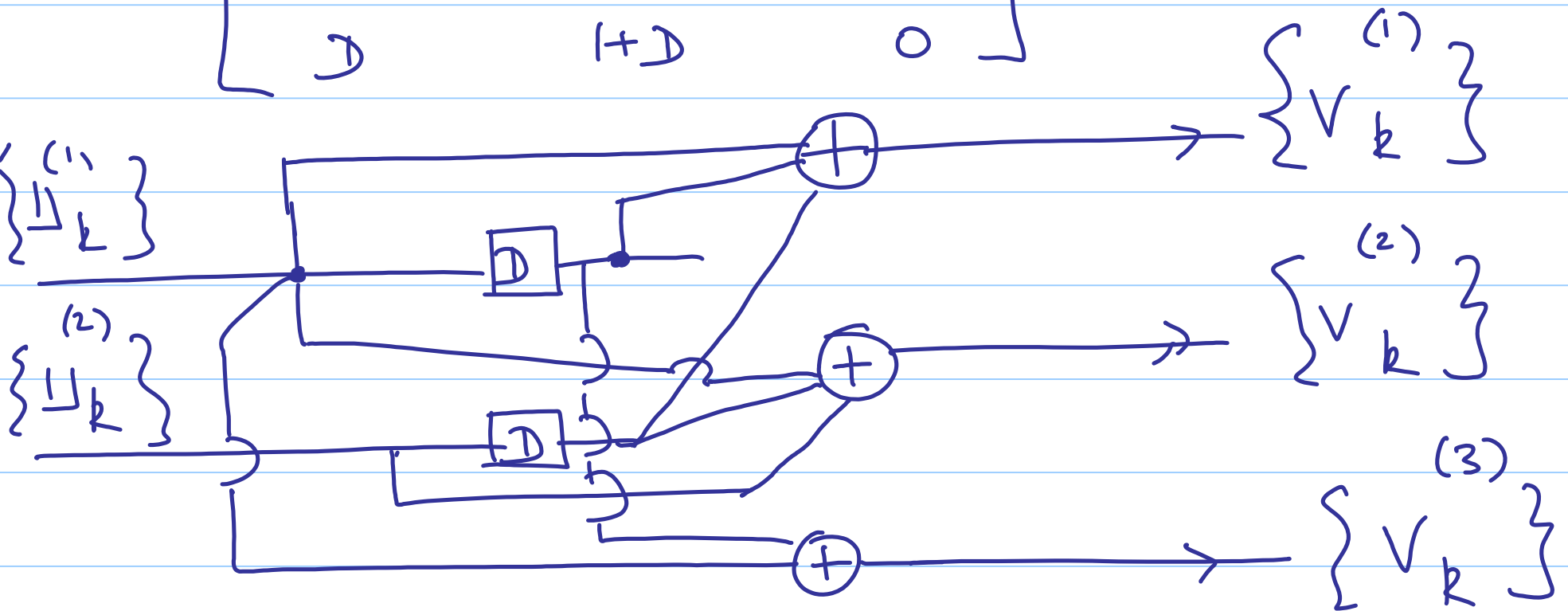
Ex  $G(D) = \begin{bmatrix} 1+D & 1 & 1+D \\ D & 1+D & 0 \end{bmatrix}$

PGM  
(2 x 3)

Encoding:

$$\begin{bmatrix} U^{(1)}(D) & U^{(2)}(D) \end{bmatrix} G(D) = \begin{bmatrix} V^{(1)}(D) & V^{(2)}(D) & V^{(3)}(D) \end{bmatrix}.$$

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



Memory of the convolutional encoder

$$G(D) = \begin{bmatrix} g_{11}(D) & \dots & g_{1n}(D) \\ \vdots & \ddots & \vdots \\ g_{k1}(D) & \dots & g_{kn}(D) \end{bmatrix}$$

$$D = \sum_{i=1}^k \max_{1 \leq j \leq n} \{v_{ij}\}$$

$$v_{ij} = \deg(g_{ij}(D))$$

In the example:

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

$$[v_i] = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -\infty \end{bmatrix}$$

$$\begin{aligned} \therefore v &= \max\{1, 1, 1\} + \max\{1, 1, -\infty\} \\ &= 1 + 1 = 2. \end{aligned}$$

---



General I/O relation:

$$V_t^{(j)} = \sum_{i=1}^K \sum_{m=0}^{v_{ij}} u_{t-m}^{(i)} g_m^{(ij)}$$

where  $g_{ij}^{(D)} \triangleq \sum_{m=0}^{v_{ij}} g_m^{(ij)} D^m$

$$V^{(j)}(D) \triangleq \sum_{t=0}^{\infty} V_t^{(j)} D^t$$

$$= \sum_{t=0}^{\infty} \sum_{i=1}^k \sum_{m=0}^{\infty} v_{ij}^{(i)} u_{t-m}^{(i)} g_m^{(ij)}$$

$$= \sum_{i=1}^k \left[ \sum_{m=0}^{\infty} g_m^{(ij)} \right] \left[ \sum_{t=0}^{\infty} u_{t-m}^{(i)} \right]$$

$$= \sum_{i=1}^k g_{ij}^{(i)} u^{(i)}$$

$$\Rightarrow \begin{bmatrix} v^{(1)} \\ \vdots \\ v^{(k)} \end{bmatrix} = \begin{bmatrix} u^{(1)} \\ \vdots \\ u^{(k)} \end{bmatrix} u^{(i)}$$

## Lec 20 { Viterbi decoder over the AWGN Channel

### Recap

- \* path information enumeration  
using generating function  
techniques
- \* { General rate  $k/n$  convolutional  
code - PAM derivation

Thm (CEP) A n.a.s.c on the PAM

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

$$\gcd(\Delta_1(D), \dots, \Delta_{\binom{n}{k}}(D)) = D^l, \quad l \geq 0$$

where

$\left\{ \Delta_p(D) \right\}_{p=1}^{\binom{n}{k}}$  is the collection of

determinants of the  $\binom{n}{k}$  ( $k \times k$ )

submatrices of  $A(D)$

---

Eg

$$G(D) = \begin{bmatrix} 1+D & 1 & 1+D \\ D & 1+D & 0 \end{bmatrix}$$

$$\left. \begin{array}{l} k=2 \\ n=3 \end{array} \right\}$$

$$(2 \times 3)$$

$$\therefore \binom{n}{k} = \binom{3}{2} = 3$$

$$\Delta_1(D) = \begin{vmatrix} 1+D & 1 \\ D & 1+D \end{vmatrix} = 1+D^2 + D$$

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

$$\Delta_3(D) = \begin{vmatrix} 1 & 1+D \\ 1+D & 0 \end{vmatrix} = (1+D)^2$$

Can be verified that

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

---

Viterbi decoding over the AWGN Channel

Channel Model:

$$s \in \{\pm \sqrt{E}\}$$



$D$

$$y = s + n$$

$$n \sim N(0, \frac{N_0}{2})$$

$\left(\frac{E}{N_0}\right)$  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:

— choose that codeword that



{ maximizes

$$p(\underline{y} | \underline{c})$$

$c_i$   $\rightarrow$  code symbol

where  $s_i = (-1)^{s_i} \sqrt{E}$



{ Symbol transmitted  
over the AWGN channel

In the convolutional code:

$$\underline{c} = \left( \overset{(1)}{V_t} \overset{(2)}{V_t} \dots \overset{(n)}{V_t} \right), t=0, 1, \dots, N-1$$

$$p(\underline{y} | \underline{c}) = \prod_{t=0}^{N-1} \prod_{j=1}^N \underbrace{p(y_t^{(j)} | v_t^{(j)})}$$

where

$$p(y_t^{(j)} | v_t^{(j)}) = \frac{1}{\sqrt{2\pi(N_0/2)}} \exp\left(-\frac{1}{2 \frac{N_0}{2}} (y_t^{(j)} - s_t^{(j)})^2\right)$$

$$s_t^{(j)} = \sqrt{E} (-1)^{v_t^{(j)}}$$

clear that MLDD calls for

minimizing

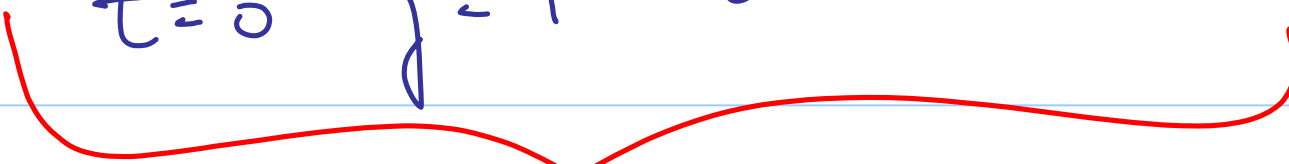
$$\sum_{t=0}^{N-1} \sum_{j=1}^n \left[ y_t^{(j)} - s_t^{(j)} \right]^2$$

$$\left[ y_t^{(j)} \right]^2 - 2 y_t^{(j)} s_t^{(j)} + \left[ s_t^{(j)} \right]^2$$

$\Downarrow$   
 $E$

and thus MLD is aimed at

maximizing the inner product:

$$\sqrt{E} \quad \sum_{t=0}^{N-1} \quad \sum_{j=1}^n y_t^{(j)} \quad (-1)^{a_t^{(j)}}$$


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  
for the fact that our branch  
metrics are now real numbers  
(+ve or -ve) as opposed to  
+ve 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 generating function  $A_{\text{END}}(L, I, D)$  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:

$$P_{be} \leq \frac{1}{k} \frac{\partial}{\partial I} A_{END}(L, I, D)$$

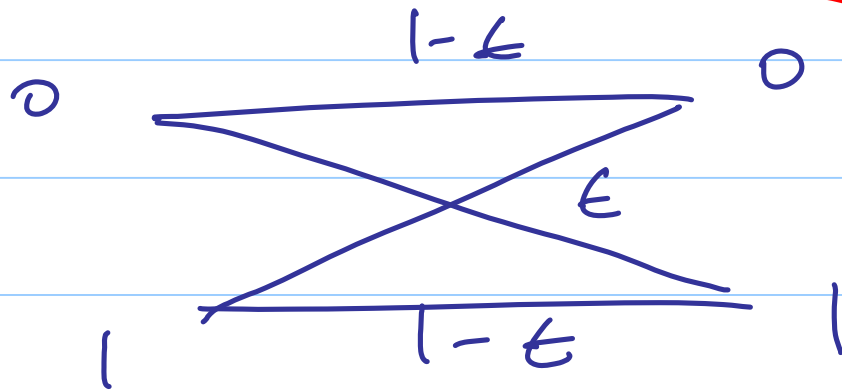
BIT ERROR

PROB.

$$L = 1$$

$$I = 1$$

$$D = 2 \sqrt{\epsilon(1-\epsilon)}$$



BSC



Case (ii) AWGN channel:

$$P_{be} \leq Q \left( \sqrt{\frac{2 E d_{free}}{N_0}} \right) \exp \left( - \frac{E d_{free}}{N_0} \right)$$

$$\star \quad \frac{1}{k} \frac{\partial}{\partial I} A_{END}(L, I, D) \quad \left| \begin{array}{l} L=1 \\ I=1 \end{array} \right.$$

$$D = \exp \left( - E / N_0 \right)$$

# lec 21 { The Generalized Distributive Law

## Recap

- \* { C E P in general, rate  $k/n$   
    { convolutional code.
- \* { Viterbi decoding over the  
    { AWGN channel
- \* { Expressions to compute  
    { bit error probability of the  
    { Viterbi decoder.

Clarification :  $d_{\text{free}} = ?$

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

Can be computed from the power series:

$A_{\text{END}}(L, I, D)$  as follows:

Set  $L = I = 1$  to get  $A_{END}(1, 1, D)$

In this power series,  $d_{free}$  is the smallest exponent of  $D$

$$\underline{\text{Eg}} \quad A_{END}(L, I, D) = \frac{L^3 I D^5}{1 - L I D(1+L)}$$

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

$$= D^5 (1 + 2D + 4D^2 + 8D^3 + \dots)$$

$$\therefore \boxed{d_{free} = 5}$$

///

G D L

Eg      $a(b+c) = ab+ac$

{ 1 addition  
1 multpln.

$\Downarrow$

{ 2 multiplications  
1 addition

Eg

$$\alpha(x, w) = \sum_{y, z} f(x, y, w) g(x, z)$$

$$\beta(y) = \sum_{x, w, z} f(x, y, w) g(x, z)$$

{ variables  $w, x, y, z$  take on values  
from a common alphabet  $A$  of  
size  $|A| = 9$

the functions  $f(\cdot)$   $g(\cdot)$  are real valued.

$$\alpha(x, w) = \sum_{y, z} f(x, y, w) g(x, z)$$

# of computations required

$$= q^2 \left\{ q^2 + (q^2 - 1) \right\}$$

$$= 2q^4 - q^2$$

$$\beta(y) = \sum_{x, w, z} f(x, y, w) g(x, z)$$

# of computations

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

$$= 2q^4 - q.$$

Invoking the distributive law:

$$\alpha(x, w) = \sum_{y, z} f(x, y, w) g(x, z)$$



$$= \left( \sum_j f(x, j, w) \right) \left( \sum_z g(x, z) \right)$$

$$\alpha(x, w) = f'(x, w) \cdot g'(x)$$

$$f' \Rightarrow \begin{matrix} \uparrow^2 \\ \downarrow \end{matrix} (L^{-1}) \quad g' \Rightarrow \begin{matrix} \uparrow \\ \downarrow \end{matrix} (L^{-1})$$

$$\alpha \Rightarrow \begin{matrix} \uparrow^2 \\ \downarrow \end{matrix}$$

total # of computations

$$= L^3 - \cancel{L^2} + \cancel{L^2} - L + L^2$$

$$= \boxed{L^3 + L^2 - L}$$

versus

$$\left( 2L^4 - L^2 \right)$$

$$\beta(z) = \sum_{x, w, z} f(xzw) g(xz)$$

$$= \sum_x \underbrace{\left( \sum_w f(xzw) \right)}_{f'(xz)} \underbrace{\left( \sum_z g(xz) \right)}_{g'(x)}$$

total # of computations

$$= q^2(q-1) + q(q-1) + q(q+q-1)$$

$$= q^3 - \cancel{q^2} + \cancel{q^2} - q + 2q^2 - q$$

$$= q^3 + 2q^2 - 2q$$

versus

$$2q^2 - q$$

///

## DETOUR: Semi rings

Defn A semiring  $(R, +, \cdot)$  is a

set  $R$  along with 2 operations  
 $(+, \cdot)$  under which the following

properties hold:

(i) Under addition:  $(R, +)$

- CLOSURE
- ASSOCIATIVE
- IDENTITY EL.
- COMMUTATIVE

(note that the additive inverse is not required)

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

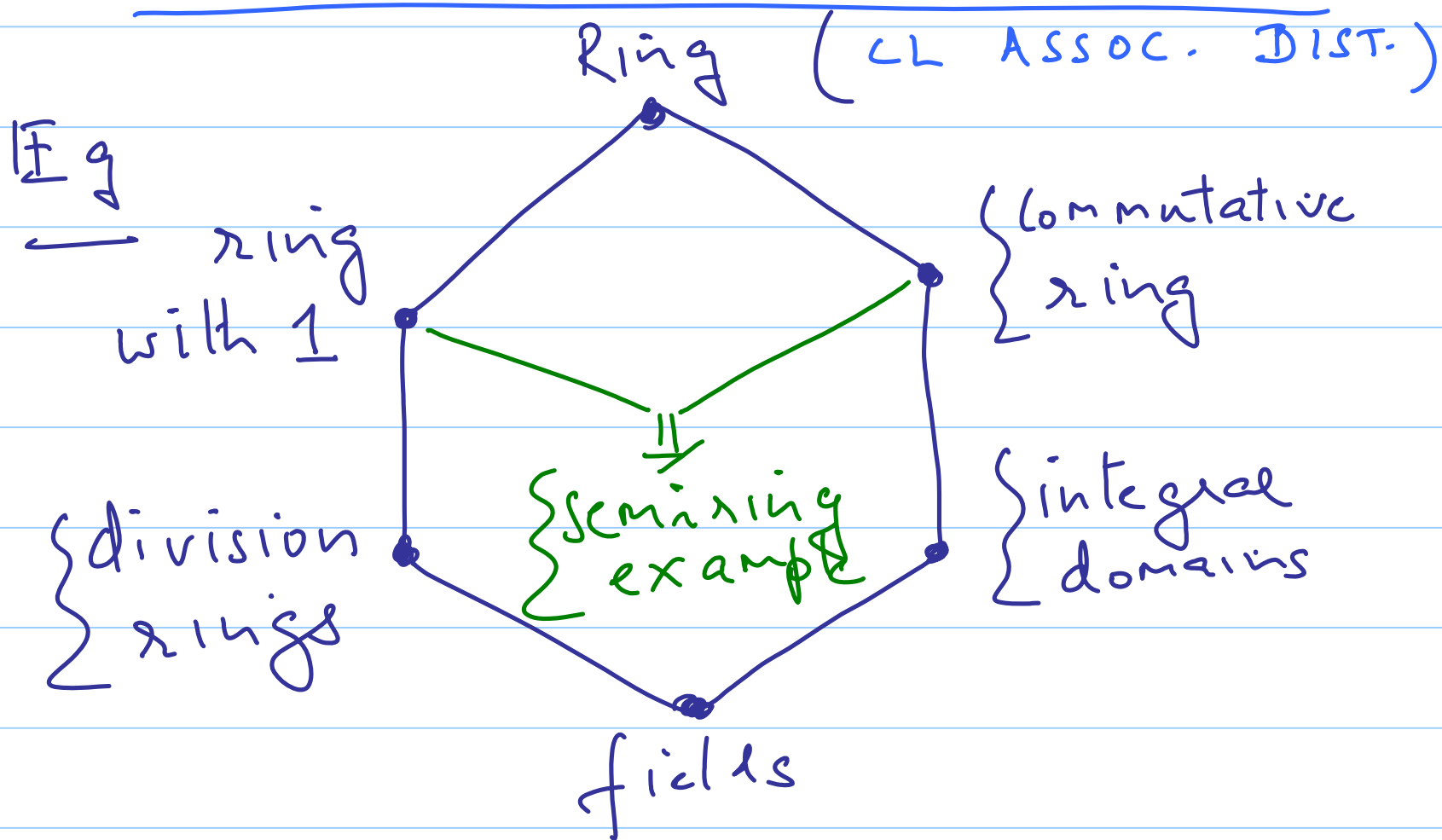
- CLOSURE
- ASSOCIATIVE
- IDENTITY EL
- COMMUTATIVE

(note that multiplicative inverses need not exist)

(iii)  $(R, +, \cdot)$  must satisfy the

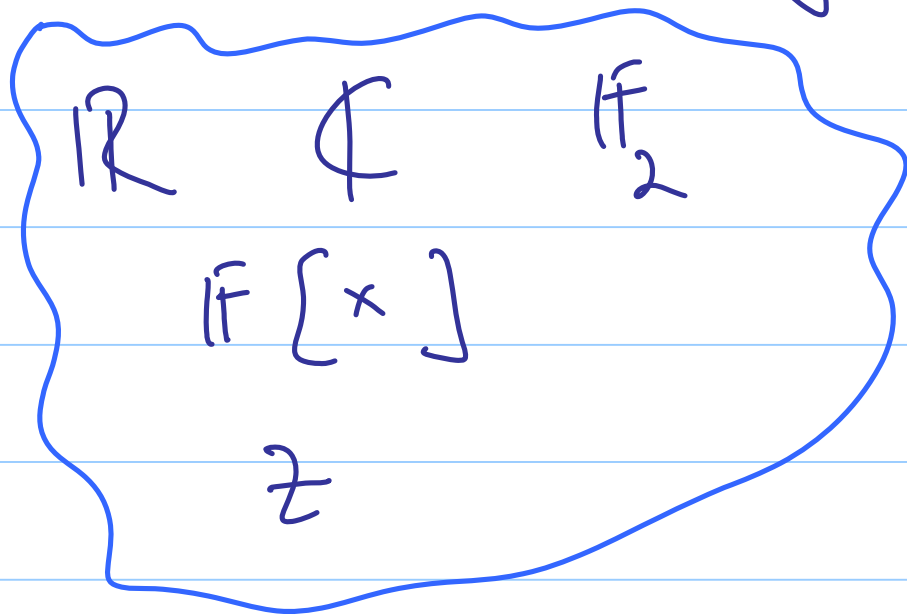
distributive kw:

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

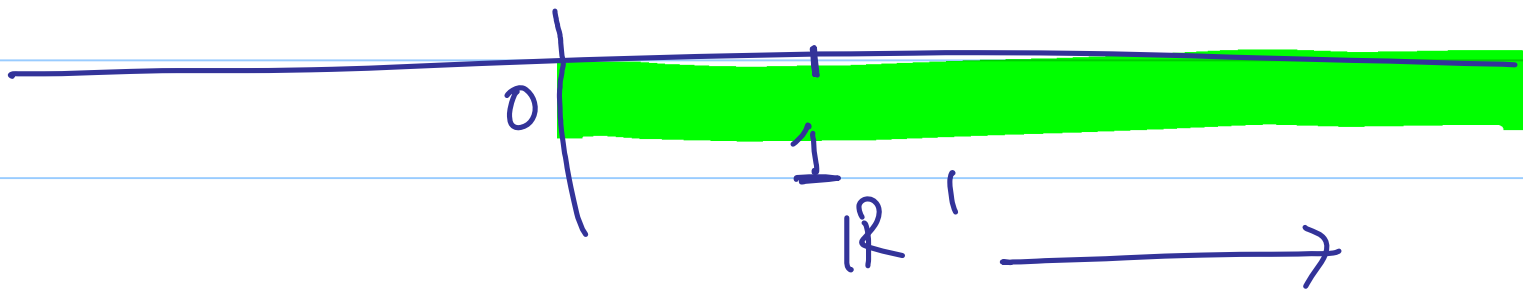


{ every commutative ring with identity  
is a semiring

{ in particular, every field is  
a semiring



$$\underline{\text{Eg}} \quad (\mathbb{R}, +, \cdot) = ([0, \infty), +, \cdot)$$



$$\underline{(\mathbb{R}, +)}$$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

$$\underline{(\mathbb{R}, \cdot)}$$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

— DIST LAW ✓

---



## Lec 22      The MPF Problem

Recap

—  $\lambda_{free}$

— { distributive law }  
  { # of computations }

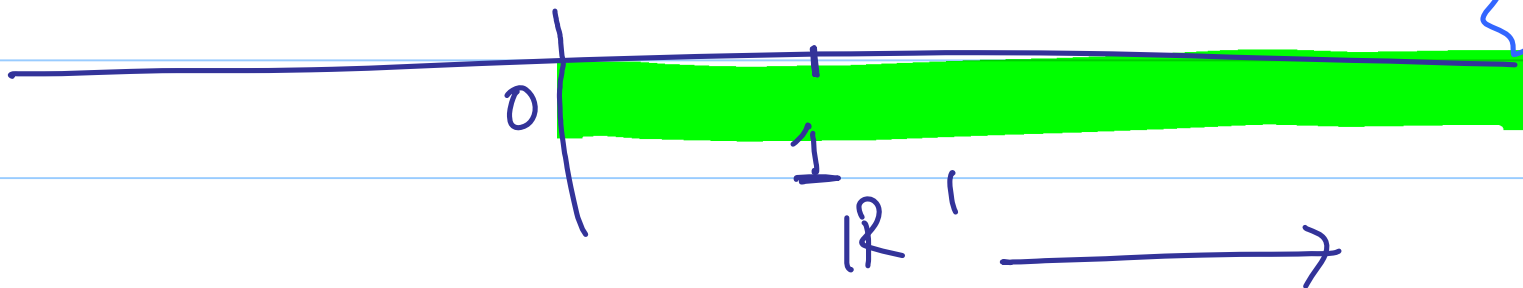
— { semiring

— examples.

Further examples of semixing

Eg 3

$$\underline{\text{Eg 3}} \quad (\mathbb{R}, \text{MAX}, \cdot) = ([0, \infty), \text{MAX}, \cdot) \quad \left. \begin{array}{l} \text{MAX} \\ \text{PRODUCT} \\ \text{SEMIKING} \end{array} \right\}$$



$(\mathbb{R}, \text{MAX})$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

$(\mathbb{R}, \cdot)$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

identity under MAX

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

$$e = 0!$$

DISTRIBUTIVE LAW?

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

SUM  
PRODUCT  
SR

$$a(\text{MAX} \{ b, c \}) = \text{MAX} \{ ab, ac \}$$

✓ ?

Eg 4  $(R, \text{MIN}, +) = ((-\infty, \infty], \text{MIN}, +)$  MIN  
SUM

SEMIKING

$(R, \text{MIN})$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

$(R, +)$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

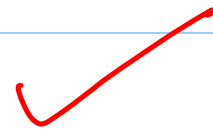
$$\text{MIN}\{e, a\} = a$$

$$e = +\infty!$$

DISTRIBUTIVE LAW?

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

$$a + \text{MIN}\{b, c\} = \text{MIN}\{a+b, a+c\}$$



?  
Y.e's!

# THE MPF PROBLEM

(marginalize a product function)

Setting:

$$S = \{1, 2, \dots, n\}$$

$$x_S = \{x_1, x_2, \dots, x_n\}$$

$x_i \in A_i$  alphabet

$$|A_i| = q_i$$

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

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

$$X_{S_j}^{\Delta} = \{x_{i_1} \ x_{i_2} \ \dots \ x_{i_{n_j}}\}$$

LOCAL  
DOMAINS



$A_{s_j} = \left\{ \begin{array}{l} \text{alphabet from which} \\ x_{s_j} \text{ is drawn} \end{array} \right.$

$$|A_{s_j}| = q_{s_j}$$

associated with each local domain

$x_{s_j}$  is a local kernel  
 $\ell_j(x_{s_j})$

$$\alpha_j : X_{s_j} \rightarrow R$$

semiring

global kernel:  $M$

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

$j^{\text{th}}$  objective function:

$$1 \leq j \leq M$$

$$\beta_j(x_{s_j}) = \sum_{x_{s_j^c}} \underbrace{\beta(x_s)}_{\text{product function}}$$

marginalization

$$s_j^c \triangleq S \setminus s_j$$

ASIDE:

$$p(x_1) = \sum_{x_2} p(x_1, x_2)$$

Fig 1 (The 8-dimensional Walsh  
- Hadamard Transform)

$$F(x_4 x_5 x_6)$$

$$= \sum_{x_1 x_2 x_3} f(x_1 x_2 x_3) \begin{matrix} x_1 x_4 & x_2 x_5 & x_3 x_6 \\ (-1) & (-1) & (-1) \end{matrix}$$

Alphabets  $A_i$

$$\left\{ \begin{array}{l} A_i = \{0, 1\} \quad \text{same for all} \\ 1 \leq i \leq M \end{array} \right.$$

$$\left\{ \begin{array}{l} |A_i| = 2 \end{array} \right.$$

{ this is thus an example of the NPF problem.

$$S = \{1, 2, 3, 4, 5, 6\} \quad X_S = \{x_1 x_2 x_3 x_4 x_5 x_6\}$$

$R =$  field of all real numbers  $= \mathbb{R}$

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

$$S_2 = \{1, 4\} \quad X_{S_2} = \{x_2 x_4\} \quad x_1 x_4$$

$$d_1(X_{S_1}) = f(x_1 x_2 x_3) \quad d_2(X_{S_2}) = (-1)$$

$$S_3 = \{2 \ 5\} \quad X_{S_3} = \{x_2 \ x_5\}$$

$$S_4 = \{3 \ 6\} \quad X_{S_4} = \{x_3 \ x_6\}$$

$$S_5 = \{4 \ 5 \ 6\} \quad X_{S_5} = \{x_4 \ x_5 \ x_6\}$$

$$d_3(X_{S_3}) = (-1)^{x_2 \ x_5}$$

$$d_4(X_{S_4}) = (-1)^{x_3 \ x_6}$$

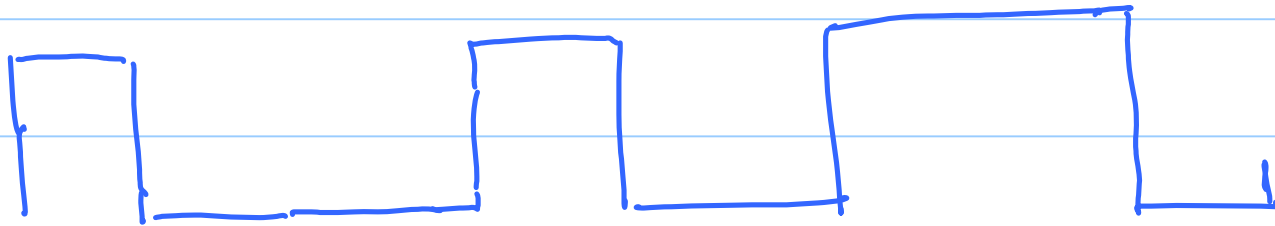
$$d_5(X_{S_5}) = 1 \quad !$$



$$\begin{bmatrix} F(000) \\ F(001) \\ \\ \\ F(111) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} f(000) \\ f(001) \\ f(010) \\ f(011) \\ : \\ f(111) \end{bmatrix}$$

{ Walsh Hadamard matrix

| - | - | | - | | | - |



Eg 2  $[7, 4, 2]$  linear block code

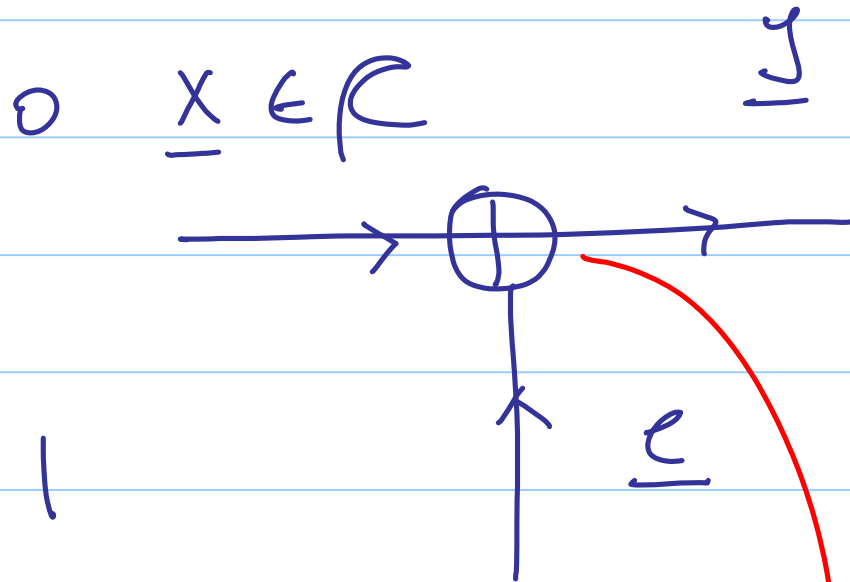
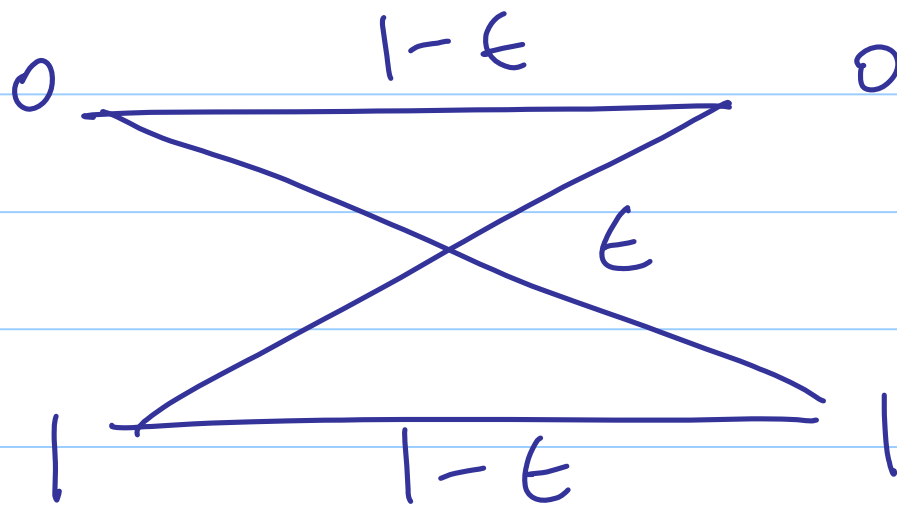
$n$   $k$   $d$

parity  
- check  
matrix

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

$(3 \times 7)$

BSC :



modulo-2  
addition

Goal: Formulate the maximum

— likelihood codeword decoding problem

as an example of MPF computation

# Lec 23 { Further Examples of the MPF Problem

## Recap

- \* completed discussion on  
semirings

- \* defined the MPF problem

- \* Examples

  - Fast Walsh Transform

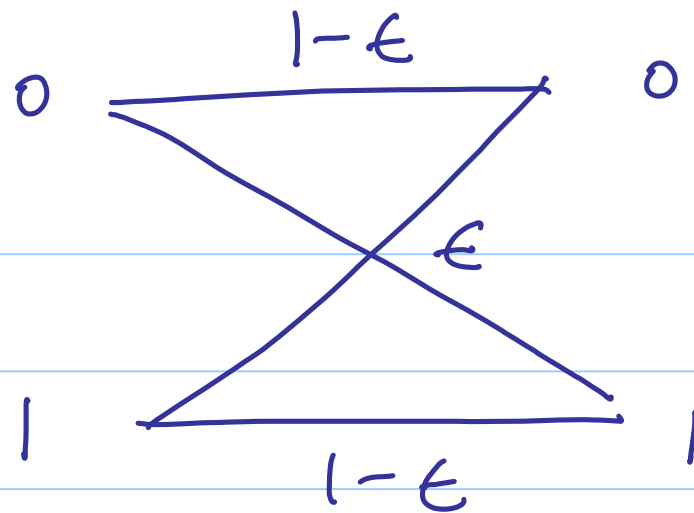
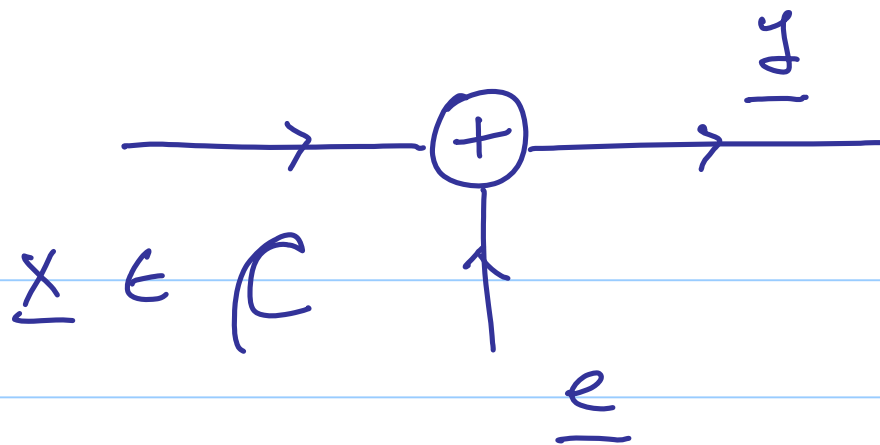
Eg 2  $[7, 4, 2]$  linear block code  
 $n$   $k$   $d$

parity  
- check  
 $m \times$

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

$(3 \times 7)$

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



Define

$$F_i(x_i) = \max_{\substack{x_1, \dots, x_{i-1} \\ x_{i+1}, \dots, x_T \\ x \in \mathcal{R}}} p(\underline{y} | \underline{x})$$



We will compute  $F_i(x_i)$  for  $i = 1, 2, \dots, 7$   
 for  $x_i \in \{0, 1\}$  example:

$F_1(0)$	$F_1(1)$
$F_2(0)$	$F_2(1)$
$F_3(0)$	$F_3(1)$
$F_4(0)$	$F_4(1)$
$F_5(0)$	$F_5(1)$
$F_6(0)$	$F_6(1)$
$F_7(0)$	$F_7(1)$

$\Leftrightarrow$

0	1	
2	8	1
1	8	1
8	6	0
2	8	1
-1	8	1
8	4	0
6	8	1

decoded code word =  $[1101101]$

$$F_i(x_i) = \max_{x_1 \dots x_{i-1}} p(\underline{y} / \underline{x})$$

$$x_{i+1} \dots x_T$$

$$\underline{x} \in \mathcal{R}$$

$$= \max_{x_1 \dots x_{i-1}} p(\underline{y} / \underline{x}) \chi_{\mathcal{R}}(\underline{x})$$

$$x_{i+1} \dots x_T$$

$$\chi_{\mathcal{R}}(\underline{x}) = \begin{cases} 1 & \underline{x} \in \mathcal{R} \\ 0 & \text{else} \end{cases}$$

$$\chi_{\mathcal{R}}(\underline{x}) = \chi_{124}(\underline{x}) \chi_{346}(\underline{x}) \chi_{457}(\underline{x})$$

$$H = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \end{matrix}$$

$$x_{124}(\underline{x}) = x_{124}(x_1 x_2 x_4) = \begin{cases} 1 & x_1 + x_2 + x_4 = 0 \\ 0 & \text{else} \end{cases}$$

$$x_{346}(\underline{x}) = x_{346}(x_3 x_4 x_6) = \begin{cases} 1 & x_3 + x_4 + x_6 = 0 \\ 0 & \text{else} \end{cases}$$

$$x_{457}(\underline{x}) = x_{457}(x_4 x_5 x_7) = \begin{cases} 1 & x_4 + x_5 + x_7 = 0 \\ 0 & \text{else.} \end{cases}$$

$$F_i(x_i) = \max_{\substack{\sim x_i \\ i=1}}^7 \prod_{i=1}^7 \phi(z_i | u_i) \quad .$$

$$\begin{array}{l} \max \\ x_1 \dots x_{i-1} \end{array} \chi_{124} (x_1 x_2 x_4) \quad .$$

$$\begin{array}{l} x_{i+1} \dots x_7 \end{array} \chi_{346} (x_3 x_4 x_6) \quad .$$

$$\chi_{457} (x_4 x_5 x_7)$$

## LOCAL DOMAINS

$$\{x_i\} \quad 1 \leq i \leq 7$$

$$\{x_1, x_2, x_4\}$$

$$\{x_3, x_4, x_6\}$$

$$\{x_4, x_5, x_7\}$$

## LOCAL KERNELS

$$\phi(I_i | x_i)$$

$$\chi_{124}(x_1, x_2, x_4)$$

$$\chi_{346}(x_3, x_4, x_6)$$

$$\chi_{457}(x_4, x_5, x_7)$$

Eg ML code-symbol decoding of the

$[7, 4, 2]$  code.

proportional to

$$p(\underline{x}_i | \underline{y}) = \sum_{\substack{\sim \underline{x}_i \\ \underline{x} \in \mathcal{R}}} p(\underline{x} | \underline{y}) \quad \textcircled{\alpha} \quad \sum_{\substack{\sim \underline{x}_i \\ \underline{x} \in \mathcal{R}}} p(\underline{x}, \underline{y})$$

$$= \sum_{\substack{\sim \underline{x}_i \\ \underline{x} \in \mathcal{R}}} p(\underline{x}) p(\underline{y} | \underline{x})$$

$\downarrow$   
 $\frac{1}{|\mathcal{R}|}$

$$\propto \sum_{\sim x_i} \prod_{j=1}^7 p(I_j | x_j) \chi_{124}(x_1 x_2 x_4) \\ \chi_{346}(x_3 x_4 x_6) \\ \chi_{457}(x_4 x_5 x_7)$$

clear that this is once again an instance of the MPPF problem.

The computation this time however, is carried out in the sum product

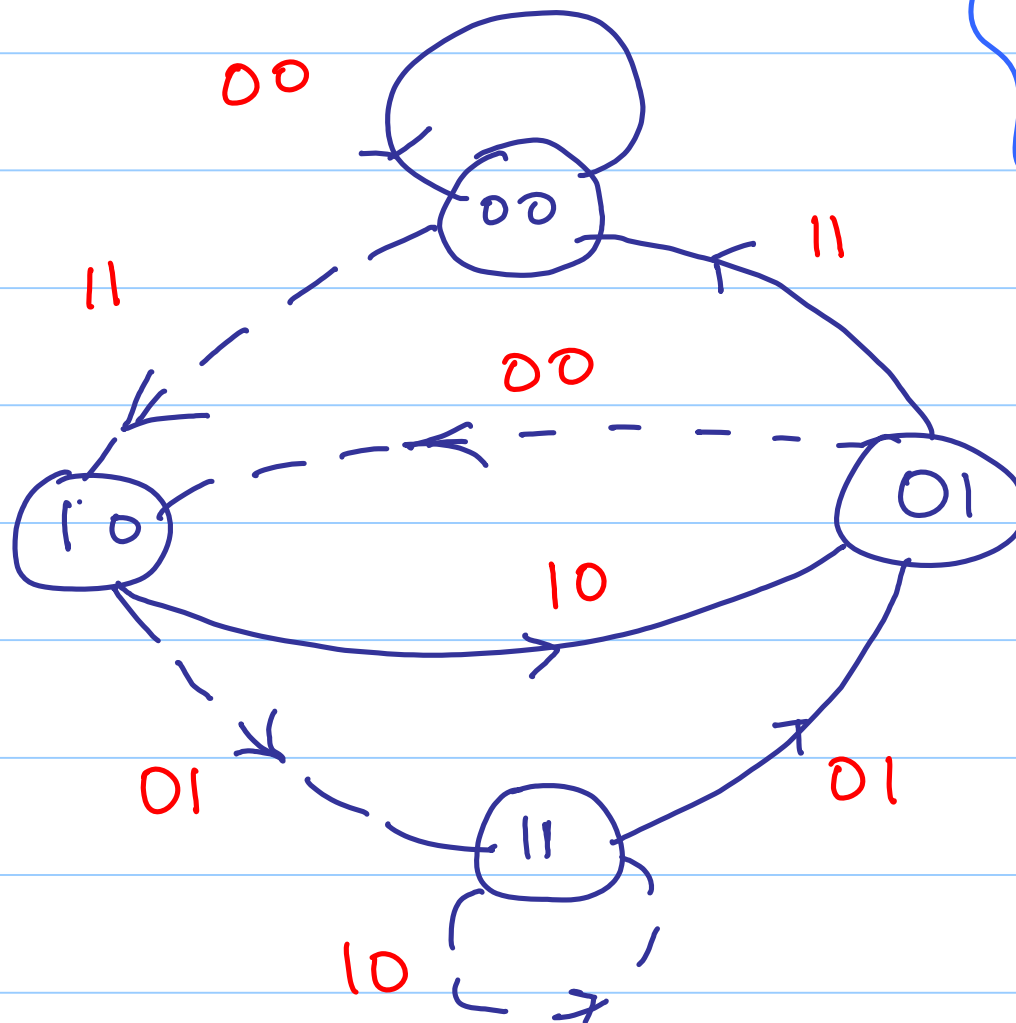


semixing. The local domains as well as local kernels remain exactly the same.

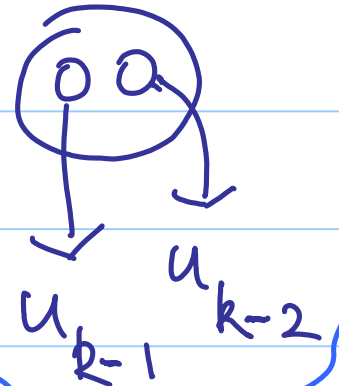
---

Eg { ML codeword decoding of  
convolutional codes.

# Finite - State Machine Description

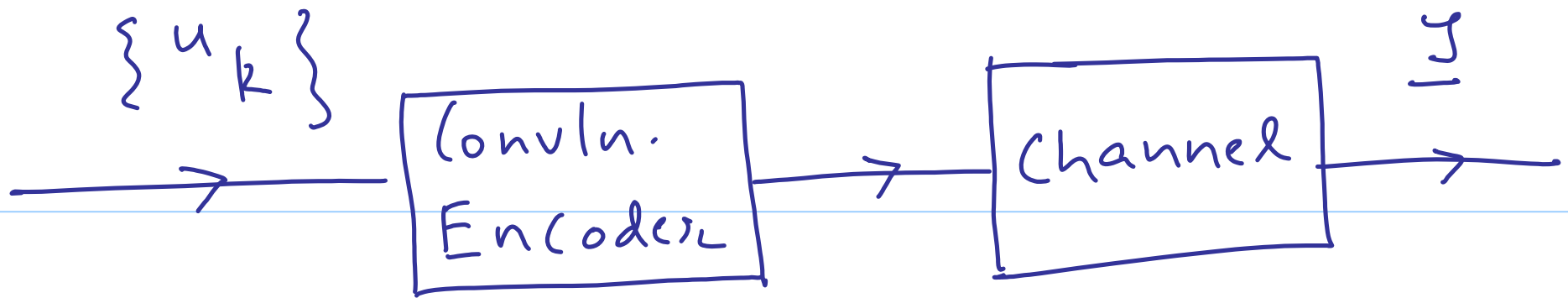


past 2 symbols



input = 0

input = 1



assume a  
rate  $1/n$   
convln code  
for simplicity

$$\left\{ \begin{matrix} (1) & (2) & & (n) \\ V_k & V_k & \dots & V_k \end{matrix} \right\}$$

Goal: ML codeword decoding, i.e.,

identifying the codeword  $\underline{v}$  such  
that  $\phi(\underline{y} | \underline{v})$  is a maximum.

which is equivalent to identifying

the message vector  $\underline{u}$  which is

such that  $\phi(\underline{y} | \underline{u})$  is a max.

$$\{u_k\}_{k=0}^{N-1}$$

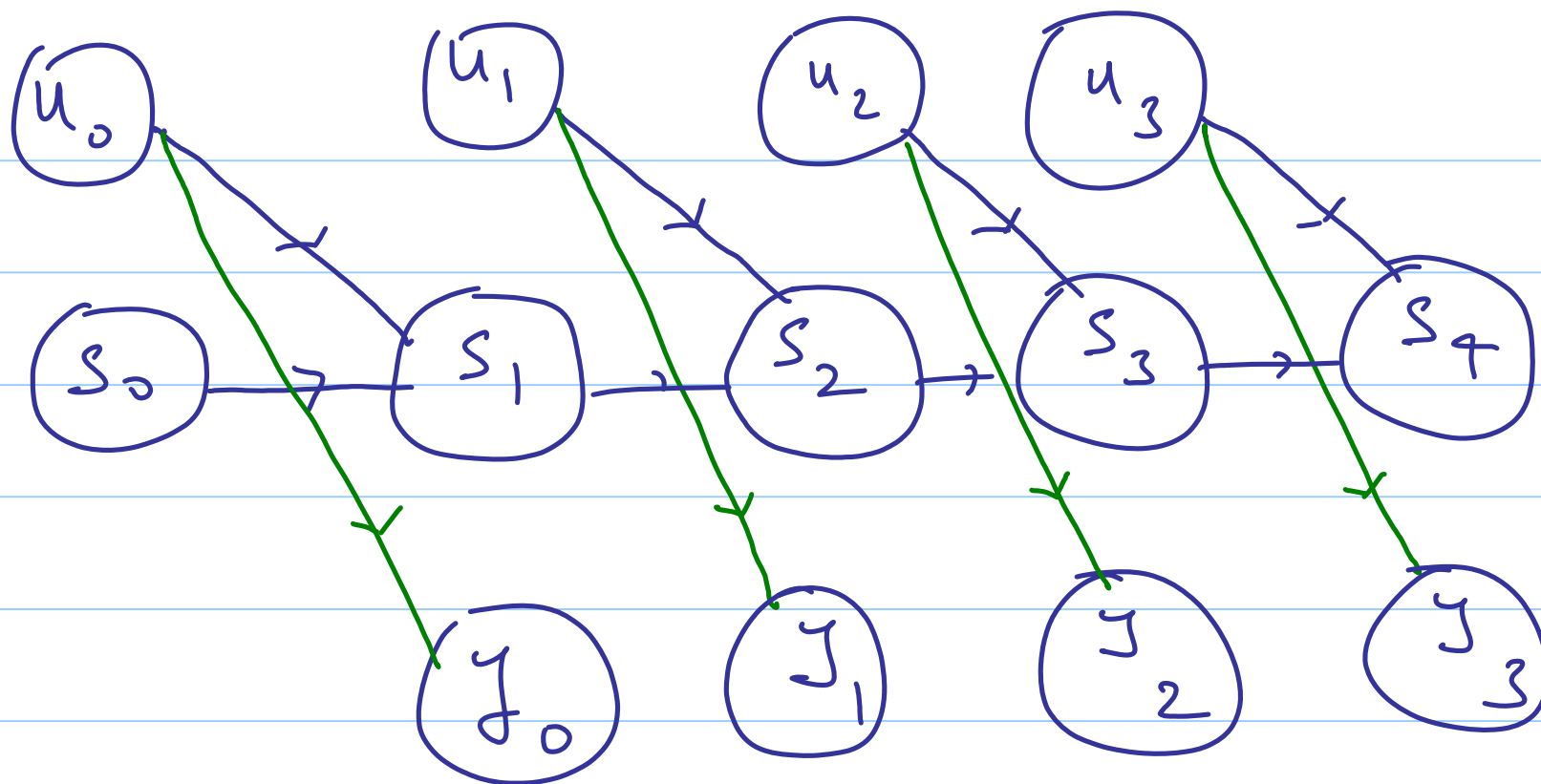
state sequence  $\{s_k\}_{k=0}^N$

output sequence  $\{y_k\}_{k=0}^{N-1}$

each  $y_k$  is an  $n$ -tuple

$$y_k = (y_{k1} \dots y_{kn})$$

(In the example below,  $N=4$ )



# Lec 24 Junction Trees

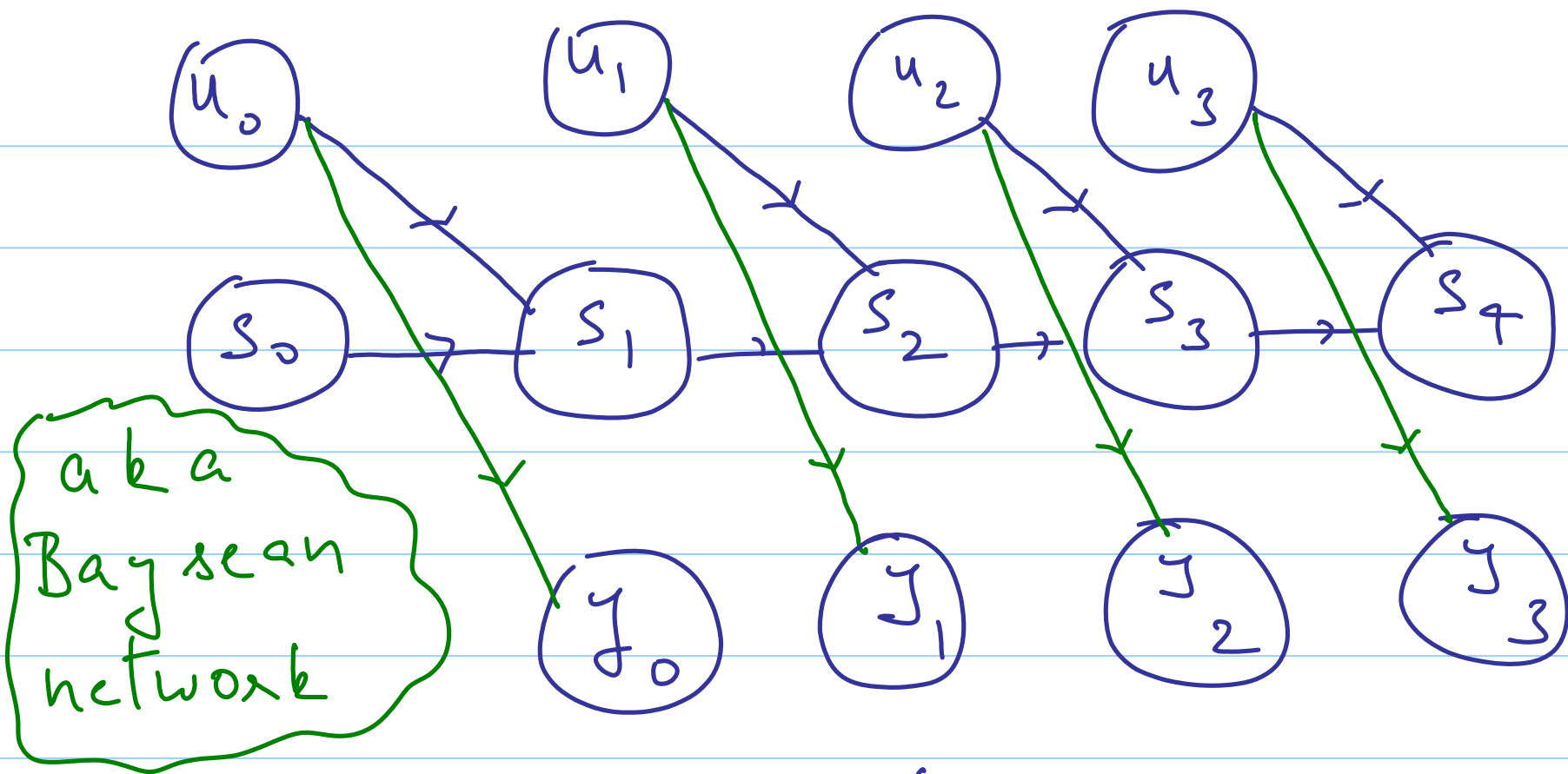
## Recap

- { cast the ML codeword decoding problem for the  $[7, 4, 2]$  code as an MPF problem
- { did the same for ML code-symbol decoding

{ of 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))

tells us how to factor the joint

prob. dist'n fn:

$$p\left(\left\{u_i\right\}_{i=0}^3, \left\{s_i\right\}_{i=0}^4, \left\{y_i\right\}_{i=0}^3\right)$$

$$= p(s_0) \prod_{i=0}^3 p(u_i) p(s_{i+1} | s_i u_i)$$

$$p(y_i | s_i u_i)$$

$$F_i(u_i) = \max_{\sim u_i} p(\underline{y} | \underline{u}) \quad 0 \leq i \leq 3$$

$$\propto \max_{\sim u_i} p(\underline{y}, \underline{u})$$

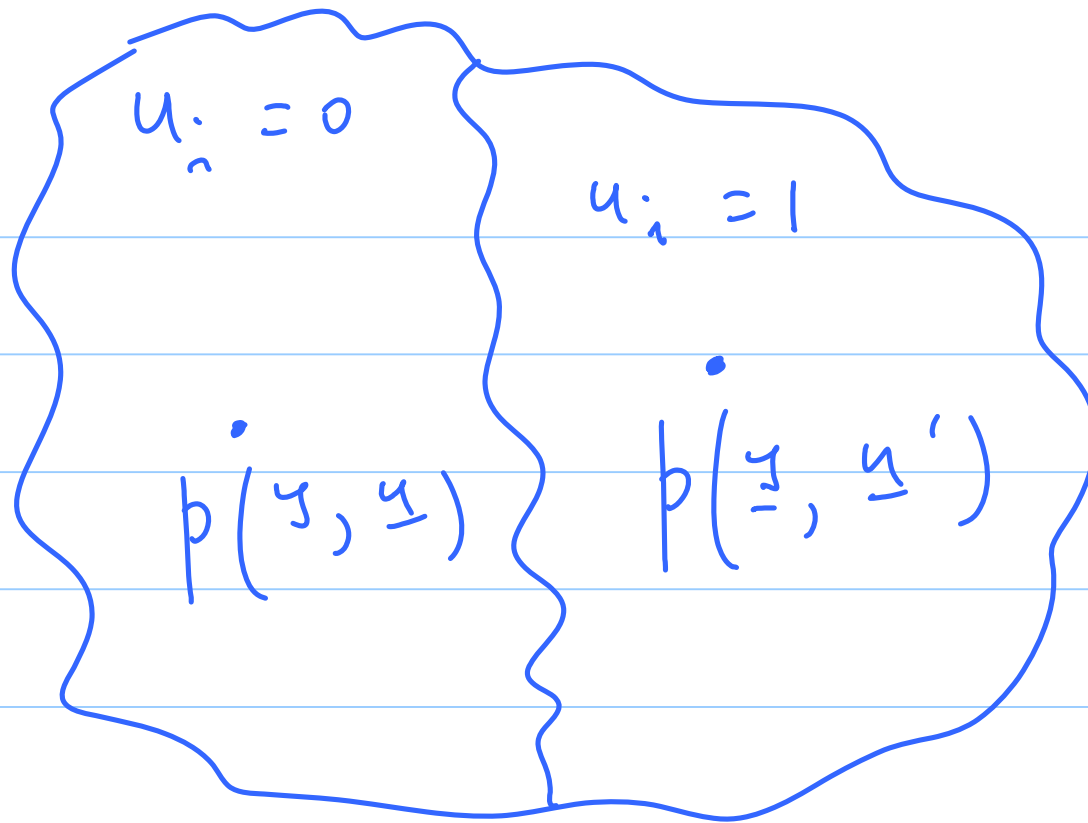
since all  
message vectors  
 $\underline{u}$  are equally  
likely

$$= \max_{\sim u_i} p(\underline{y}, \underline{u}, \underline{s})$$

$\underline{s}$

$$\begin{aligned}
 &= \max_{u_i} p(s_0) \prod_{i=0}^3 p(u_i) p(s_{i+1} | s_i u_i) \\
 &\quad \leq p(y_i | s_i u_i)
 \end{aligned}$$

$\frac{LD}{\{u_i\}}$	$\frac{LK}{p(u_i)}$	$0 \leq i \leq 3$
$\{s_0\}$	$p(s_0)$	
$\{s_{i+1} \ s_i \ u_i\}$	$p(s_{i+1}   s_i \ u_i)$	
$\{s_i \ u_i\}$	$p(y_i   s_i \ u_i)$	

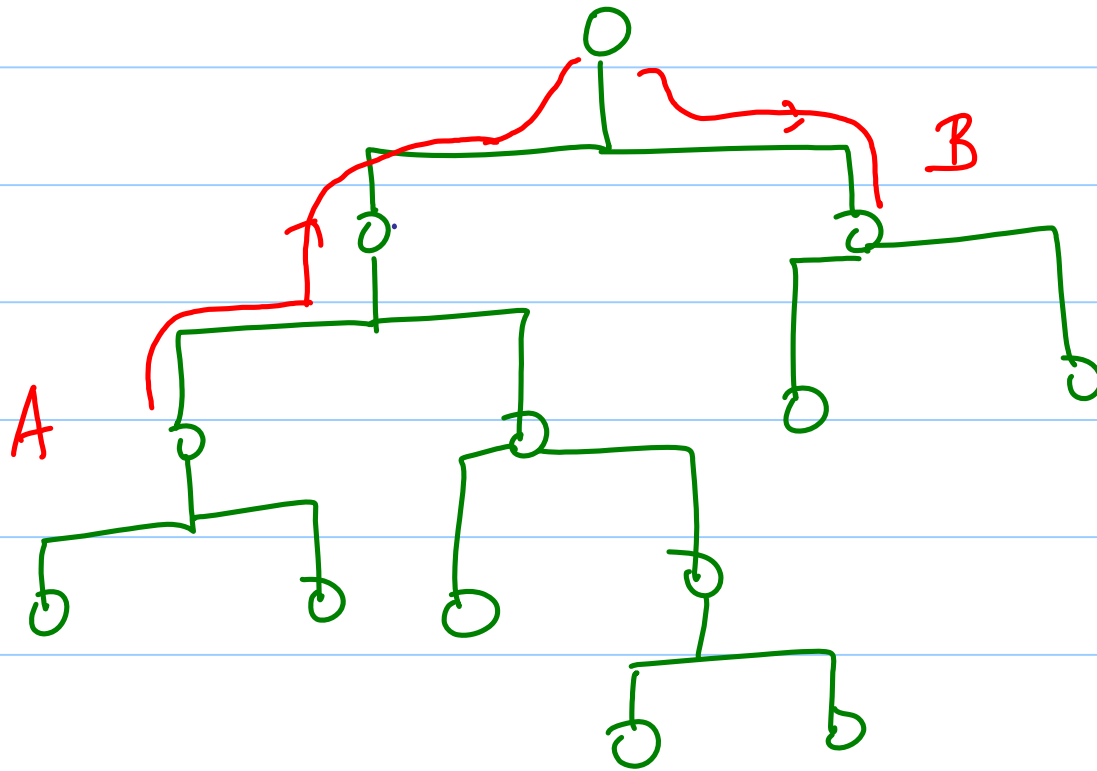


{ We have thus formulated the ML  
code word decoding of a convolution code as  
an MPF problem.

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.



# of nodes  
= 13

# of edges  
= 12

Note: \* In any tree we have:

$$\text{the \# of nodes} - \text{\# of edges} = 1$$

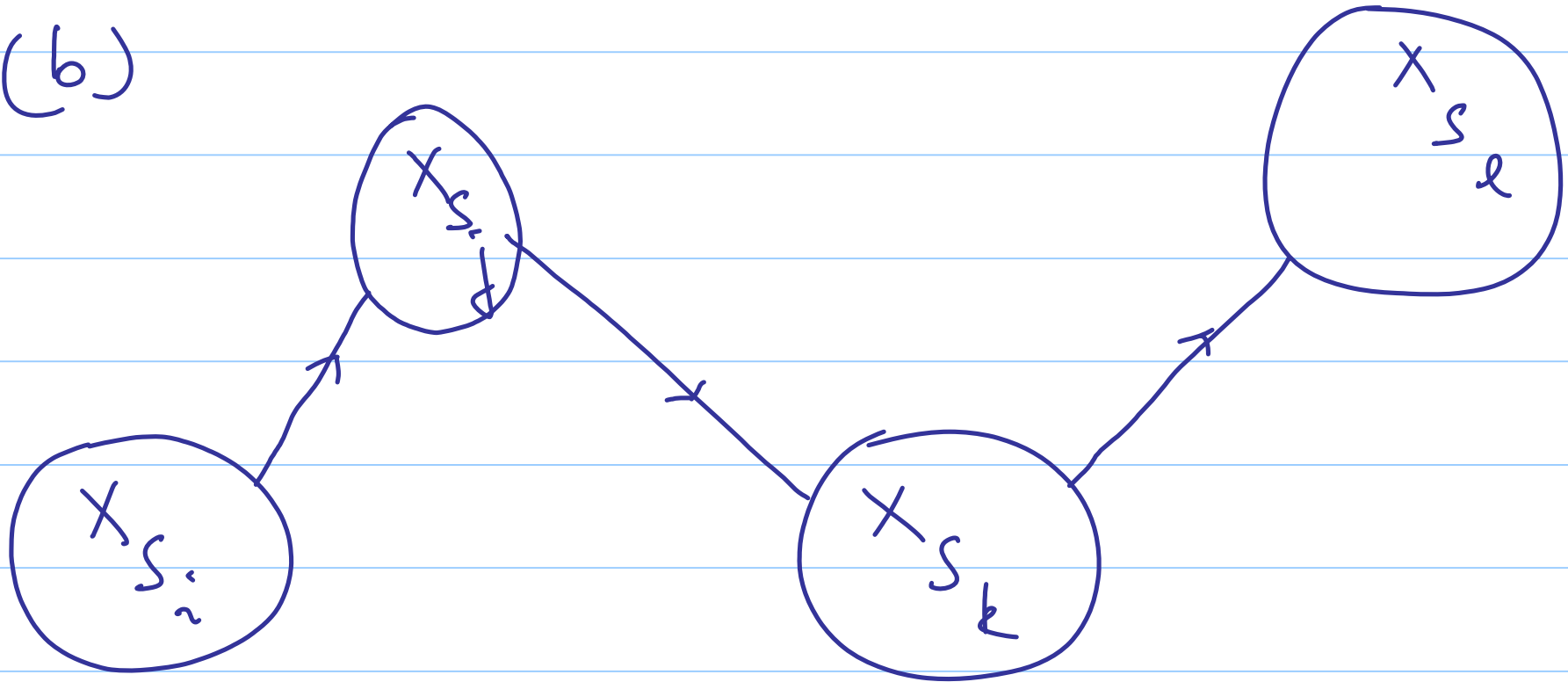
\* { In any tree there is a unique  
path between any two distinct  
nodes.



Defn. In the setting of the MPF problem, a join tree is a graph whose nodes are in 1-1 correspondence with the local domains  $\{X_i\}_{i=1}^M$  and where edges are drawn between nodes in such a way that

(a) The graph is a tree

(b)



For every node  $x_{s_i}$  on the  
unique path lying between nodes  
 $x_{s_i}$  and  $x_{s_e}$ , it must be that

$$s_j \subseteq s_i \cap s_e$$

Eg 2  $[7, 4, 2]$  linear block code

parity  
- check  
matrix

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

$(3 \times 7)$

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

## LOCAL DOMAINS

$$\{x_i\} \quad 1 \leq i \leq 7$$

$$\{x_1, x_2, x_4\}$$

$$\{x_3, x_4, x_6\}$$

$$\{x_4, x_5, x_7\}$$

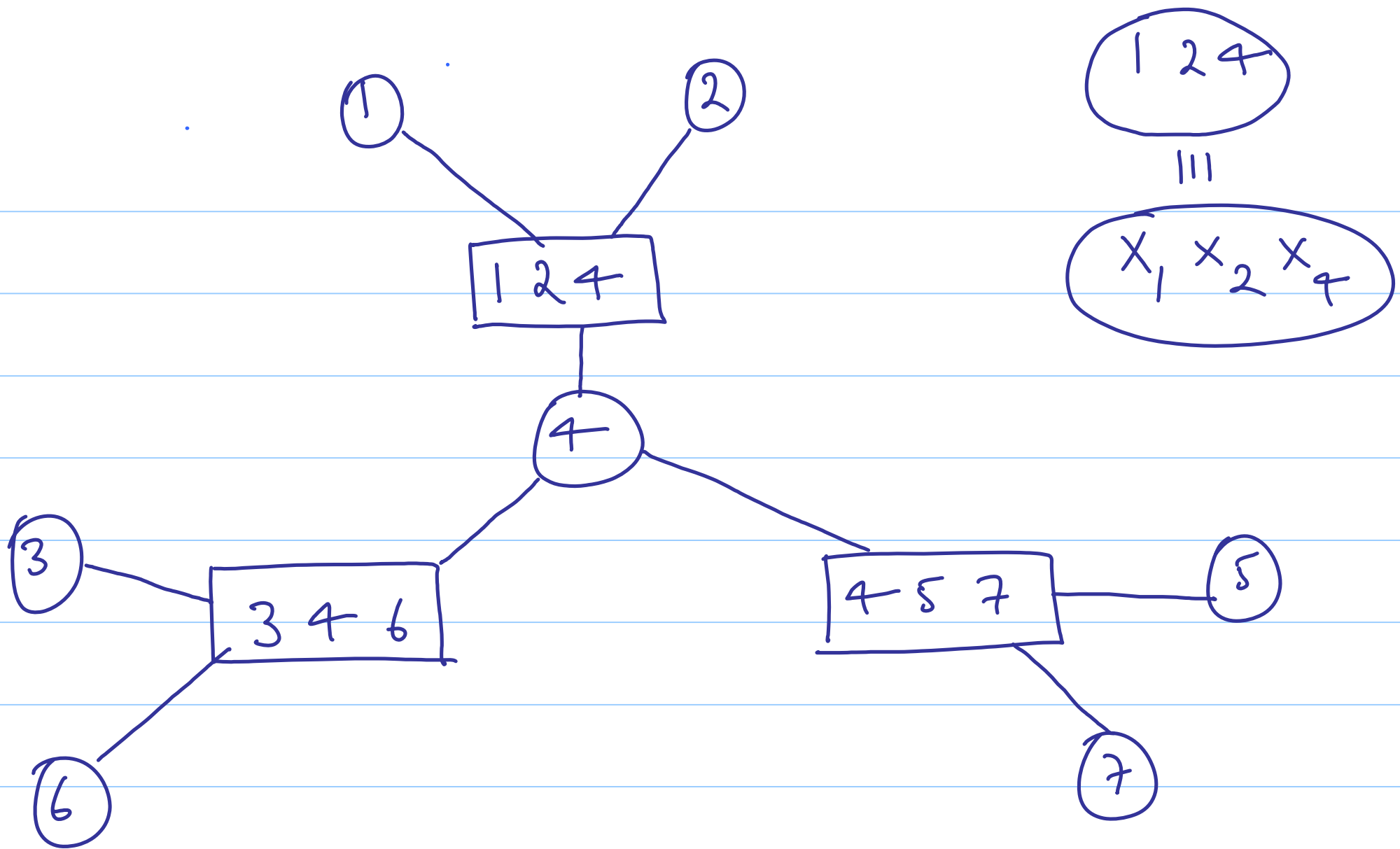
## LOCAL KERNELS

$$\phi(I_i | x_i)$$

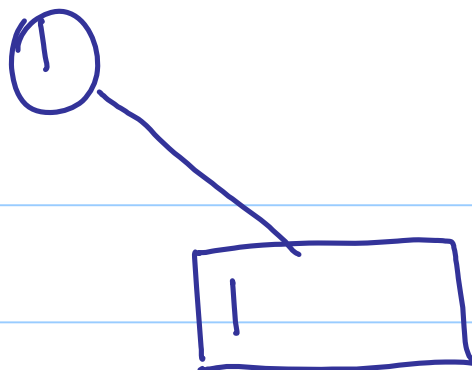
$$\chi_{124}(x_1, x_2, x_4)$$

$$\chi_{346}(x_3, x_4, x_6)$$

$$\chi_{457}(x_4, x_5, x_7)$$



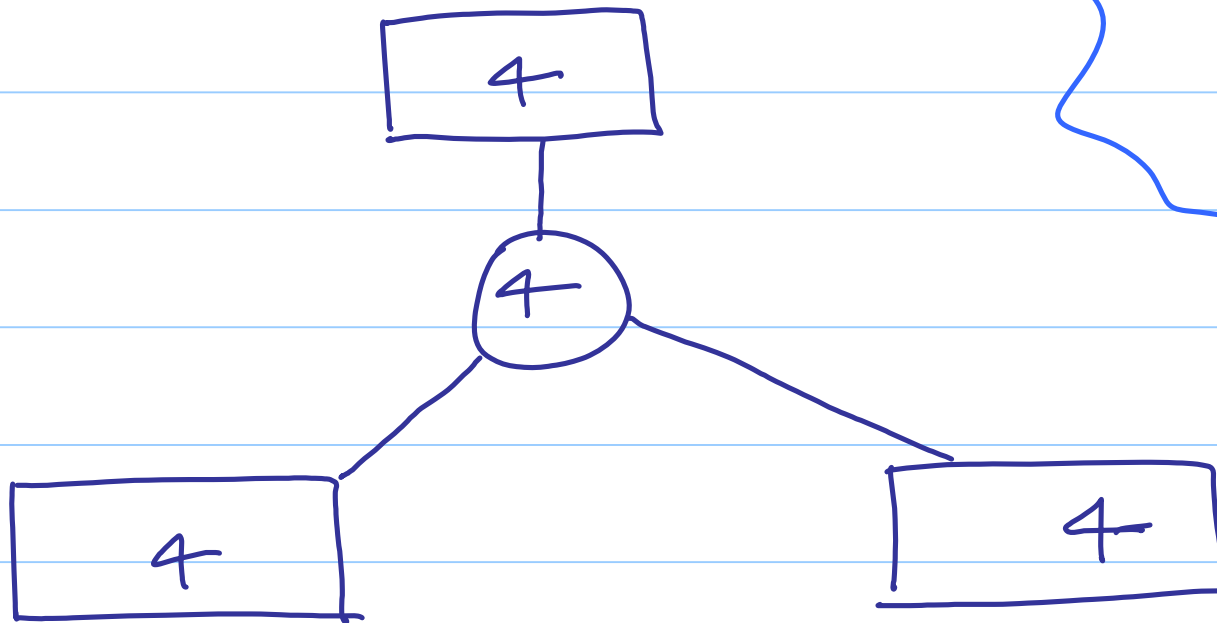
Note: An alternative defn. of a  $jn$   
tree is a tree which when projected  
onto each individual variable also  
yields a tree.



projection  
onto  
 $x_1$



projection  
onto  
 $x_4$



Qn { How does one construct a  $gn$   
 { tree?

Ans { By constructing a maximal  
 { - wt spanning tree

Thm If  $g$  is a graph whose  
 nodes correspond to the local domains  
 of an MPP problem, and which

is also a tree, then it must be that

$$\text{edge weight}_{f, g} \leq \text{node weight}_{f, g} - n$$

where  $n = \# \text{ of variables}$   
 $= |S|.$

---

# lec 25 { Examples of Jn Tree Construction }

## Recap

- \* { formulating the decoding of a  
convolutional code as an  
MPF problem }

- \* Defined "junction tree"

  - example

  - stated theorem

Thm If  $G$  is a graph whose nodes correspond to the local domains of an MPF problem, and which

is also a tree, then it must be that

$$\begin{array}{ccc} \text{edge weight} & \leq & \text{node weight} - n \\ \uparrow \quad \uparrow & & \text{of } y \\ y & & y \end{array}$$

where  $n = \#$  of variables  
 $= |S|.$

---

Pf. 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  $x_{S_j} = |S_i \cap S_j|$ .

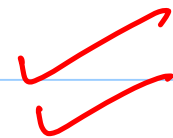
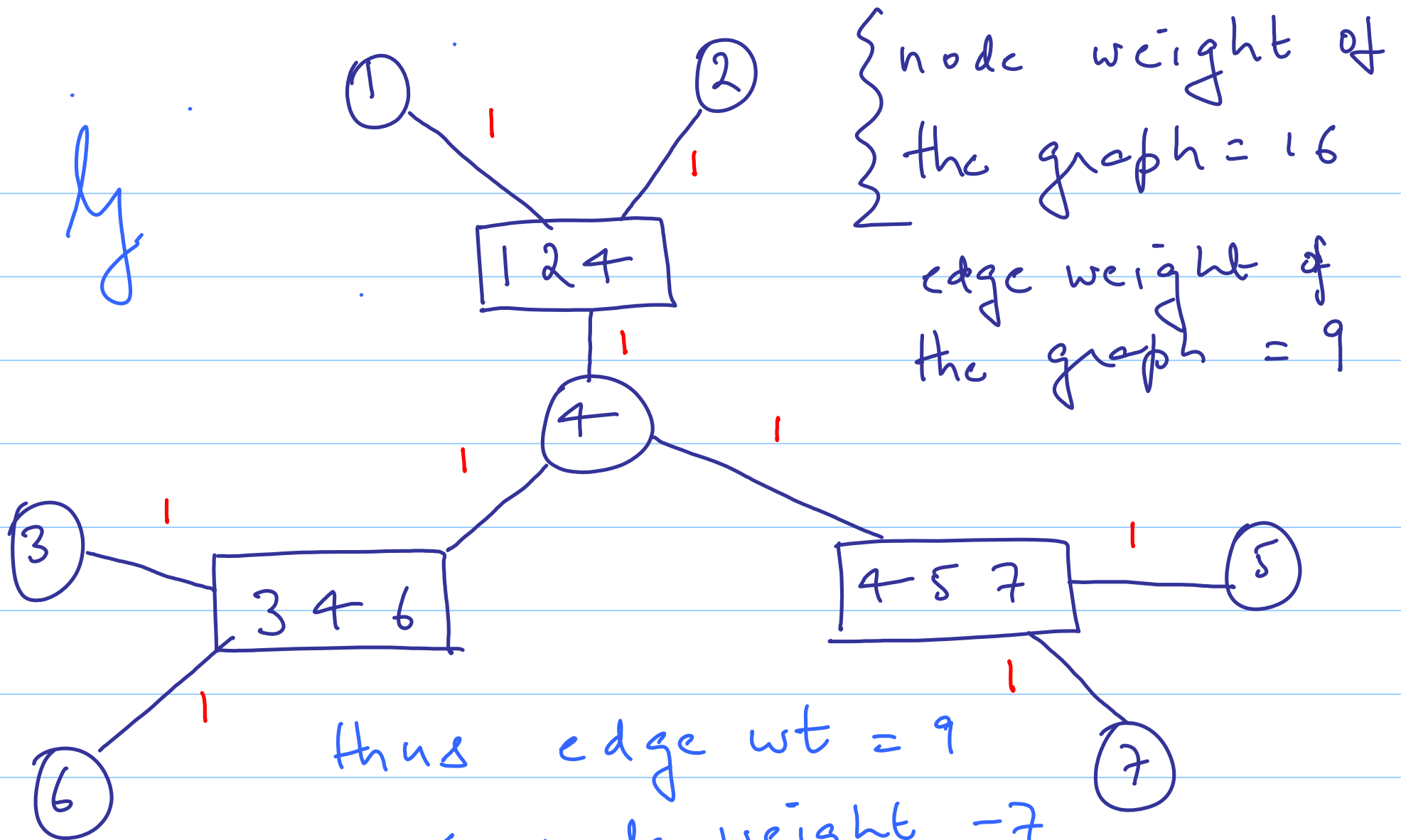
node weight of the graph

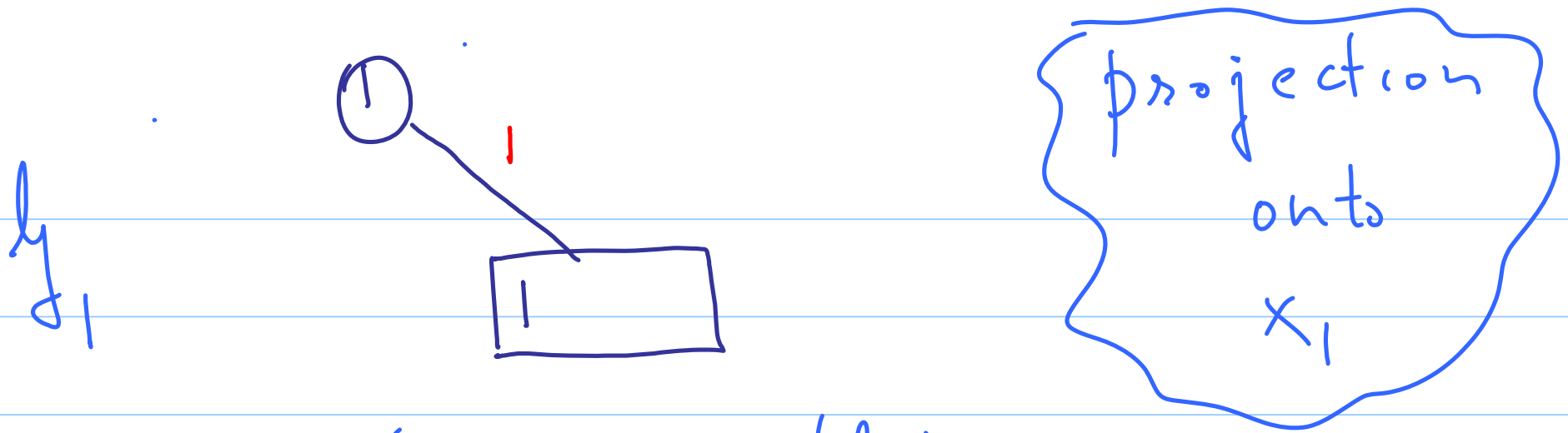
=  $\sum$  of the node weight of the  
 $\sum$  nodes in the graph.

edge weight of the graph

=  $\sum$  of the edge weights of  
 $\sum$  edges in the graph.







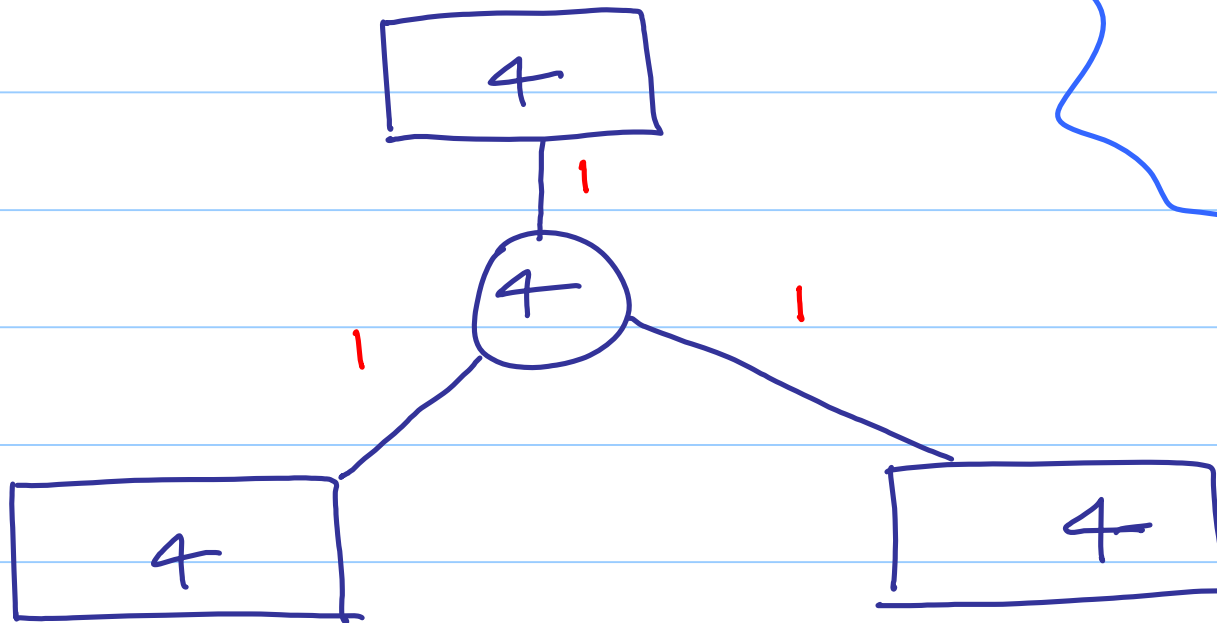
$$EW(y) = NW(y) - 1$$

$\left\{ \begin{array}{l} \text{edge} \\ \text{weight} \end{array} \right\}$ 
 $\updownarrow$ 
 $\left\{ \begin{array}{l} \text{node} \\ \text{weight} \end{array} \right\}$

$$\Leftrightarrow \# \text{ of edges in } y_1 = \left\{ \begin{array}{l} \# \text{ of nodes} \\ \text{in } y_1 - 1 \end{array} \right.$$

(this is true since  $y_1$  is a tree)

projection  
onto  
 $x_4$



$NW(y_4) = 4 = \# \text{ of nodes in } y_4$

$EW(y_4) = 3 = \# \text{ of edges in } y_4$

$\therefore \boxed{EW(y_4) = NW(y_4) - 1}$

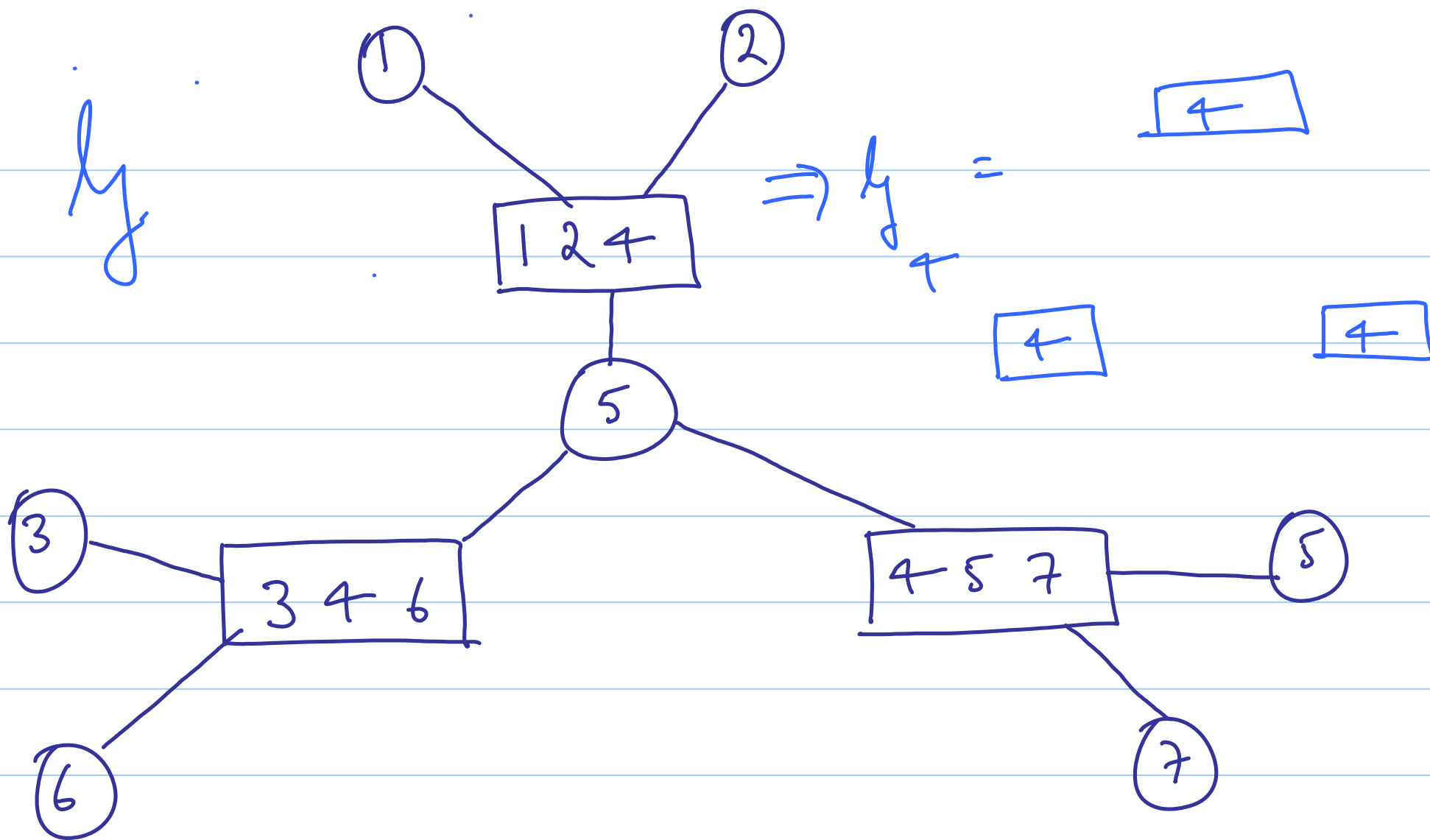
Pf. When  $T$  is a tree we have

$$EW(T) = \sum_{i=1}^n EW(T_i)$$

$$= \sum_{i=1}^n (NW(T_i) - 1)$$

$$= \sum_{i=1}^n NW(T_i) - n$$

$$\therefore \boxed{EW(T) = NW(T) - n} \quad (1)$$



When  $g$  is not a jn tree, then at least one of the  $g_i$  will fail to be a tree (will be the union of  $\geq 2$  trees instead) and hence

$$EW(g_i) < NW(g_i) - 1$$

If we now proceeded to argue as when deriving (1) we would end up with

$$EW(y) < NW(y) - n$$

the theorem follows.

---

This suggests that if the local domains can be organized into a  $jn$  tree, then that  $jn$  tree represents a maximal wt spanning tree for the collection of local domains.

---

A maximal wt spanning tree (MST) can be constructed using Prim's



greedy algorithm which we will  
now illustrate.

---

Eg 1

$$\alpha(x, w) = \sum_{y, z} f(x, y, w) g(x, z)$$

objective  
functions

$$\beta(y) = \sum_{x, w, z} f(x, y, w) g(x, z)$$

LD

$$\{x, y, w\}$$

$$\{x, z\}$$

$$\{x, w\}$$

LK

$$f(x, y, w)$$

$$g(x, z)$$

$$1$$

LD

$$\{y\}$$

LK

$$1$$

$$w \times y \times z \in A$$

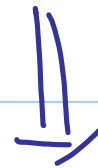
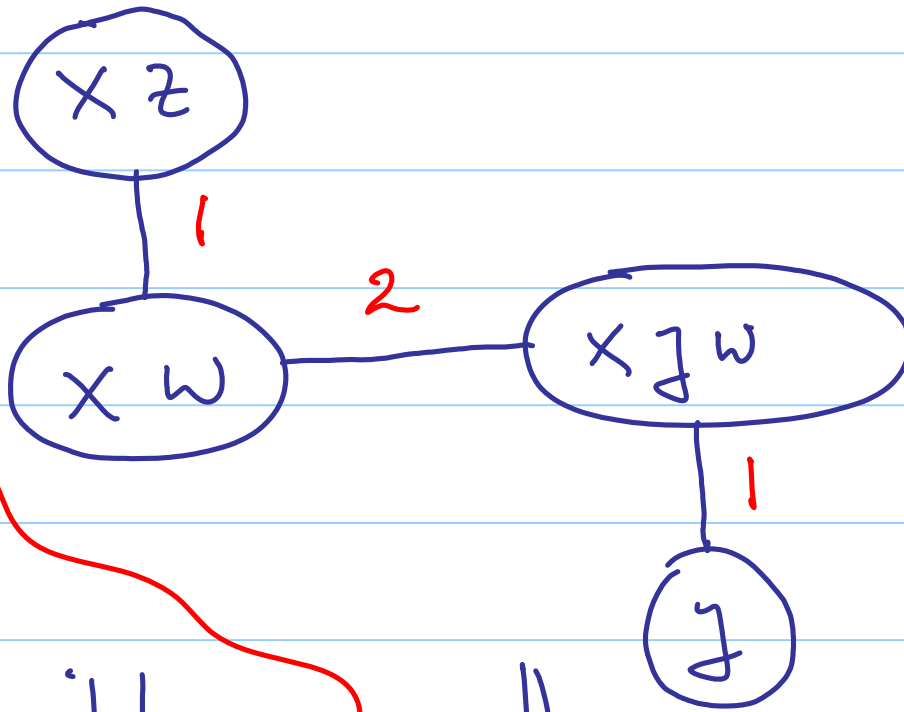
$$|A| = 9$$

$$NW(y) = 8$$

$$EW(y) = 4$$

$$\text{diff} = 4$$

= # of variables



thus this is the  
 { in tree for this  
 example

ASIDE

It turns out that an edge  
connecting nodes  $x_{s_i}$  and  $x_{s_j}$

incurs a cost =

$$q_{s_i} + q_{s_j} - q_{s_i \cap s_j} \quad (2)$$

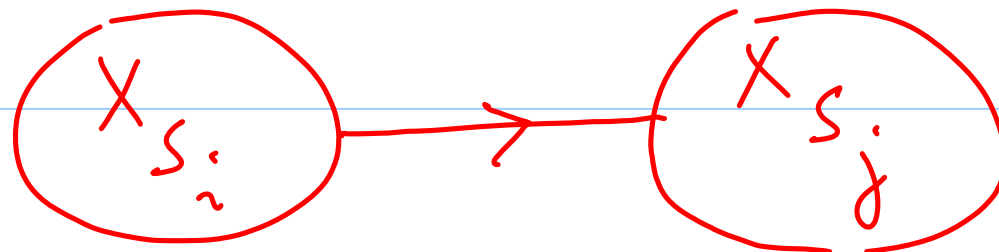


Fig 2 (The 8-dimensional Walsh  
- Hadamard Transform)

$$F(x_4 x_5 x_6)$$

$$= \sum_{x_1 x_2 x_3} f(x_1 x_2 x_3) \begin{matrix} x_1 x_4 & x_2 x_5 & x_3 x_6 \\ (-1) & (-1) & (-1) \end{matrix}$$

LD

$$\{x_1 x_2 x_3\}$$

$$\{x_1 x_4\}$$

$$\{x_2 x_5\}$$

$$\{x_3 x_6\}$$

$$\{x_4 x_5 x_6\}$$

LK

$$f(x_1 x_2 x_3)$$

$$x_1 x_4$$

$$(-1)$$

$$x_2 x_5$$

$$(-1)$$

$$x_3 x_6$$

$$(-1)$$

$$F(x_4 x_5 x_6)$$

LD

$\{x_1 x_2 x_3\}$

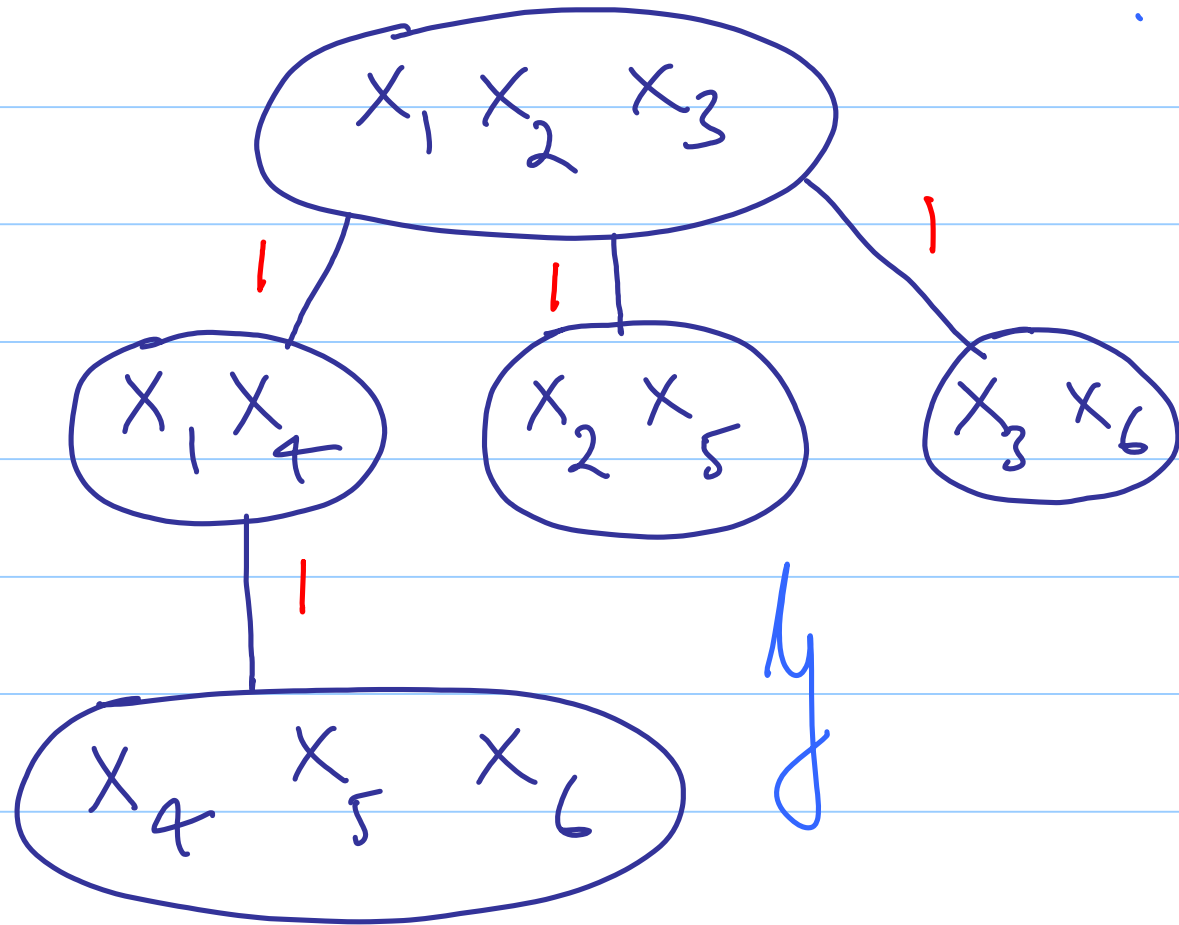
$\{x_1 x_4\}$

$\{x_2 x_5\}$

$\{x_3 x_6\}$

$\{x_4 x_5 x_6\}$

$$\left. \begin{array}{l} NW(y) = 12 \\ EW(y) = 4 \end{array} \right\} \begin{array}{l} \text{diff} = 8 \\ \# \text{ of} \\ \text{variables!} \end{array}$$



Lec 26

# Message passing on the Jn Tree

Recap

- \* completed proof showing that a jn tree (if <sup>i.e.</sup> the local domains can be organized into a jn tree) is necessarily a maximal wt spanning tree (MST)



- we would invoke Prim's (greedy) algorithm to construct a MST
- Eg - pair of simple computations
  - Walsh transform

Eg 3 [7, 4, 2] Code

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

$$p(x_4 | \underline{y}) \propto \sum_{\substack{\sim x_4 \\ 7}} p(\underline{y} | \underline{x}) \chi_{\mathcal{R}}(\underline{x})$$

$$= \sum_{\sim x_4} \prod_{j=1}^7 p(j_j | x_j) \chi_{124}(x_1 x_2 x_4) \\ \chi_{346}(x_3 x_4 x_6)$$

$$\chi_{457}(x_4 x_5 x_7)$$

LD

$$\{x_j\}_{j=1}^7$$

$$\{x_1 x_2 x_4\}$$

$$\{x_3 x_4 x_6\}$$

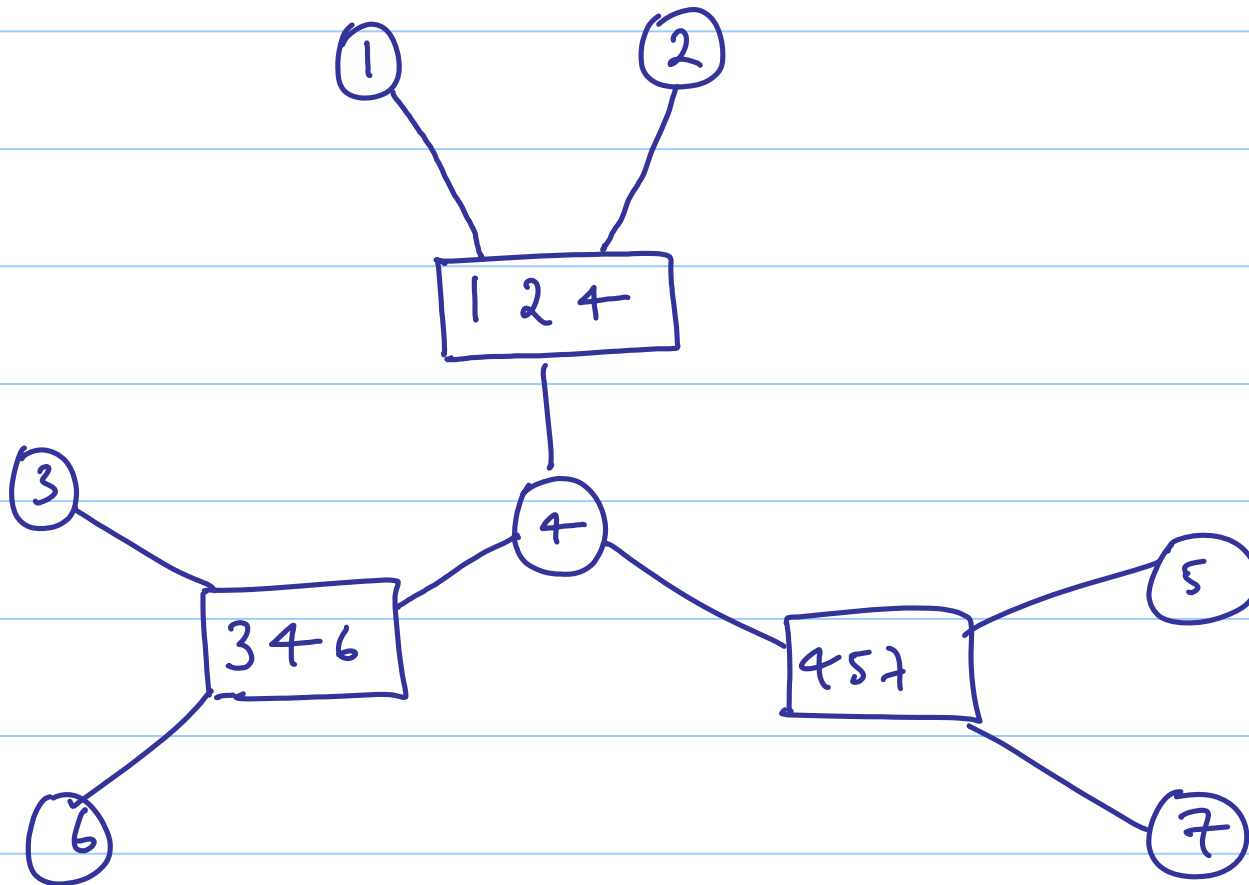
LK

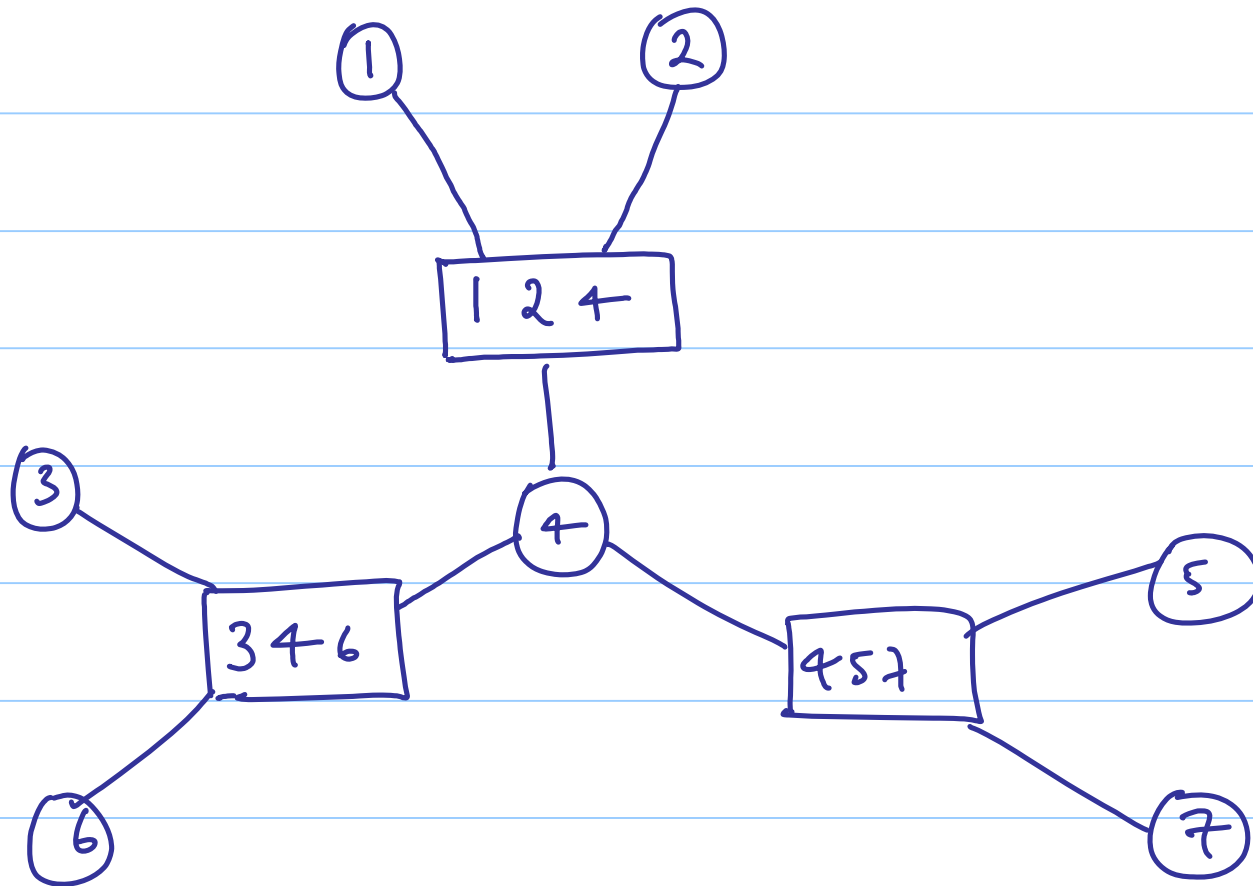
$$p(j_j | x_j)$$

$$\chi_{124}(x_1 x_2 x_4)$$

$$\chi_{346}(x_3 x_4 x_6)$$

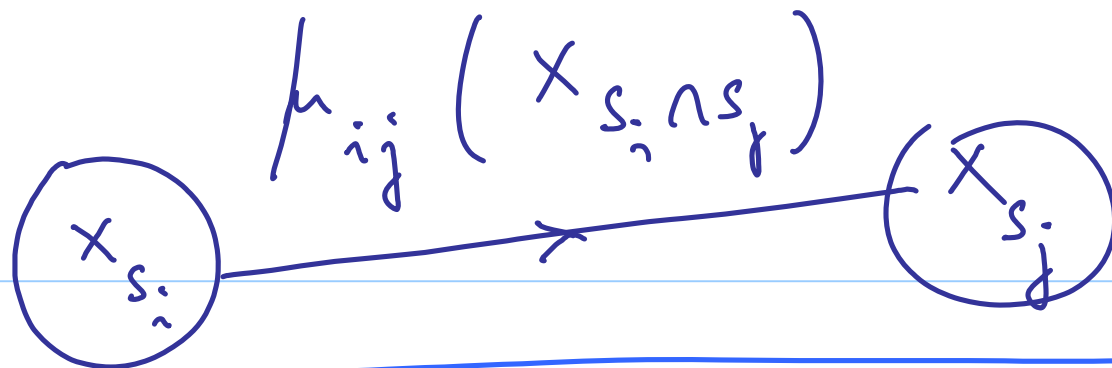
$$\{x_4, x_5, x_7\} \quad \chi_{457} (x_4, x_5, x_7)$$





Step 1 { in solving (where possible) the  
MPF problem is to organize  
the local domains into a join tree.

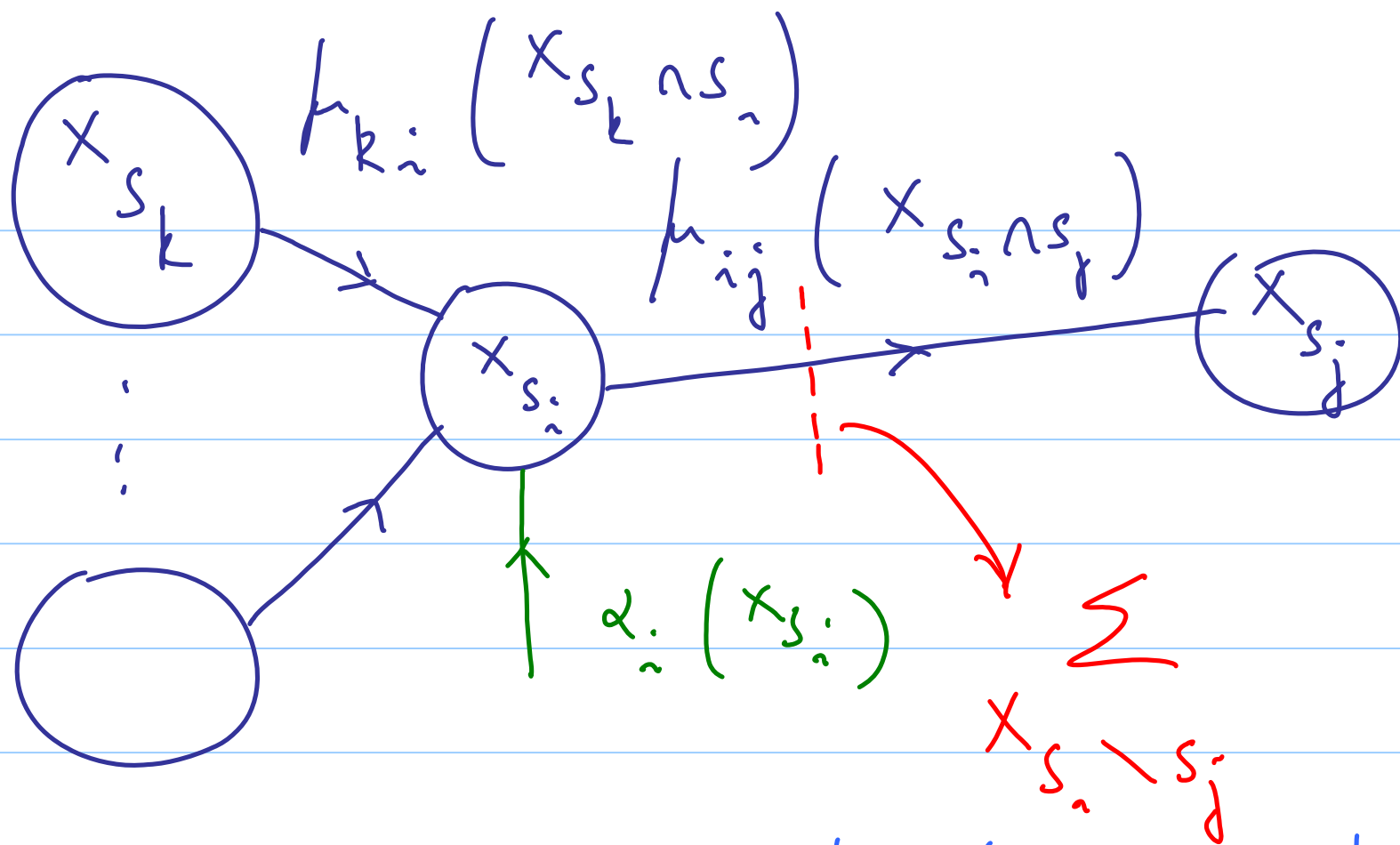
Step 2 { Pass messages along the  
edges of the join tree in  
accordance with some  
schedule



$$h_{ij}(x_{s_i} \cap x_{s_j}) = \sum_{x_{s_i} \cap s_j} d_i(x_{s_i})$$

$$\prod_{\substack{k \in N_i \\ k \neq j}} h_{ki}(x_{s_k} \cap x_{s_i})$$

$N_i$  = set of neighbors of  $x_{s_i}$



Note: the message  $h_{ij}(x_{s_i}, s_j)$  passed on by a leaf node is simply the marginalization of its local band.

Eg 3 [7, 4, 2] Code

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

xmtd  
codeword

$$\underline{x} = (0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0)$$

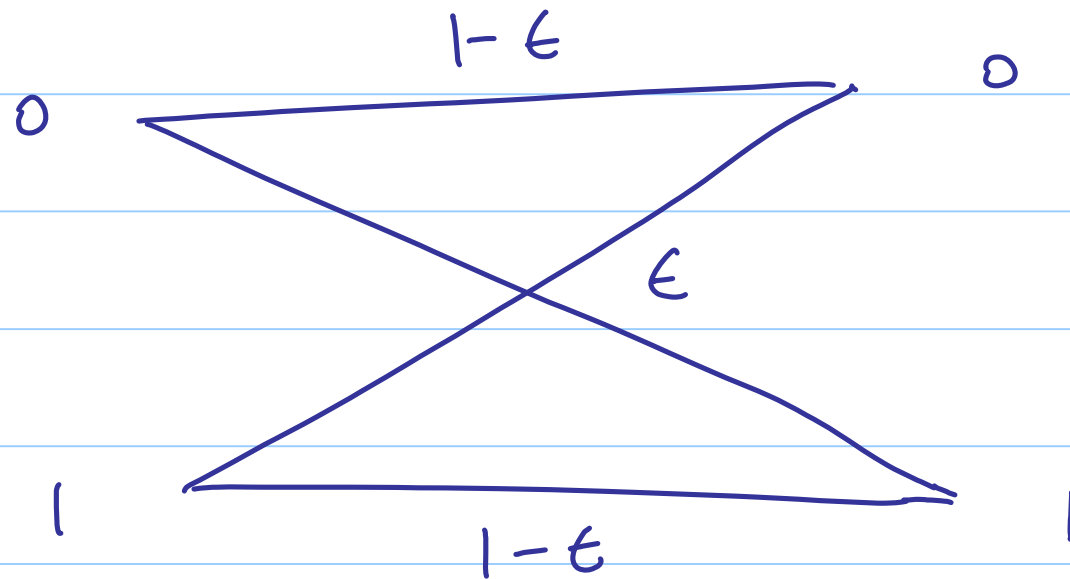
$$\underline{e} = (0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0)$$

{received  
vector

$$\underline{y} = (0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0)$$



Assume a BSC



$$\underline{y} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{pmatrix}$$

$$p(\underline{y}_1 | x_1) = \begin{bmatrix} p(y_1 | x_1 = 0) \\ p(y_1 | x_1 = 1) \end{bmatrix} \left\{ \begin{array}{l} \text{This is typical} \\ \text{in that it is a} \\ \text{vector representation} \\ \text{of the fn.} \end{array} \right.$$

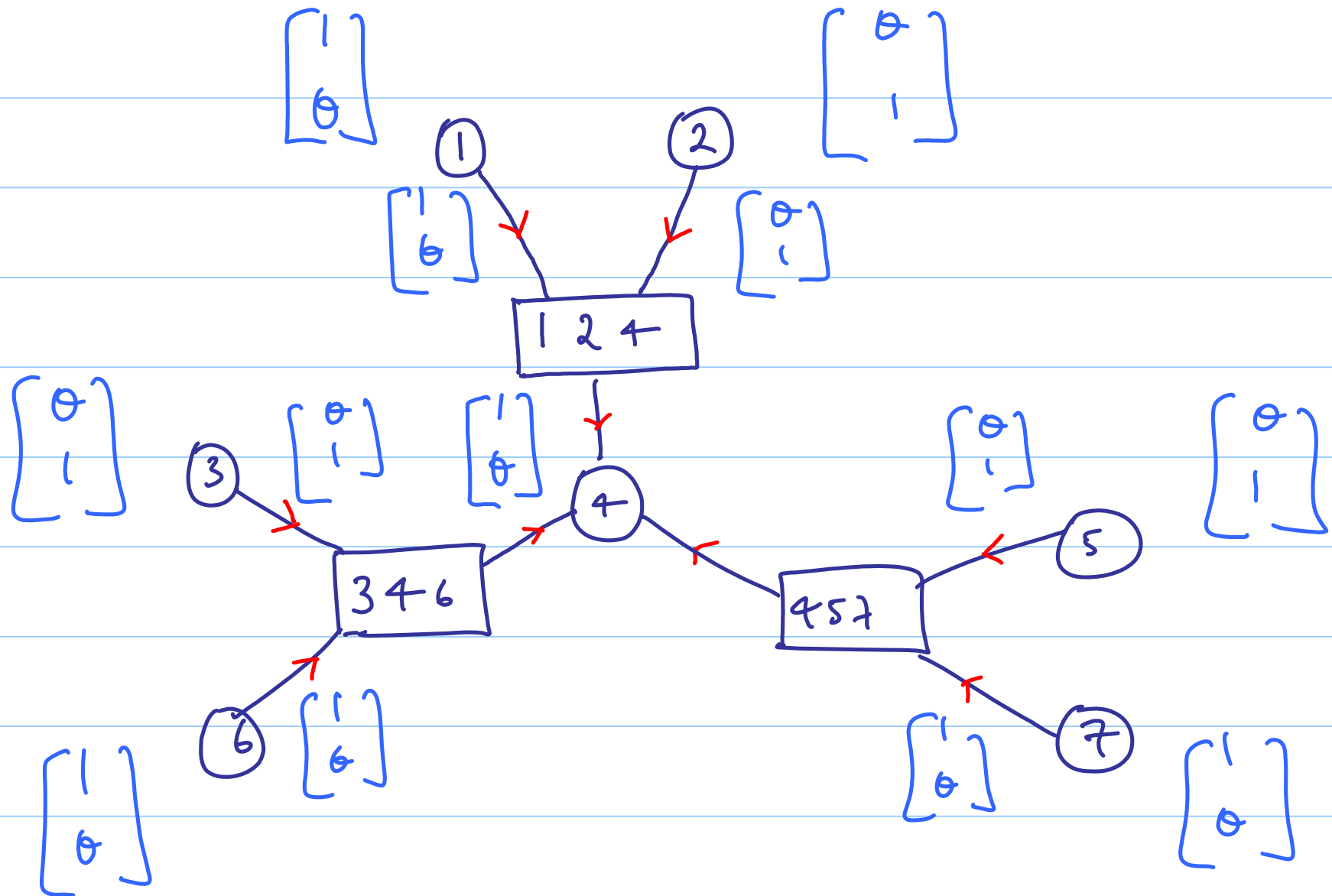
$$\underline{y} = \begin{matrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} \end{matrix}$$

$$p(\underline{y}_1 | x_1) \Rightarrow \begin{bmatrix} p(y_1 | x_1 = 0) \\ p(y_1 | x_1 = 1) \end{bmatrix} = \begin{bmatrix} 1 - \epsilon \\ \epsilon \end{bmatrix}$$

scaling by  $(1 - \epsilon)$   
yields:

$$\text{where } \theta = \begin{pmatrix} \epsilon \\ 1 - \epsilon \end{pmatrix}$$

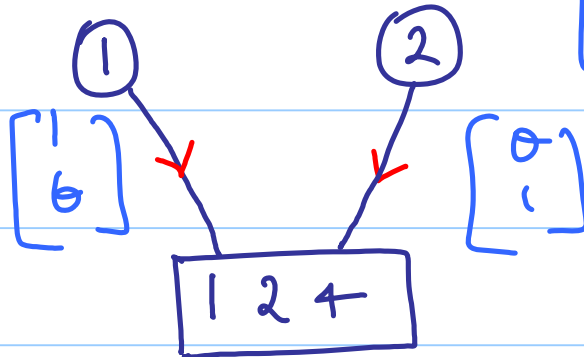
$$\begin{bmatrix} 1 \\ \theta \end{bmatrix}$$



In the case of the single-vertex problem, there is just a single objective fn. to be computed, so one orients all edges in the jn tree towards the corresponding local domain.

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

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



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

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

$$\begin{bmatrix} 2 & 0 \\ 1 + 0^2 \end{bmatrix}$$

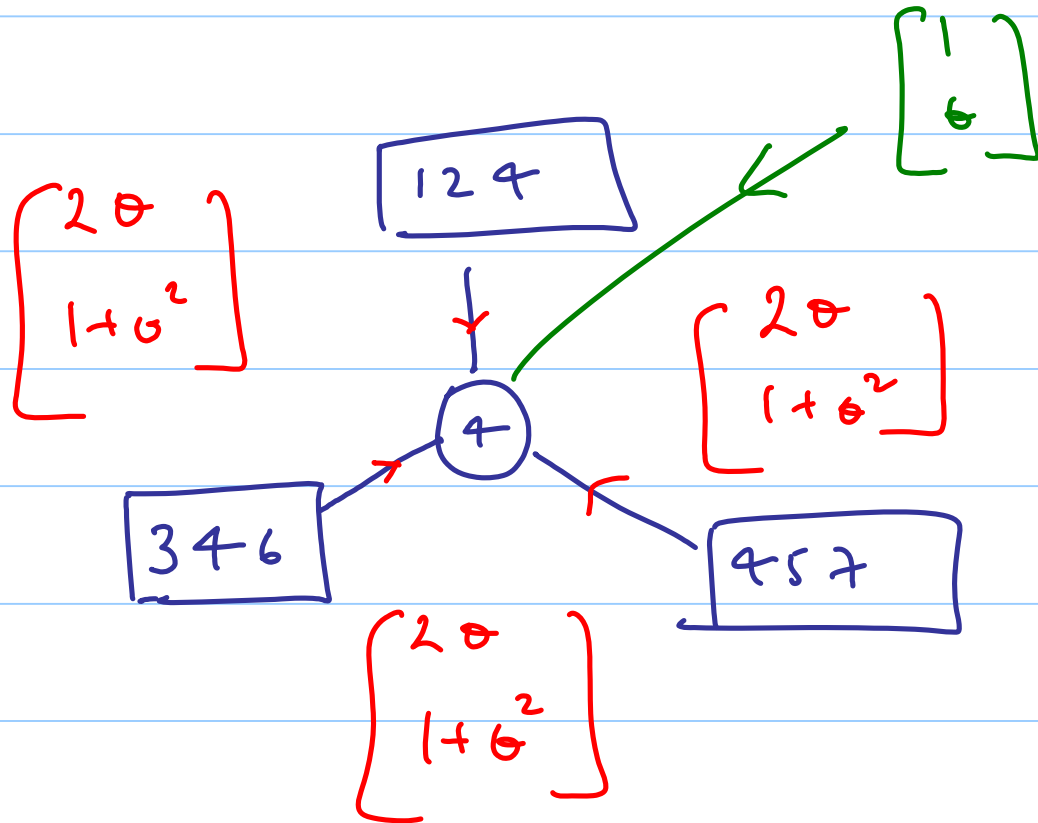
$$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\sum_{x_1 x_2} p(y_1 | x_1) p(y_2 | x_2)$$

$$\chi_{124}(x_1 x_2 x_4) \\ \parallel \\ g(x_4)$$

$$\beta_4(x_4) = \begin{bmatrix} (2\theta)^3 \\ \theta(1+\theta^2)^3 \end{bmatrix} \approx \begin{bmatrix} 8\theta^3 \\ \theta \end{bmatrix}$$

$$\theta = \frac{\epsilon}{1-\epsilon}$$

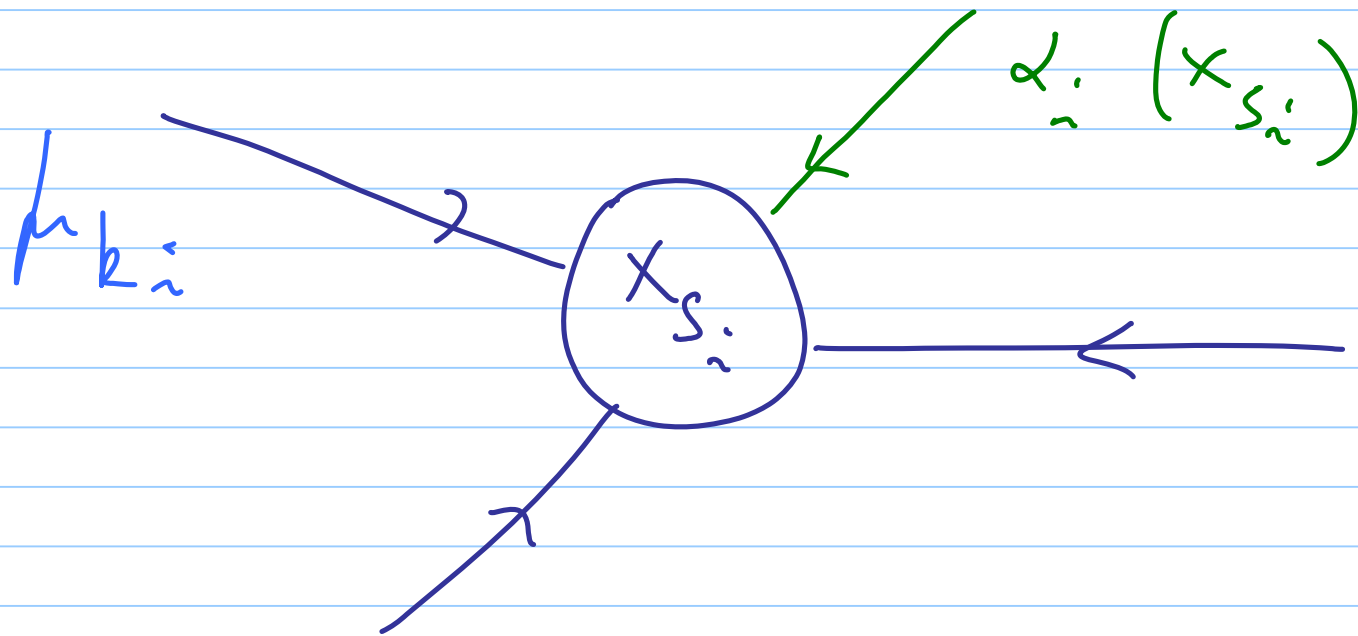


# lec 27 { GDL Approach to Decoding convolutional codes

## Recap

- { final example of jn-tree  
construction
- { message passing
  - Eg  $[7, 4, 2]$  code decoding

A node is ready to compute its  
objective fn. once it has received  
messages from all of its neighbors



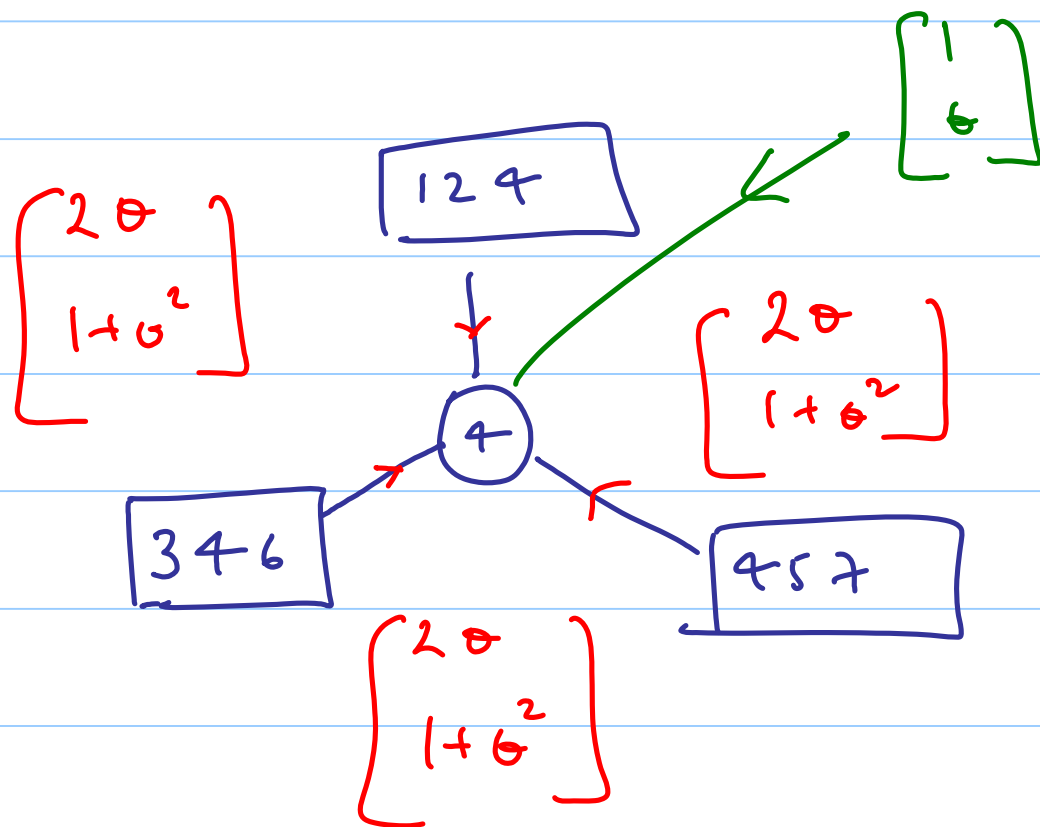


At this stage the node computes its objective fn. simply by computing the product of the incoming messages and the local kernel.

$$\beta_j(x_j) = \alpha_j(x_j) \prod_{k \in N_j} h_{kj}(x_k, x_j)$$

$$\beta_4(x_4) = \begin{bmatrix} (2\theta)^3 \\ \theta(1+\theta^2)^3 \end{bmatrix} \approx \begin{bmatrix} 8\theta^3 \\ \theta \end{bmatrix}$$

$$\theta = \frac{\epsilon}{1-\epsilon}$$



$$\begin{bmatrix} 2\theta \\ 1+\theta^2 \end{bmatrix} \odot \begin{bmatrix} 2\theta \\ 1+\theta^2 \end{bmatrix} \odot \begin{bmatrix} 2\theta \\ 1+\theta^2 \end{bmatrix} \odot \begin{bmatrix} 1 \\ \theta \end{bmatrix}$$

Schur product

$$= \begin{bmatrix} 8\theta^3 \\ \theta(1+\theta^2)^3 \end{bmatrix} \approx \begin{bmatrix} 8\theta^3 \\ \theta \end{bmatrix} \begin{matrix} x_4 = 0 \\ x_4 = 1 \end{matrix}$$

$$p_4(x_4) \propto p(x_4 | \underline{y})$$

{ Since  $\theta \ll 1$  (typically) the ML  
code-symbol decoder will decode  
 $X_q$  to  $= 1$

From the example and the jn tree  
property it is apparent why  
marginalization at intermediate stages  
of message passing as determined by

the GDL is justified.

---

ML codeword decoding of the  $[7, 4, 2]$

---

code using the GDL

---

$$F_i(x_i) = \max_{\sim x_i} \prod_{i=1}^7 \phi(z_i | u_i)$$

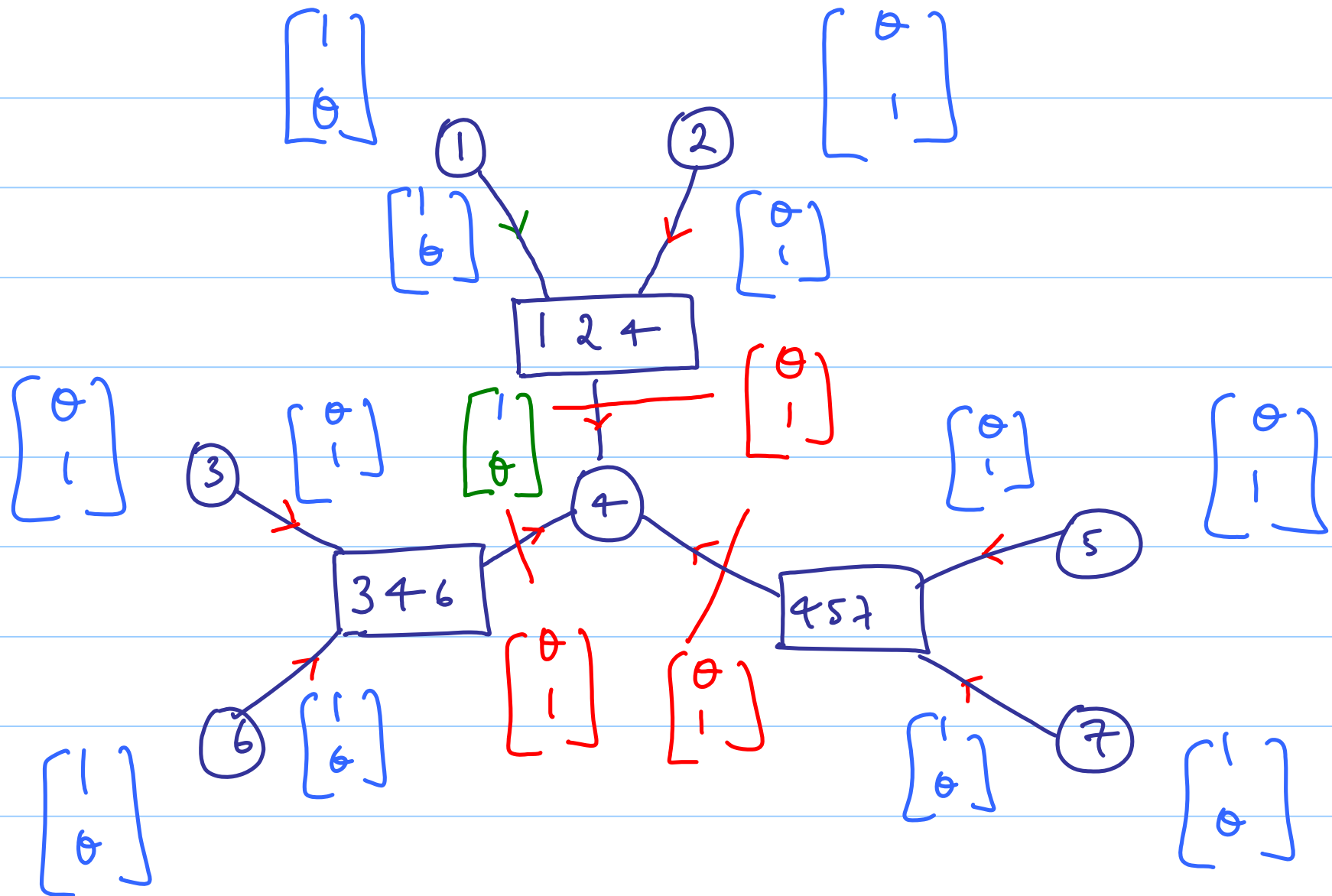
$$\max_{x_1 \dots x_{i-1}} \chi_{124}(x_1 x_2 x_4)$$

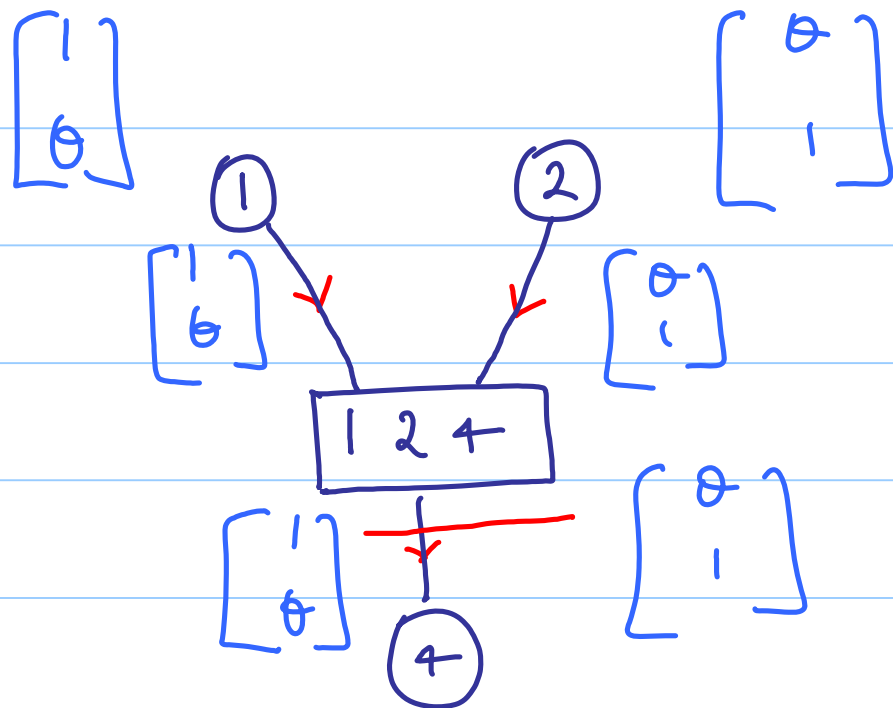
$$x_{i+1} \dots x_7 \quad \chi_{346}(x_3 x_4 x_6)$$

$$\chi_{457}(x_4 x_5 x_7)$$

Ex decode  $[7, 4, 2]$  code  $\underline{z} = (0 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0)$

↑





$$g(x_q) = \max_{x_1, x_2}$$

$$p(y_1/x_1) p(y_2/x_2) \chi_{12q}(x_1 x_2 x_q)$$

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



$$g(0) = \max \{ 1.0, 0.1 \} = 1$$

$$g(1) = \max \{ 1.1, 0.0 \} = 1$$

$$\underbrace{LD}_{\{x_j\}_{j=1}^7}$$

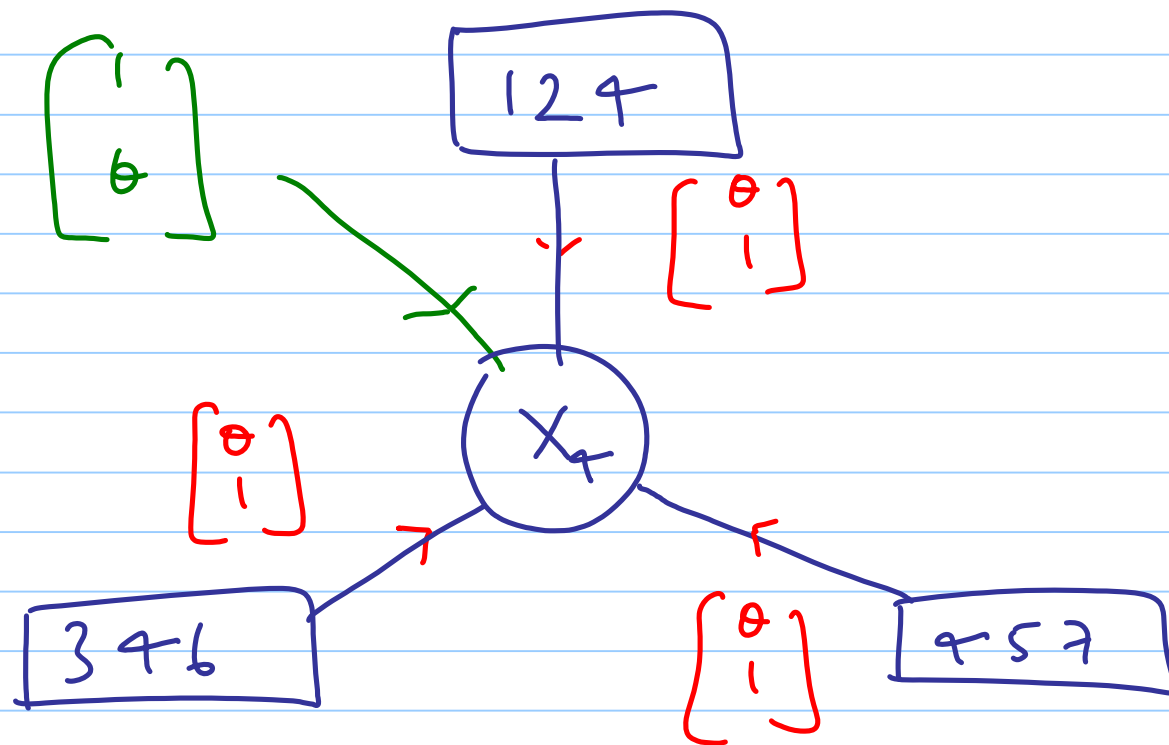
$$\overline{p(j_j | x_j)}$$

$$\{x_1, x_2, x_4\}$$

$$x_{124} (x_1, x_2, x_4)$$

$$\{x_3, x_4, x_6\}$$

$$x_{346} (x_3, x_4, x_6)$$

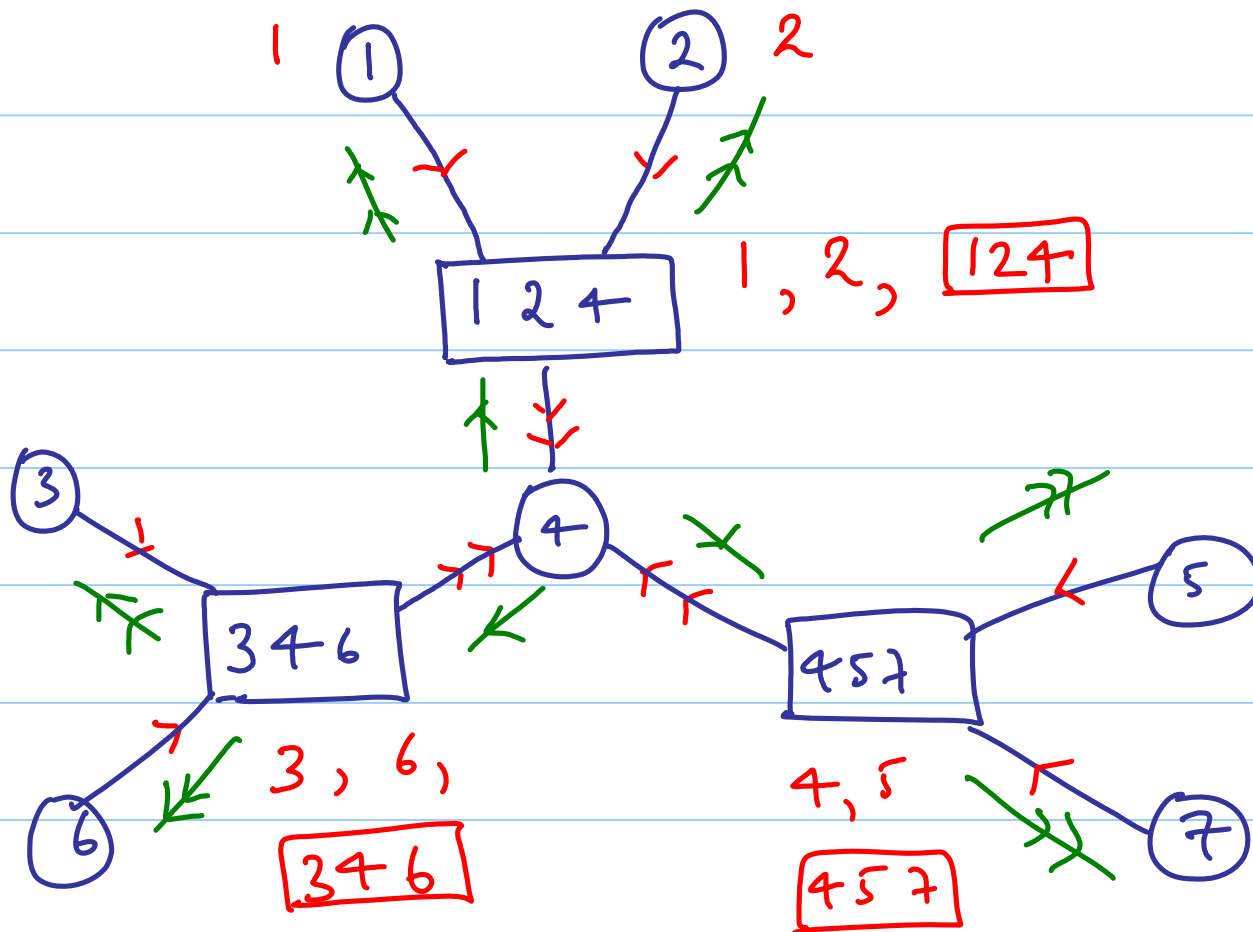


$$\therefore F_q(x_q) = \left[ \begin{array}{l} 0^3 \\ 0 \end{array} \right] \left. \begin{array}{l} x_q = 0 \\ x_q = 1 \end{array} \right\}$$

$\therefore$  Given the ML code word decoder

$\{ \text{decodes } x_q = 1$

---



for more details on the scheduling

please see

The Generalized distributive

Law , S. M. Aji {  
R. J. McEliece

IEEE Trans. Inform. Theory

March 200

Complexity of implementing the GDL:

The complexity for the single-vertex implementation of the GDL

$$= \sum_{\substack{\text{edges} \\ E_{ij}}} \left( l_{s_i} + l_{s_j} - l_{s_i \wedge s_j} \right)$$

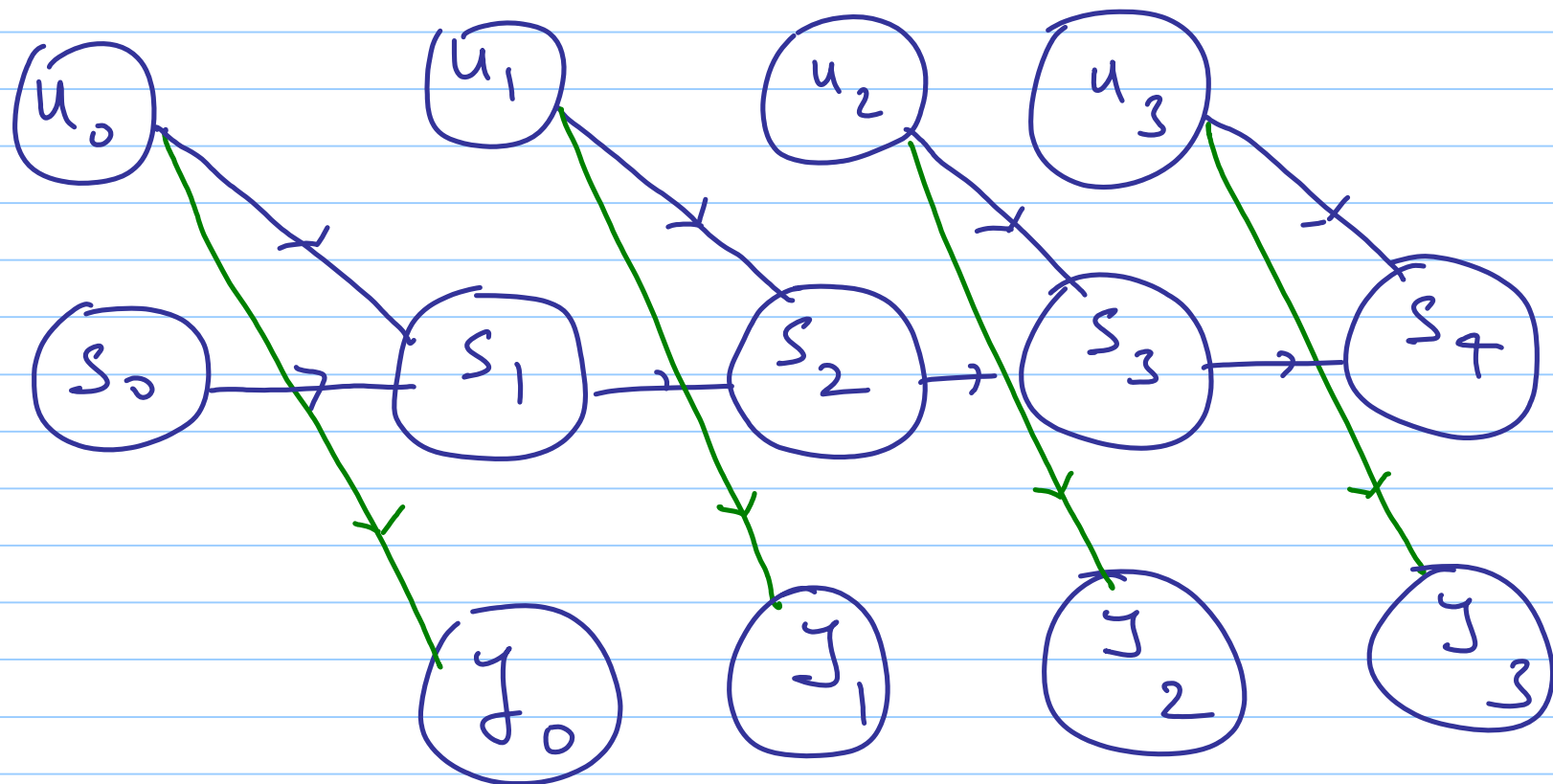
(additions } multiplications)

Turns out that the complexity involved  
in computing the objective fn. at all  
nodes is bdd. above by

$$4 \left( \begin{array}{c} \text{single-vertex} \\ \text{complexity} \end{array} \right).$$

---

# 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.

$$p(u_k | \underline{y}) \propto p(u_k, \underline{y}) = \sum p(\{u_k\}_{k=0}^3, \underline{y})$$

$$= \sum_{\sim u_k} \sum_{\underline{s}} p\left(\{s_k\}_{k=0}^4, \{u_k\}_{k=0}^3, \{y_k\}_{k=0}^3\right)$$

$$= \sum_{u_k} \sum_{\underline{1}} p(s_0) \prod_{k=0}^3 p(u_k) p(s_{k+1} | s_k u_k) \\ p(z_k | s_k u_k)$$

Lec 28 { ML Code - Symbol Decoding  
of the convolutional code  
(BCJR Algorithm)

Recap

\* { formalizing the final step in the  
ADL - computing the objective  
function

Eg Decoding  $[7, 4, 2]$   
(ML code - symbol decoding)

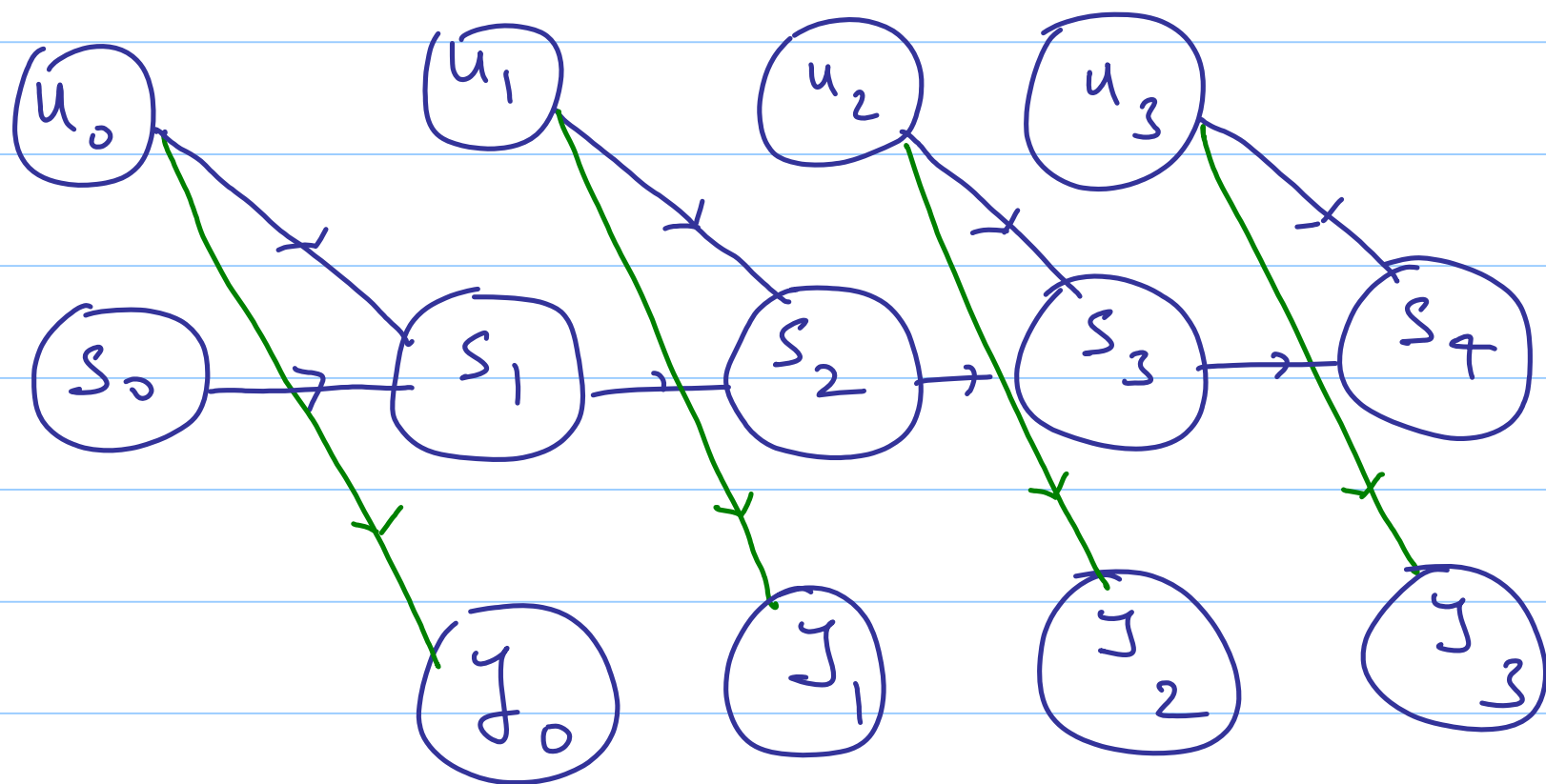
\* {ML code word decoding of the  
    {[7, 4, 2] code

\* {complexity (in terms of the  
    # 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 code-symbol  
 decoding.

$$p(u_k | \underline{y}) \propto p(u_k, \underline{y}) = \sum p(\{u_k\}_{k=0}^3, \underline{y})$$

$$= \sum_{\sim u_k} \sum_{\underline{s}} p\left(\{s_k\}_{k=0}^4, \{u_k\}_{k=0}^3, \{y_k\}_{k=0}^3\right)$$

$$= \sum_{u_k} \sum_{s_0} p(s_0) \prod_{k=0}^3 p(u_k) p(s_{k+1} | s_k u_k) \\ p(y_k | s_k u_k)$$

$$\frac{LD}{\{u_i\}}$$

$$\{s_0\}$$

$$\{s_{i+1} \quad s_i \quad u_i\}$$

$$\{s_i \quad u_i\}$$

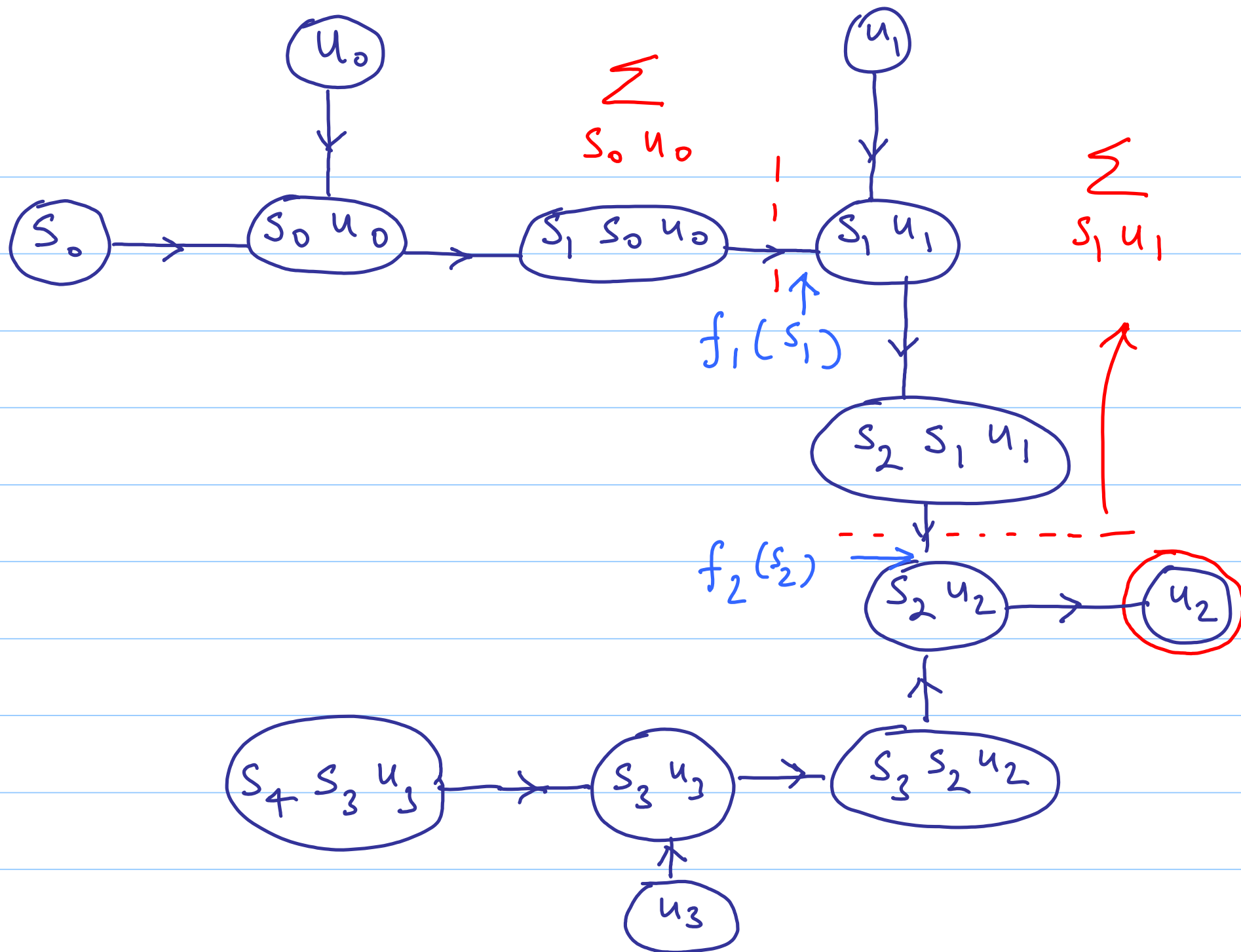
$$\frac{LK}{p(u_i)}$$

$$p(u_i)$$

$$p(s_0)$$

$$p(s_{i+1} | s_i \quad u_i) \quad 0 \leq i \leq 3$$

$$p(y_i | s_i \quad u_i) \quad 0 \leq i \leq 3$$





$$f_1(s_1) = \sum_{s_0, u_0} p(u_0) p(s_0)$$

$$p(y_0 | s_0, u_0) p(s_1 | s_0, u_0)$$

$$f_2(s_2) = \sum_{s_1, u_1} f_1(s_1)$$

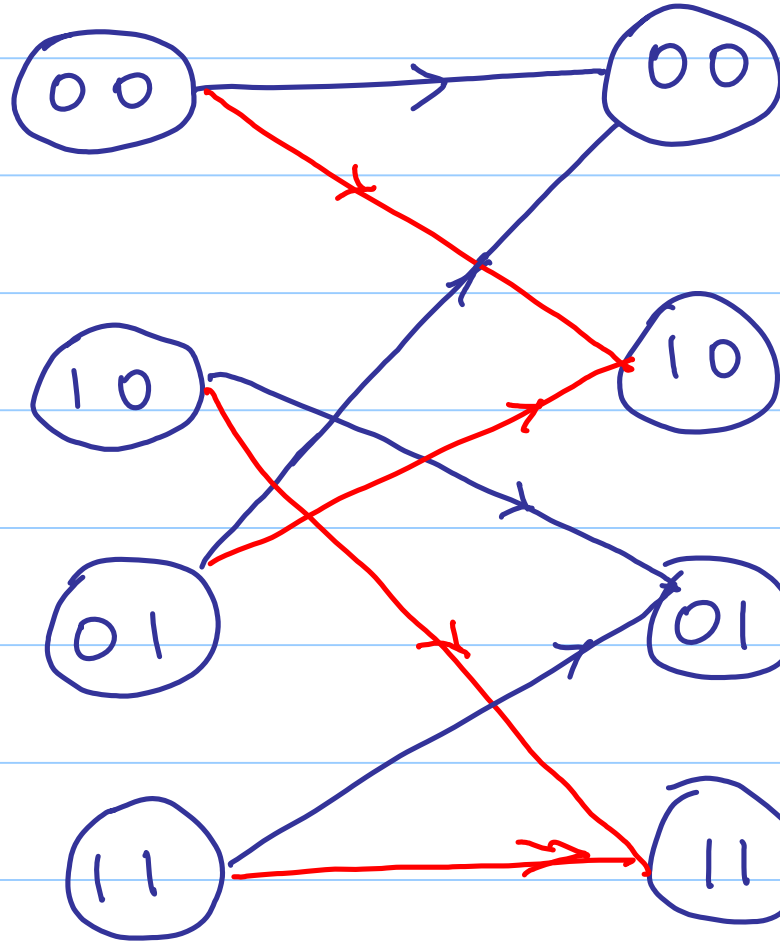
$$p(u_1) p(y_1 | s_1, u_1) p(s_2 | s_1, u_1)$$

generic  
iterative  
step in the  
forward  
direction

$$\underline{f}_g \quad h(D) = \begin{bmatrix} 1 + D + D^2 & 1 + D^2 \end{bmatrix}$$

$s_1$

$s_2$



$f_1(s_1)$

$f_2(s_2)$



$$f_2(s_2) = \sum_{s_1, u_1} f_1(s_1) p(u_1) p(y_1 | s_1, u_1) p(s_2 | s_1, u_1)$$

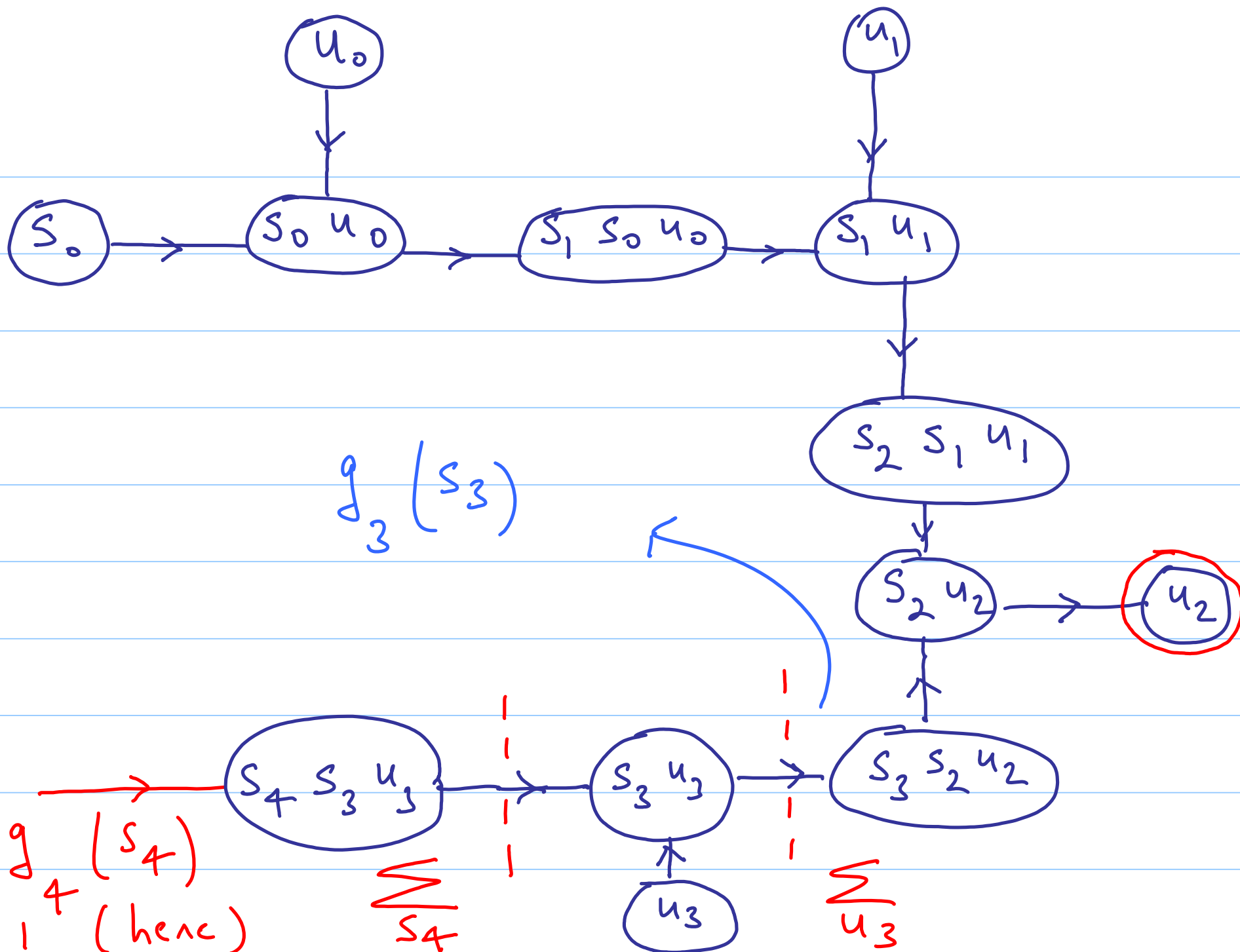
$$f_2(s_2) = \sum_{s_1} f_1(s_1) \left\{ \sum_{u_1} \overset{\uparrow \uparrow = 1/2}{p(u_1) \cdot p(y_1 | s_1, u_1) \cdot p(s_2 | s_1, u_1)} \right\}$$

$\nwarrow$   
 $\uparrow(s_2, s_1)$

$\Downarrow$   
 is  $\{0, 1\}$ -valued  
 fn.

$\uparrow (s_2, s_1)$  $f_1(s_1)$  $f_2(s_2)$ 

$$\begin{bmatrix} f_2(00) \\ f_2(10) \\ f_2(01) \\ f_2(11) \end{bmatrix} = \begin{bmatrix} * & 0 & * & 0 \\ * & 0 & * & 0 \\ 0 & * & 0 & * \\ 0 & * & 0 & * \end{bmatrix} \begin{bmatrix} f_1(00) \\ f_1(10) \\ f_1(01) \\ f_1(11) \end{bmatrix}$$



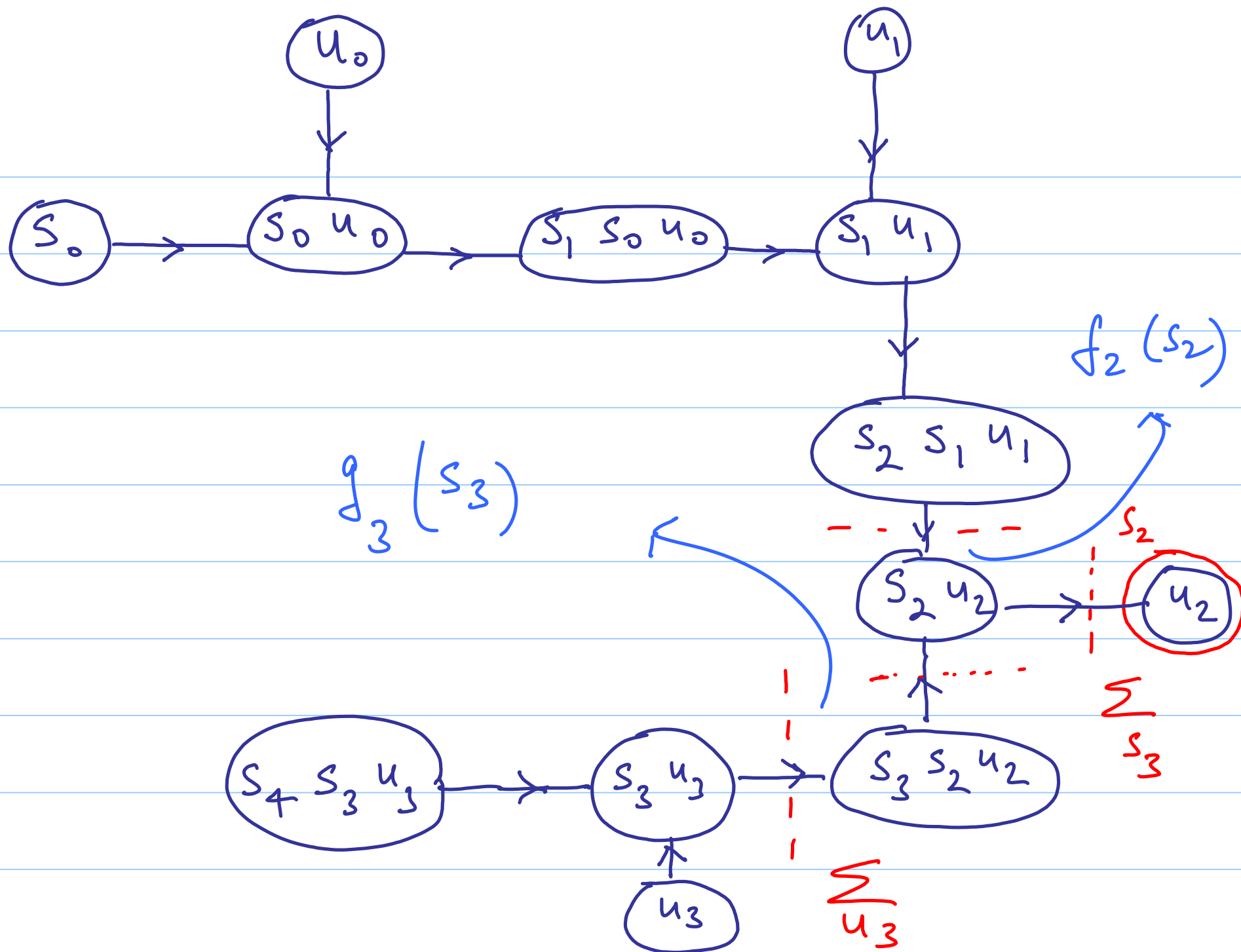
$$g_3(s_3) = \sum_{u_3} p(y_3 | s_3, u_3) p(u_3) \cdot \sum_{s_4} p(s_4 | s_3, u_3) g_4(s_4)$$

$$= \sum_{s_4} g_4(s_4) \left\{ \sum_{u_3} p(u_3) \cdot p(y_3 | s_3, u_3) \cdot p(s_4 | s_3, u_3) \right\}$$

$\Delta(s_3, s_4)$

$$g_3(s_3) = \Delta(s_3, s_4) g_4(s_4)$$

Conclusion: { both forward and  
backward recursions  
correspond to  $m \times$   
multiplication





$$p(u_k | \underline{y}) = p(u_2) \sum_{s_2} p(y_2 | s_2 u_2) f_2(s_2) \\ \sum_{s_3} g_3(s_3) p(s_3 | s_2 u_2)$$

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

$$\frac{k}{n} = R, \quad 0 < R < 1$$

$$\text{and } \frac{d_{\min}}{n} = \delta, \quad 0 < \delta < 1.$$

$$H = \begin{bmatrix} \phantom{0} \end{bmatrix}$$

$(n-k \times n)$

Thus in a typical p.c. mx,

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 # 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  $\mathcal{C}^\perp$ . This can be used to show that once again this implies that the nullspace of  $H$  is precisely the original code  $\mathcal{C}$ .

A  $(d_r, d_c)$ -regular LDPC code

is one in which each row of  $H$  has  $d_c$  1's and each column of  $H$  contains  $d_v$  1's -  $d_v$  1's

$$H = \begin{bmatrix} & & & & \\ & & & & \\ & & & & \\ & & & & \\ & & & & \end{bmatrix} \quad \begin{matrix} \nearrow \\ d_c \text{ 1's} \end{matrix}$$

(m \times n)

$$\therefore m d_c$$

$$= n d_v$$

$$\therefore \boxed{\frac{m}{n} = \frac{d_v}{d_c}}$$

$$\text{rk}(H) \leq m$$

(2)

$$\therefore \dim(\varphi(H)) \geq n-m$$

$$\therefore \dim(\mathcal{L}) \geq n-m$$

$$\therefore k \geq n-m$$

$$\therefore \boxed{\frac{k}{n} \geq 1 - \frac{m}{n}} \quad \dots (1)$$

$$\therefore \boxed{\frac{k}{n} \geq 1 - \frac{d_u}{d_L}} \quad (3)$$

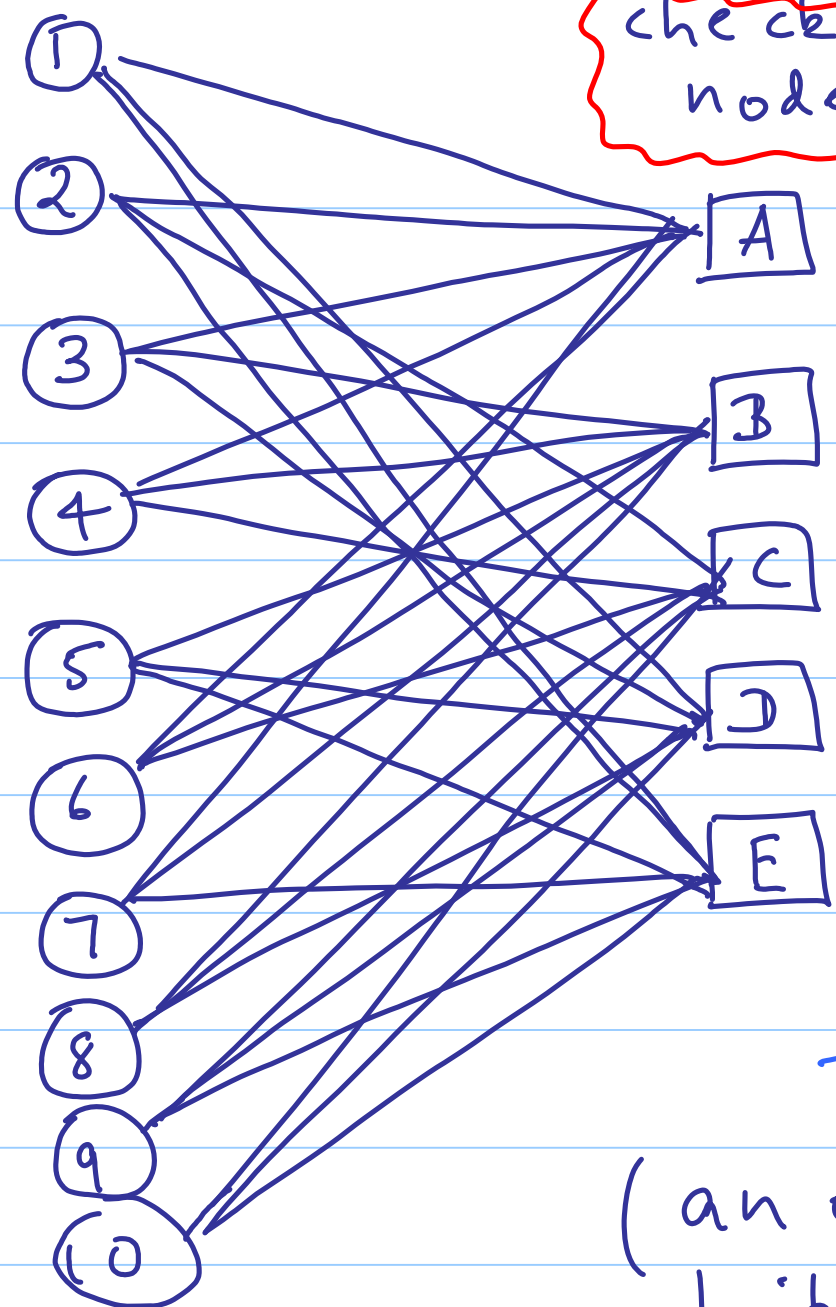
follows from  
(1) and (2)

variable nodes

check nodes

degree on the right = 6

degree of each node on the left = 3



1 2 3 4 6 7

3 4 5 6 7 8

2 4 6 8 9 10

1 3 5 8 9 10

1 2 5 7 9 10

Tanner graph  
(an example of a bipartite graph)

$H =$

1	2	3	4	5	6	7	8	9	10	
1	1	1	1	0	1	1	0	0	0	A
0										B
0										C
1										D
1										E

(5 × 10)

this is an instance of a ( $d_v = 3, d_c = 6$ )  
 - regular code

$$\therefore \frac{k}{n} \geq 1 - \frac{d_v}{d_c} = 1 - \frac{3}{6} = \frac{1}{2}$$

$$\therefore \boxed{\text{rate}(R) \geq \frac{1}{2}} \quad \begin{array}{l} \text{designed} \\ \text{rate} \end{array}$$

Note that the  $(d_v, d_c)$  constraint implies that the total # of is in

$$\text{the p.c. } mx = nd_v = md_c$$

and hence is of order  $n$

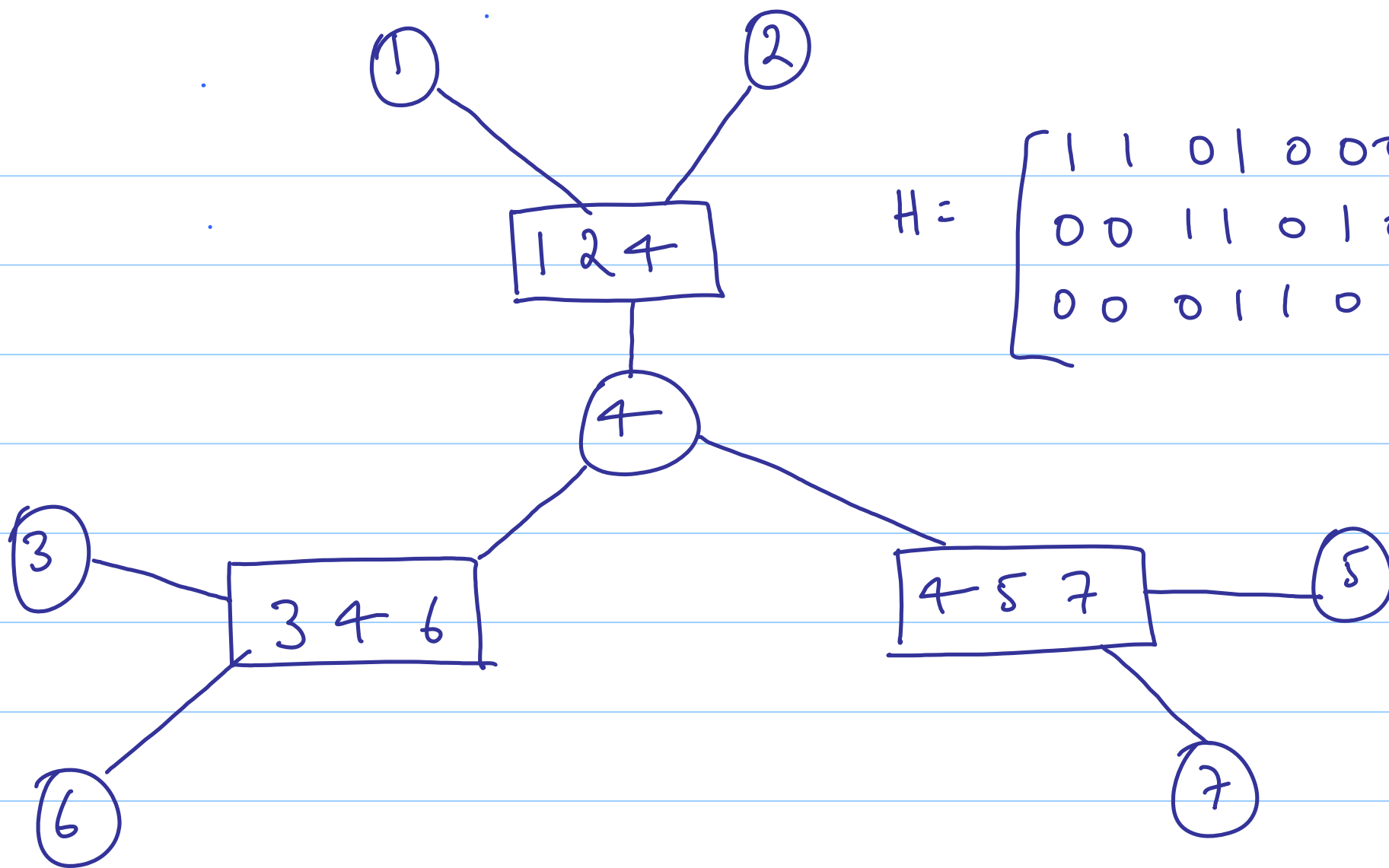
(since  $(d_v, d_c)$  are fixed and

independent of  $n$ )

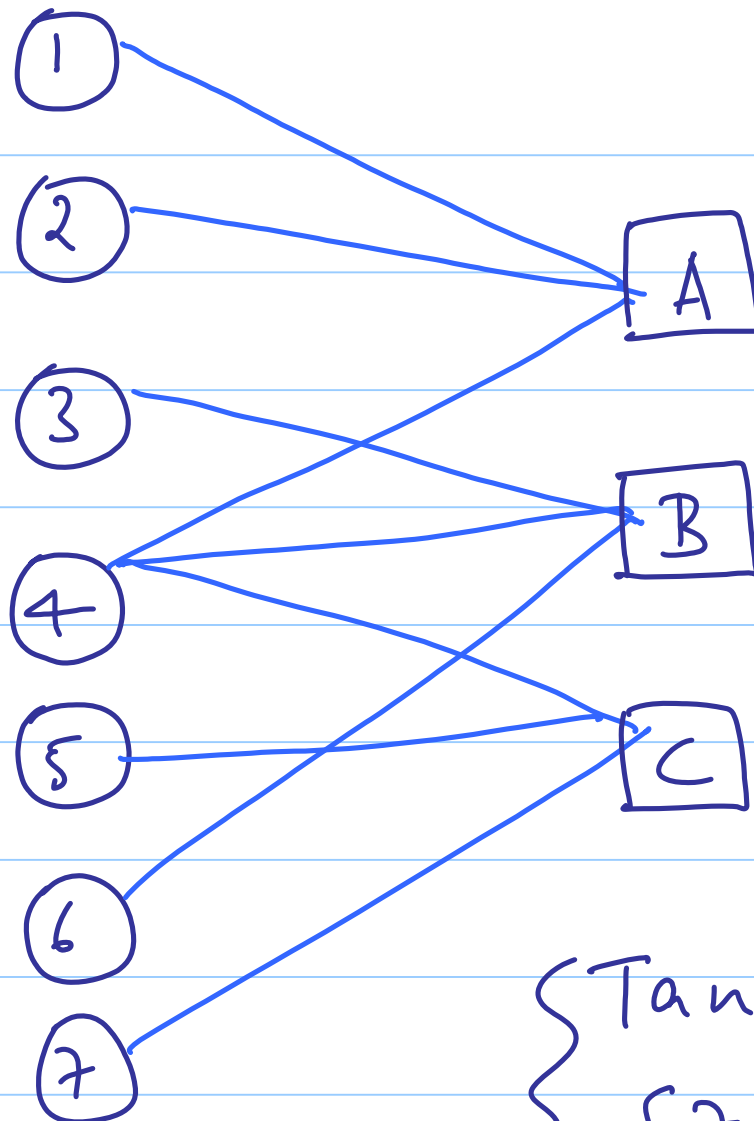
The graphical, message-passing nature of the decoding algorithm in the case of an LDPC code implies that the complexity of the decoding algorithm is proportional to the # of edges in the Tanner

graph of the code and hence is  
linear (!!!) in the block length  
of the code.





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



1 2 4

3 4 6

4 5 7

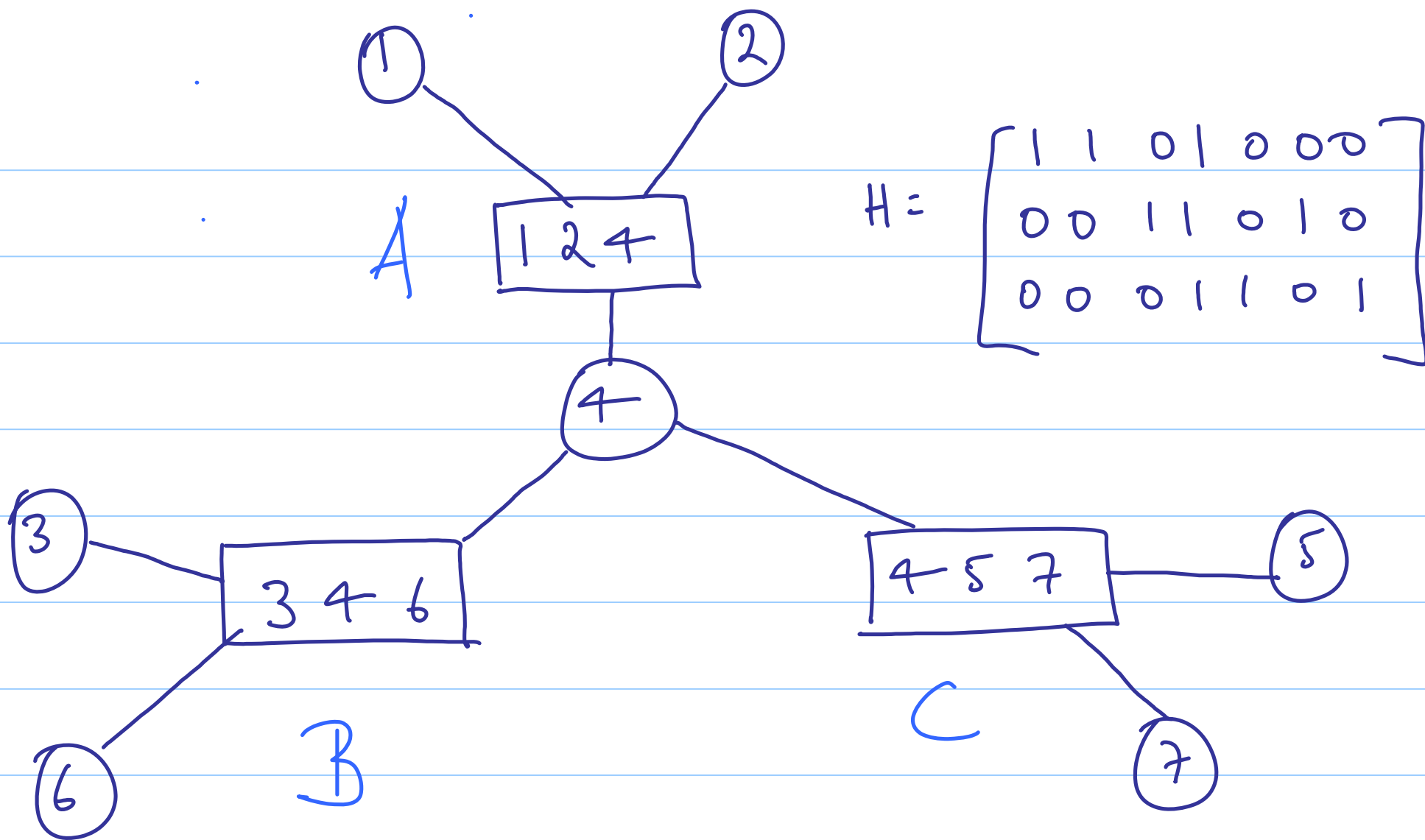
note: this

code is  
not

$(d_v, d_c)$

regular

{ Tanner graph of the  
[7, 4, 2] code



Lec 30

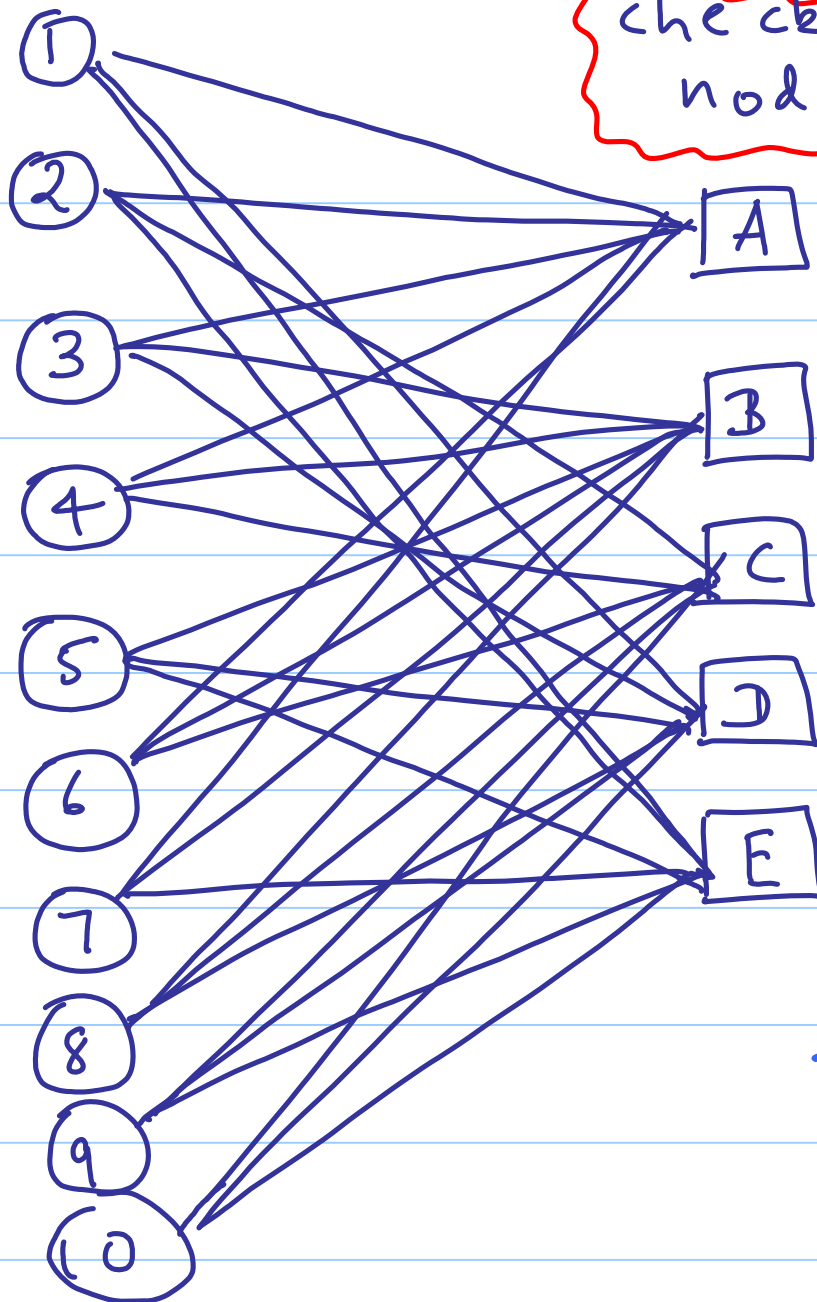
## LDPC Code Terminology

### Recap

- \* introduced LDPC codes
  - $(d_v, d_c)$  regular code
  - rate
  - Tanner graph
  - decoding complexity

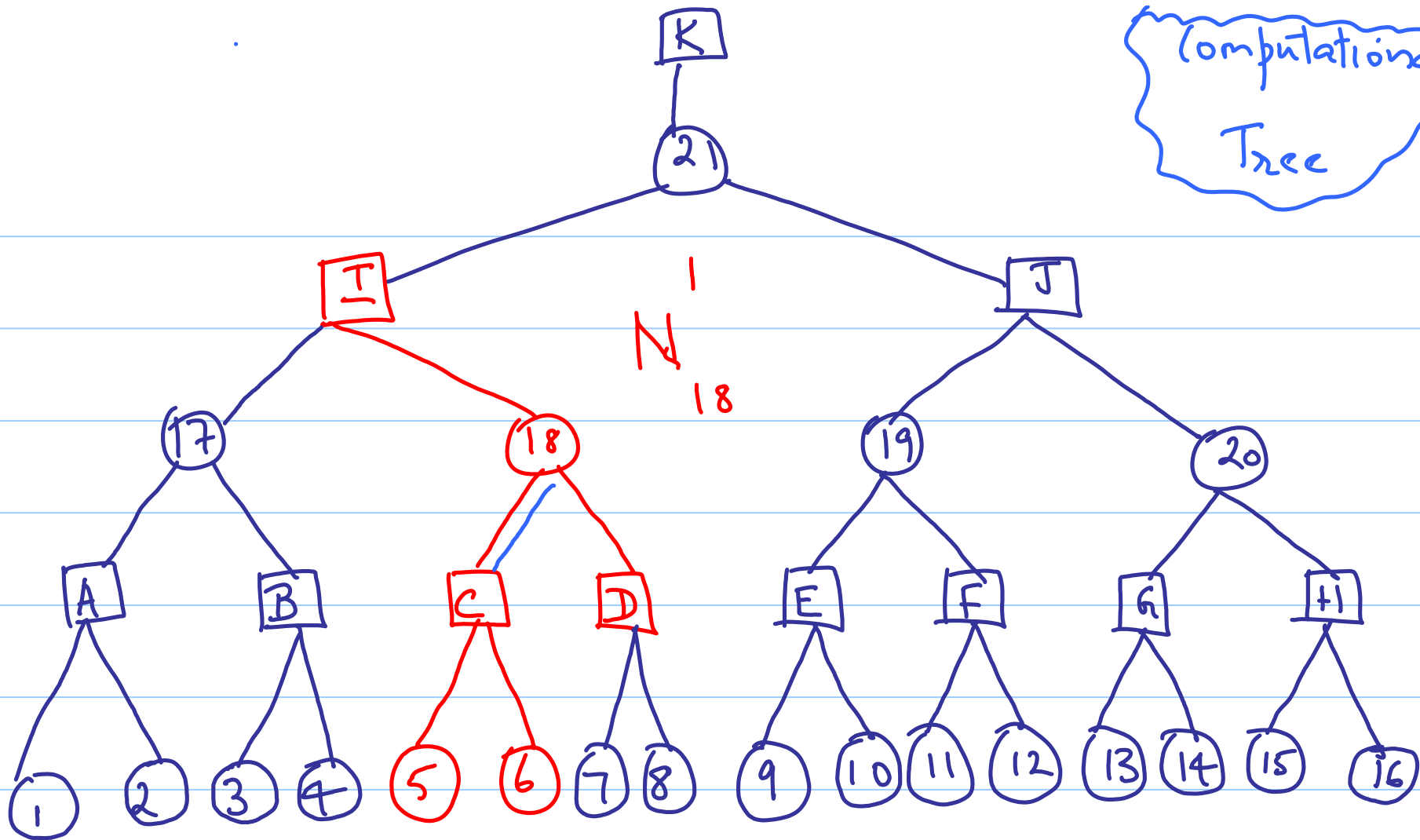
variable  
nodes

check  
nodes

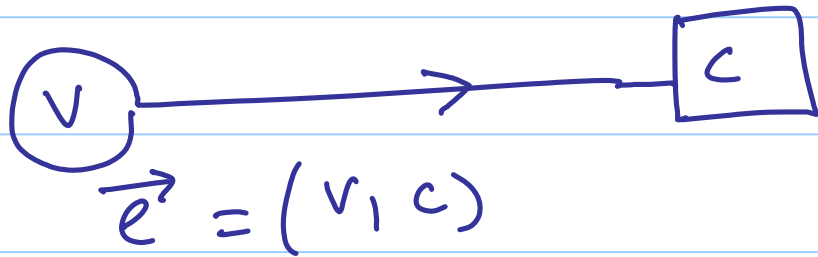


Tanner graph

Computational  
Tree



# Edges, Paths & Neighborhoods



A path in the graph (Tanner graph) is a directed sequence of directed edges

$\vec{e}_1 \rightarrow \vec{e}_2 \rightarrow \dots \rightarrow \vec{e}_k$  s.t. if

$\vec{e}_i = (u_i, u'_i)$ , then  $\vec{e}_{i+1} = (u_{i+1}, u'_{i+1})$

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

distance  $d$  if they are connected by a path of length  $d$ , but not by a



path of length  $< d$ .

$N_u^d$  = nbhd of node  $u$  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

$$u_1 \in N_{u_2}^d \Leftrightarrow u_2 \in N_{u_1}^d$$

If  $\vec{e} = (v, c)$  then the undirected

nbhd to depth  $d$  of  $\vec{e} = N_v^d \cup N_c^d$

$$= N_c^d$$

$$e \in N_{e'}^d \Leftrightarrow e' \in N_e^d$$

The directed nbhd to depth  $d$  of  
edge  $\vec{e} = (v, c)$  denoted by  $N_{\vec{e}}^d$

= the induced subgraph containing all

edges and nodes on paths  $\vec{e}_1, \vec{e}_2, \dots$

$\vec{e}_d$  starting from  $v$ , but with

$\vec{e}_1 \neq \vec{e}'$ .

# Channel Models

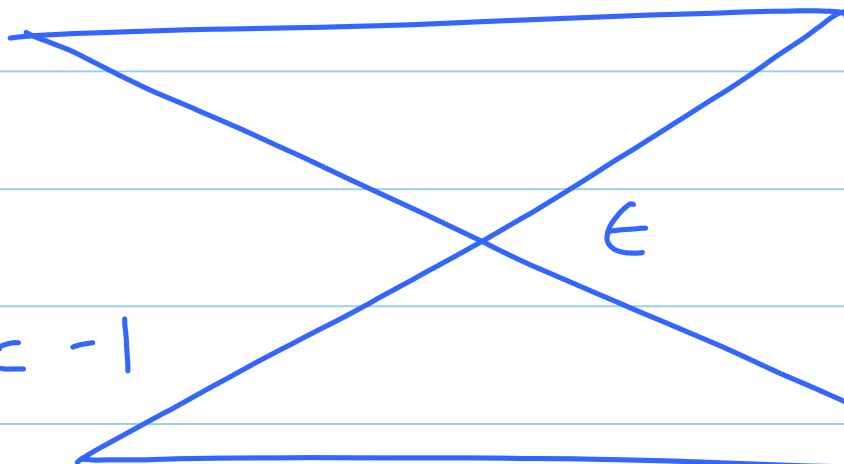
BSC

$$x_t = 1$$

$$1 - \epsilon$$

$$y_t$$

$$1$$



$$x_t = -1$$

$$1 - \epsilon$$

$$-1$$

$x_t = \pm 1$   
 $u_t \in \{0, 1\}$

$$y_t = x_t z_t$$

$$z_t \in \{\pm 1\}$$

$$p_{z_t}(-1) = \epsilon$$

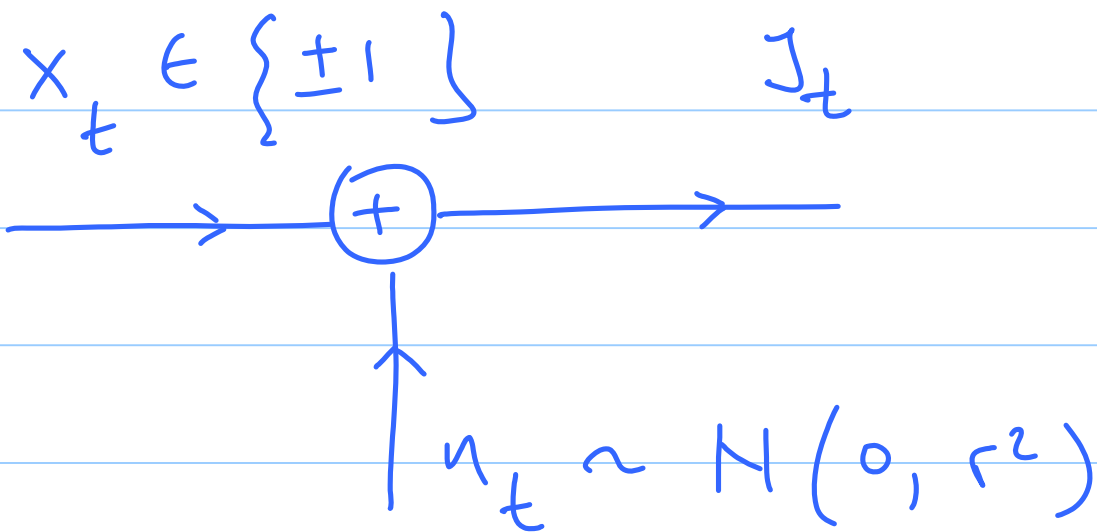
$$p_{z_t}(+1) = 1 - \epsilon$$

$$p_{Y_t | X_t} (y | x) = p_{Z_t} \left( \frac{y}{x} \right)$$

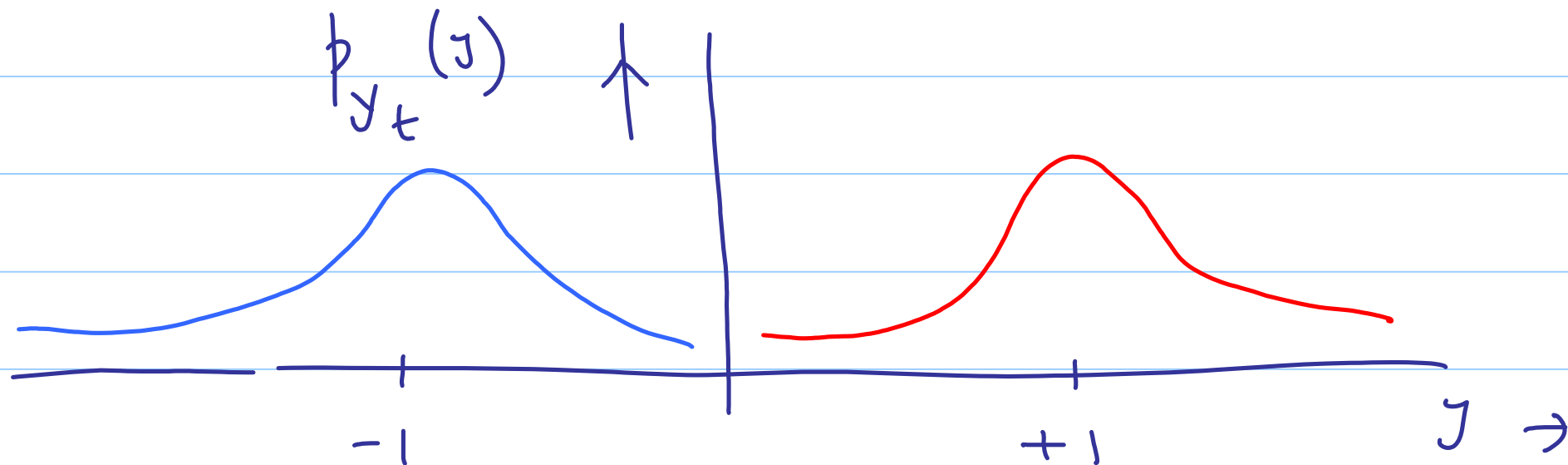
$$= p_{Y_t | X_t} (-y | -x)$$

known as the channel symmetry condition  
(as applied to the BSC)

# Binary - Input AWGN Channel



$$y_t = x_t + n_t$$



Can write:

$$y_t = x_t z_t$$

$$p_{y_t | x_t}(y | x)$$

$$= p_{z_t}\left(\frac{y}{x}\right)$$

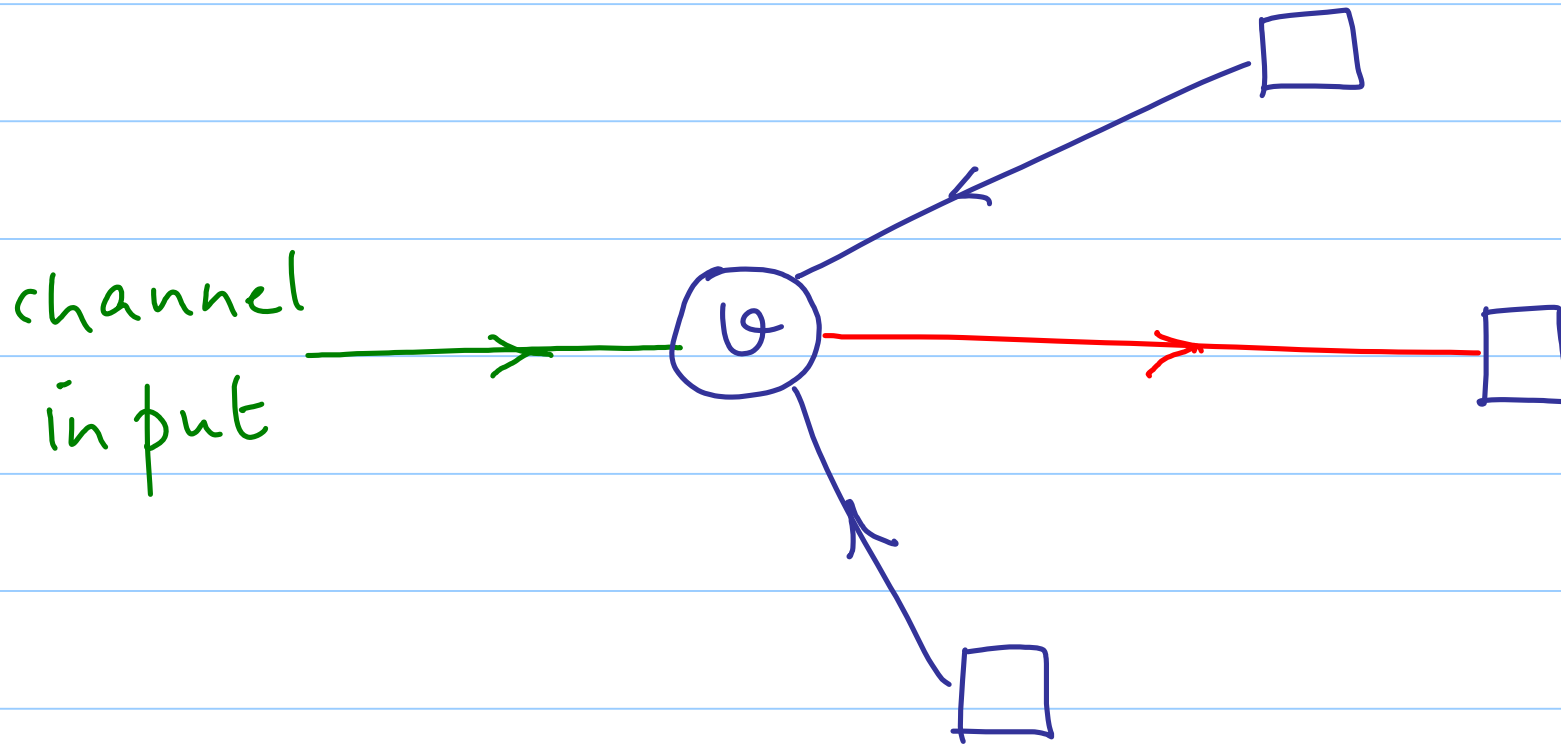
$$z_t \in N(1, \sigma^2)$$

$z_t$  independent of  $x_t$

$$= p_{y_t | x_t}(-y | -x) = p_{z_t}\left(\frac{-y}{-x}\right) = p_{z_t}\left(\frac{y}{x}\right)$$

(called the channel symmetry condition)

## Message-Passing Terminology





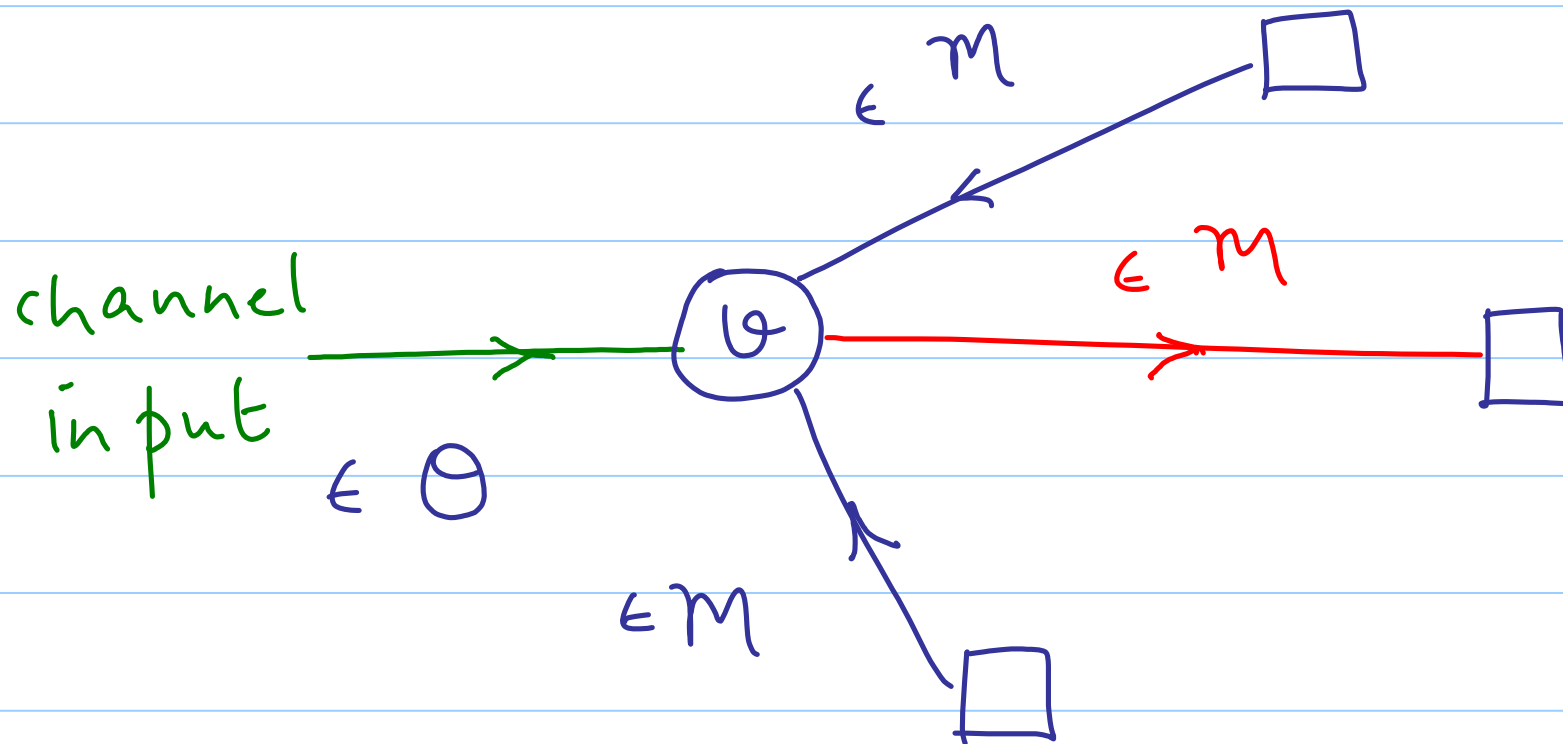
# Lec 31 Gallager Decoding Algorithm

## Recap

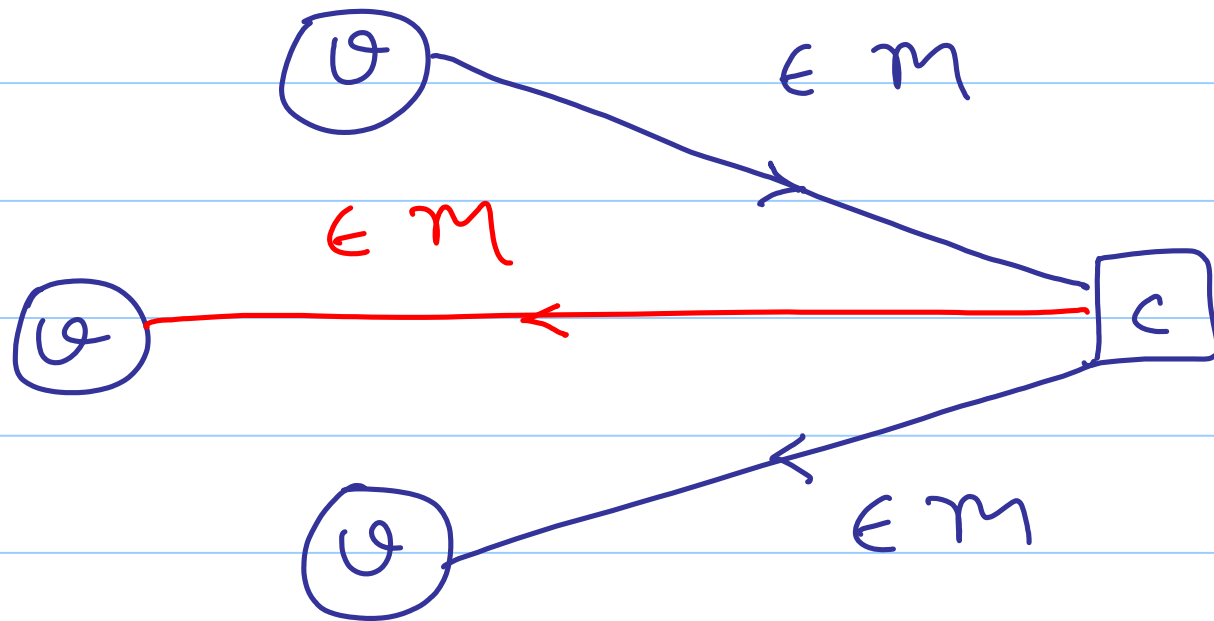
- \* {terminology relating to LDPC  
codes:
  - edges, paths, nbhds
  - channel models

(called the channel symmetry condition)

## Message-Passing Terminology



$\Theta$  = output alphabet of the channel  
 $\mathcal{M}$  = the common alphabet employed to  
pass messages from variable node  
to check nodes or vice-versa.



$$\psi_v^{(0)} : \Theta \rightarrow \mathcal{M} \quad \left\{ \begin{array}{l} \text{the initial} \\ \text{message map} \end{array} \right.$$

$$\psi_c^{(e)} : \mathcal{M}^{d_c-1} \rightarrow \mathcal{M}$$

$l = \#$  of  
the  
iteration

$$\psi_v^{(e)} : \Theta \times \mathcal{M}^{d_v-1} \rightarrow \mathcal{M}$$

Assumptions concerning message  
 passing at a variable / check node:

$$\psi_u^{(0)}(bm) = \left[ \psi_u^{(0)}(m) \right] b, \quad b \in \{\pm 1\}$$

(1)  $m \in \mathcal{M}$

$$\psi_u^{(2)}(bm_0, bm_1, \dots, bm_{d_v-1})$$

$$= b \psi_u^{(2)}(m_0, m_1, \dots, m_{d_v-1})$$

(2)  $b \in \{\pm 1\}$

$$\psi_c^{(e)}(b_1^{m_1}, b_2^{m_2}, \dots, b_{d_c-1}^{m_{d_c-1}})$$

$$= \left[ \prod_{j=1}^{d_c-1} b_j \right] \psi_c^{(e)}(m_1, m_2, \dots, m_{d_c-1}) \quad (3)$$

$$b_j \in \{\pm 1\}$$

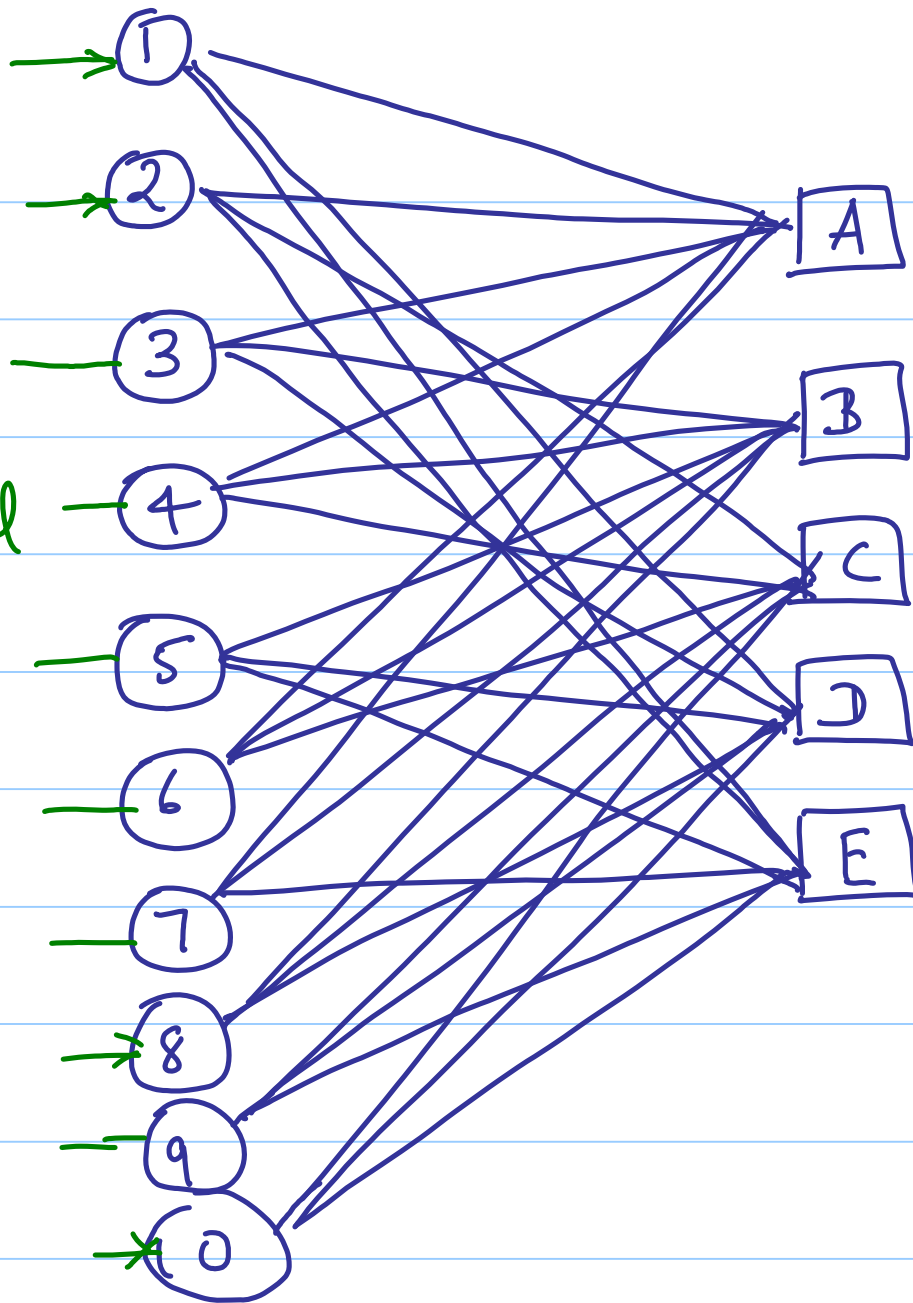
$$m_j \in \mathcal{M}.$$

Equations (1) - (3) are known as the

variable and check-node symmetry

conditions.

channel  
input



we will use  
the sign of  
the message  
 $v \rightarrow c$   
as an  
indication  
as to the  
value  $f(v)$ .

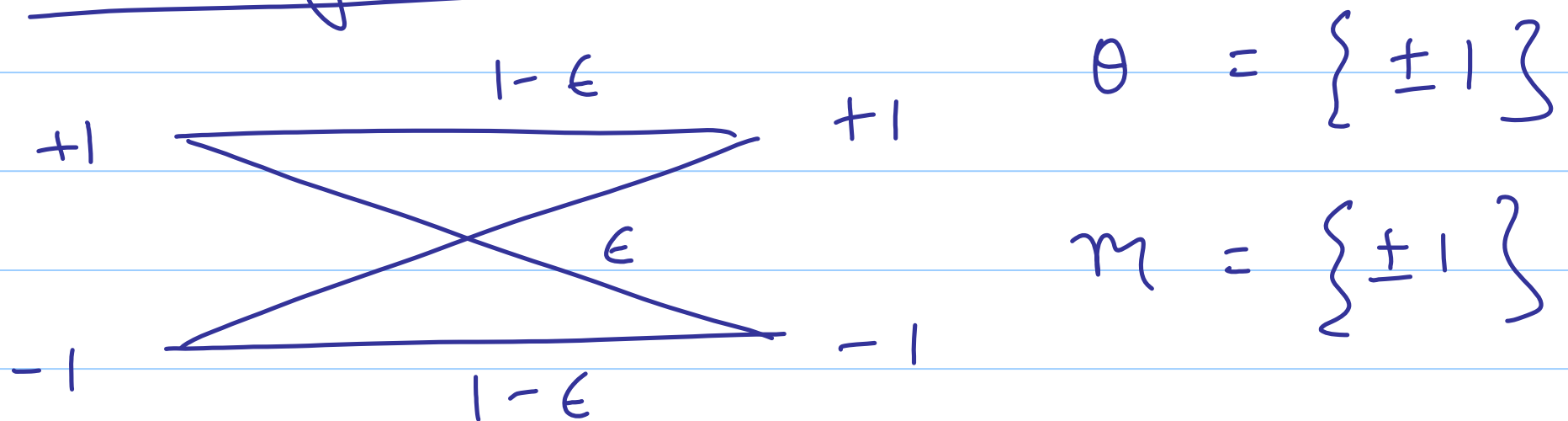


{ the message along an edge is said to be  
in error if its sign is not the true sign  
of the associated code symbol.

{ We will now proceed to show that the  
# of incorrect messages passed  
along the edges of the Tanner graph  
during each iteration is independent

[of the transmitted codeword.  
(deferred to Lec 32)]

## Gallager Decoding Algorithm A



$$\psi_{(0)}^{(m)} = m$$

$$\psi_v^{(2)}(m_0, m_1, \dots, m_{d_v-1})$$

$$= \begin{cases} -m_0 & \text{if } m_j = -m_0 \text{ all } 1 \leq j \leq d_v-1 \\ m_0 & \text{else} \end{cases}$$

$$\psi_c^{(2)}(m_1, m_2, \dots, m_{d_c-1}) = \prod_{j=1}^{d_c-1} m_j$$

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 1 codeword was transmitted.

$$p_1^{(0)} = \begin{cases} \text{prob that the message passed by} \\ \text{a variable node during the } 0^{\text{th}} \\ \text{iteration} = +1 \end{cases}$$

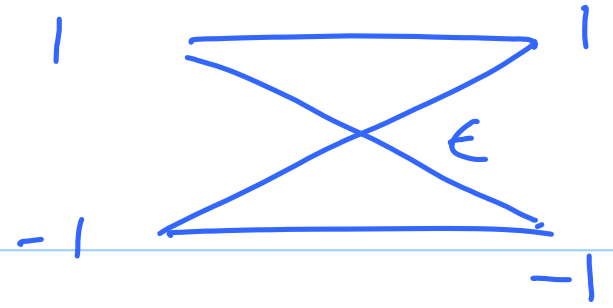
$$p_{-1}^{(0)} = \text{similarly for } -1$$

$$q_1^{(e)} = \begin{cases} \text{prob that on the } e^{\text{th}} \text{ iteration} \\ \text{the check node message} = 1 \end{cases}$$

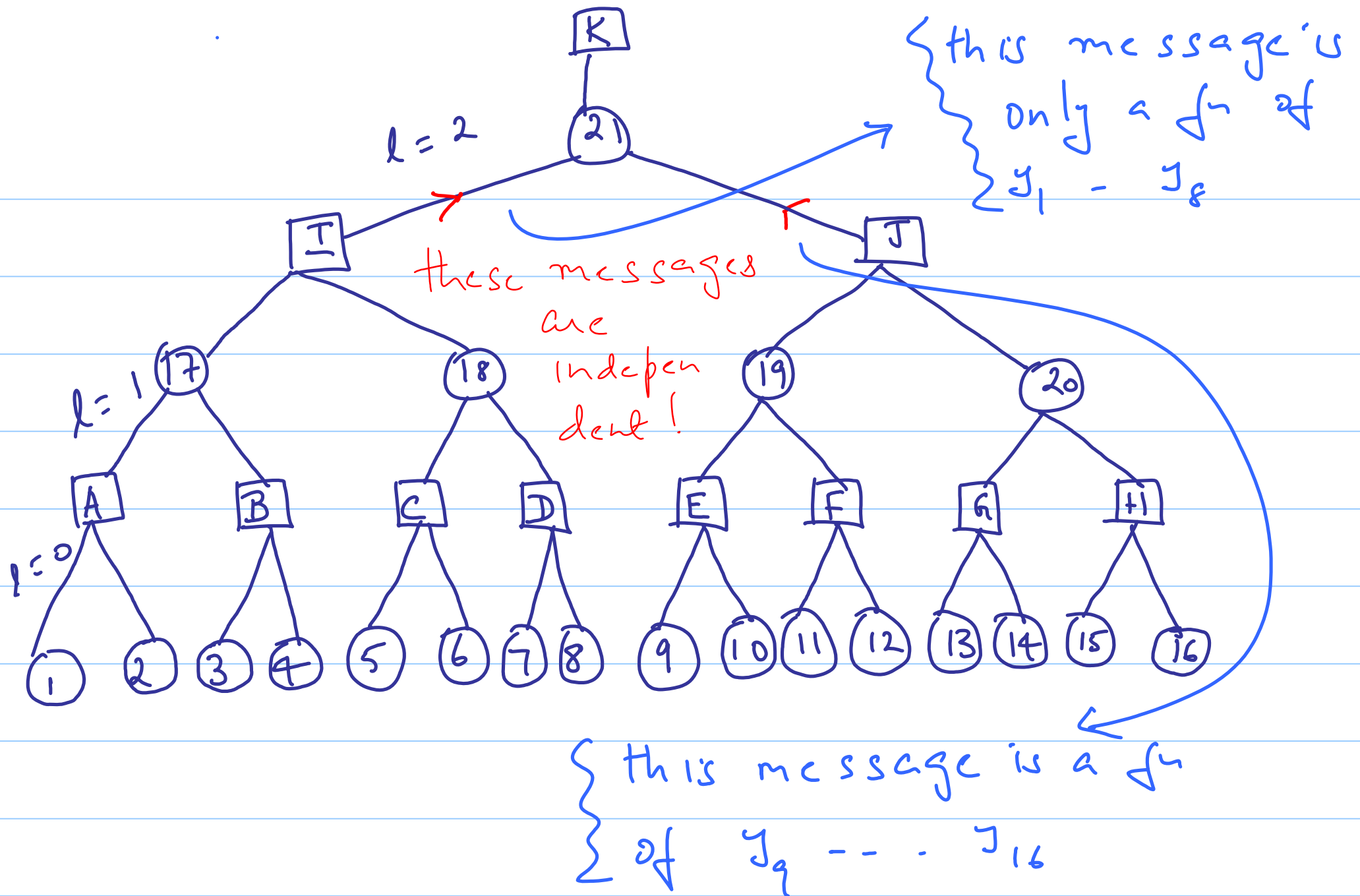
$$q_{-1}^{(e)} = \text{similarly for } -1$$

$$p_{11}^{(0)} = 1 - \epsilon$$

$$p_{-1-1}^{(0)} = \epsilon$$



note that by symmetry the probability mass fn of messages along an edge only depends upon the number of the iteration (i.e. on  $l$ ) as well as the direction of the message (variable node to check node or vice-versa).



$$p_1^{(0)} = 1 - \epsilon$$

$$p_{-1}^{(0)} = \epsilon$$

$$p_{-1}^{(L)} = p_{-1}^{(0)} \left\{ 1 - (1 - q_{-1}^{(L)}) d_{-1}^{-1} \right\} + p_1^{(0)} [q_{-1}^{(L)}] d_{-1}^{-1}$$

represents the probability that not all incoming messages are  $= +1$

(uses independence)



# Lec 32 {BP Decoding of LDPC Codes

thm

A

(BP  $\equiv$  Belief Propagation)

## Recap

C

- \* {Alphabets arising in  
message passing
- \* independence assumptions
- \* {Gallager decoding algorithm  
A

$$\therefore h_{d_c} = \prod_{j=1}^{d_c-1} h_j$$

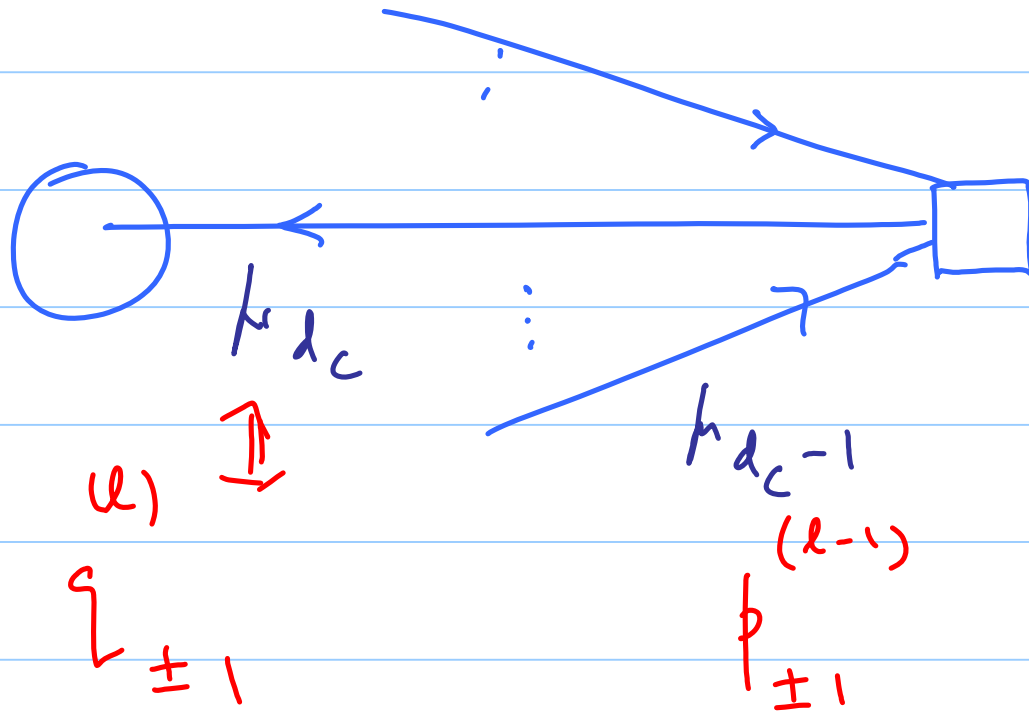
$$h_j \in \{\pm 1\}$$

$(l-1)$

$$h_1 \Leftrightarrow$$

$$p_{\pm 1}$$

$u_i$   
-1)



$$\therefore \boxed{\mu_{d_c} = \prod_{j=1}^{d_c-1} \mu_j}$$

$$\therefore \mathbb{E} \{ \mu_{d_c} \} = \prod_{j=1}^{d_c-1} \mathbb{E} \{ \mu_j \} \quad \dots (1)$$

$$\begin{cases} P_n(\mu_j = 1) = p_1^{(l-1)} \\ P_n(\mu_j = -1) = p_{-1}^{(l-1)} \end{cases}$$

$\therefore$  (1) gives us that

$$\begin{bmatrix} q_{-1}^{(x)} - q_{-1}^{(x)} \end{bmatrix} = \begin{bmatrix} p_{-1}^{(x-1)} - p_{-1}^{(x-1)} \end{bmatrix} d_c - 1$$

$$\therefore 1 - 2 q_{-1}^{(x)} = \begin{bmatrix} 1 - 2 p_{-1}^{(x-1)} \end{bmatrix} d_c - 1$$

$$\therefore q_{-1}^{(x)} = \frac{1}{2} \left\{ 1 - \begin{bmatrix} 1 - 2 p_{-1}^{(x-1)} \end{bmatrix} d_c - 1 \right\}$$

.... (2)

$$p_{-1}^{(2)} = p_{-1}^{(0)} \left\{ 1 - (1 - q_{-1}^{(2)}) d_{-1}^{-1} \right\} + p_{-1}^{(0)} [q_{-1}^{(2)}] d_{-1}^{-1}$$

- (3)

From (2) and (3) we get:

$$p_{-1}^{(l)} = p_{-1}^{(0)} \left\{ 1 - \left[ \frac{1}{2} \left\{ 1 + \left[ 1 - 2 p_{-1}^{(l-1)} \right]^{d_c-1} \right\} \right]^{d_v-1} \right\} \\ + \left[ 1 - p_{-1}^{(0)} \right] \left\{ \left[ \frac{1}{2} \left[ 1 - \left[ 1 - 2 p_{-1}^{(l-1)} \right]^{d_c-1} \right] \right]^{d_v-1} \right\}$$

{ This is the desired expression for  
density evolution

(can be shown that

a)  $p_{-1}^{(x)}$  { increases monotonically with  
increase in  $p_{-1}^{(0)}$

b) { when  $p_{-1}^{(0)}$  is below a certain  
threshold,  $p_{-1}^{(x)} \rightarrow 0$  as the # of  
iterations increases.

Ex when  $(d_v, d_c) = (3, 6)$

then designed rate satisfies:

$$\frac{k}{n} > 1 - \frac{3}{6} = \frac{1}{2}$$

and the threshold exhibited by  
Gallager decoding algorithm A is

$$\epsilon = 0.04,$$

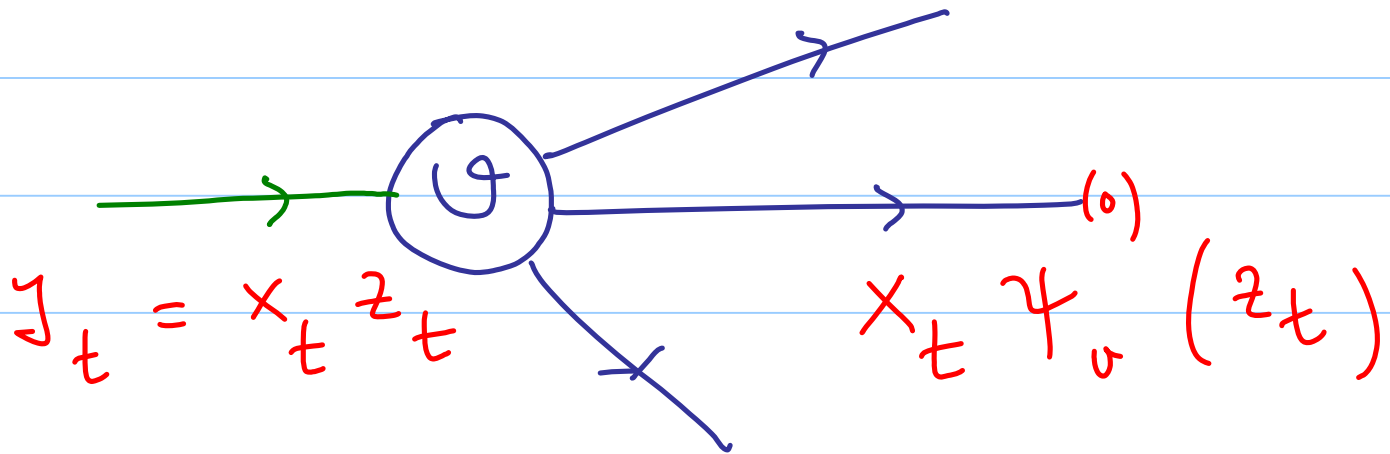
(in comparison channel capacity

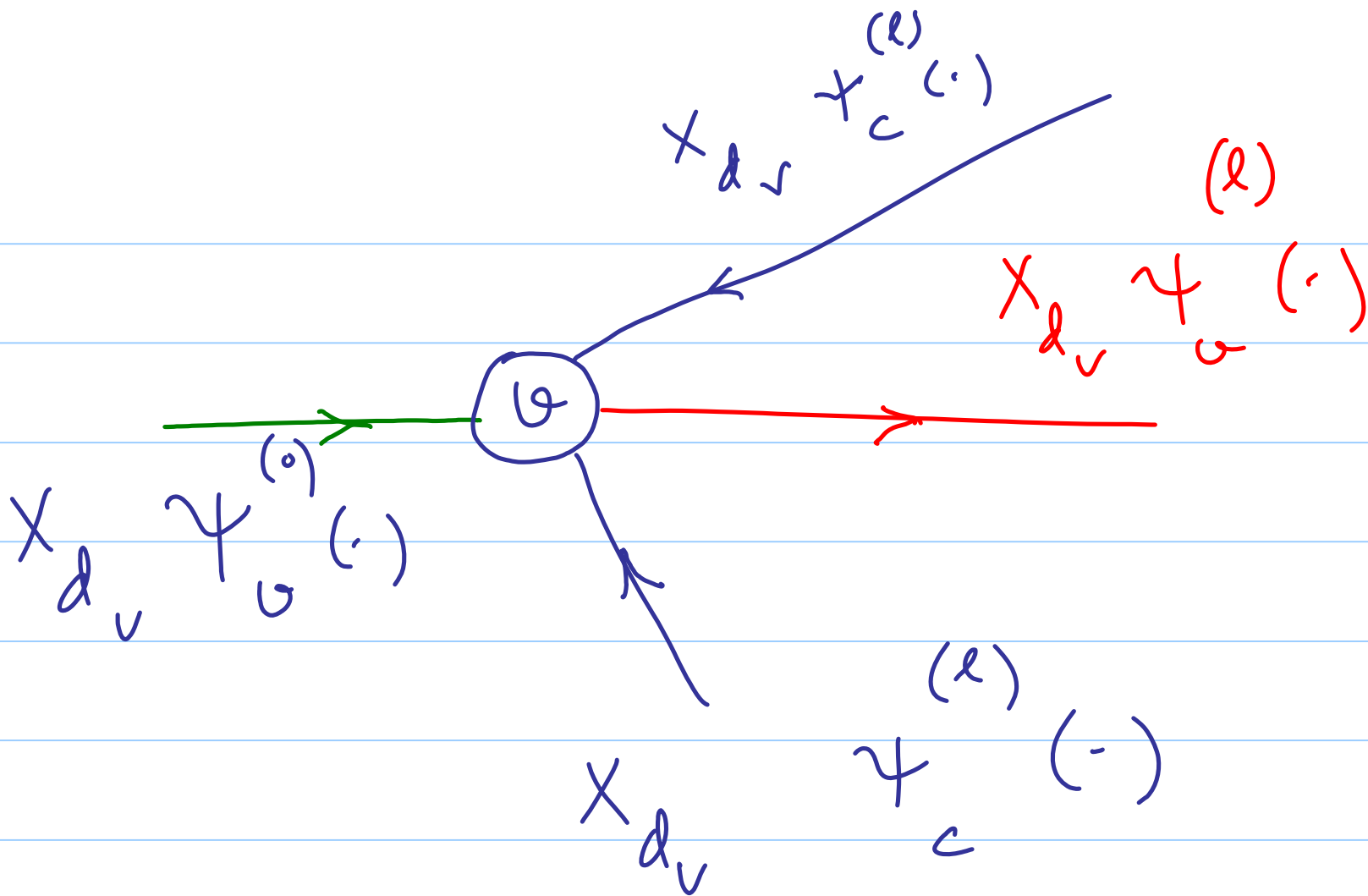


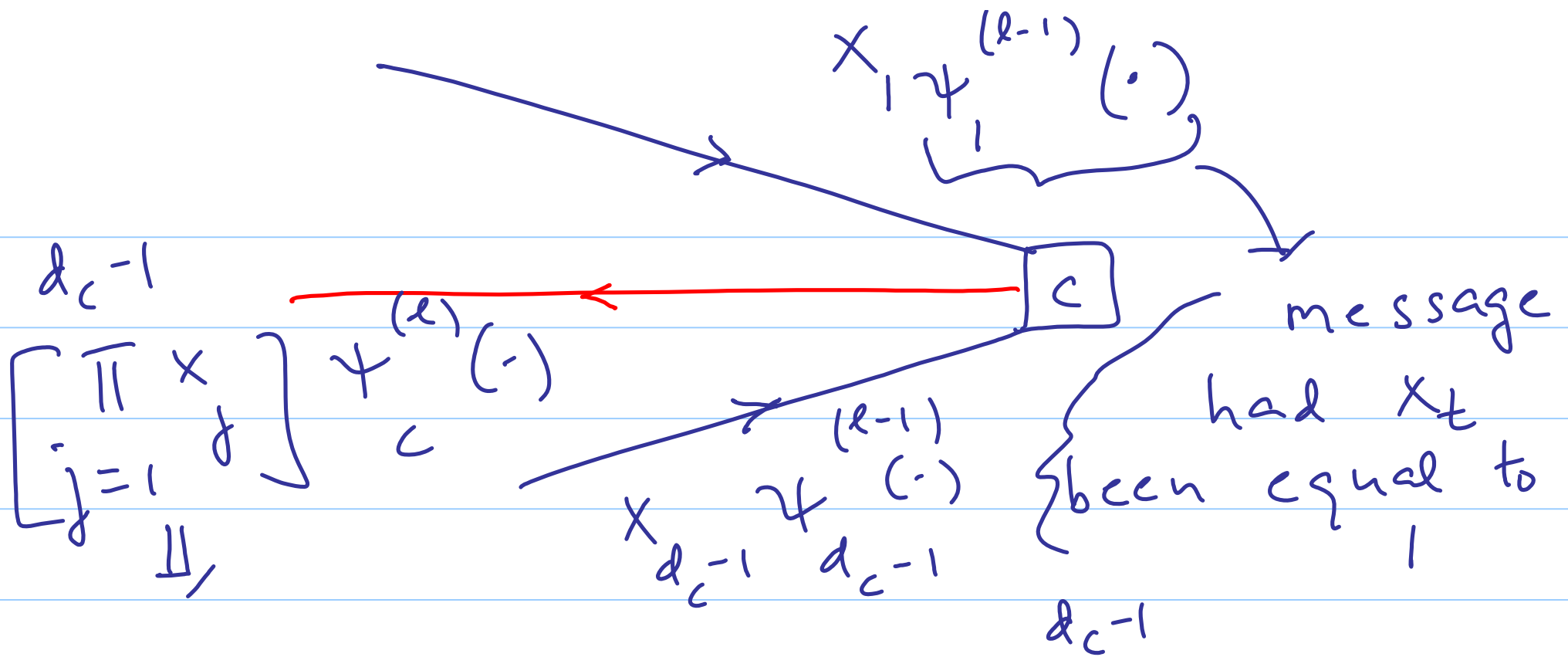
dictates that we should be able  
to operate provided  $\epsilon \leq 0.11$ )

---

We will now proceed to show that the  
 # of incorrect messages passed  
 along the edges of the Tanner graph  
 during each iteration is independent  
 of the transmitted codeword.







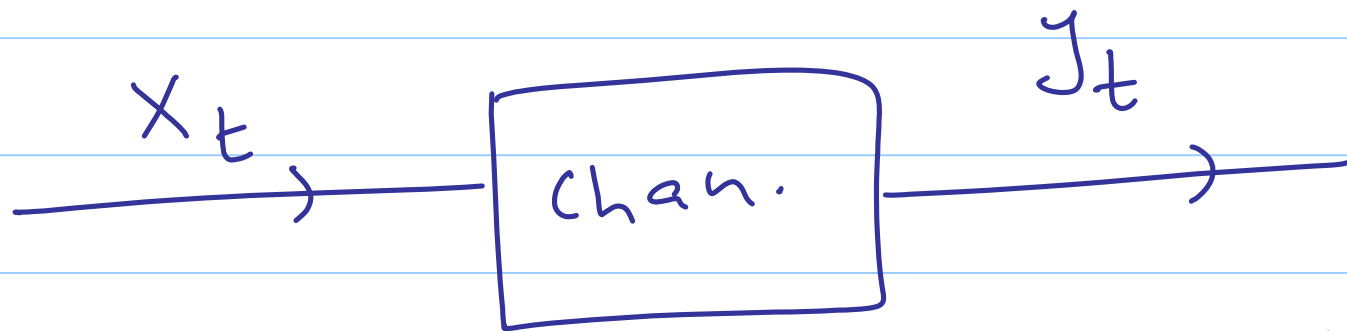
$= x_{d_c}$   
 (because of the parity check)

$$u_{d_c} = \sum_{j=1}^{d_c-1} u_j \pmod{2}$$

$$x_j = (-1)^{u_j}$$

## Channel Symmetry Assumption

$$p(y_t | x_t) = p(-y_t | -x_t) \quad (*)$$



Define  $y_t = x_t z_t$

$$x_t \in \{\pm 1\}$$

{ we will now show that  $x_t$  is  
independent of  $z_t$ .

$$p_{z/x}(z_t | x_t = 1) = p_{y/x}(z_t | x_t = 1)$$

$$p_{z/x}(z_t | x_t = -1) = p_{y/x}(-z_t | x_t = -1)$$

$$= p_{y/x}(z_t | x_t = 1)$$

by (4)

$$\therefore p_{z/x}(z_t | x_t) = p_z(z_t) \quad //$$

## Lec 33 BP Decoding (continued)

### Recap

- \* { Gallager's decoding algorithm A  
– 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 codeword was transmitted.

---



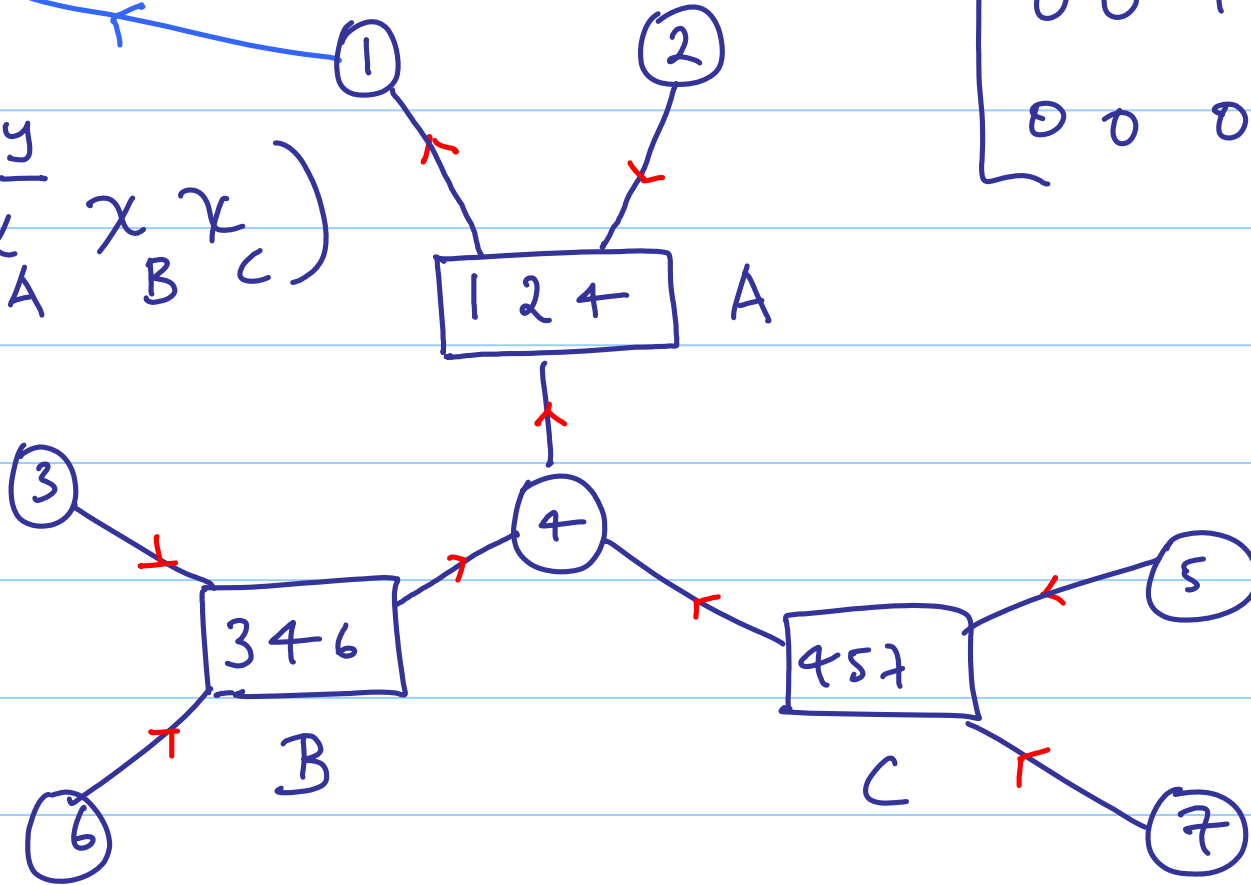
objective fn.

$\beta_1(x_1)$

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

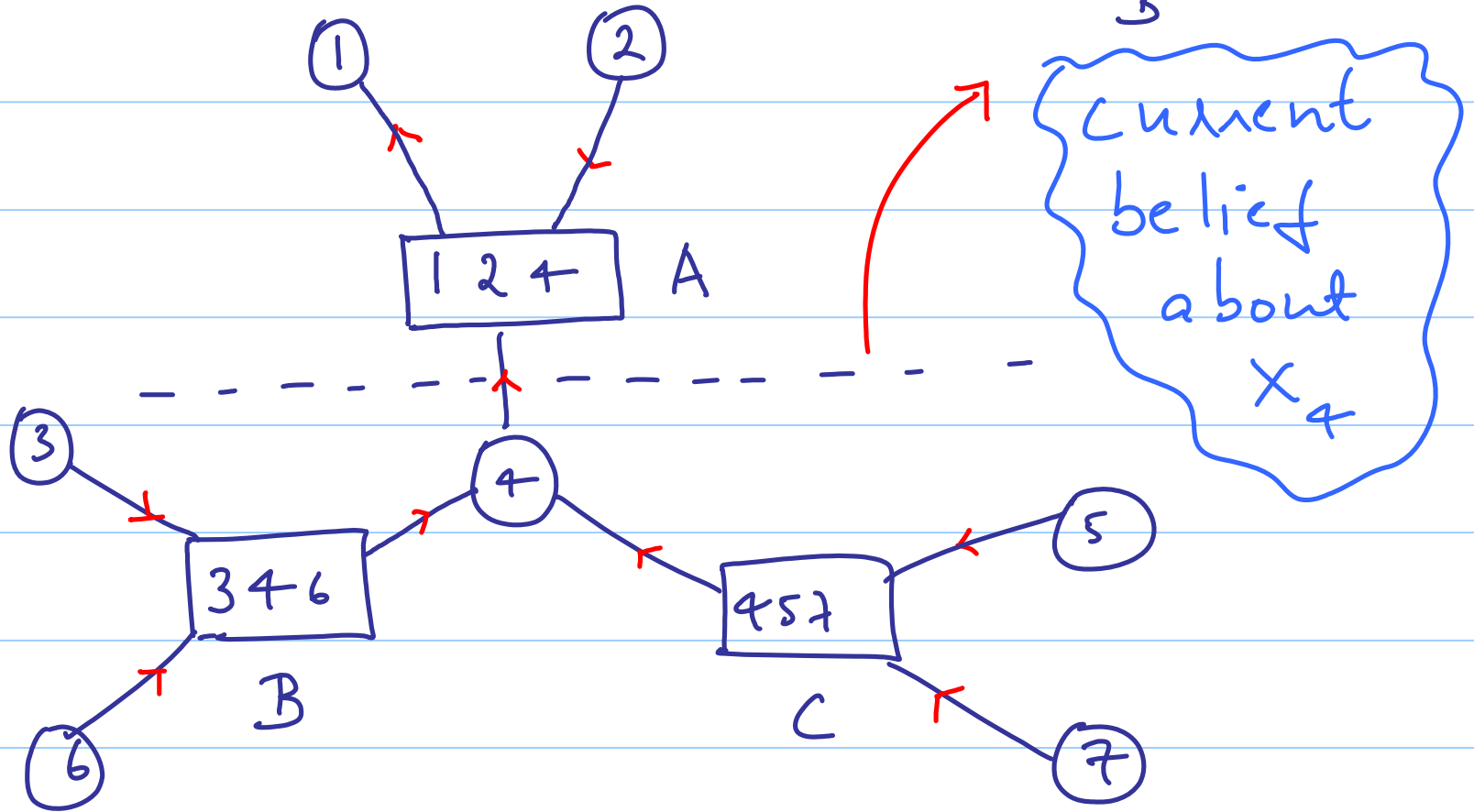
H

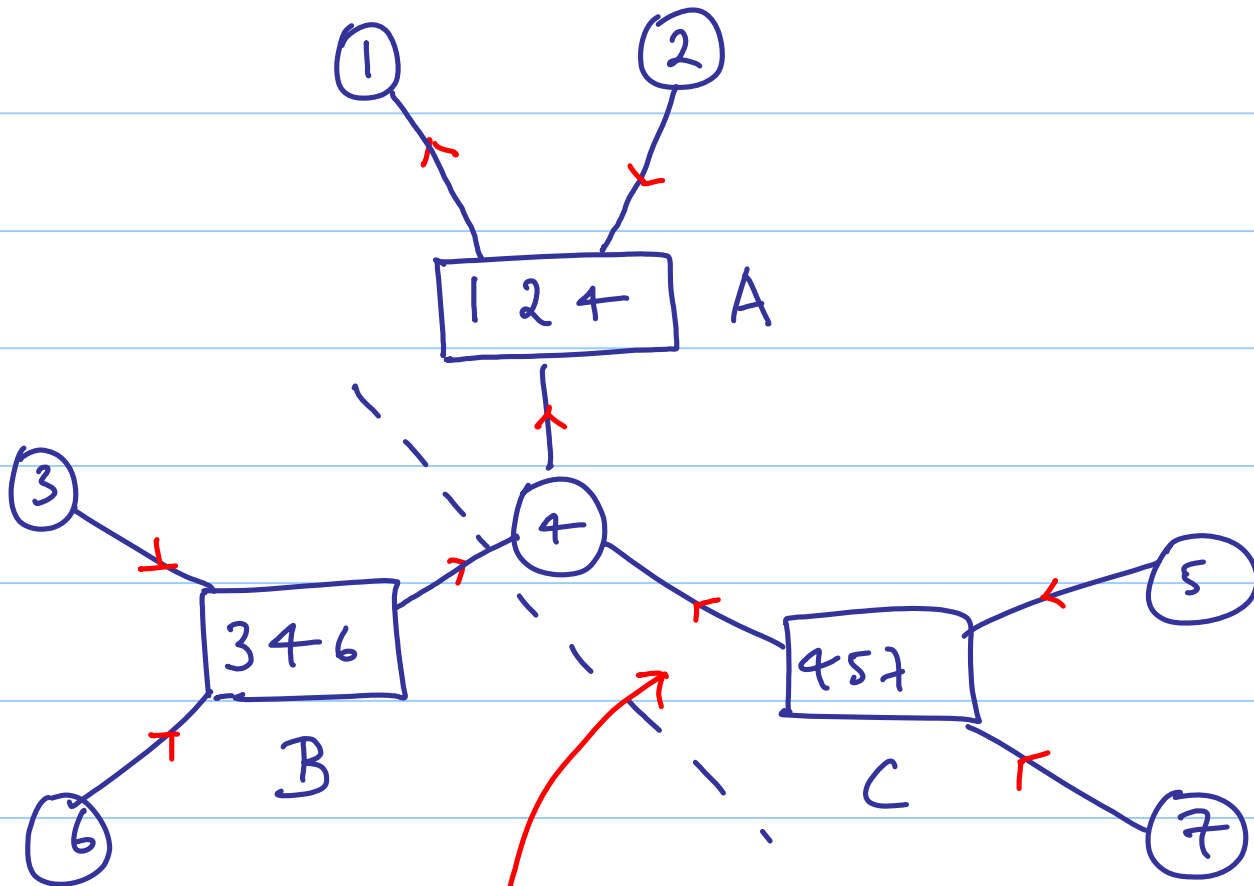
$$\alpha p(x_1 | \underline{y} x_A x_B x_C)$$



$$\propto p(x_4 | \gamma_3 \gamma_4 \gamma_5 \gamma_6 \gamma_7)$$

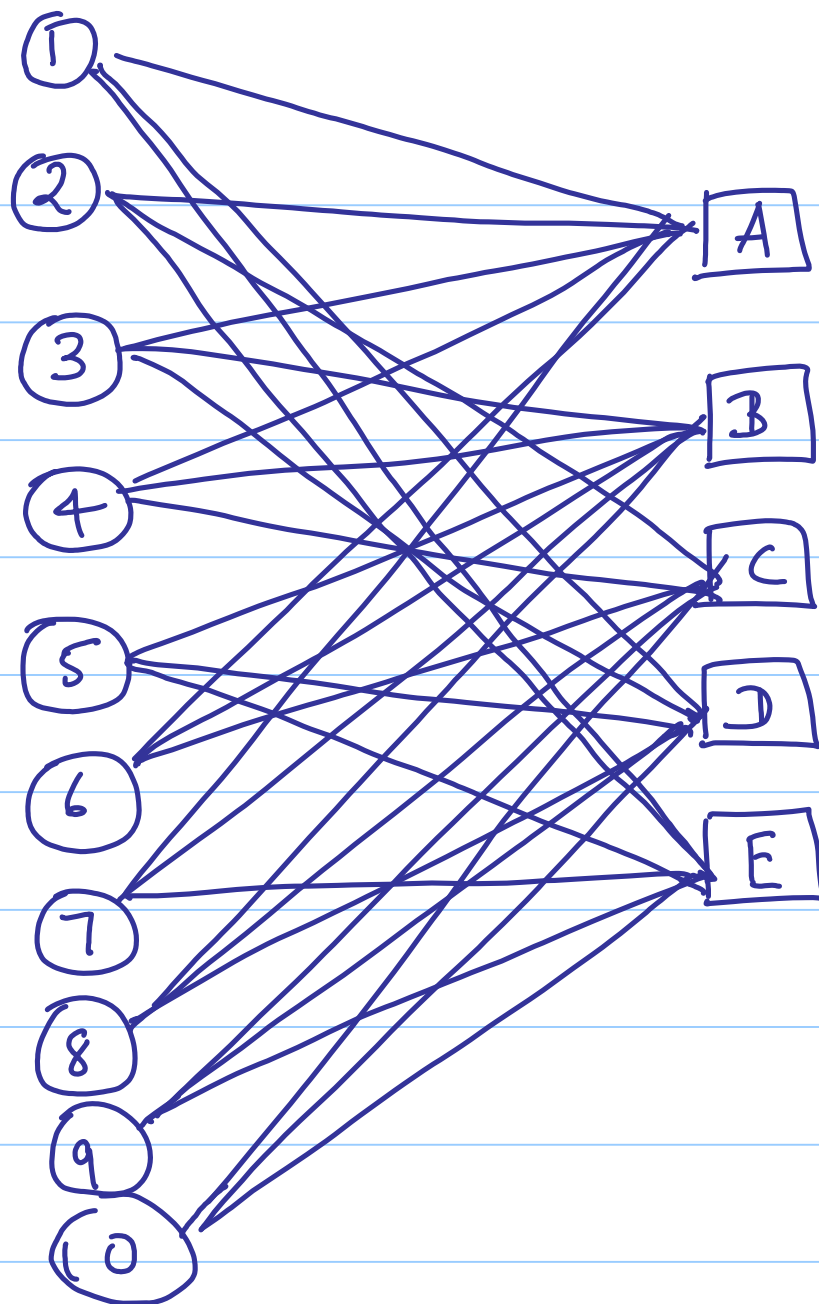
$\gamma_B \quad \gamma_C$





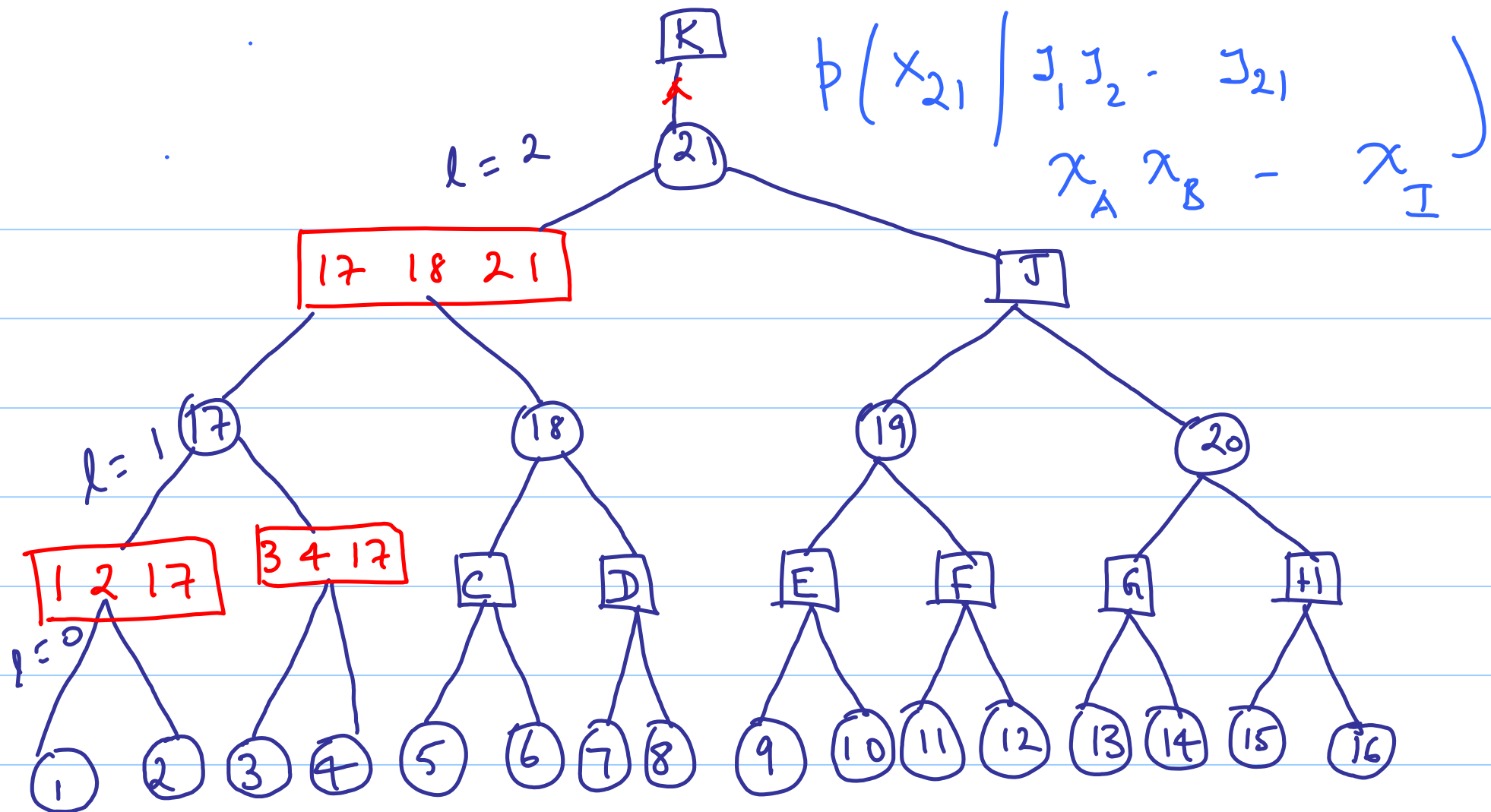
$$p(x_4 / y_3 y_6 x_B)$$

current belief  
about the prob  
of  $x_4$



Tanner }  
graph }

we assume  
that the  
nbhd of every  
edge is tree like  
to depth  $2\ell$   
 $N_e^{2\ell}$



this is the jn tree that one would  
 obtain if one posed the problem

of computing  $p(x_{21} | y_1, y_2, \dots, y_{21}, x_A, x_B, \dots, x_I)$

as an MPF problem!

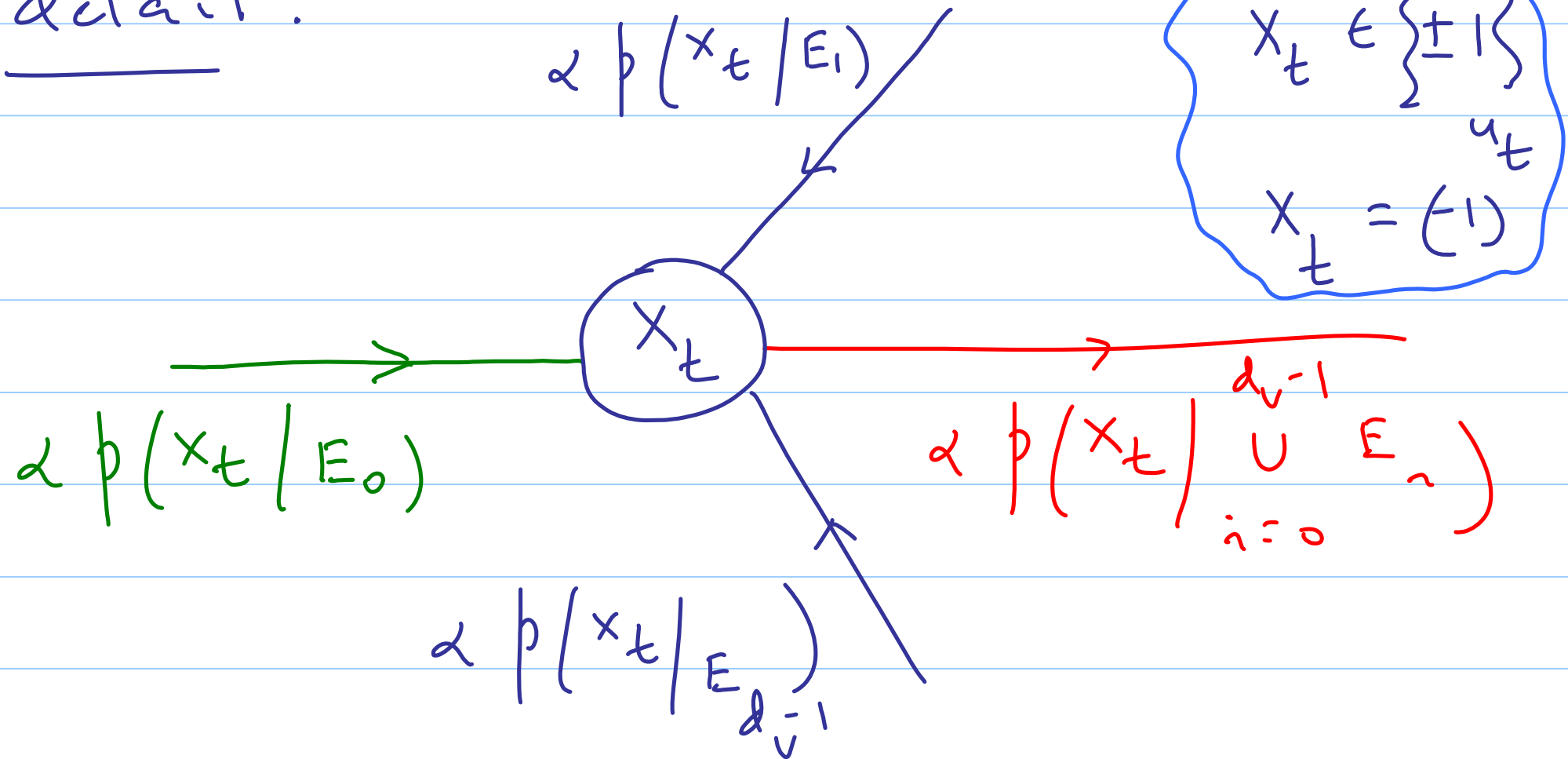
Hence if we pass messages along the edges of the Tanner graph in exactly the same manner as in the case of messages passed along the edges of the tree associated with the example

$[7 \ 4 \ 2]$  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 mcmc

detail:





Under the LDL we have:

$$p(x_t | \bigcup_{i=0}^{d_v-1} E_i) \propto \prod_{j=0}^{d_v-1} p(x_t | E_j)$$

$$\Rightarrow \frac{p(x_t = 1 | \bigcup_{i=0}^{d_v-1} E_i)}{p(x_t = -1 | \bigcup_{i=0}^{d_v-1} E_i)} = \frac{\prod_{j=0}^{d_v-1} p(x_t = 1 | E_j)}{\prod_{j=0}^{d_v-1} p(x_t = -1 | E_j)}$$

$$\Rightarrow \frac{p(u_t = 0 \mid \cup E_i)}{p(u_t = 1 \mid \cup E_i)} = \prod_{j=0}^{d_v-1} \frac{p(u_t = 0 \mid E_j)}{p(u_t = 1 \mid E_j)}$$

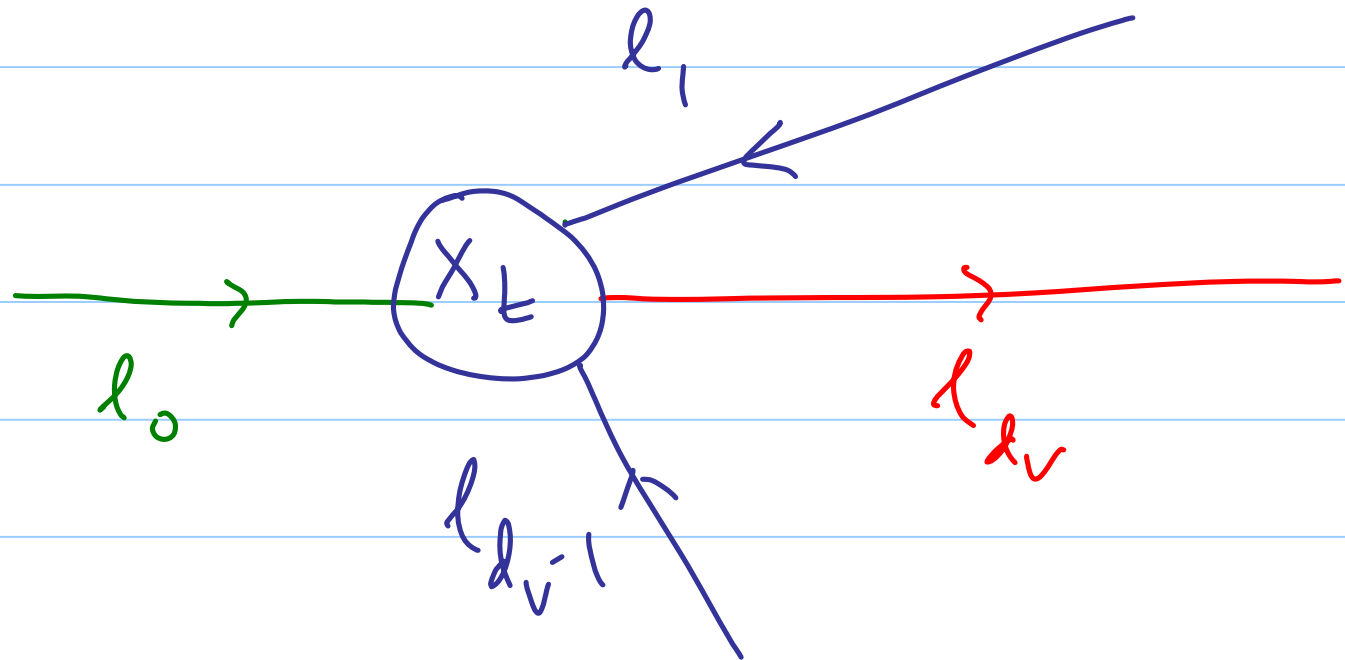
$$l_j = \ln \left\{ \frac{p(u_t = 0 \mid E_j)}{p(u_t = 1 \mid E_j)} \right\} \quad \dots \textcircled{1}$$

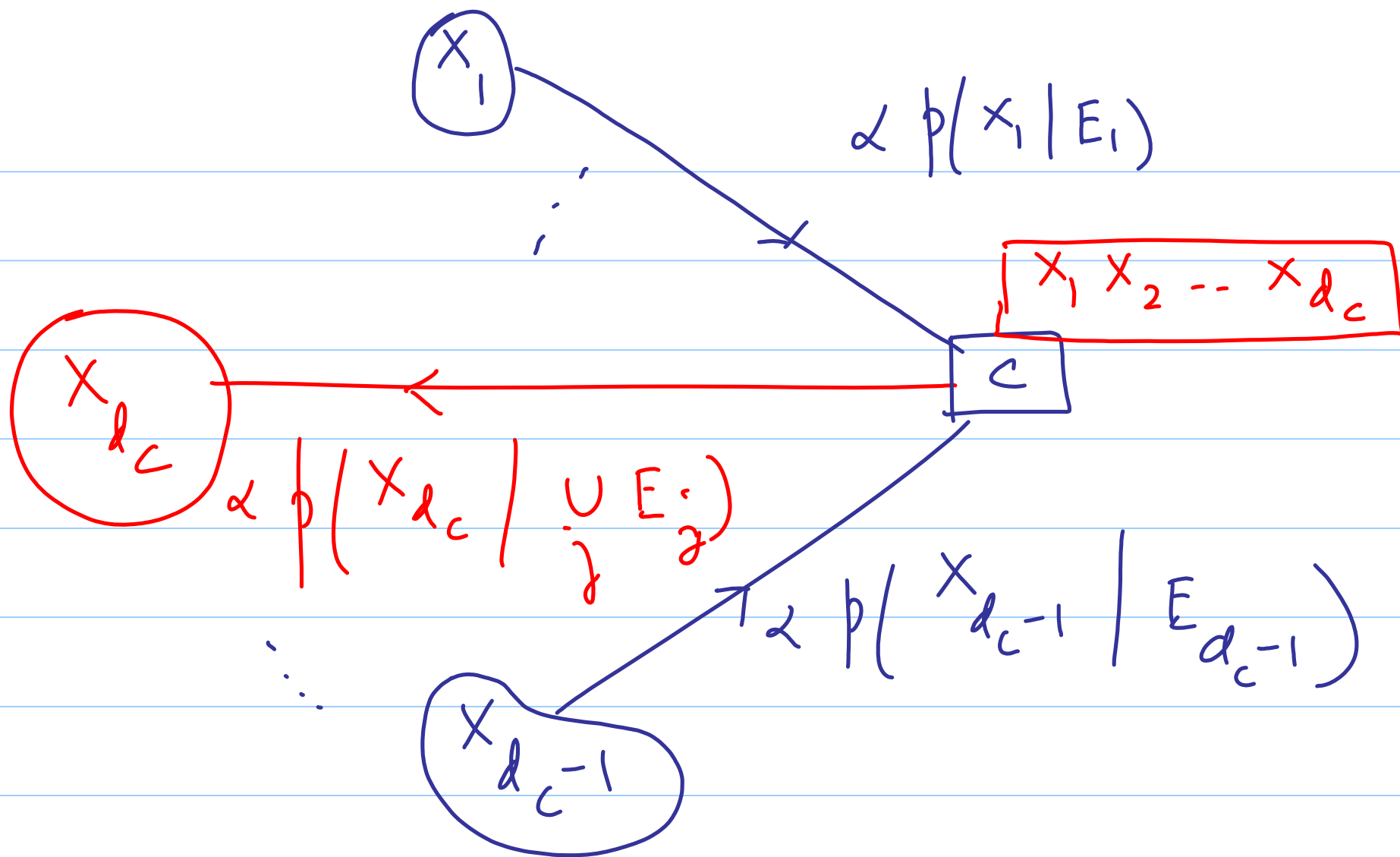
$$E_{d_v} = \bigcup_{i=0}^{d_v-1} E_i$$

Taking logs on both sides of ①

gives us:

$$l_{d_v} = \sum_{i=0}^{d_v-1} l_i$$





$$p(x_{d_c} | v E_j) \propto \sum_{\substack{\sim x_{d_c} \\ j=1}}^{d_c-1} \prod p(x_j | E_j)$$

$$\sum_{u_{d_c}=0}^1 p(u_{d_c} | v E_j) (-1)^{u_{d_c}}$$

$$x_j = (-1)^{u_j}$$

$$\sum_{u_{d_c}=0}^1 (-1)^{u_{d_c}} \sum_{u_1 \dots u_{d_c-1}}^{d_c-1} \prod_{j=1}^{d_c-1} p(u_j | E_j)$$

$$\sum_{j=1}^{d_c-1} u_j = u_{d_c}$$

$$= 2 \sum_{u_1 \dots u_{d_c}} \prod_{j=1}^{d_c-1} (-1)^{\sum u_j} p(u_j | E_j)$$

$$\therefore \frac{p(u_{d_c}=0 | E_{d_c}) - p(u_{d_c}=1 | E_{d_c})}{p(u_{d_c}=0 | E_{d_c}) + p(u_{d_c}=1 | E_{d_c})}$$

$$= \prod_{j=1}^{d_c-1} \frac{p(u_j=0 | E_j) - p(u_j=1 | E_j)}{p(u_j=0 | E_j) + p(u_j=1 | E_j)}$$

In terms of log-likelihood ratios,  
reduces to:

$$\frac{e^{l_{dc}} - 1}{e^{l_{dc}} + 1} = \prod_{j=1}^{d_c-1} \frac{e^{l_j} - 1}{e^{l_j} + 1}$$

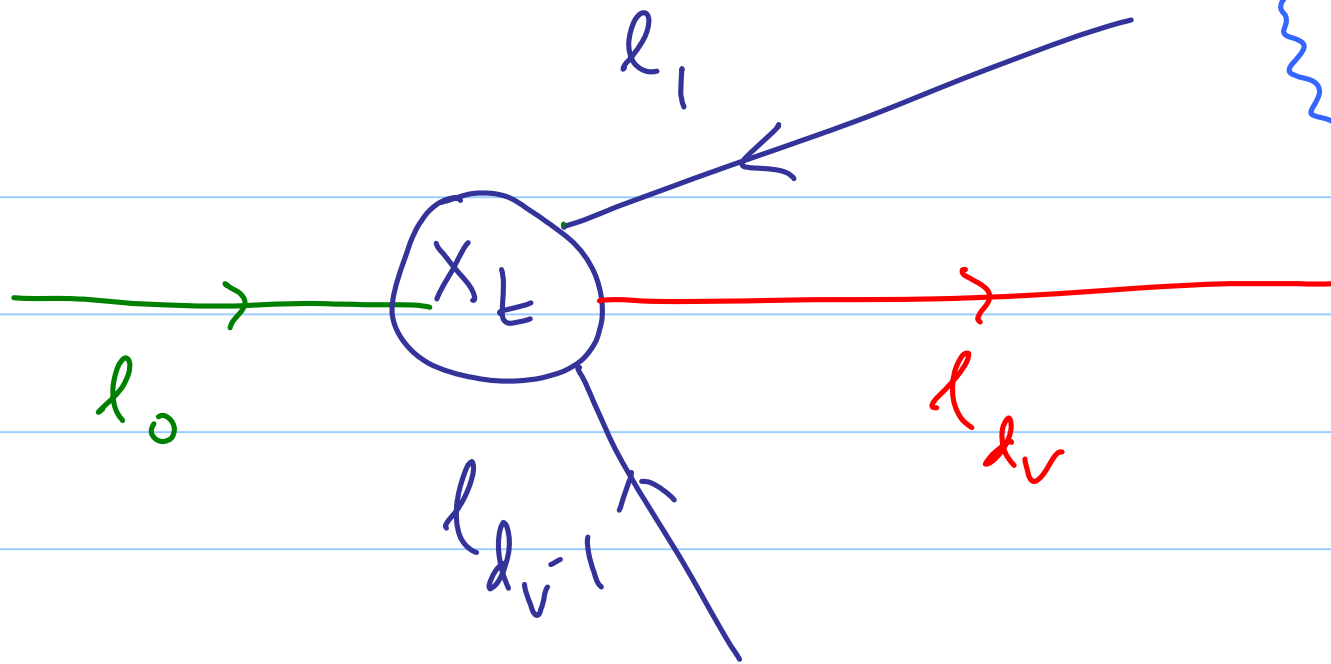
$$\therefore \tanh\left(\frac{l_{dc}}{2}\right) = \prod_{j=1}^{d_c-1} \tanh\left(\frac{l_j}{2}\right)$$

# Lec 34 Density Evolution under BP decoding

## 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

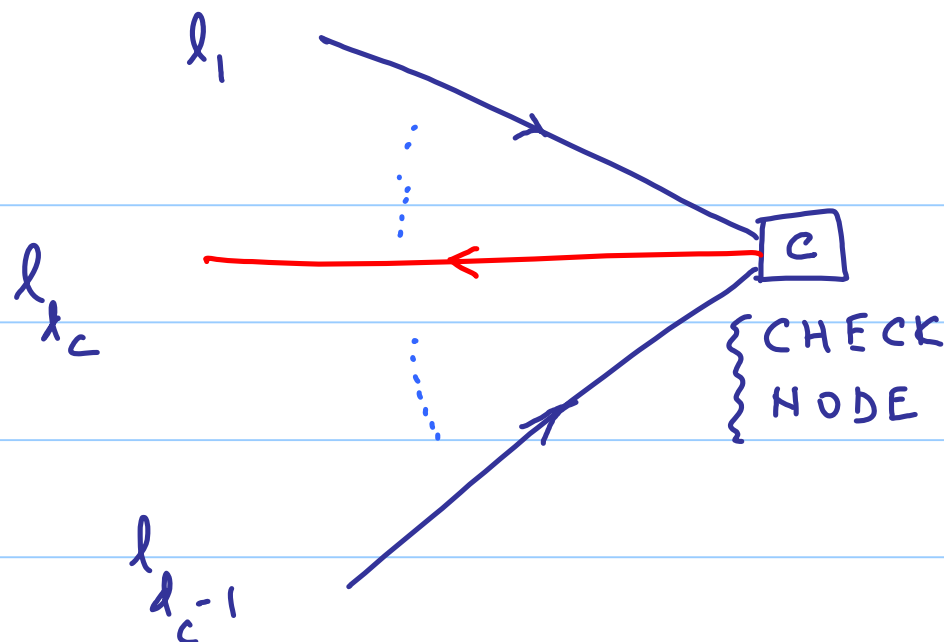
$$l_{d_v} = \sum_{i=0}^{d_v-1} l_i$$

Note: variable  
- node symmetry  
condition is  
met.

$$\tanh\left(\frac{l_{d_c}}{2}\right)$$

$$= \prod_{j=1}^{d_c-1} \tanh\left(\frac{l_j}{2}\right)$$

(2)



Or,

(3)

$$l_{d_c} = 2 \tanh^{-1} \left\{ \prod_{j=1}^{d_c-1} \tanh\left(\frac{l_j}{2}\right) \right\}$$

Can verify that if  $l_j \Rightarrow l_j b_j$   
 Then  $b_j \in \{\pm 1\}$

$$l_{d_c} = 2 \tanh^{-1} \left\{ \prod_{j=1}^{d_c-1} \tanh \left( \frac{b_j l_j}{2} \right) \right\}$$

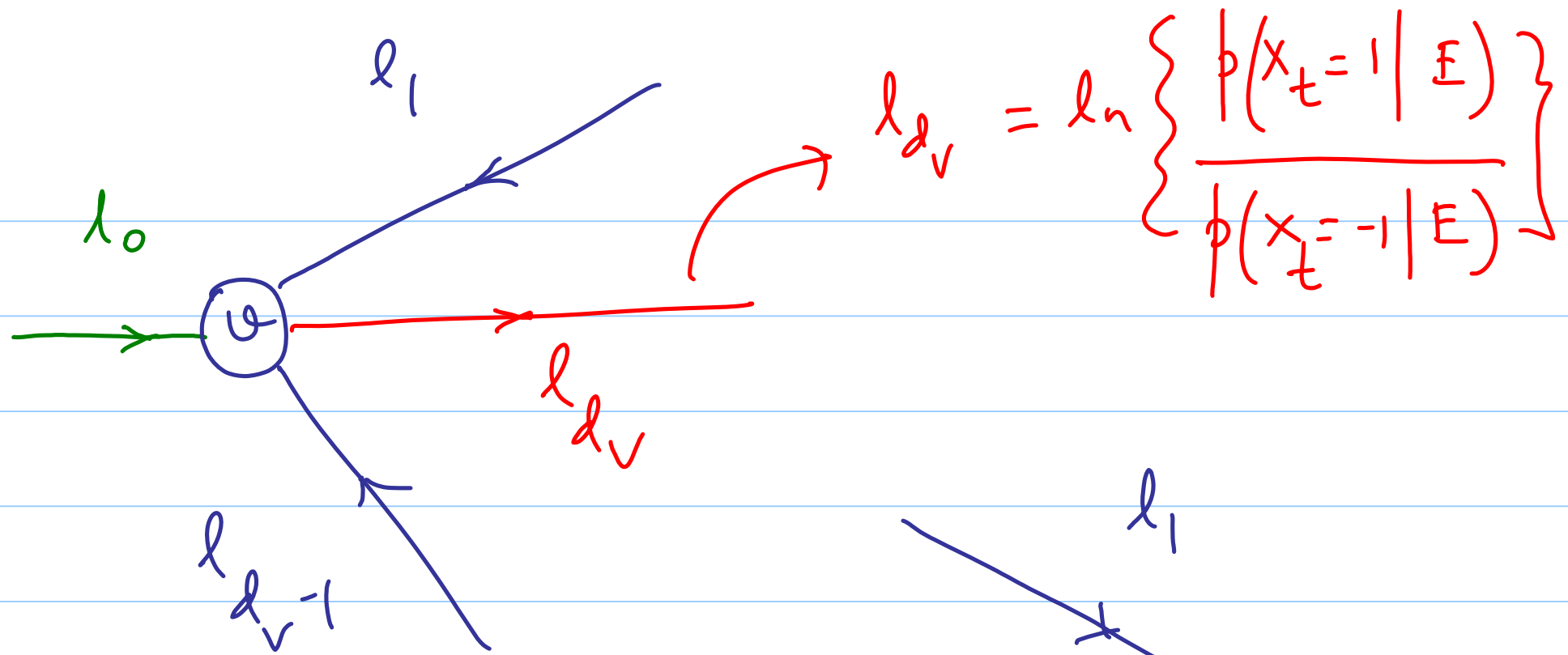
$$= \left( \prod_{j=1}^{d_c-1} b_j \right) 2 \tanh^{-1} \left\{ \prod_{j=1}^{d_c-1} \tanh \left( \frac{l_j}{2} \right) \right\}$$

and hence the check-node symmetry  
 condition is also met.

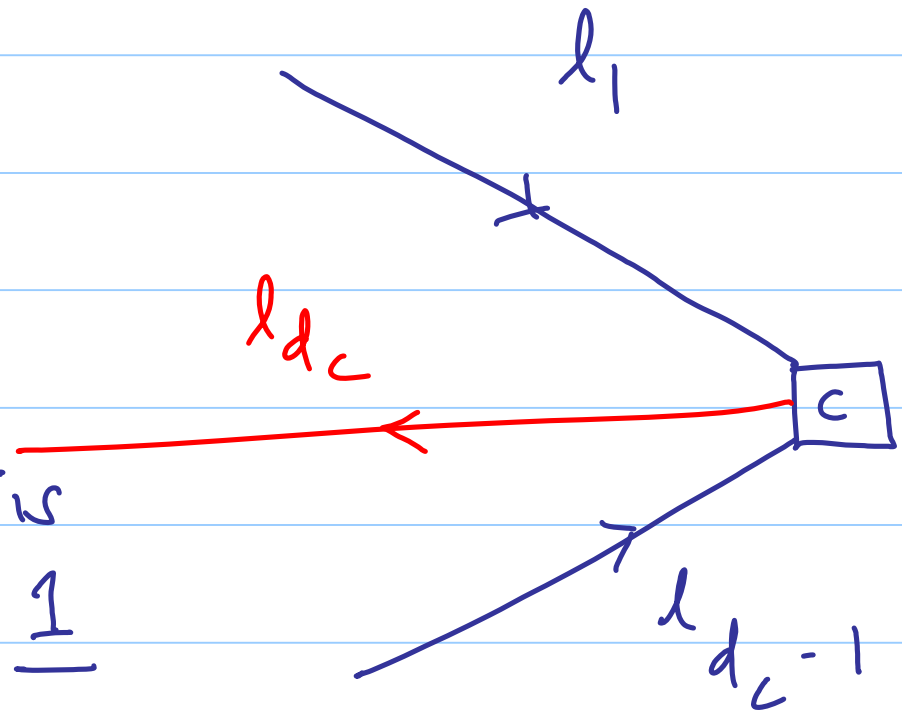
With this we are done with  
the description of BP-based  
message-passing-decoding of LDPC  
codes.

We now turn our attention  
to performance analysis.

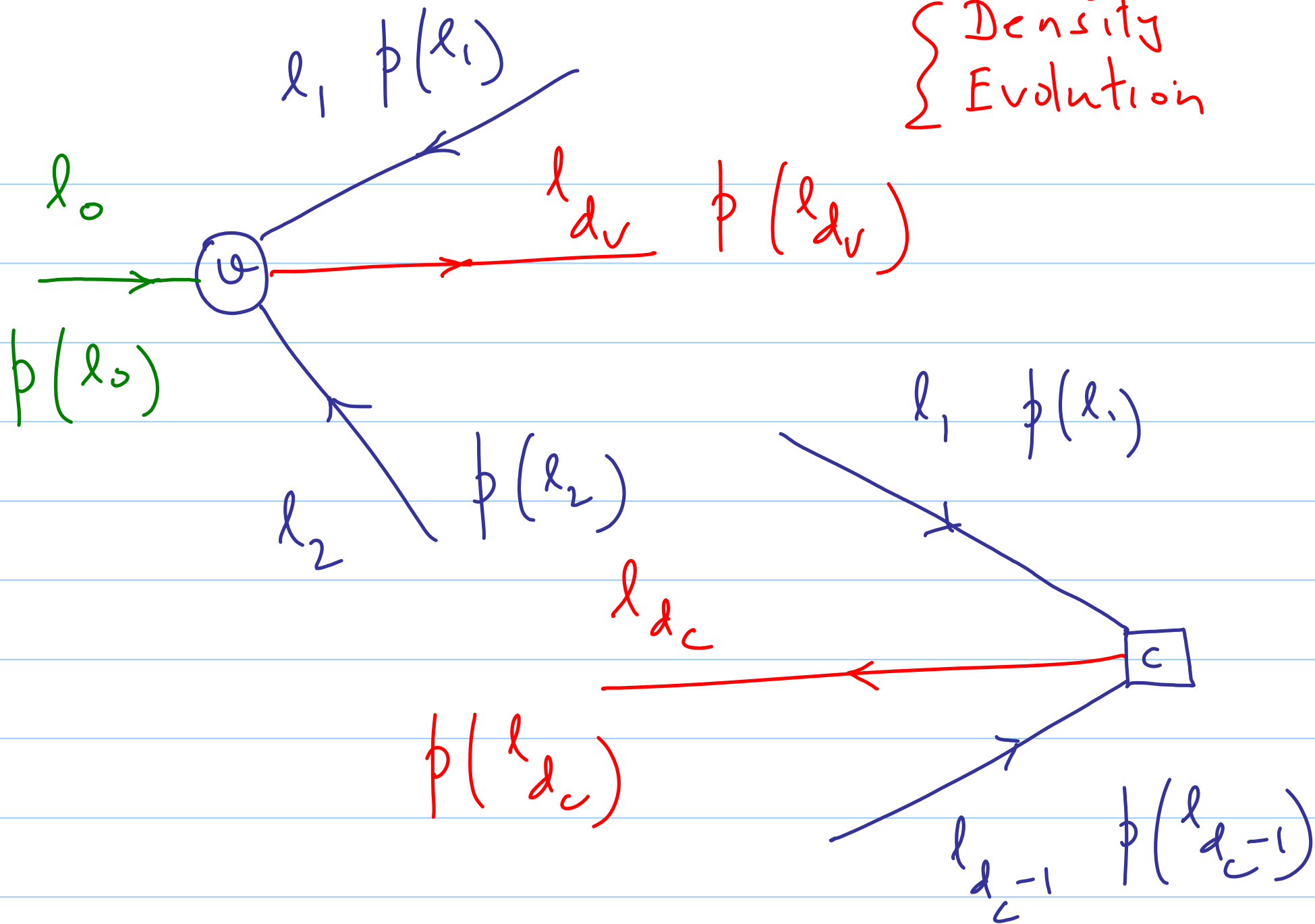
⇓  
to be carried out using density  
evolution. }



{ In carrying out  
performance analysis  
we assume that 1  
was transmitted



Density  
Evolution



Replace  $\tanh\left(\frac{l}{2}\right)$  by  $(x, y)$

where:

$$x = \operatorname{sgn}(l)$$

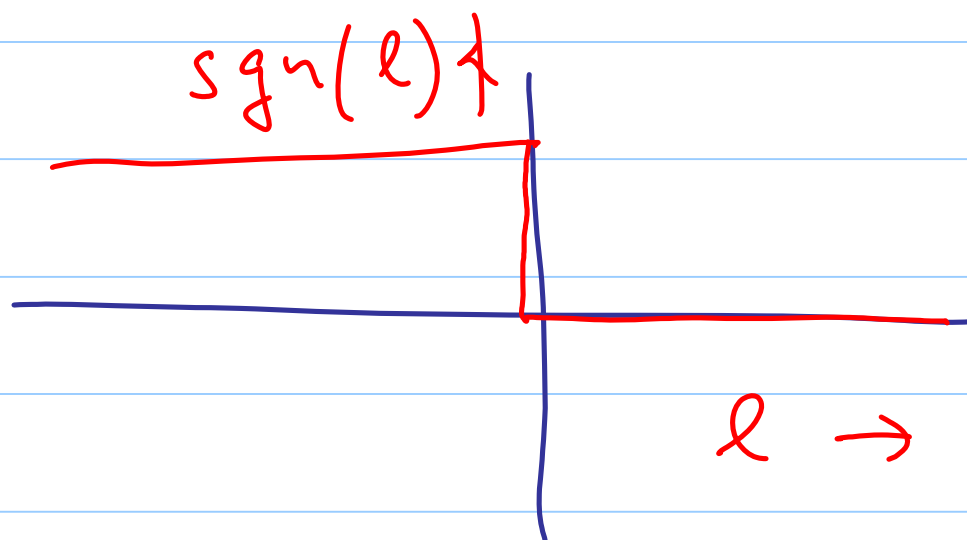
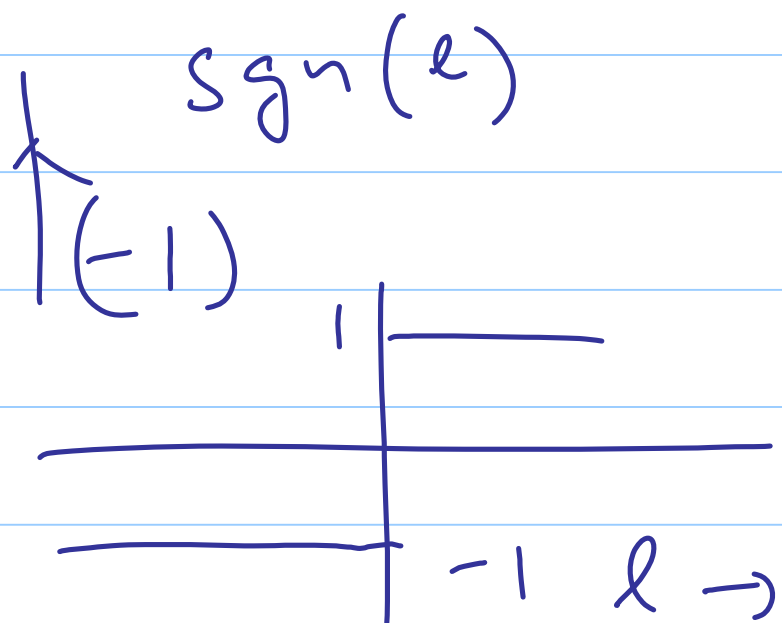
$$y = -\ln \left| \tanh\left(\frac{l}{2}\right) \right|$$

keeps track of the  
sign of  $\tanh\left(\frac{l}{2}\right)$

keeps track of the magnitude of  
 $\tanh\left(\frac{l}{2}\right)$

where

$$\operatorname{sgn}(l) = \begin{cases} 0 & l \geq 0 \\ 1 & l < 0 \end{cases}$$





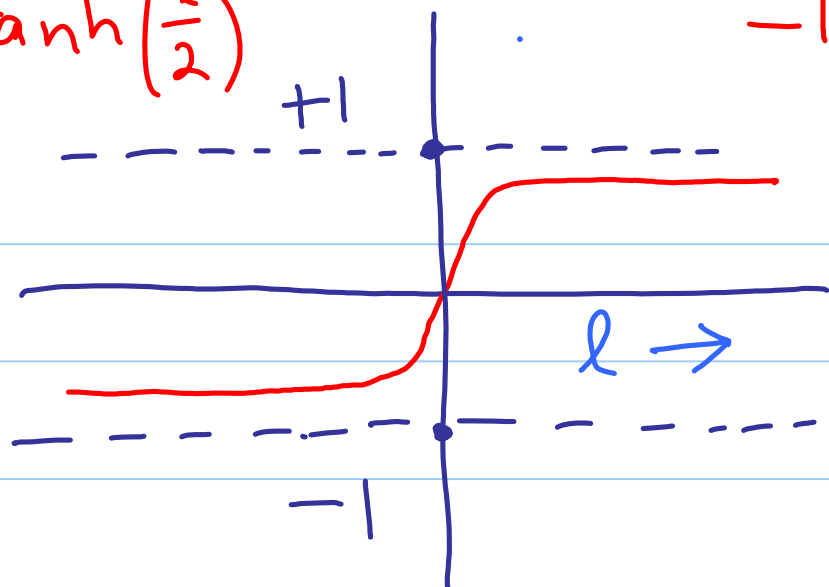
Thus

$$\tanh\left(\frac{l_c}{2}\right) = \prod_{j=1}^{l_c-1} \tanh\left(\frac{l_j}{2}\right)$$

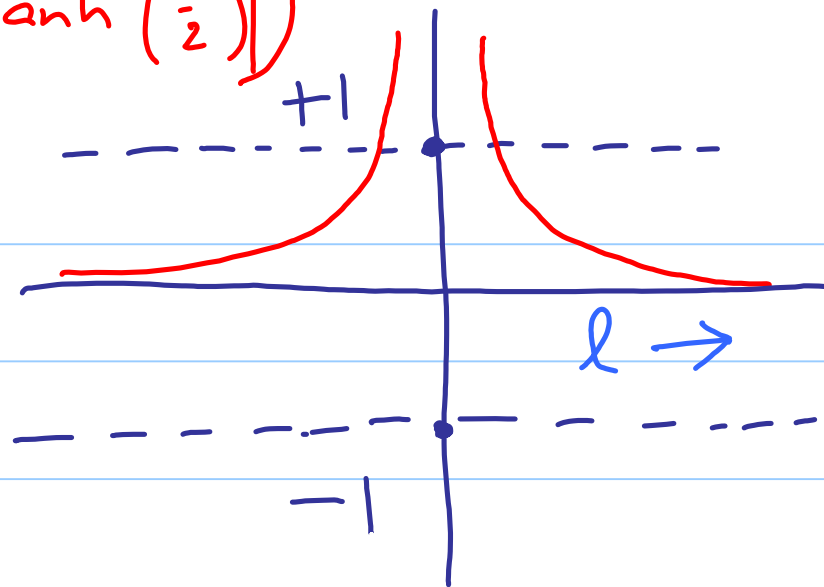
is equivalent to:

$$(x_{d_c}, y_{d_c}) = \left( \sum_{i=1}^{d_c-1} x_i \pmod{2}, \sum_{i=1}^{d_c-1} y_i \right)$$

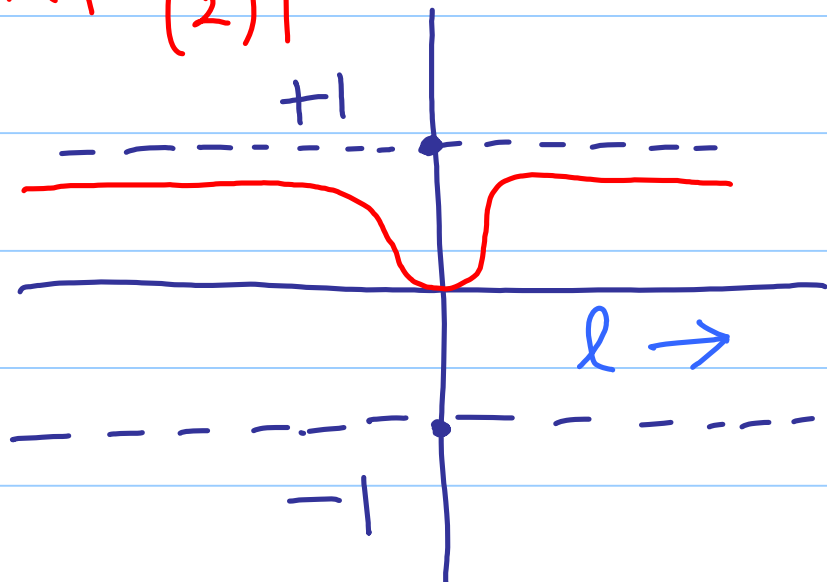
$$\tanh\left(\frac{l}{2}\right)$$



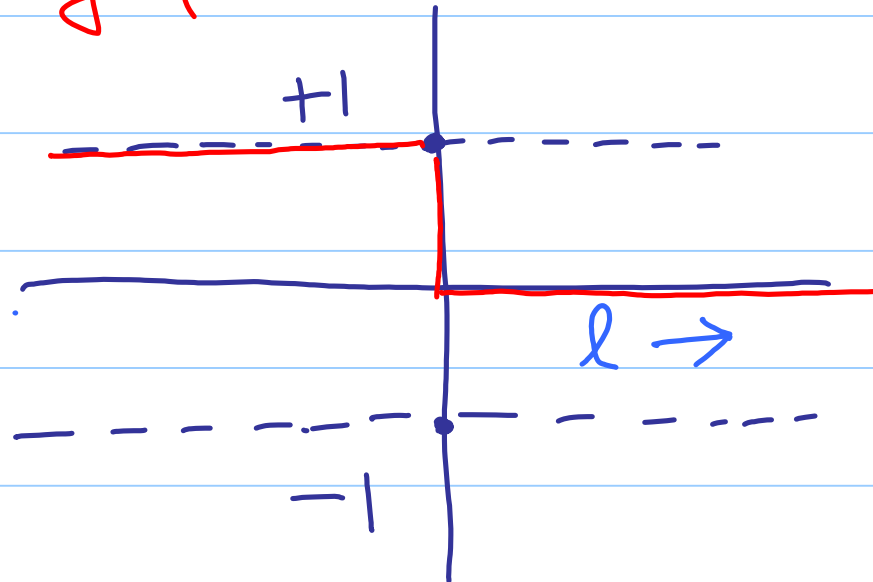
$$-\ln\left(\left|\tanh\left(\frac{l}{2}\right)\right|\right)$$

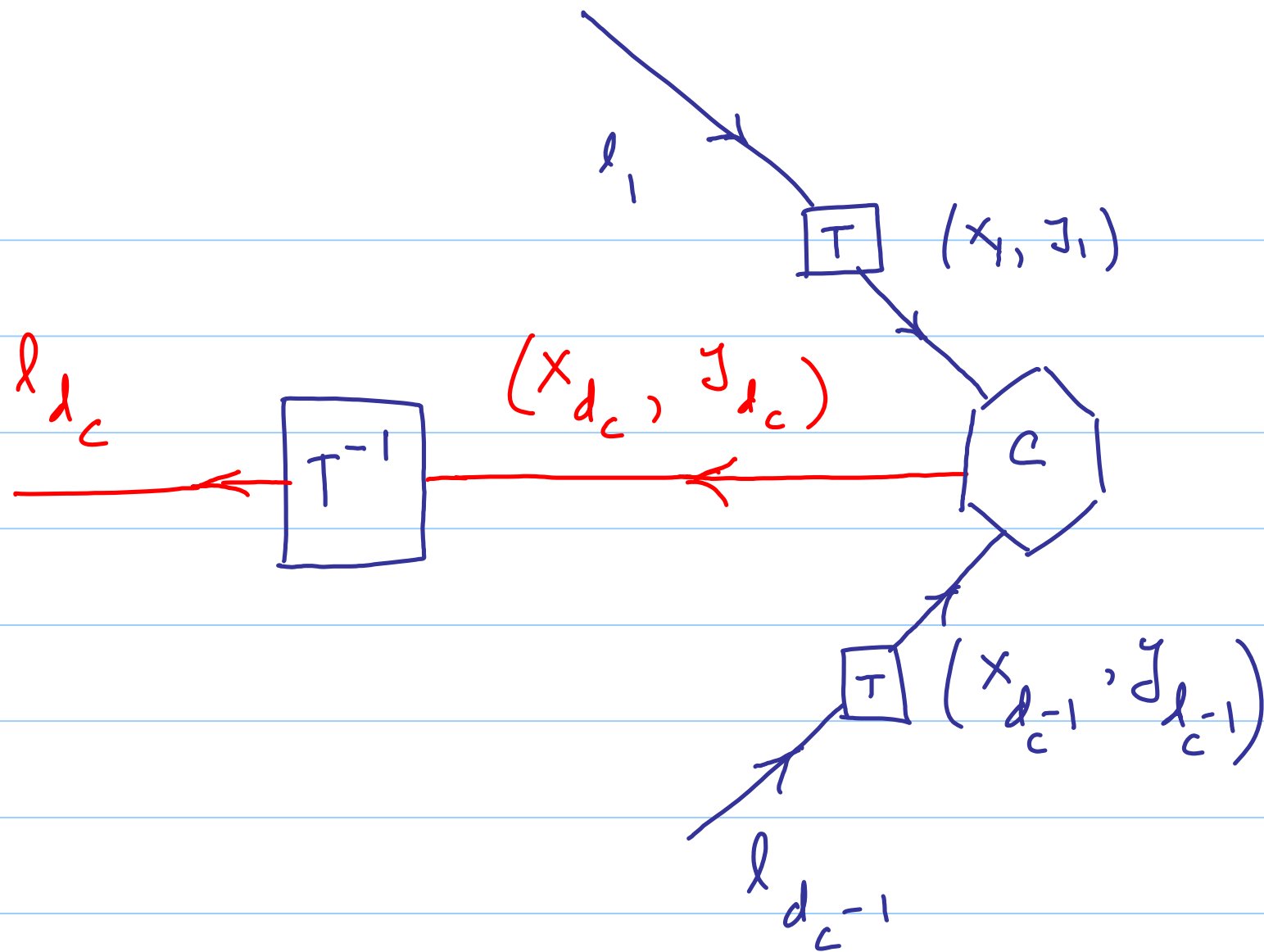


$$\left|\tanh\left(\frac{l}{2}\right)\right|$$



$$\text{sgn}(l)$$





(0)

$$\psi_0 : \Theta \rightarrow \mathcal{M}$$

$$* \psi_0^{(0)} : \prod_{\Theta} \rightarrow \prod_{\mathcal{M}}$$

corresponding  
map in terms of  
density functions

(l)

$$\psi_l : \Theta \times \mathcal{M}^{d_v-1} \rightarrow \mathcal{M}$$

$$* \psi_l^{(l)} : \prod_{\Theta} \times \prod_{\mathcal{M}}^{d_v-1} \rightarrow \prod_{\mathcal{M}}$$

<sup>th</sup>  
l  
iteration

$\psi_c^{(l)}$

$m^{d_c-1}$



$m$

$* \psi_c^{(l)}$

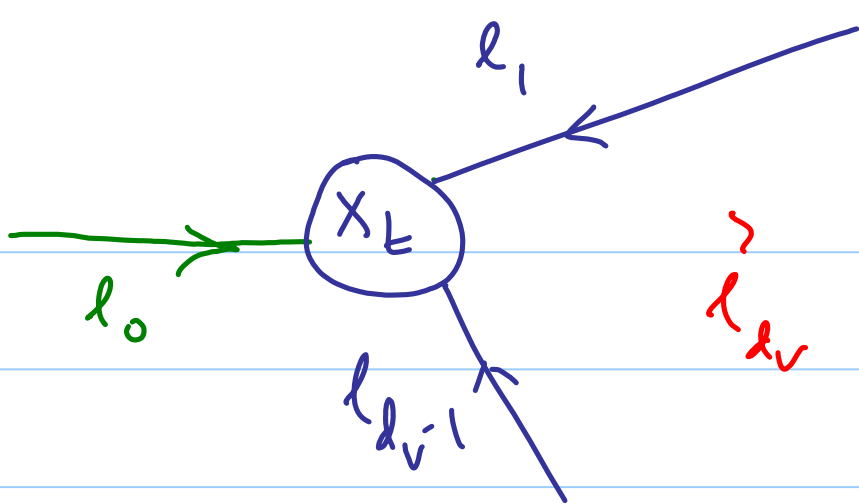
$\pi_m^{d_c-1}$



$\pi_m$

$l^{\text{th}}$   
iteration

(at a check node)



$$l_{d_v} = \sum_{i=0}^{d_v-1} l_i$$

Assume that  $\perp$  was transmitted.

The  $\{l_i\}$  are random since they are a function of the particular channel realization. Consider  $l$  rounds

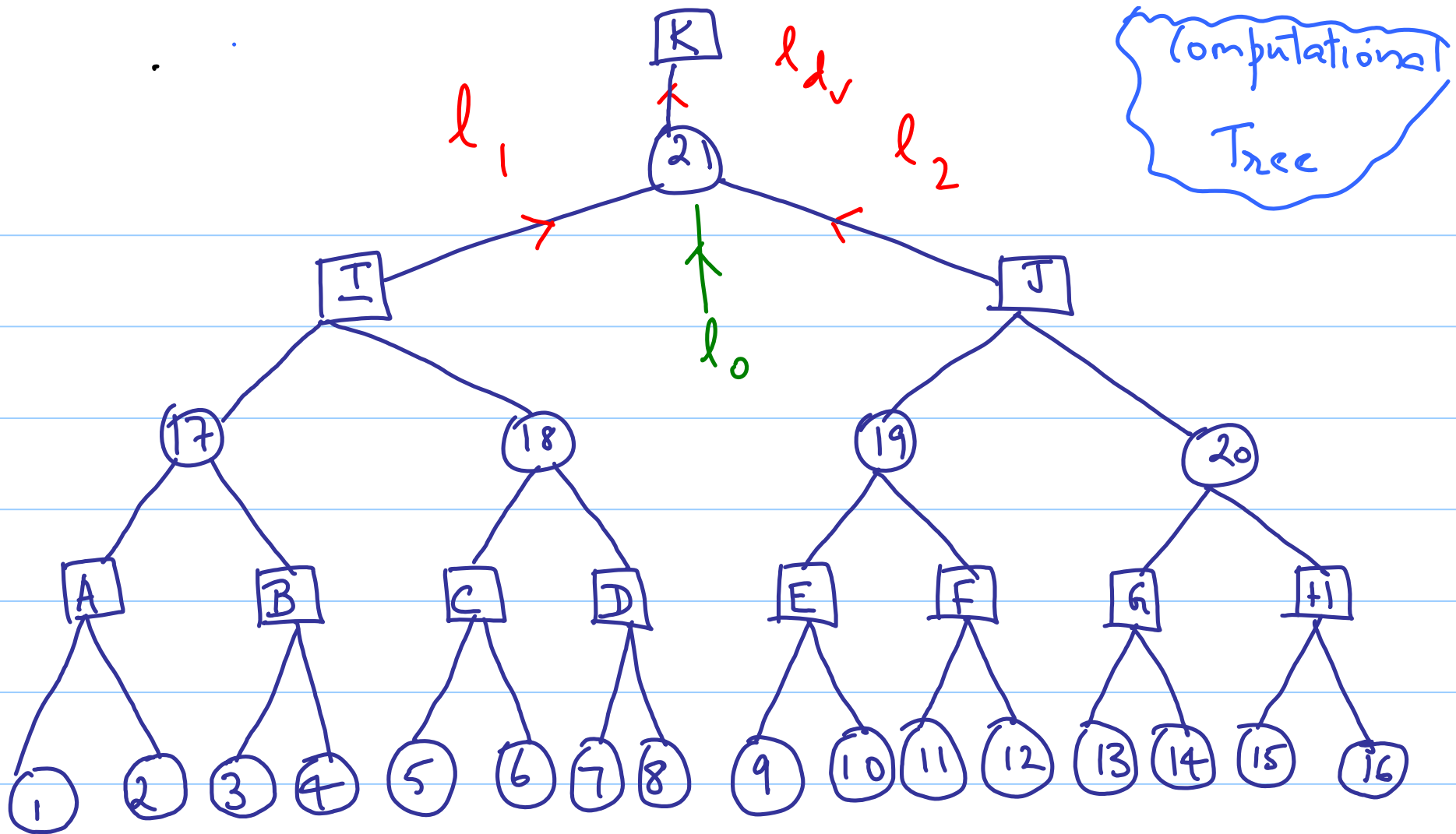
↓ message passing and assume  
the Tanner graph to be such that the  
nbhd of every node to depth 2ℓ  
is tree-like (as in figure on page  
following). Then the differences  
 $l_i, 1 \leq i \leq d_v - 1$  are linearly

independent as they are functions

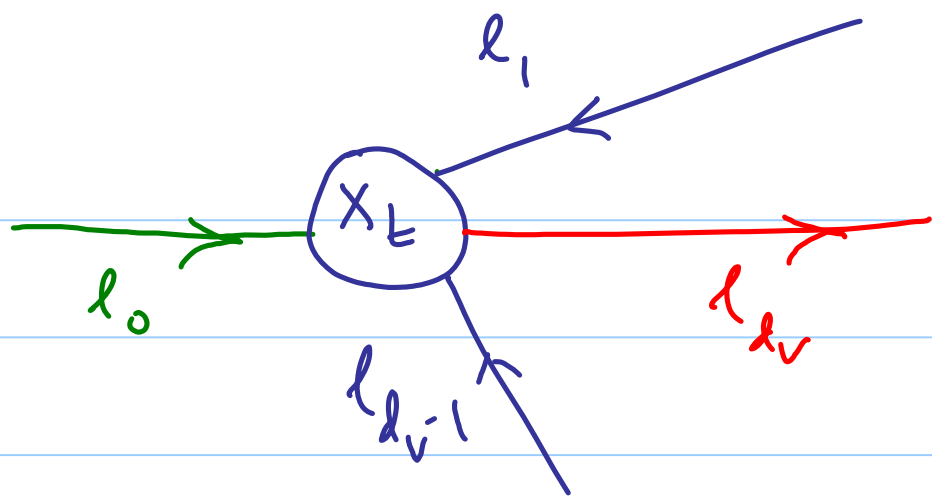
of disjoint subsets of received

variables  $\{y_j\}_{j=1}^n$ .





Scenario where the nblhd is tree-like.

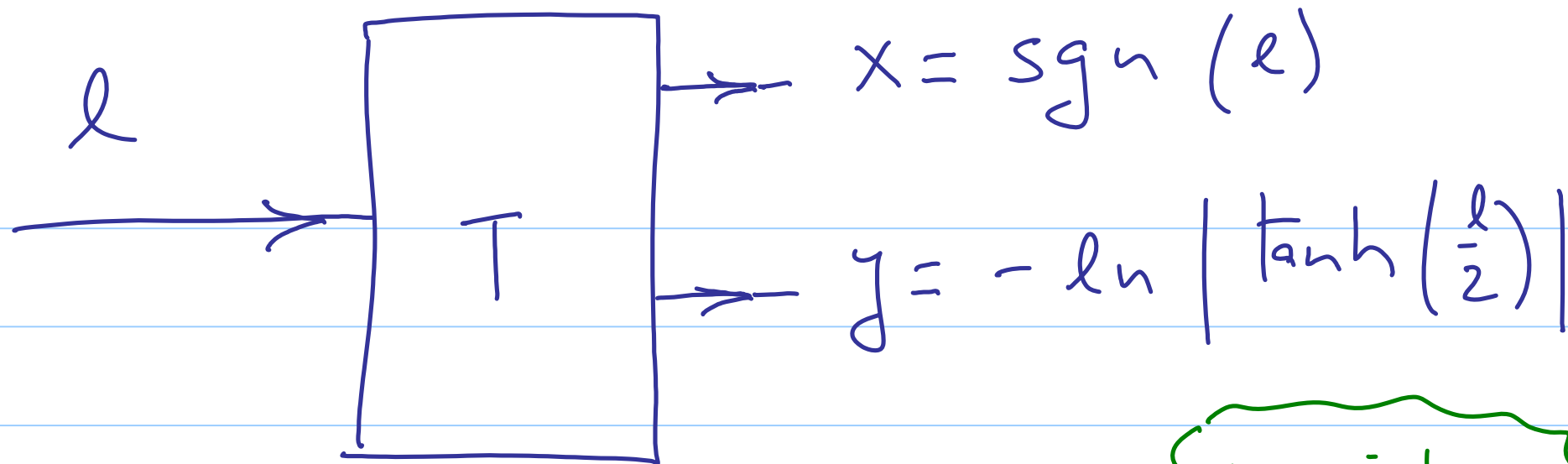


$$l_v = \sum_{i=0}^{d_v-1} l_i$$

$$\mathbb{F} \left\{ e^{-j\omega l_0} \right\} = \mathbb{F} \left\{ e^{-j\omega \sum_{i=0}^{d_v-1} l_i} \right\}$$

$$= \prod_{i=0}^{d_v-1} \mathbb{F} \left\{ e^{-j\omega l_i} \right\}$$





$$p_{xy}(0, y) = \frac{p_L\left(-\ln \tanh\left(\frac{y}{2}\right)\right)}{\sinh(y)}$$

$$p_{xy}(1, y) = \frac{p_L\left(\ln \tanh\left(\frac{y}{2}\right)\right)}{\sinh(y)}$$

densities  
after  
the

transfmn

Define

$$\phi_j(\lambda, s) = \mathbb{E} \left\{ (-1)^{\lambda x_j} e^{-s y_j} \right\} \left. \begin{array}{l} \text{joint} \\ \text{char.} \\ \text{fn.} \end{array} \right\}$$

$$= \sum_{n=0}^{\infty} (-1)^{\lambda n} \int e^{-s y} p_{x_j, r_j}(n, y) dy$$

$$= \mathbb{I}(p_{x_j, r_j}(0, y)) + (-1)^{\lambda} \mathbb{I}(p_{x_j, r_j}(1, y))$$

↙  
Laplace transform

Since the  $\{l_j\}$  and hence the  $\{(x_j, y_j)\}$  are statistically independent it

follows that:

$$\begin{aligned} x_{d_c} &= \sum_j x_j \pmod{2} \\ y_{d_c} &= \sum_j y_j \end{aligned}$$

$$\phi_{d_c}(\lambda, s) = \mathbb{E} \left\{ (-1)^{\lambda x_{d_c} - s y_{d_c}} \right\}$$

$$= \prod_{j=1}^{d_c-1} \phi_j(\lambda, s)$$

Hence

$$\left\{ \phi_{x_j, \gamma_j}(x, y) \right\}_{j=1}^{d_c-1}$$

$$\Downarrow$$

$$\left\{ \phi_j(\lambda, s) \right\}_{j=1}^{d_c-1}$$

$$\Downarrow$$

$$\phi_{d_c}(\lambda, s)$$

$$\Downarrow$$

$$\phi_{x_{d_c}, \gamma_{d_c}}(x, y)$$

The last computation use that:

$$\phi_{d_c}(\lambda s)$$

$$= \int (\phi_{x_{d_c} \tau_{d_c}}(0, y)) + (-1)^{\hat{\lambda} y} \int (\phi_{x_{d_c} \tau_{d_c}}(1, y))$$

$$\therefore \phi_{x_{d_c} \tau_{d_c}}(0, y) =$$

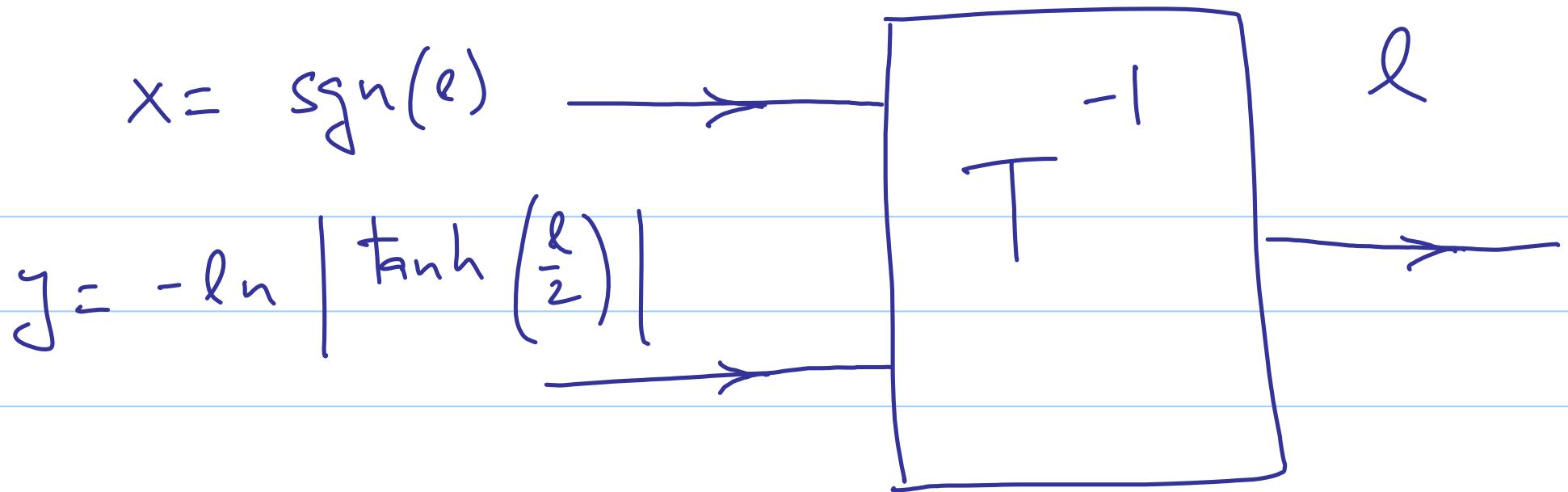


$$I^{-1} \left\{ \frac{\phi_{d_c}(0, s) + \phi_{d_c}(1, s)}{2} \right\}$$

$$\phi_{x_{d_c} \gamma_{d_c}}(1, s) =$$

$$I^{-1} \left\{ \frac{\phi_{d_c}(0, s) - \phi_{d_c}(1, s)}{2} \right\}$$

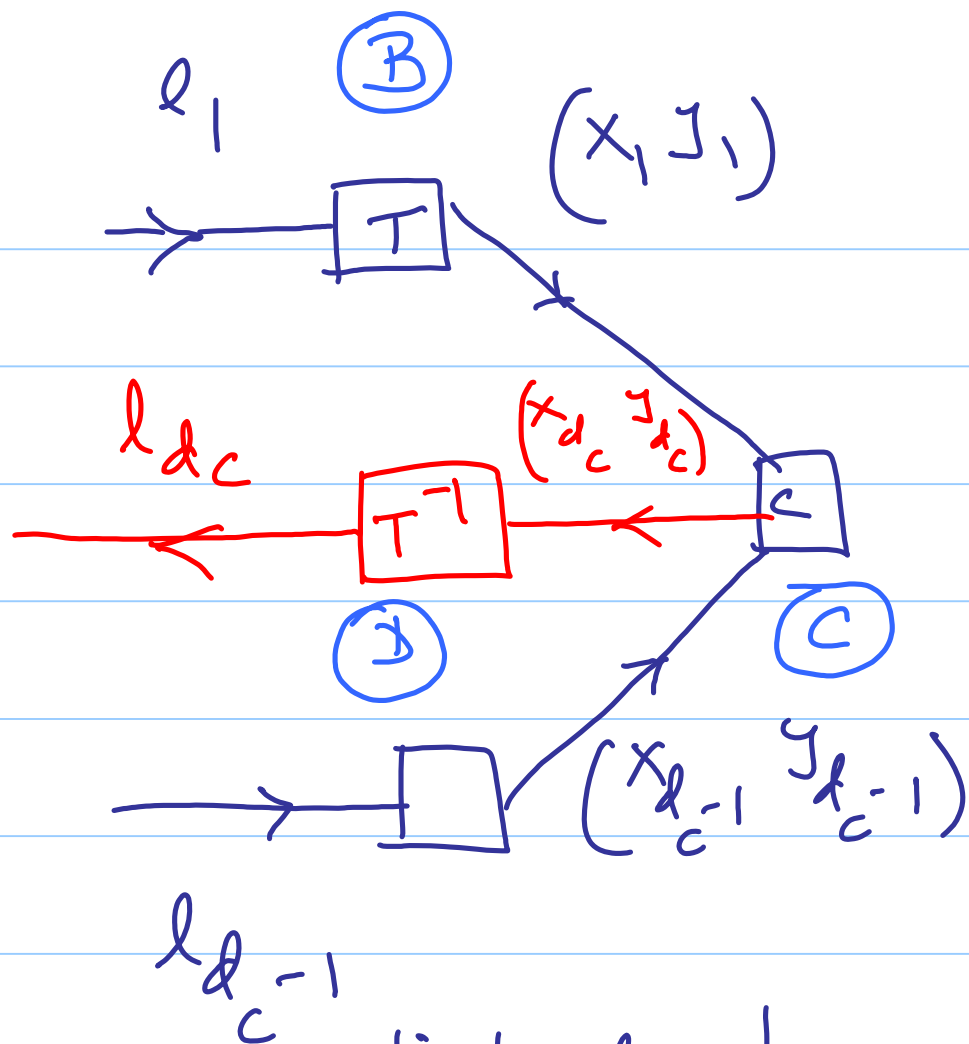
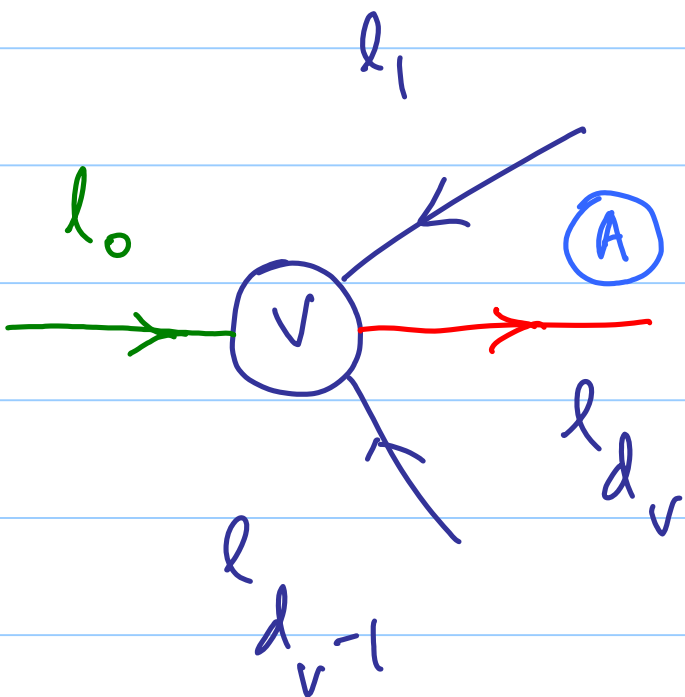
Letting  $x = x_{d_c}$   $\gamma = \gamma_{d_c}$   $l = l_{d_c}$ :



Finally

$$p_L(l) = \frac{\phi\left(0, -\ln \tanh\left(\frac{l}{2}\right)\right)}{\sinh(l)} \quad l > 0$$

$$= \frac{\phi\left(1, -\ln \tanh\left(-\frac{l}{2}\right)\right)}{-\sinh(l)} \quad l \leq 0$$



SUMMARY:

density evolution was accomplished by  
tracking densities across locations:

$A$   $B$   $C$   $D$

Lec 35 { Convergence } Concentration  
{ Theorem - LDPC codes

Recap \* { Completed discussion of  
density evolution w.r.t.  
BP decoding of LDPC codes

" The Capacity of Low-Density

Parity-check Codes Under Message

# - Passing Decoding<sup>u</sup>

T. J. Richardson & Ruediger

L. Urbanke

{ IEEE Trans. on Inform. Theory  
Feb. 2001  
}

---

Theorem For any  $\epsilon > 0$ ,

$$- \beta \epsilon^2 n$$

$$a) \quad \mathbb{P}_n \left\{ |z - \mathbb{E}(z)| > \frac{nd_v \epsilon}{2} \right\} \leq 2e \quad (1)$$

(concentration around the mean) (1)

b) For any  $\epsilon > 0$  and  $n > \frac{2\gamma}{\epsilon}$ ,

$$| \mathbb{E}(z) - nd_v p | < \frac{nd_v \epsilon}{2} \quad (2)$$

(convergence to the cycle-free case)

c) For any  $\epsilon > 0$  and  $n > \frac{2\gamma}{\epsilon}$ ,

we have:

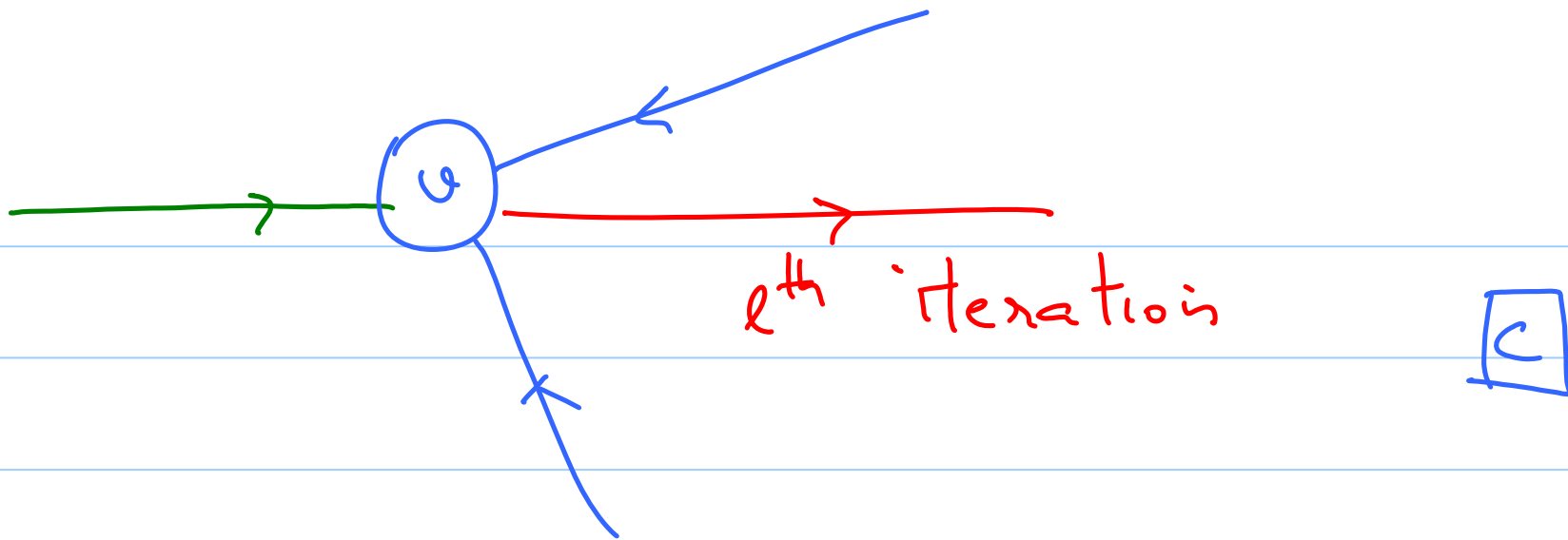
$$\mathbb{P}_n \left\{ |z - nd_V \epsilon| > nd_V \epsilon \right\} \leq 2e^{-\beta \epsilon^2 n} \quad (3)$$

(concentration around the cycle-free case)

Where:

- (i) the probabilities are computed over all choices of  $(t_v, d_c)$  - regular codes and over all channel realizations
- (ii)  $z = \#$  of incorrect messages passed from the  $n_{d_v}$  variable nodes to the  $n_{d_v}$  check nodes in the  $l^{\text{th}}$  iteration, ( $n$  denotes the  $\#$  of check nodes and  $n_{d_v} = n d_c$ ).





$p$  below is the probability of an incorrect message being passed during the  $\ell^{\text{th}}$  iteration in the tree-like case.

b) For any  $\epsilon > 0$  and  $n > \frac{2\gamma}{\epsilon}$ ,

$$\left| \mathbb{E}(z) - nd_v p \right| < \frac{nd_v \epsilon}{2} \quad (2)$$

$p$  derived from density evolution

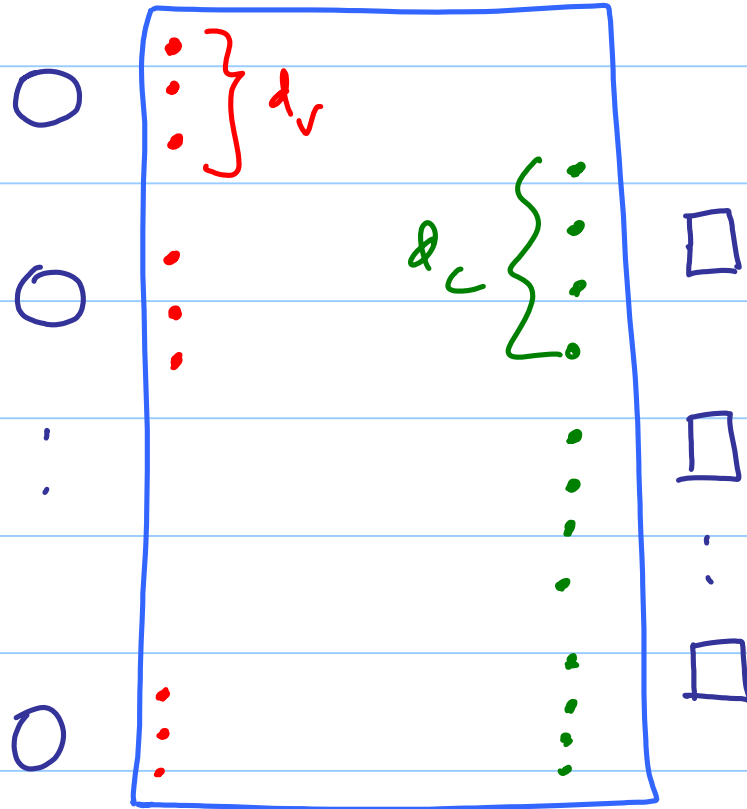
Also, in the theorem,

$$\beta = \beta(d_v, d_c, \ell) \text{ and } \gamma = \gamma(d_v, d_c, \ell)$$

are constants.

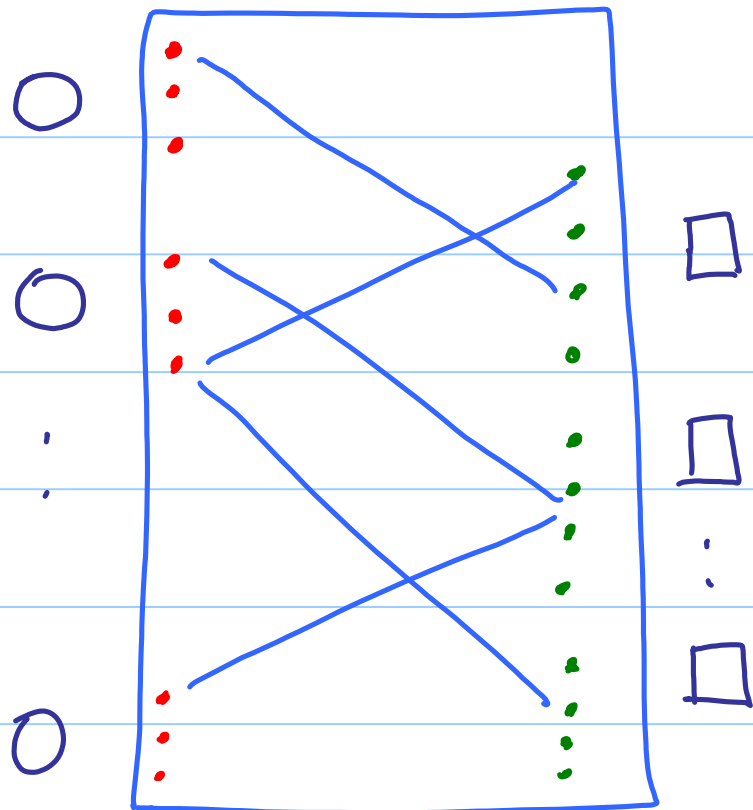
{ The ensemble of  $(d_v, d_c)$  - regular LDPC  
codes.

$n$  variable  
nodes, each  
of degree  
 $= d_v$



$n$  check  
nodes,  
each of  
degree  $= d_c$

each  
variable  
is associated  
with  $d_v$   
"sockets"



each  
check  
node is  
linked to  
 $d_c$   
"sockets"

$$(nd_v)!$$

$$nd_v = rd_c$$

possible Tanner  
graphs preserving the  
 $(d_v, d_c)$ -regular property.

Pf (of part b) of Theorem).

Let  $z_i$  be the # of incorrect messages  
passed along the  $i^{\text{th}}$  edge  $e_i$  during  
the  $\ell^{\text{th}}$  iteration.  $z_i \in \{0, 1\}$ .

$$z = \sum_{i=1}^{nd_v} z_i$$

$$\therefore \mathbb{E}\{z\} = \sum_{i=1}^{nd_v} \underbrace{\mathbb{E}\{z_i\}}_{\text{(by symmetry)}} = nd_v \mathbb{E}\{z_1\}$$



focus on this

$$\mathbb{E} \{ z_1 \} = \mathbb{E} \left\{ z_1 \mid N_{e_1}^{2\ell} \text{ is T-L} \right\} \rightarrow \begin{matrix} \phi \\ \text{T-L} \end{matrix}$$

$$P_2 \left\{ N_{e_1}^{2\ell} \text{ is T-L} \right\} \left\{ \begin{matrix} \text{true} \\ \text{like} \end{matrix} \right.$$

$$1 - \frac{\gamma}{n} \leq \cdot \leq 1 + \mathbb{E} \left\{ z_1 \mid N_{e_1}^{2\ell} \text{ is not T-L} \right\} \leq 1$$

$$\leq \frac{\gamma}{n} \leftarrow P_2 \left\{ N_{e_1}^{2\ell} \text{ is not T-L} \right\}.$$

When  $n$  is large, turns out that

$$\left\{ P_2 \left\{ N_e^{2\ell} \text{ is T-L} \right\} \geq 1 - \frac{\gamma}{n} \right\}$$

$\gamma$  is a constant.

$$p - \frac{\gamma}{n} \leq p \left(1 - \frac{\gamma}{n}\right) \leq \mathbb{E}\{z_1\} \leq p + \frac{\gamma}{n}$$

$$\therefore \quad \left| \mathbb{E}\{z_1\} - p \right| \leq \frac{\gamma}{n}$$

$$\therefore \quad \left| \mathbb{E}\{z\} - nd_v p \right| \leq nd_v \left( \frac{\gamma}{n} \right)$$

We assume  $n$  large enough so that

$$\frac{\gamma}{n} < \frac{\epsilon}{2}$$

$\therefore$

$$\left| \mathbb{E}\{z\} - nd_v p \right| \leq \frac{nd_v \epsilon}{2}$$



This proves ②. We will skip the lengthy proof of ①.

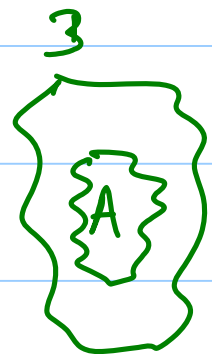
To prove ③ given ① and ②, we argue as follows:

$$P_n \left\{ |z - nd_v| > nd_v \epsilon \right\}$$

$$= P_n \left\{ \left| \left[ z - \mathbb{E}\{z\} \right] + \left[ \mathbb{E}\{z\} - nd_v \right] \right| > nd_v \epsilon \right\}$$

but for large enough  $n$  when

$$\left| \mathbb{E}\{z\} - nd_v \right| < \frac{nd_v \epsilon}{2}$$



$$\leq P_n \left\{ \left| z - \mathbb{E} \{ z \} \right| > \frac{n d_v \epsilon}{2} \right\}$$

$$\leq 2c \quad \text{from part a)}$$

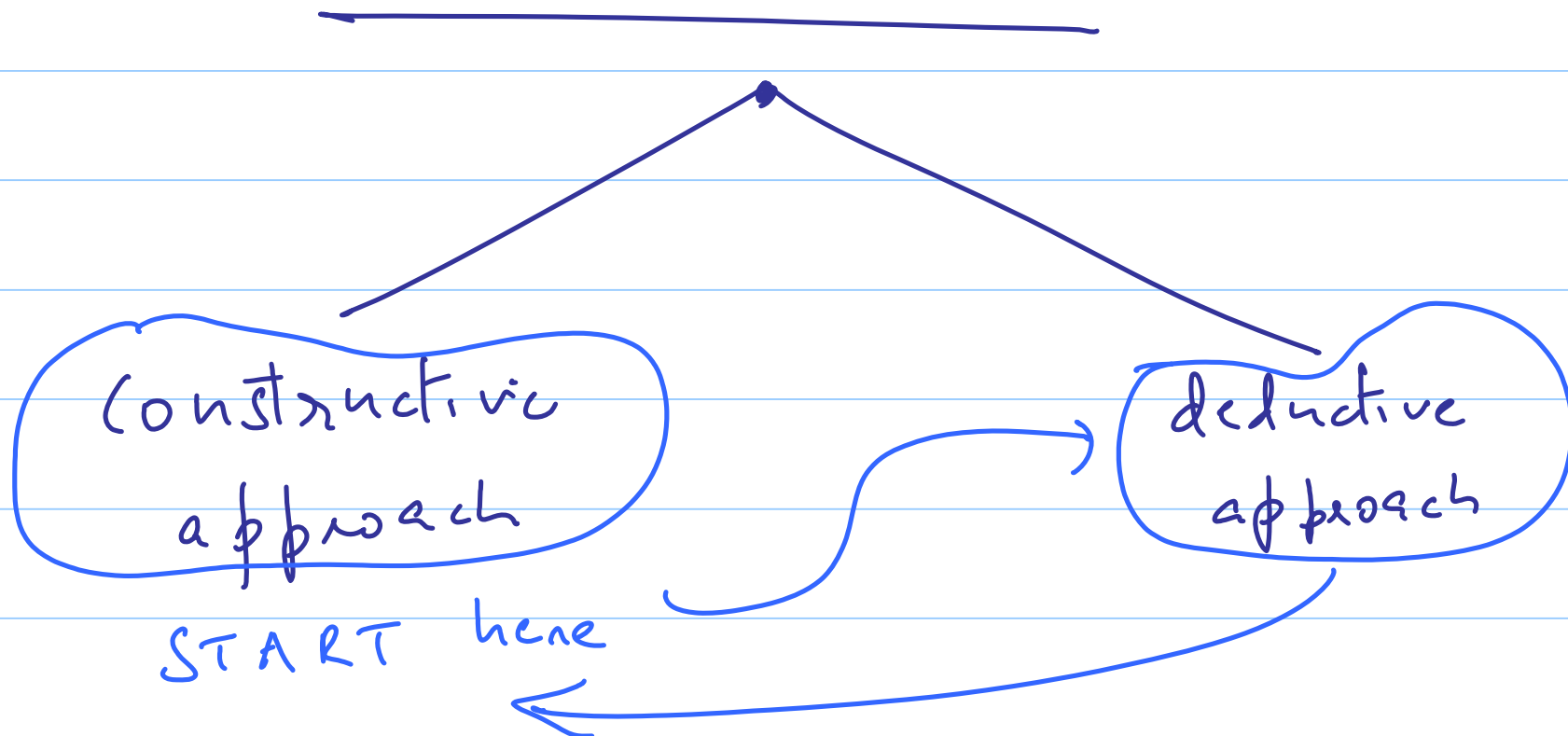
---

Lec 36 { A construction for  
Finite Fields

Recap \*

- { reviewed discussion of  
density evolution w.r.t  
the BP decoding of LDPC  
codes
- { concentration & convergence  
theorems

\* { finite fields are the basis upon  
which the widely-used classes of  
BCH and Reed-Solomon codes are  
built



Eg of a finite field  $\rightarrow$  imaginary

$$\mathbb{C} = \mathbb{R}[i], \quad i^2 = -1 \quad \sqrt{-1}$$

{ collection of all polynomials in  
i having real coefficients

$$\mathbb{R}[x] / (x^2 + 1)$$

$$\mathbb{R}[x] \pmod{x^2 + 1}$$

$\mathbb{R}[x]$  the ring of all  
 polynomials in  $x$   
 over the real numbers

forms  
 a field  
 $\mathbb{Z}_p$

$$\mathbb{Z} \pmod{n}$$

$$\mathbb{Z} \pmod{p}$$

$$\Downarrow$$

$$(p \text{ prime})$$

$$I_n \quad \mathbb{R}[x] / (x^2 + 1)$$

$$x^2 = (x^2 + 1) - 1 = -1$$



$$\mathbb{R}[i]$$

$$\boxed{i^2 + 1 = 0}$$

$$= \left\{ a + ib \mid \begin{array}{l} a, b \\ \in \mathbb{R} \end{array} \right\}$$

$$i^2 = -1$$

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  $m = 1$ ,  $\mathbb{Z}_p$

$$\mathbb{Z}_p = \left\{ \text{set of integers modulo } p \right\}$$

is a finite field of size  $p$ .



We focus therefore on the case  $m \geq 2$ .

\* { Let  $f(x)$  be a (monic) irreducible  
polynomial of degree  $m$  over  $\mathbb{F}_p$

(monic  $\Rightarrow$  highest degree coefficient  
 $= 1$ )

$$\text{Let } f(x) = \sum_{i=0}^m f_i x^i$$

we say that  $f(x)$  is monic of  
degree  $m$  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:

$$f(x) = g(x)h(x)$$

where

$$0 < \deg(g(x)), \deg(h(x)) < \deg(f(x))$$

Eg  $p = 2$   $\mathbb{F}_2 = \{0, 1\}$

degree $d$	list of irreducible poly. of degree $= d$
1	$x, x+1$
2	$x^2 + x + 1$
3	$x^3 + x + 1, x^3 + x^2 + 1$
4	$x^4 + x + 1, x^4 + x^3 + 1,$ $x^4 + x^3 + x^2 + x + 1$

{ It can be shown, that for every prime  $p$ ,  
and every integer  $m \geq 1$ , irreducible  
polynomials of  $\deg = m$  exist.

{ Next, consider the set

$$\mathbb{F}_p[x] / (f(x))$$

where  $f(x)$  is irreducible of degree

$$= m \geq 2.$$

$\mathbb{F}_p[x] / (f(x))$  is the collection of  
equivalence classes where we define

$g_1 \sim g_2 \Leftrightarrow f(x) \mid g_1(x) - g_2(x)$
--

Ex Verify that the reflexive, symmetric  
& 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

$$(\mathbb{F}_q, +, \cdot)$$

ie.  $\overset{\Delta}{=} 0$

$i.e. = 1$

$\mathbb{F}_2$  contains 1,  $\therefore$  it contains

1,  $1+1$ ,  $\underbrace{1+1+1}_3$ ,  $1+1+1+1$ ,  $\dots$

This list must repeat entries.

$\therefore$   $\boxed{m = n}$  some  $n > m$ .

$\underbrace{1+1+1+\dots+1}_{m \text{ times}}$

$$\Rightarrow \boxed{(n-m)1 = 0}$$

Let  $p$  be the smallest integer s.t.

$p \cdot 1 = 0$ . Then,  $p$  must be prime,

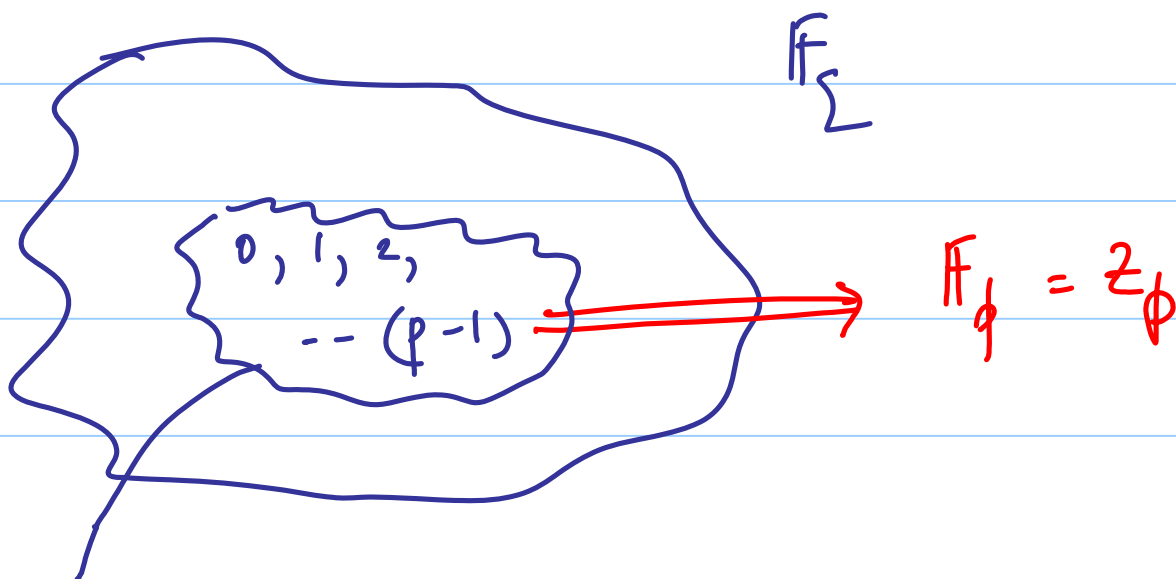
else  $p = p_1 p_2 \Rightarrow p_1 p_2 = 0$  in the ff

$\Rightarrow p_1 = 0$  or  $p_2 = 0$ .

This contradicts the minimality of  $p$ , hence  $p$  is prime.

---

{ This prime number is called the characteristic of the finite field.





↙ { in the finite field, within this  
set, one carries out arithmetic  
modul a prime  $p$ .

## Lec 37

{ Finite Fields

{ A deductive approach

### Recap

\* Motivating a construction  
for finite fields

$$\begin{aligned} * & \left\{ \begin{array}{l} \mathbb{F}_p[x] / (f(x)) \\ \deg(f(x)) = m. \end{array} \right. \end{aligned}$$

$\mathbb{F}_p[x] / (f(x))$  is a collection of equivalence classes

where we define

$$g_1(x) \sim g_2(x) \text{ if } f(x) \mid g_1(x) - g_2(x)$$

Exercise Verify that this is an

equivalence relation.

We will denote the equivalence of  $g_1(x)$

by  $[g_1(x)] = [g_1]$ .

Thm Let  $p$  be prime and  $f(x)$  be monic, irreducible of degree  $m$  over  $\mathbb{F}_p$ . Set

$$R \stackrel{\Delta}{=} \mathbb{F}_p[x] / (f(x))$$

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 ✓

$$[a(x)] + [b(x)] = [a(x) + b(x)].$$

$$\text{i.e.} = [0] = \left\{ \begin{array}{l} \text{set of all} \\ \text{multiples of} \\ f(x) \end{array} \right\}$$

$$\text{inverse of } [a(x)] = [-a(x)]$$

Step 2 Under  $(R, \cdot)$  satisfies:

- CLOSURE ✓
- ASSOCIATIVE ✓
- i.e. ✓
- COMMUTATIVE ✓
- INVERSE

$$[a(x)] \cdot [b(x)] = [a(x) \cdot b(x)]$$

$$\text{i.e.} = [1].$$

{ We will demonstrate through example, the  
presence of a multiplicative inverse.

Ex  $p = 2 \quad f(x) = x^2 + x + 1$

$$[a(x)] = [x + 1]$$

$$[a(x)]^{-1} = ?$$

$$\begin{bmatrix} a(x) \\ b(x) \end{bmatrix} = \begin{bmatrix} a(x) & b(x) \\ 1 \end{bmatrix}.$$

We make use of the extended Euclidean division algorithm (EDA).



	$f(x) =$ $x^4 + x + 1$	$(x + 1)$	quotient
$x^4 + x + 1$	1	0	
$x + 1$	0	1	$x^3 + x^2 + x$
(Rem) 1	1	$x^3 + x^2 + x$	
0			

$$\begin{array}{r}
 x^3 + x^2 + x \\
 \hline
 x+1 \quad \overline{) \quad x^4 + x^3 + 1}
 \end{array}$$

$$\begin{array}{r}
 x^4 + x^3 \\
 \hline
 \end{array}$$

$$\begin{array}{r}
 x^3 + x + 1
 \end{array}$$

$$\begin{array}{r}
 x^3 + x^2 \\
 \hline
 \end{array}$$

$$\begin{array}{r}
 x^2 + x + 1
 \end{array}$$

$$\begin{array}{r}
 x^2 + x \\
 \hline
 1
 \end{array}$$

$$\therefore 1 = 1(x^4 + x + 1) + (x+1)(x^3 + x^2 + x)$$

$$\therefore [1] = [(x+1)(x^3 + x^2 + x)]$$

$$= [(x+1)][(x^3 + x^2 + x)]$$

and hence  $[(x+1)]^{-1} = [(x^3 + x^2 + x)]$ .

---

hence in this way, we are always  
guaranteed to find an inverse.

---

hence  $\mathbb{F}_p[x] / (f(x))$  is a field.

---

Ex ( of the construction of a finite field  
of size = 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  $\alpha$  which is a zero of  $x^4 + x + 1$ :

$$\alpha^4 + \alpha + 1 = 0$$

$$\mathbb{F}_2[x] / (x^4 + x + 1) = \mathbb{F}_2[\alpha]$$

---

$$\mathbb{F}_2[\alpha] = \left\{ \sum_{i=0}^3 a_i \alpha^i \mid a_i \in \{0, 1\} \right\}$$

0

$$\alpha^4 = \alpha + 1$$

$$\alpha^{11} = \alpha^3 + \alpha^2 + \alpha$$

$$\alpha^0 = 1$$

$$\alpha^5 = \alpha^2 + \alpha$$

$$\alpha^{12} = \alpha^4 + \alpha^3 + \alpha^2 = \alpha^3 + \alpha^2 + \alpha + 1$$

$$\alpha^1 = \alpha$$

$$\alpha^6 = \alpha^3 + \alpha^2$$

$$\alpha^{13} = \alpha^4 + \alpha^3 + \alpha^2 + \alpha = \alpha^3 + \alpha^2 + \alpha + 1$$

$$\alpha^2 = \alpha^2$$

$$\alpha^7 = \alpha^4 + \alpha^3$$

$$\alpha^{14} = \alpha^4 + \alpha^3 + \alpha = \alpha^3 + 1$$

$$\alpha^2 = \alpha^2$$

$$= \alpha^3 + \alpha + 1$$

$$\alpha^3 = \alpha^3$$

$$\alpha^8 = \alpha^4 + \alpha^2 + \alpha$$

$$= \alpha^2 + 1$$

$$\alpha^{15} = \alpha^4 + \alpha = \alpha + \alpha + 1$$

$$= 1 \quad !!$$

$$\alpha^9 = \alpha^3 + \alpha$$

$$\alpha^{10} = \alpha^4 + \alpha^2$$

$$= \alpha^2 + \alpha + 1$$

From the table, we observe that the finite field also has representation in terms of the element  $\alpha$  and the various powers of  $\alpha$ :

powers of  $\alpha$ :

$$\mathbb{F}_2 = \{0\} \cup \{\alpha^i \mid 0 \leq i \leq 14\}$$



# Lec 38 { Deductive Approach to Finite Fields

Recap A completed discussion of the general

$$\mathbb{F}_p[x] / (f(x))$$

construction of finite fields

## 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.  
i.e. = 1  
inverse  
comm. }

— { multiplication  
distributes over addition }

$$1 \in \mathbb{F}_\Sigma \Rightarrow$$

$$1, 1+1, 1+1+1, \dots, 1+1+1, \dots \left\{ \begin{array}{l} \text{all} \\ \in \mathbb{F}_\Sigma \end{array} \right.$$

$$\Rightarrow \quad m = n \quad \text{some } n > m$$

$$\stackrel{\Delta}{=} \underbrace{1+1+\dots+1}_{m \text{ times}} \Rightarrow \boxed{n-m=0}$$

Defn. The characteristic  $p$  of a finite field  $\mathbb{F}_\Sigma$  is the smallest integer  $p$  s.t.

$$p = \underbrace{1+1+\dots+1}_{p \text{ times}} = 0 \quad \text{in } \mathbb{F}_p$$

Thm 1 The char.  $p$  is a prime.

pf. Suppose  $p = a \cdot b$ ,  $1 < a, b < p$

Then  $a \cdot b = 0 \Rightarrow a = 0$  or  $b = 0$

which contradicts the minimality of  $p$ .

$\mathbb{F}_q$  contains the set  $\{0, 1, 2, \dots, q-1\}$

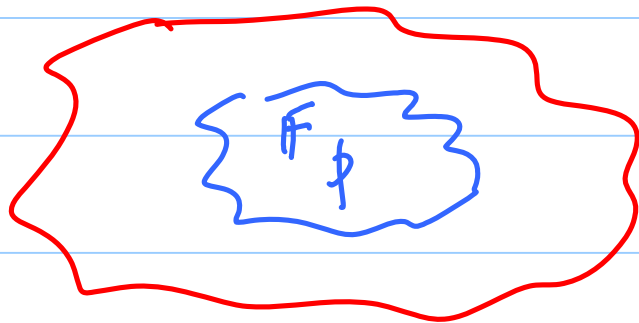
the arithmetic used to operate on these

elements is  $(\text{mod } q)$  arithmetic since

$q \equiv 0$  in  $\mathbb{F}_q$ . Hence

$\mathbb{F}_q$  contains a copy of

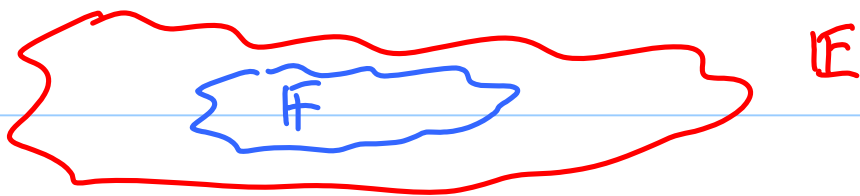
$\mathbb{F}_p$ .



$\mathbb{F}_q$

$\mathbb{F}_p = \mathbb{Z}_p$

It can be shown that  $E, F$  are fields and  $E \supseteq F$ , 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

since

$$F_\Sigma = \left\{ \sum_{i=1}^m a_i \gamma_i \mid a_i \in F_p \right\}$$

$\begin{matrix} \swarrow \\ \in F_\Sigma \end{matrix}$

where  $\{\alpha_1 \dots \alpha_m\}$  is a basis for  $\mathbb{F}_q / \mathbb{F}_p$ ,

it follows that  $\mathbb{F}_q$  is of size  $p^m$ .

Thm Every ff (finite field)  $\mathbb{F}_q$  has  
size  $q$  of the form  $q = p^m$ ,  $p$  prime,  
(more over  $p$  is the characteristic of  $\mathbb{F}_q$ )  
 $m \geq 1$

$$q \in \{2, 2^2, 3, 3^2, 2^3, \dots\}$$

# Multiplicative Structure of $\mathbb{F}_\Sigma$

Let  $\beta \in \mathbb{F}_\Sigma^* \triangleq \{x \in \mathbb{F}_\Sigma \mid x \neq 0\}$ .

Then  $\mathbb{F}_\Sigma^*$  contains

$$1 = \beta^0, \beta, \beta^2, \dots, \beta^a, \dots, \beta^b, \dots$$

by finiteness of  $\mathbb{F}_\Sigma$ ,  $\beta^a = \beta^b$  for some

integers  $b > a \Rightarrow \beta^{b-a} = 1$ .

This motivates:



Defn. The (multiplicative) order of  
nonzero  
 $\beta \in \mathbb{F}_\ell^\times$  is the smallest exponent  $e$  s.t.  
 $\beta^e = 1$

Lemma 1 Let  $\beta \in \mathbb{F}_\ell^\times$  have order  $e$ .

Then  $\beta^l = 1$  iff  $e \mid l$ .

Pf Let  $l = ue + v$ ,  $0 \leq v \leq e-1$

$\uparrow$                        $\uparrow$   
 quotient              remainder

Then,

$$\beta^l = \beta^{ue} \cdot \beta^v = (\beta^e)^u \cdot \beta^v$$

$$= (1)^u \cdot \beta^v = \beta^v = 1$$

but this contradicts the minimality of  $e$  unless  $v = 0$ .

hence  $e \mid l$

///

Lemma 2 Let  $\beta \in \mathbb{F}_{\Sigma}^*$  have order  $e$ .

Then  $\beta^l$  has order  $= \frac{e}{(l, e)}$ .

Eg  $e = 15$   $l = 10$   
 $\rightarrow \text{order}(\beta^{10}) = \frac{15}{(15, 10)} = 3$

Pf. (Exercise, straightforward).

Lemma 3 Let  $\beta, \gamma \in \mathbb{F}_{\Sigma}^*$  have orders

$a, b$  s.t.  $(a, b) = 1$ . Then

$(\beta\gamma)$  has order  $= ab$ .

Q8. Let  $(\beta\gamma)$  have order  $= l$ .

Then  $\beta^l = \gamma^{-l}$

$$\Rightarrow \frac{a}{(a, l)} = \frac{b}{(b, l)}$$

{ since  $\gamma, \gamma^{-1}$   
have the same  
order }

$$\Rightarrow a(b, l) = b(a, l)$$

$$\therefore a \mid (a, l) \Rightarrow a \mid l$$

Similarly  $b \mid (b, l) \Rightarrow b \mid l$

$$\therefore ab \mid l.$$

On the other hand,

$$(p\gamma)^{ab} = (p^a)^b \cdot (\gamma^b)^a = 1$$

hence  $l \mid ab$ .  $\therefore l \leq ab \quad //$

---

Amongst all the elements in  $\mathbb{F}_\Sigma^*$ , let

$\beta$  have maximal order  $\leq n$ .

Claim: Every element  $\theta \in \mathbb{F}_\Sigma^*$  has

Order dividing  $n$ .

If Let  $\theta$  have order  $= l$ ,  $l \nmid n$ .  
 (via Lemma 3)  
 Then we can write

$$n = p_1^{e_1} \cdots p_s^{e_s}$$

$$l = p_1^{a_1} \cdots p_s^{a_s}$$

with  $a_i > e_i$  some  $i$ . WLOG assume  $i = 1$ .

Then,

$$\left. \begin{aligned} n &= p_1^{e_1} (n_1) \\ l &= p_1^{a_1} (l_1) \end{aligned} \right\} \text{(say)}$$

Now consider

$$\underbrace{\beta \phi_1^{e_1}}.$$

$$\ominus \phi_1$$

has order

$$= \frac{l}{(l, l_1)} = \phi_1^{a_1}$$

has order

$$= \frac{\pi}{(\pi, \phi_1^{e_1})} = \frac{\pi}{\phi_1^{a_1}} = \pi_1$$

Since  $(\pi_1, \phi_1^{a_1}) = 1$ , it follows

that  $\beta \phi_1^{e_1} \ominus \phi_1$  has order

$$= \phi_1^{a_1} \cdot \pi_1 > \phi_1^{b_1} \pi_1 = \pi$$

which contradicts the maximality of  $\pi$ .

Hence,  $\ell \mid \pi$ . ///

Lemma The maximal order of an

element  $\beta \in \mathbb{F}_{\Sigma}^* = \mathbb{F}_q^* = \mathbb{F}_p^m - 1$ .



Pf. Let  $\beta$  have maximal order  $= n$ .

To show that  $n = p^m - 1$ . Since  $\theta \in \mathbb{F}_p^*$

$\Rightarrow \theta$  has order dividing  $n$ ,

$$\theta^n = 1 \Rightarrow x^n - 1 \Big|_{x=\theta} = 0.$$

It follows that every nonzero element in

$\mathbb{F}_p$  is a zero of  $(x^n - 1)$ .

$$\therefore \pi \geq p^m - 1.$$

On the other hand, consider

$$1, p, p^2, \dots, p^i, \dots, p^j$$

$$\left. \begin{array}{l} \text{clearly of some } a, b, \\ b > a \end{array} \right\} \Leftrightarrow \begin{array}{l} p^a = p^b \\ p^{b-a} = 1 \end{array}$$

$$\text{with } b-a \leq p^m - 1.$$

$$\text{Hence } \pi \leq p^m - 1. \quad \therefore \boxed{\pi = p^m - 1} //$$

Corollary Every  $\mathbb{F}_q$  contains an

element  $\alpha$  of order  $= q-1$ . In terms of

$\alpha$ ,  $\mathbb{F}_q$  has the representation:

$$\mathbb{F}_q = \{0\} \cup \{\alpha^i \mid 0 \leq i \leq q-2\}$$

Defn. An element  $\alpha \in \mathbb{F}_q^*$  of order  
 $= (q-1)$  is called a primitive element of  $\mathbb{F}_q$ .

Eg consider  $\mathbb{F}_{2^4} = \mathbb{F}_2[x] / (x^4 + x + 1)$

( $x^4 + x + 1$  is irreducible over  $\mathbb{F}_2$ ),

Write  $\alpha$  for the equivalence class

$$\alpha = [x] \text{ in } \mathbb{F}_2[x] / (x^4 + x + 1).$$

(alternately we may regard  $\alpha$  as an imaginary

element satisfying  $\alpha^4 + \alpha + 1 = 0$

$\Uparrow$   
more practical!)

0

$$\alpha^5 = \alpha^2 + \alpha$$

1

$$\alpha^6 = \alpha^3 + \alpha^2$$

$\alpha$

$$\alpha^7 = \alpha^4 + \alpha^3$$

$\alpha^2$

$$= \alpha^3 + \alpha + 1$$

$\alpha^3$

$$\alpha^8 = \alpha^4 + \alpha^2 + \alpha$$
$$= \alpha^2 + 1$$

$$\alpha^4 = \alpha + 1$$

$$\alpha^9 = \alpha^3 + \alpha$$

$$\alpha^{10} = \alpha^4 + \alpha^2$$
$$= \alpha^2 + \alpha + 1$$

$$\alpha^{11} = \alpha^3 + \alpha^2 + \alpha$$

$$\alpha^{12} = \alpha^4 + \alpha^3 + \alpha^2$$
$$= \alpha^3 + \alpha^2 + \alpha + 1$$

$$\alpha^{13} = \alpha^4 + \alpha^3 + \alpha^2$$
$$+ \alpha =$$

$$\alpha^3 + \alpha^2 + 1$$

$$\alpha^{14} = \alpha^4 + \alpha + \alpha^3$$
$$= \alpha^3 + 1$$

$$\alpha^{15} = \alpha^4 + \alpha = 1!$$

Thus  $\alpha$  as defined above, is a  
p.e. (primitive element) of  $\mathbb{F}_{16}$ .

---

{ Preview of  
Minimal Polynomials (by example)

---

Eg  $p = 2$   $f(x) = x^4 + x + 1$

$$\mathbb{F}_2 = \mathbb{F}_2[x] / (x^4 + x + 1)$$

$$\mathbb{F}_q = \left\{ \sum_{i=0}^3 a_i d^i \mid a_i \in \{0,1\} \right\}$$

where  $d \stackrel{\Delta}{=} [x]$  in  $\mathbb{F}_2[x]/(x^4+x+1)$

and hence satisfies  $d^4 + d + 1 = 0$

---

$m_{\beta}(x)$  $\beta$  $x$ 

0

 $(x+1)$ 

1

2

4

8

$$(x+2)(x+2^2)(x+2^4) = (x^4 + x + 1)$$

$$(x+2^8)$$

 $2$  $2$  $2$  $2$ 

$$(x^4 + x^3 + x^2 + x + 1)$$

 $2^3$  $2^6$  $2^{12}$  $2^9$ 

$$(x^2 + x + 1)$$

 $2^5$  $2^{10}$ 

$$(x^4 + x^3 + 1)$$

 $2^7$  $2^{14}$  $2^{13}$  $2^{11}$ 

$$\mathbb{F}_2$$



$$\therefore (x^4 - x) = (x)(x+1) \cdot$$

$$(x^2 + x + 1) \cdot$$

$$(x^4 + x + 1)(x^4 + x^3 + 1)(x^4 + x^3 + x^2 + x + 1)$$

# Minimal Polynomials

Note: Every element  $\beta \in \mathbb{F}_\Sigma^*$  is a  
zero of  $x^{2-1} - 1$ . Hence every

element of  $\mathbb{F}_\Sigma$  is a zero of  
 $x^2 - x$ .

This motivates:

Defn      The minimal polynomial

$m_{\beta}(x)$  of  $\beta \in \mathbb{F}_{\Sigma}^*$  is the smallest  
degree monic polynomial of which  
 $\beta$  is a zero.

Lemma  $m_\beta(x)$  is irreducible

pf Suppose not. Then

$$m_\beta(x) = f(x) g(x) \text{ with}$$

$$0 < \deg(f), \deg(g) < \deg(m_\beta).$$

But

$$m_\beta(\beta) = 0 \Rightarrow f(\beta) g(\beta) = 0$$

$\Rightarrow$  either  $f(\beta) = 0$  or  $g(\beta) = 0$ .

This contradicts the minimality of  $m_\beta(x)$ . //

Lemma  $f(\beta) = 0 \Rightarrow m_\beta(x) \mid f(x)$

Pf Use the Euclidean division algorithm

to arrive at:

$$f(x) = \underset{\substack{\uparrow \\ \text{quotient}}}{a(x)} m_{\beta}(x) + \underset{\substack{\downarrow \\ \text{remainder of} \\ \deg < m_{\beta}(x)}}{b(x)}$$

$$\therefore f(\beta) = 0 \Rightarrow a(\beta) m_{\beta}(\beta) + b(\beta) = 0$$

$$\Rightarrow b(\beta) = 0$$

but this once again,

contradicts the minimality of  $m_{\beta}(x)$   
 unless  $b(x) = 0$ . Hence  $m_{\beta}(x) \mid f(x) //$

Corollary  $m_{\beta}(x) \mid x^2 - x$

Pf. Every element in  $\mathbb{F}_2$  is a zero of  $x^2 - x$ . Hence  $\beta^2 - \beta = 0$

$$\Rightarrow m_{\beta}(x) \mid x^2 - x. \quad //$$

---

# Lec 39 Subfields of a finite field

## Recap

- \* characteristic of a finite field
- \* multiplicative order of an element
- \* primitive elements
- \* minimal polynomials



Defn  $\mathbb{F}_q^* = \mathbb{F}_q \setminus \{0\}$  (set of all non zero elements)

Lemma 1 Let  $\beta \in \mathbb{F}_q^*$ ,  $q = p^m$ . Then  
 $\deg(m_\beta(x)) \leq m$ .

Pf Consider

$1, \beta, \beta^2, \beta^3, \dots, \beta^m$   
( $m+1$ ) elements

$$\begin{array}{c} \mathbb{F}_{p^m} \\ | \\ \mathbb{F}_p \end{array}$$

These cannot all be linearly independent  
for this would mean that the  $p^{m+1}$

elements

$$\sum_{i=0}^m a_i \beta^i, \quad a_i \in \mathbb{F}_p$$

are all distinct, which is impossible since

$$|\mathbb{F}_\Sigma| = p^m. \quad \text{Thus there exists a}$$

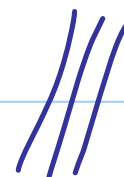
linear dependence expression of the form:

$$\sum_{i=0}^m c_i \beta^i = 0 \quad \left\{ \begin{array}{l} \text{with at least} \\ \text{one } c_i \neq 0 \end{array} \right.$$

$$\Leftrightarrow \beta \text{ is a zero of } \sum_{i=0}^m c_i x^i,$$

$$\text{Hence } m_{\beta}(x) \mid \sum_{i=0}^m c_i x^i$$

$$\Rightarrow \boxed{\deg(m_{\beta}(x)) \leq m}$$



Lemma 2 If  $\beta \in \mathbb{F}_q^*$ ,  $q = p^m$ , is

a primitive element, then

$$\deg(m_\beta(x)) = m.$$

Pf. Let  $\deg(m_\beta(x)) = s$ . From

Lemma 1,  $s \leq m$ . Since  $\beta$  is a p.e.

of  $\mathbb{F}_q$ , every element  $\theta$  in  $\mathbb{F}_q$  can be

expressed as a polynomial in  $\beta$  and

hence has an expression of the form

$$\theta = \sum_{i=0}^{s-1} a_i \beta^i, \quad a_i \in \mathbb{F}_p$$

Hence  $p^s \leq p^m$  by counting

$$\therefore s \leq m$$

$$\therefore \boxed{s = m}$$



Defn. The minimal polynomial  $m_\beta(x)$

of a primitive element  $\beta$  in  $\mathbb{F}_q$ ,

$q = p^m$ , is called a primitive polynomial  
(note that  $m_\beta(x)$  has degree =  $m$ )

Eg  $q = 2^4$   $\mathbb{F}_q = \mathbb{F}_2[x] / (x^4 + x + 1)$

$$= \mathbb{F}_2[\alpha] \quad \text{where } \alpha = [x]$$

and thus satisfies

$$\boxed{\alpha^4 + \alpha + 1 = 0.}$$

As seen before  $\alpha$  is a p.e.

$$\alpha^k \text{ has order } = \frac{p^m - 1}{(p^m - 1, k)} = \frac{15}{(15, k)}$$

and hence is primitive iff  $(15, k) = 1$ .

Given an integer  $n = \prod_{i=1}^r p_i^{e_i}$  the number of

integers  $\lambda$ ,  $0 \leq \lambda \leq n-1$  s.t.

$$(\lambda, n) = 1 \text{ equals } \phi(n) = \prod_{i=1}^r p_i^{(e_i-1)} (p_i - 1)$$

(Euler's totient fn.)

where  $\{p_i\}$  are all primes.

$$\therefore \phi(15) = \phi(5 \cdot 3) = (5-1)(3-1) = 8.$$

Hence  $\mathbb{F}_{16}$  contains 8 p.e.:

PRIMITIVE ELEMENT

COMMON MIN. POLY.

$$\{\alpha, \alpha^2, \alpha^4, \alpha^8\} \Leftrightarrow X^4 + X + 1$$

$$\{\alpha^7, \alpha^{11}, \alpha^{13}, \alpha^{14}\} \Leftrightarrow X^4 + X^3 + 1$$



Hence  $(x^4 + x + 1)$  and  $(x^4 + x^3 + 1)$  are

the only primitive polynomials  
associated to (this)  $\mathbb{F}_{16}$ .

---

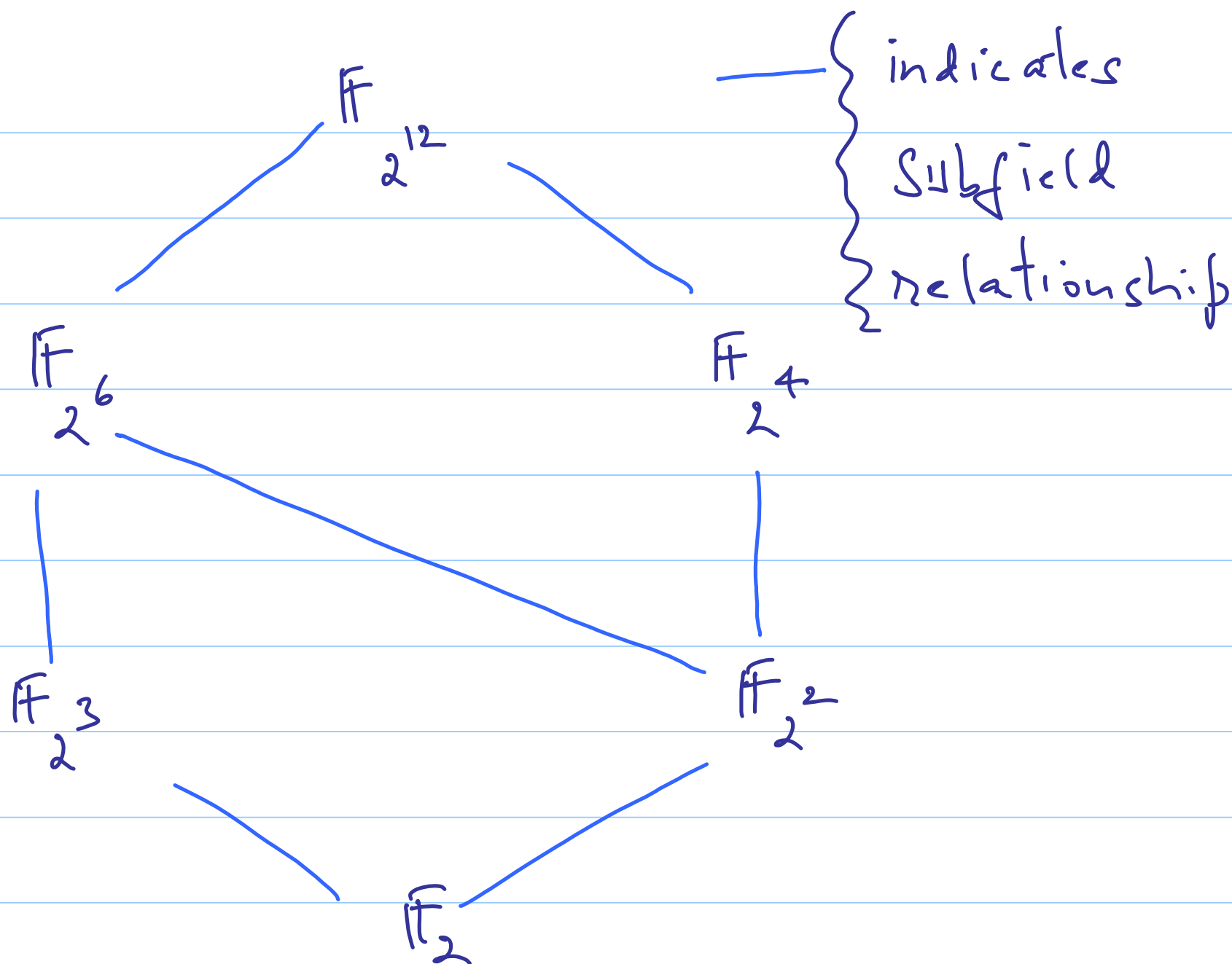
# Subfields of a finite field

Goal: characterize all the subfields of

a ff  $\mathbb{F}_{p^m}$ .

Ex  $p = 2 \quad m = 12$

$\mathbb{F}_2$  has the following subfield structure:



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:

Thm  $\mathbb{F}_{p^d} \subseteq \mathbb{F}_{p^m}$  iff  $d \mid m$ .

The proof is provided at the  
end of the lecture notes in the  
form of an appendix.

## A Useful Lemma

Lemma In any field  $\mathbb{F}$

characteristic  $p$ ,

$$(x + y)^p = x^p + y^p.$$

pf.

$$(x + y)^p = \sum_{i=0}^p \binom{p}{i} x^i y^{p-i}$$

$$= (x^p + y^p) \text{ because:}$$

$$\binom{p}{i} = \begin{cases} 1 & i = 0 \text{ or } i = p \\ 0 \pmod{p} & \text{else} \end{cases}$$

since:

$$\binom{p}{i} = \frac{p!}{(p-i)! \cdot i!} = p \frac{(p-1) \cdots (p-i+1)}{1 \cdot 2 \cdots i}$$



## Test for membership in a subfield

Thm Let  $\mathbb{F}_{p^d} \subseteq \mathbb{F}_{p^m}$ . Then

$\theta \in \mathbb{F}_{p^m}$  belongs to  $\mathbb{F}_{p^d}$  iff  
 $\theta^{p^d} = \theta.$

(proof is in the Appendix)



Ex  $\mathbb{F}_{16} = \mathbb{F}_2[\alpha]$  with  $\alpha^4 + \alpha + 1 = 0$

consider the subfield  $\mathbb{F}_2$ . Test:  $x^4 = x$ .

$\Rightarrow x = 0$  or else  $x = \alpha^k$  with

$$(\alpha^k)^4 = \alpha^k \Rightarrow \alpha^{3k} = 1 \Rightarrow$$

$$k \in \{0, 5, 10\}$$

$$\therefore \mathbb{F}_4 = \{0, 1, \alpha^5, \alpha^{10}\}$$

$\mathbb{F}_{2^4}$   
 $|$   
 $\mathbb{F}_{2^2}$   
 $|$   
 $\mathbb{F}_2$

Similarly,

$$F_2 \Rightarrow \boxed{\text{Test: } x^2 = x}$$

$\Rightarrow x=0$  or else  $x=\alpha$  with

$$\alpha^2 = \alpha \Rightarrow \alpha = 1 \Rightarrow k=0$$

$$\therefore F_2 = \{0, 1\}.$$

$$\mathbb{F}_{2^4} = \{0\} \cup \{\alpha^k \mid 0 \leq k \leq 15\}$$

$$\mathbb{F}_{2^2} = \{0, 1, \alpha^5, \alpha^{10}\}$$

$$\mathbb{F}_2 = \{0, 1\}$$

Thm Finite fields of size  $p^m$  exist  
for every prime  $p$ ,  $m \geq 1$

Pf When  $m = 1$ , the set of integers  
mod  $p$ ,  $\mathbb{Z}_p$ , is an example of a ff  
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  $x^{p^m} - x$ . Then this  
collection of  $p^m$  zeros can be shown to form the  
desired finite field.

(further details may be found in the  
Appendix)

---

Thm The polynomial  $x^{\frac{p^m}{p}} - x$  over  $\mathbb{F}_p$

has the factorization:

$$x^{\frac{p^m}{p}} - x = \prod_{\substack{1 \leq d \leq m \\ d \mid m}} \prod_{\substack{f(x) \\ \text{irred.} \\ \deg(f) = d}} f(x)$$

We illustrate by example. The proof may be found in the Appendix.

Eg. Let  $q = 2^4$  so  $p = 2$ . Then

$$x^4 - x = \underbrace{(x)(x+1)}_{\deg = 1} \underbrace{(x^2 + x + 1)}_{\deg = 2}$$

$$\Rightarrow \underbrace{(x^4 + x + 1)(x^4 + x^3 + 1)(x^4 + x^3 + x^2 + x + 1)}_{\deg = 4}$$

irreducible factors

Thm. Any two finite fields  $F_q, F_{q'}$  of the same size  $q = q'$  are isomorphic.

Pf. (sketch) Let  $\beta$  be a p.e. of  $F_q$  and  $m_\beta(x)$  its min. poly.

Then  $m_\beta(x) \mid x^{q'} - x$ .



Over  $\mathbb{F}_p$ , every element is a

zero of  $x^p - x$ . Hence some element

$\theta$  must be a zero of  $m_p(x)$ . Then

the map  $\phi: \beta \rightarrow \theta$

$$\sum_{i=0}^{m-1} a_i \beta^i \rightarrow \sum_{i=0}^{m-1} a_i \theta^i$$

$(a_i \in \mathbb{F}_p)$

can be shown to be an isomorphism.

We illustrate below with an example.

---

Ex Let  $q = 2^4$ .

—

We note from the factorization:

$$x^4 - x = (x)(x+1)(x^2+x+1)$$

$$(x^4+x+1)(x^4+x^3+1)(x^4+x^3+x^2+x+1)$$

that there are 3 different irreducible polynomials of degree 4 over  $\mathbb{F}_2$ .

Let

$$\mathbb{F}_2 = \mathbb{F}_2[x] / (x^4 + x^3 + x^2 + x + 1)$$

and

$$\mathbb{F}_2' = \mathbb{F}_2[x] / (x^4 + x + 1) .$$

Let  $\beta = [x]$  in  $\mathbb{F}_\Sigma$  so that

$$m_\beta(x) = x^4 + x^3 + x^2 + x + 1$$

and

$\alpha = [x]$  in  $\mathbb{F}'_\Sigma$  so that

$$m_\alpha(x) = x^4 + x + 1.$$

Then in  $\mathbb{F}_2'$  the element

$\theta = \alpha^3$  has min poly

$$m_{\theta}(x) = x^4 + x^3 + x^2 + x + 1.$$

Hence the map:

$$\phi : \begin{cases} \mathbb{F}_2' \\ \beta \end{cases} \longrightarrow \begin{cases} \mathbb{F}_2' \\ \theta \end{cases}$$

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.



Eg  $\phi = 2 \quad q = 2^4$

$$\mathbb{F}_q = \mathbb{F}_2[x] / (x^4 + x + 1)$$

with  $\alpha = [x]$  so that  $\alpha^4 + \alpha + 1 = 0$ .

Then

$$\mathbb{F}_q = \{0\} \cup \{\alpha^i \mid 0 \leq i \leq 14\}.$$

# POLYNOMIAL REPRESENTATION

0	$\alpha^4 = \alpha + 1$	$\alpha^{11} = \alpha^3 + \alpha^2 + \alpha$
$\alpha^0 = 1$	$\alpha^5 = \alpha^2 + \alpha$	$\alpha^{12} = \alpha^4 + \alpha^3 + \alpha^2 = \alpha^3 + \alpha^2 + \alpha + 1$
$\alpha^1 = \alpha$	$\alpha^6 = \alpha^3 + \alpha^2$	$\alpha^{13} = \alpha^4 + \alpha^3 + \alpha^2 + \alpha = \alpha^3 + \alpha^2 + 1$
$\alpha^2 = \alpha^2$	$\alpha^7 = \alpha^4 + \alpha^3$	$\alpha^{14} = \alpha^4 + \alpha^3 + \alpha = \alpha^3 + 1$
$\alpha^3 = \alpha^3$	$\alpha^8 = \alpha^4 + \alpha^2 + \alpha$	$\alpha^{15} = \alpha^4 + \alpha = \alpha + \alpha + 1 = 1 \quad !!$
	$\alpha^9 = \alpha^3 + \alpha$	
	$\alpha^{10} = \alpha^4 + \alpha^2$	
	$\alpha^{10} = \alpha^2 + \alpha + 1$	

With the aid of the polynomial

representation given above, one can

create the add-1 table given below:

# The add-1 table

$x$	$1+x$	$x$	$1+x$	$x$	$1+x$
0	1	$2^5$	$2^{10}$	$2^{11}$	$2^{12}$
1	0	$2^6$	$2^{13}$	$2^{13}$	$2^6$
2	$2^4$	$2^7$	$2^9$	$2^{14}$	$2^3$
$2^2$	$2^8$	$2^8$	$2^2$		
$2^3$	$2^{14}$	$2^9$	$2^7$		
$2^4$	2	$2^{10}$	$2^5$		

The add-1 table is frequently used in conjunction with thanes's method to add a string of powers of 2: Ex:

$$\begin{aligned}
 2^3 + 2 + 2^7 + 2^8 &= 2 + 2^3 + 2^7 + 2^8 \\
 &= 2 \left( 1 + 2^2 \left( 1 + 2^4 \left( 1 + 2 \right) \right) \right) = 2^2
 \end{aligned}$$

The diagram illustrates the factoring process with red curly braces and labels:

- A red brace under  $(1 + 2)$  is labeled  $2^0$ .
- A red brace under  $1 + 2^4(1 + 2)$  is labeled  $2^4$ .
- A red brace under  $1 + 2^2(1 + 2^4(1 + 2))$  is labeled  $2^2$ .
- A red brace under the entire expression  $2(1 + 2^2(1 + 2^4(1 + 2)))$  is labeled  $2$ .

# CYCLOTOMIC COSETS

These cosets will be used to explain the structure of minimal polynomials as well as to understand cyclic codes.

Let  $p$  be prime,  $m \geq 1$ .

If  $\alpha$  is a g.e. of  $\mathbb{F}_{p^m}$ , then

all arithmetic in the exponent of  $\alpha$

is conducted  $(\text{mod } p^m - 1)$  since

$\alpha$  has order  $= p^m - 1$ .

Eg  $\alpha^{p^m + 1} = \alpha^2$  etc.

Defn. Define  $a \sim b$  in  $\mathbb{Z}_{p^m-1}$

if 
$$a = p^k b \pmod{p^m-1}.$$

This can be verified to be an equivalence relation. The resulting equivalence classes are called the  $p$ -cyclic cosets  $\pmod{p^m-1}$ .



## Proof of equivalence:

$$(i) \quad a = p^0 a \quad \therefore a \sim a \quad \text{REFLEXIVE}$$

$$(ii) \quad a = p^k b \Rightarrow b = p^{n-k} a \quad (\text{mod } p^{n-1})$$

$$(iii) \quad a = p^k b, \quad b = p^l c \Rightarrow a = p^{k+l} c$$

TRANSITIVE

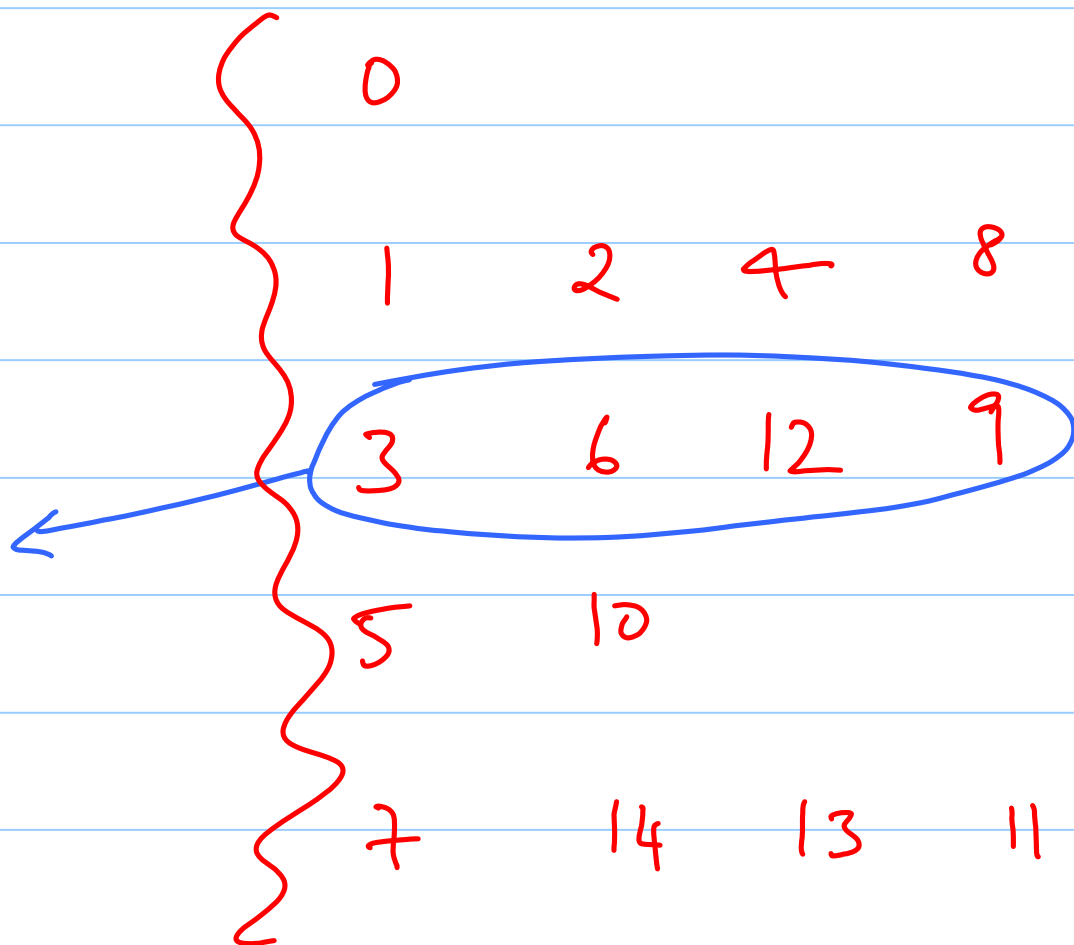
---

Ex  $\phi = 2$   $\eta = 2^4$

The 2-cyclotomic cosets (mod  $2^m - 1$ )  
are shown

alongside  $\Rightarrow$

is an  
equivalence  
class



The smallest element within each cyclic coset is called a coset leader.

Thus here,

0	}	are the coset leaders.
1		
3		
5		
7		

Thm. Let  $\beta \in \mathbb{F}_{p^m}$ . Then

$$m_\beta(x) = \prod_{l=0}^{d-1} (x - \beta^{p^l})$$

where  $d$  is the smallest integer such

that  $\beta^{p^d} = \beta$ , i.e.,  $\mathbb{F}_{p^d}$  is

the smallest subfield in  $\mathbb{F}_{p^m}$  which  
 $\beta$  is an element.

Pf (see the Appendix)

Defn. The elements  $\beta^{\phi^l}$ ,  
 $1 \leq l \leq d-1$  are called the  
conjugates of  $\beta$ .

Thus the theorem also asserts  
that all the conjugates of  $\beta$   
share the same minimal  
polynomial.

---

# APPENDIX

— { proofs to complete the  
lecture notes

Lemma 1 Let  $n \geq 2$ ,  $r, s \geq 1$ , then

$$n^r - 1 \mid n^s - 1 \quad \text{iff} \quad r \mid s$$

Pf Write:  $s = ar + b$   $b < r$

$$(n^s - 1) \pmod{n^r - 1}$$

$$= (n^{ar+b} - 1) \pmod{n^r - 1}$$



$$= \left( (n^2)^b \cdot n^b - 1 \right) \pmod{n^n - 1}$$

$$= (n^b - 1) = 0 \quad \text{iff} \quad b = 0$$

i.e., iff  $2 \mid 8$  //

---

Lemma 2 Let  $s \geq r \geq 1$ . Then

$$(x^r - 1) \mid (x^s - 1) \quad \text{iff} \quad r \mid s.$$

Pf  $(x^s - 1) \pmod{x^r - 1}$

$$= (x^{ar+b} - 1) \pmod{x^r - 1}$$

(letting  $s = ar + b$  as before)

$$= (x^b - 1) \pmod{x^n - 1}$$

$$= 0 \text{ iff } b = 0$$

$$\text{i.e. iff } n \mid b.$$

///

Corollary  $x^d - 1 \mid x^m - 1$  iff  $d \mid m$ .

Thm  $\mathbb{F}_{p^d} \subseteq \mathbb{F}_{p^m}$  iff  $d \mid m$ .

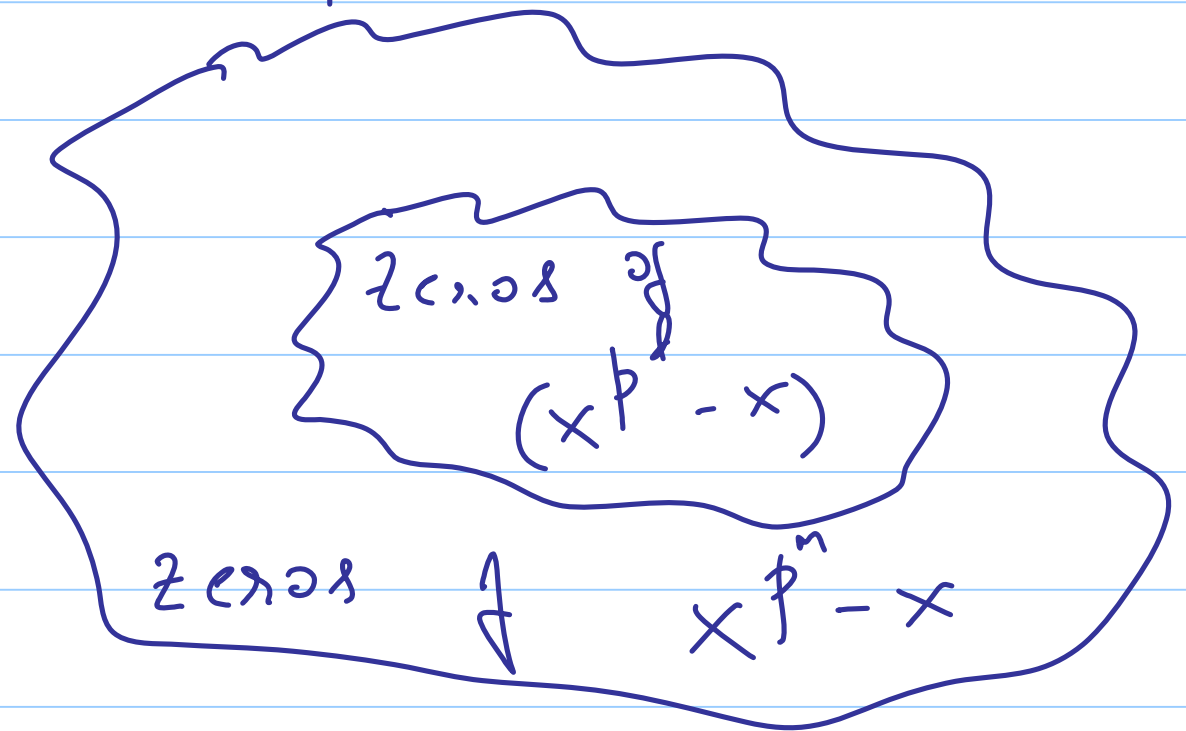
Pf a) Suppose  $d \mid m$

$$\Rightarrow (x^{p^d} - 1) \mid (x^{p^m} - 1)$$

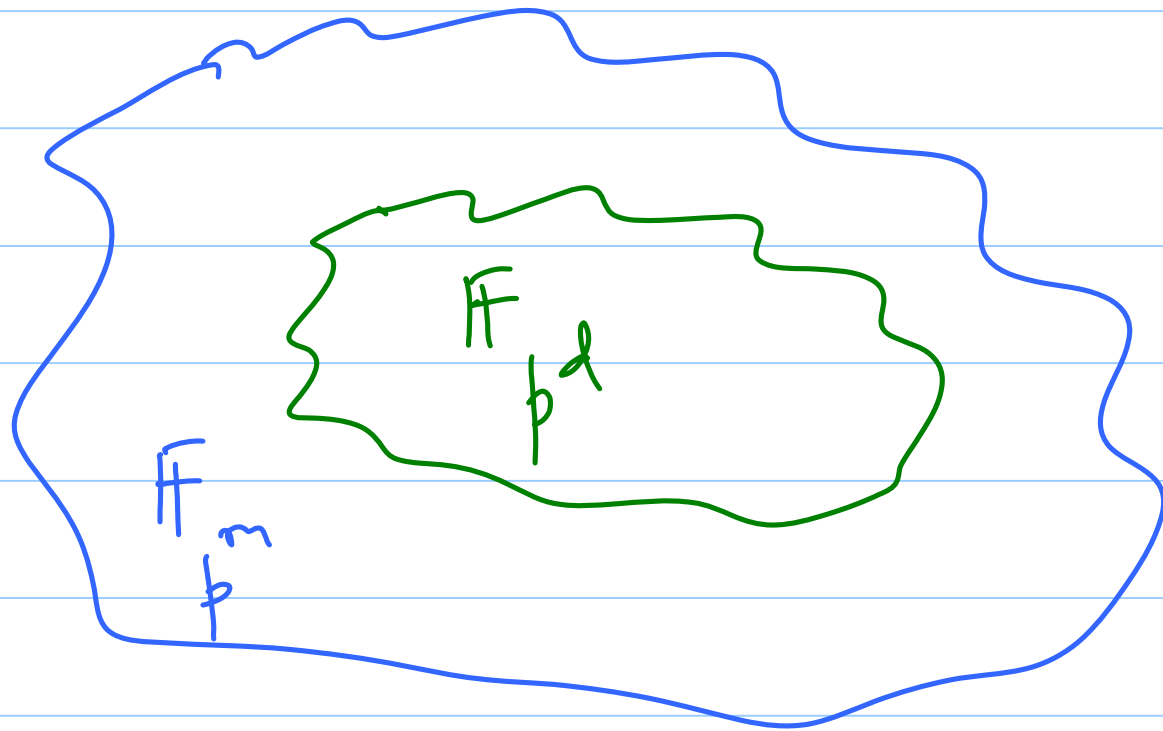
$$= (x^{p^d} - x) \mid (x^{p^m} - x)$$

But the elements of  $\mathbb{F}_{2^m}$  are  
precisely the  $2^m$  zeros of  $(x^{2^m} - x)$ .

Thus we have the picture:



and Lemma 3 will show that this picture has a subfield interpretation:



Lemma 3: The collection of  $p^d$  roots of  $x^{p^d} - x$  in  $\mathbb{F}_{p^m}$  form a ff of size  $p^d$ .

PS Let

$$S = \left\{ \theta \in \mathbb{F}_{p^m} \mid \theta^{p^d} - \theta = 0 \right\}.$$

To show that  $(S, +)$  is an Abelian group, it suffices to show that

$$a, b \in S \Rightarrow a - b \in S.$$

but

$$\begin{aligned}
 (a - b)^{\phi} &= (a + (-b))^{\phi} \\
 &= a^{\phi} + (-1)^{\phi} b^{\phi} \\
 &= a^{\phi} - b^{\phi} \\
 &= a - b
 \end{aligned}$$

$\therefore (a - b)$  is also a zero  $\dagger$



$$(x^{p^k} - x) \quad \therefore (a - b) \in S$$

$(S, \cdot)$  Hence it suffices to show that

$$a, b \in S \Rightarrow ab \in S \text{ and } a \in S \Rightarrow$$

$$a^{-1} \in S. \text{ But}$$

$$a^{p^k} = a, \quad b^{p^k} = b = (ab)^{p^k} = ab$$

$$a^{p^t} = a \Rightarrow (a^{-1})^{p^t} = a^{-1}$$

$$\therefore a^{-1} \in S$$

$\therefore S$  is a subfield of size  $\mathbb{F}_{p^d}$ .

---

b) next, suppose  $\mathbb{F}_{p^t} \subseteq \mathbb{F}_{p^m}$ .

Let  $\beta$  be a p.e. of  $\mathbb{F}_{p^d}$ .

Then  $\beta$  has order  $= p^t - 1$ .

Sei also  $\beta \in \mathbb{F}_p^m$ ,  $\beta^{p^m-1} = 1$

$$\therefore (p^k - 1) \mid (p^m - 1) \Rightarrow d \mid m \quad //$$

---

Thm Let  $\mathbb{F}_{p^d} \subseteq \mathbb{F}_{p^m}$ . Then

$\theta \in \mathbb{F}_{p^m}$  belongs to  $\mathbb{F}_{p^d}$  iff

$$\theta^{p^d} = \theta.$$

(test for membership in the subfield)

Pf. Suppose  $\theta^{p^d} = \theta$

$\Rightarrow \theta$  is a zero of  $x^{p^d} - x$ .

But the collection of zeros of

$x^{p^d} - x$  in  $F_{p^m}$  forms a subfield

$F_{p^d}$  of size  $p^d$ .  $\therefore \theta \in F_{p^d}$ .

(conversely, suppose  $\Theta \in \mathbb{F}_p^d$

$$\Rightarrow \Theta^p = \Theta.$$



Thm Finite fields of size  $p^m$  exist  
for every prime  $p$ ,  $m \geq 1$

Pf When  $m = 1$ , the set of integers  
mod  $p$ ,  $\mathbb{Z}_p$ , is an example of a ff  
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  $x^{p^m} - x$ . Then this  
collection of  $p^m$  zeros can be shown to form the  
desired finite field.

We begin with  $\mathbb{F}_p = \mathbb{Z}_p$ . Let us  
factorize  $x^{p^m} - x$  over  $\mathbb{F}_p$  to get:



$$(x^p - x) = \prod_{i=1}^r l_i(x) \prod_{j=1}^s f_j(x)$$

where the  $\{l_i(x)\}$  are irreducible polynomials of degree 1 (i.e., of the form  $(x-a)$ ) and the  $\{f_j(x)\}$  are also irreducible polynomials but of degree  $d_j \geq 2$ .

Then the field  $\mathbb{F}_{p^{d_1}} = \mathbb{F}_p[x] / (f_1(x))$ .

Then it follows that the number of zeros of  $(x^{p^m} - x)$  over  $\mathbb{F}_{p^{d_1}}$  is greater than the

# of zeros over  $\mathbb{F}_p$  since the

equivalence class  $[x]$  in  $\mathbb{F}_p[x] / f_1(x)$

is an additional zero.

Continuing in this fashion by picking an irreducible factor  $f$  of  $(x^{p^m} - x)$  of degree  $> 2$  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  $x^{p^m} - x$ . 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  $x^{\overset{m}{p}} - x$  over  $\mathbb{F}_p$

has the factorization:

$$x^{\overset{m}{p}} - x = \prod_{\substack{1 \leq d \leq m \\ d \mid m}} \prod_{\substack{f(x) \\ \text{irred.} \\ \deg(f) = d}} f(x)$$

Qd Let  $f(x)$  be irred  $f$   $\deg = d$  with

$d \mid m$ .

Then

$$\mathbb{F}_p[x] / (f(x)) \stackrel{\sim}{=} \mathbb{F}_{p^d}.$$

Let  $\beta = [x]$  in  $\mathbb{F}_p[x] / (f(x))$ . Then

$f(\beta) = 0$  and since  $f(x)$  is irred.  $f(x)$

is the min. poly.  $m_\beta(x)$ .

But  $m_{\beta}(x) \mid x^{p^d} - x \mid x^{p^m} - x$  since  $d \mid m$ .

hence  $f(x) \mid x^{p^m} - x$ .

Next let  $f(x)$  be irreducible of  $\deg = d$   
and let  $f(x) \mid x^{p^m} - x$ . T.S.:  $d \mid m$

Again, let  $\beta = [\alpha]$  in  $\mathbb{F}_p[x] / (f(x))$

$\cong \mathbb{F}_{p^d}$ . Then  $m_{\beta}(x) = f(x)$  in  $\mathbb{F}_{p^d}$ .

$$f(x) \mid x^p - x \Rightarrow \beta^p = \beta.$$

Let  $\alpha$  be a d.e. of  $\mathbb{F}_p$ . Then

$$\alpha = \sum_{i=0}^d a_i \beta^i, \quad a_i \in \mathbb{F}_p$$

$$\begin{aligned} \Rightarrow \alpha^p &= \left[ \sum_{i=0}^d a_i \beta^i \right]^p = \sum_{i=0}^d a_i^p \beta^{pi} \\ &= \sum_{i=0}^d a_i \beta^i = \alpha. \end{aligned}$$



Thus  $(p^d - 1) \mid p^m - 1 \Rightarrow d \mid m$

---

Thm. Let  $\beta \in \mathbb{F}_{p^m}$ . Then

$$m_\beta(x) = \prod_{l=0}^{d-1} (x - \beta^{p^l})$$

where  $d$  is the smallest integer such

that  $\beta^{p^d} = \beta$ , i.e.,  $\mathbb{F}_{p^d}$  is

the smallest subfield in  $\mathbb{F}_{p^m}$  which  
 $\beta$  is an element.

Pf Let

$$m_{\beta}(x) = \sum_{i=0}^d g_i x^i, \quad (g_i \in \mathbb{F}_p)$$

Then

$$m_{\beta}(\beta) = \sum_{i=0}^d g_i \beta^i = 0$$

$$\Rightarrow m_{\beta}(\beta^{p^l}) = \sum_{i=0}^d g_i \beta^{p^l \cdot i} = 0$$

$$= \sum_{i=0}^d g_i (\beta^{\phi^i})^i = 0$$

$\Rightarrow \beta^{\phi^i}$  is also a zero of  $m_{\beta}(x)$

On the other hand, define

$$h(x) = \sum_{j=0}^d h_j x^j = \prod_{i=0}^{d-1} (x - \beta^{\phi^i})$$

The coefficients of  $h(x)$  are elem.

Symm. fns  $f \left\{ \beta \phi^l \right\}_{l=0}^{d-1}$ .

$$\underline{\text{Eg}} \quad h_{d-1} = - \sum_{l=0}^{d-1} \beta \phi^l$$

$$\therefore h_{d-1}^{\phi} = - \sum_{l=0}^{d-1} \beta \phi^{l+1} = - \sum_{i=0}^{d-1} \beta \phi^i$$

$$\text{since } \beta \phi^d = \beta.$$

$$\text{Hence } h(x) \in \mathbb{F}_p[x]$$

$$\Rightarrow m_{\beta}(x) = \prod_{l=0}^{d-1} (x - \beta^l)$$

---

# lec 40-41 { Transform Approach to Cyclic Codes

Recap

\* { completed discussion on finite  
fields

\* completed discussion on min. poly.

\* subfield structure

$$* (x+y)^p = x^p + y^p$$

$$* \theta \in \mathbb{F}_{p^d} \Leftrightarrow \theta^{p^d} = \theta$$

\* finite fields of every size  $p^m$  exist

$$* x^{p^m} - x = \prod_{\substack{d|m \\ \deg f = d, f \text{ irreducible}}} f(x)$$



\* any two ffs of the same size  
are isomorphic

\* add-1 table

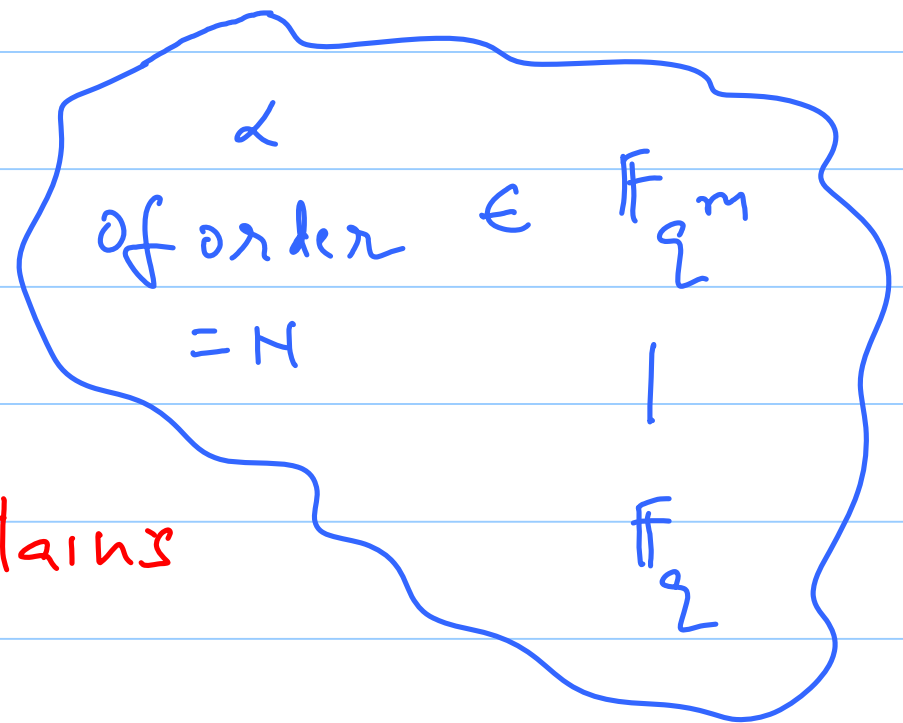
\*  $p$ -cyclotomic roots (mod  $p^m - 1$ )

$$* m_p(x) = \prod_{l=0}^{d-1} (x - \beta^{p^l})$$

---

Our interest is in cyclic codes of  
 block length  $N$  over  $\mathbb{F}_q$ ,  $q = p^e$ ,  
 $(q, N) = 1$ .

Goal: Find the  
 smallest field  $\mathbb{F}_{q^m}$   
 $\supseteq \mathbb{F}_q$  that also contains



an element  $\alpha \in \mathbb{F}_{\Sigma^m}$  of order  $= H$ .

Consider

$$1, \alpha, \alpha^2, \dots, \alpha^k, \dots, \alpha^l \pmod{N}$$

Since  $\mathbb{Z}_H$  is finite,

$$\text{Some } \alpha^k = \alpha^l, \quad l > k, \\ \pmod{N}$$

i.e.,

$$g^{l-k} = 1 \pmod{N}.$$

Let  $m$  be the smallest power of  $g$

s.t.  $\boxed{g^m = 1 \pmod{N}},$   $m$  is then

called the multiplicative order of  $g$

$\pmod{N}$ .

Note: this implies that

$$q^m = 1 \pmod{N}$$

$$\Rightarrow N \mid (q^m - 1)$$

$\Rightarrow \mathbb{F}_q^m$  contains an element of order

$N$ , for example,  $\alpha = \beta^{\left(\frac{q-1}{N}\right)}$

where  $\beta$  is a p.e. of  $\mathbb{F}_q^m$ .

$$\underline{\text{Eg}} \left\{ \begin{array}{l} q = 2 \quad H = 15 \Rightarrow m = 4 \\ \alpha = \beta^1, \quad \beta \text{ primitive in } \mathbb{F}_{2^4} \end{array} \right.$$

$$\left\{ \begin{array}{l} q = p^e \quad H = (q-1) \Rightarrow m = 1 \\ \alpha = \beta, \quad \beta \text{ p.c. of } \mathbb{F}_{p^e} \end{array} \right.$$

FINITE FIELD

TRANSFORM

Defn Let  $(q, N) = 1$ ,  $m$  be the

multiplicative order of  $q \pmod{N}$ .

Let  $\alpha$  be an element of order  $N$

in  $\mathbb{F}_\Sigma$ . Let  $(a_t)_{t=0}^{N-1}$  be a vector

of length  $N$  over  $\mathbb{F}_m$ . Then

$$\hat{a}_\lambda = \sum_{t=0}^{N-1} a_t \alpha^{\lambda t}, \quad 0 \leq \lambda \leq N-1,$$



is called the finite field transform  
(fft) of  $(a_t)$ .

### Properties

1). Linearity:

$$(a_t) \Leftrightarrow (\hat{a}_\lambda) \quad (b_t) \Leftrightarrow (\hat{b}_\lambda)$$

$$\Rightarrow (a_t + \theta b_t) \Leftrightarrow (\hat{a}_\lambda + \theta \hat{b}_\lambda)$$

(Pf. Exercise!)

all  $\theta \in \mathbb{F}_{2^m}$ .

## CYCLIC SHIFTS

2). Let

$$b_t = a_{(t-\tau) \pmod N} \quad 0 \leq t \leq N-1$$

for some  $0 \leq \tau \leq N-1$ . Then

$(b_t)$  is called a cyclic shift of  $(a_t)$ .

In the sequel, we will always assume that subscripts of  $a$  are

are always computed  $(\text{mod } H)$   
 and will therefore simply write  
 $(a_{t-\tau})$  in place of  $(a_{t-\tau} \text{ mod } H)$ .

Now

$$\sum_{t=0}^{H-1} a_{t-\tau} \alpha^{\lambda t}$$

$$= \sum_{t=0}^{H-1} a_{t-\tau} \alpha^{\lambda(t-\tau)} \alpha^{\lambda \tau}$$

$$= \alpha^{\lambda T} \left[ \sum_{s=0}^{H-1} a_s \alpha^{\lambda s} \right],$$

Setting  
 $t - \uparrow = s$   
 $(\text{mod } H)$

$$= \alpha^{\lambda T} \hat{a}_\lambda.$$

Thus

$$(a_t) \Leftrightarrow (\hat{a}_\lambda)$$

$$\Rightarrow (a_{t-\uparrow}) \Leftrightarrow \alpha^{\lambda T} (\hat{a}_\lambda)$$

3)

## INVERSION FORMULA:

$$a_t = (N)^{-1} \sum_{\lambda=0}^{N-1} \hat{a}_\lambda \omega^{-\lambda t}$$

Pf. RHS =  $(N)^{-1} \sum_{\lambda=0}^{N-1} \sum_{s=0}^{N-1} a_s \omega^{\lambda(s-t)}$

$$= (N)^{-1} \sum_{s=0}^{N-1} a_s \left[ \sum_{\lambda=0}^{N-1} \omega^{\lambda(s-t)} \right]$$

= N (when  $s=t$ )

$$= \begin{bmatrix} N(s-t) & -1 \\ \omega^{(s-t)} & -1 \end{bmatrix} \quad s \neq t$$

$$= 0 \quad \text{since} \quad \alpha^N = 1$$

$$= a_t$$


---

#### 4). CYCLIC CONVOLUTION

$$(a_t) \Leftrightarrow (\hat{a}_\lambda) \quad (b_t) \Leftrightarrow (\hat{b}_\lambda)$$

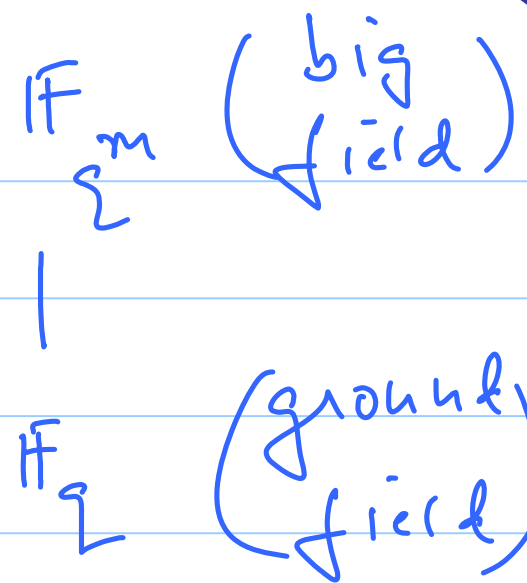
$$\Rightarrow (\hat{u}_\lambda) = (\hat{a}_\lambda \hat{b}_\lambda) \quad \text{where} \quad u_t = \sum_{\tau=0}^{N-1} a_{t-\tau} b_\tau$$

Pf. (Exercise!)

5). CONJUGATE

Suppose  $a_t \in \mathbb{F}_\Sigma$ ,  
all  $t$ , i.e.,

$(a_t)$  lies in the  
"ground field"





Then

$$\hat{a}_{\lambda \ell} = \left[ \hat{a}_{\lambda} \right]_{\ell} \quad \forall \lambda$$

conjugacy  
relation

( where  $\lambda \ell$  in the subscript is  
computed  $(\text{mod } \pi)$  ).

conversely  $\hat{a}_{\lambda \ell} = \left[ \hat{a}_{\lambda} \right]_{\ell} \quad \text{all } \lambda$

2)  $a_t \in \mathbb{F}_{\ell}$ , all  $t$ .

# PERIODIC SEQUENCE INTERPRETATION

since arithmetic in the  
subscript of  $a_t$  or else  $\hat{a}_x$  is  
always computed (mod  $\pi$ ), it is often  
convenient to think of both  
 $(a_t)$  as well as  $(\hat{a}_x)$  as being  
periodic sequences over  $\mathbb{F}_{2^m}$  of

period  $H$ ,  $i \in \mathbb{C}$ ,

$$\dots \hat{q}_{H-1} \quad \underbrace{\hat{q}_0 \quad \hat{q}_1 \quad \hat{q}_2 \quad \dots \quad \hat{q}_{H-1}}_{\text{one period}} \quad \hat{q}_0 \quad \hat{q}_1 \quad \dots \quad \hat{q}_{H-1} \quad \hat{q}_0 \quad \dots$$

$$\dots \hat{\hat{q}}_{H-1} \quad \underbrace{\hat{\hat{q}}_0 \quad \hat{\hat{q}}_1 \quad \hat{\hat{q}}_2 \quad \dots \quad \hat{\hat{q}}_{H-1}}_{\text{one period}} \quad \hat{\hat{q}}_0 \quad \hat{\hat{q}}_1 \quad \dots \quad \hat{\hat{q}}_{H-1} \quad \hat{\hat{q}}_0 \quad \dots$$

This is consistent with the defn:

$$\hat{a}_x = \sum_{t=0}^{N-1} a_t \alpha^t$$

$$\hat{a}_{x+N} = \hat{a}_x$$

since  $\alpha$  has order  $N$ .

---

Pf (of  $\hat{a}_{\lambda \mathbb{Z}} = \begin{bmatrix} \hat{a} \\ \lambda \end{bmatrix}^{\mathbb{Z}}$  when  $a_t \in \mathbb{F}_{\mathbb{Z}}$  all  $t$ )

$$\hat{a}_{\lambda \mathbb{Z}} = \sum_{t=0}^{N-1} a_t \alpha^{\lambda \mathbb{Z} t}$$

$$= \sum_{t=0}^{N-1} (a_t \alpha^{\lambda t})^{\mathbb{Z}}$$

since  
 $a_t^{\mathbb{Z}} = a_t$

$$= \left[ \sum_{t=0}^{N-1} a_t \alpha^{\lambda t} \right]^{\mathbb{Z}}$$

since  
 $\mathbb{Z} = \mathbb{F}^e$

$$= \left[ \hat{a}_{\lambda} \right]^q$$


---

Converse: Next suppose

$$\hat{a}_{\lambda}^q = \left[ \hat{a}_{\lambda} \right]^q \quad \text{all } \lambda, \text{ then}$$

$$\left[ a_t \right]^q = \left[ N^{-1} \sum_{\lambda} \hat{a}_{\lambda} \lambda^{-\lambda t} \right]^q$$

$$= (N^{-1})^q \sum_{\lambda} \left[ \hat{a}_{\lambda} \right]^q \lambda^{-\lambda q t}$$

$$= N^{-1} \sum_{\lambda=0}^{N-1} \hat{a}_{\lambda q} \alpha^{-\lambda q t}$$

Set  $\mu \equiv \lambda q \pmod{N}$ . Then  
 as  $\lambda$  varies over  $\{0, 1, \dots, N-1\}$ ,  
 so does  $\lambda q$ .

$$= N^{-1} \sum_{\mu=0}^{N-1} \hat{a}_{\mu} \alpha^{-\mu t} = a_t$$

Thus  $[a_t]^2 = a_t$  all  $t$

$\Rightarrow a_t \in \mathbb{F}_2$  all  $t$ .

---



# CYCLIC CODES

Defn A linear cyclic code  $\mathcal{C}$  of block length  $N$  over  $\mathbb{F}_2$  is a collection of  $N$ -tuples  $(c_t)_{t=0}^{N-1}$  such

that :

(i) LINEARITY

$$\begin{pmatrix} c_t^{(1)} \\ c_t^{(2)} \end{pmatrix} \in \mathbb{C}$$

$$\Rightarrow \begin{pmatrix} c_t^{(1)} + \theta c_t^{(2)} \end{pmatrix} \in \mathbb{C}, \quad \forall \theta \in \mathbb{F}_2$$

(thus  $\mathbb{C}$  is a vector space over  $\mathbb{F}_2$ )

## (ii) CYCLIC SHIFTS

$$c_t = (c_0 \ c_1 \ \dots \ c_{N-1}) \in \mathbb{C}$$

$$\Rightarrow c_{t-\tau} = (c_{N-\tau} \ c_{N-\tau+1} \ \dots \ c_{N-1}) \in \mathbb{C}$$

(i.e.,  $\mathcal{C}$  is closed under cyclic shifts).

---

Given a codeword  $(c_t) \in \mathbb{F}_2^N$  in  $\mathcal{C}$ ,

we define the transform  $(\hat{c}_x)$  of

$(c_t)$  via

$$\hat{c}_x = \sum_{t=0}^{N-1} c_t \alpha^{\lambda t}$$

where  $\alpha$  has order  $N$  in  $\mathbb{F}_{\Sigma^m}$ ,

in which  $m$  is the multiplicative

order of  $\Sigma \pmod{N}$ .

Note: Thus while the code symbols  
belong to the ground field  $\mathbb{F}_2$ ,

their transforms lie in  $\mathbb{F}_{\sum^m}^H$  in general.

In this context, we will regard  $\hat{c}_\lambda$  as the transform

coefficient of  $c_t$  at frequency  $\lambda$ .

# EQUIVALENCE RELATION ON FREQUENCIES

---

Define

$$\lambda_2 \sim \lambda_1 \quad 0 \leq \lambda_1, \lambda_2 \leq H-1$$

if  $\lambda_2 = \sum_i \lambda_1 \pmod{H}$  some  $i \geq 0$ .

This can be verified to be an  
equivalence relation.

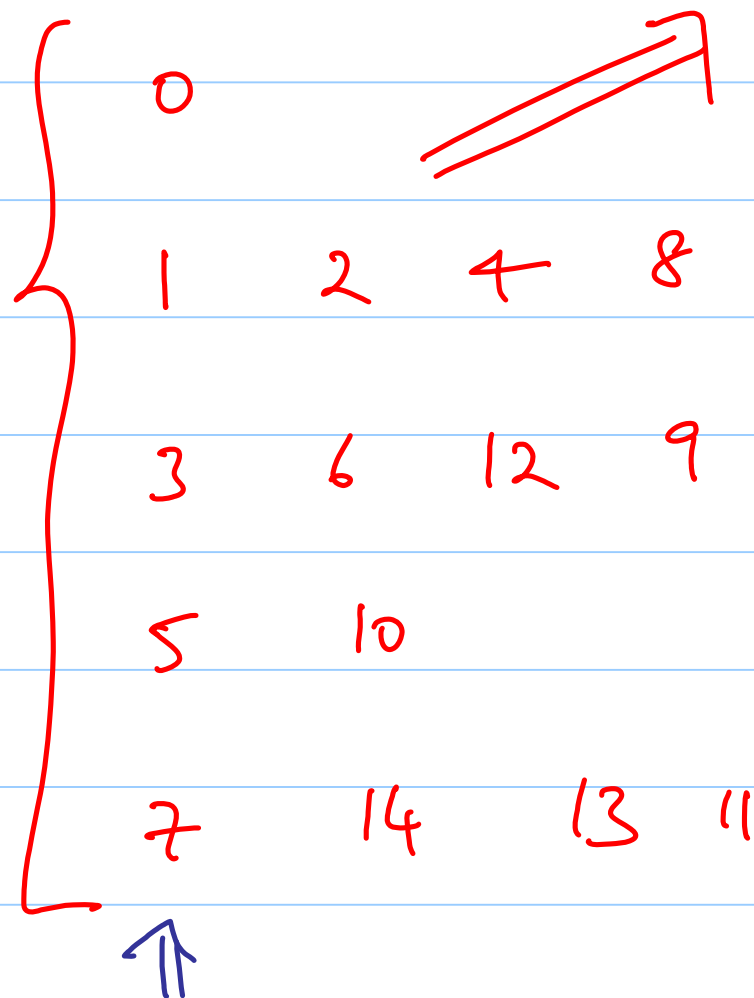
The corresponding equivalence

classes are called the

$q$ -cyclotomic cosets  $(\text{mod } N)$ .

Ex  $q = 2$   $H = 15$

there are  
5 equivalence  
classes  $\Rightarrow$



frequencies

$\{\lambda\}$

$\lambda_1 \sim \lambda_2$

$\lambda_1 = 2 \lambda_2$

(mod H)

coset leader



The smallest element within  
each coset is called  
the coset leader.

In general, if there  $k$  cosets,  
with  $n_i$  elements in the  $i^{\text{th}}$  coset,  
then 
$$\sum_{i=1}^k n_i = H.$$

Defn. A subset

$$S \subseteq \{0, 1, 2, \dots, N-1\}$$

is said to be a closed set of

frequencies if

$$\lambda \in S \Rightarrow q\lambda \pmod{N} \in S.$$

It is straight forward to show  
that every closed set is the  
union of cyclic sets.

## NULL SPECTRUM

Defn Let  $\mathcal{C}$  be a linear, cyclic code of block length  $N$  over  $\mathbb{F}_2$ .

Then the collection of frequencies

$$NS_{\mathcal{C}} = \left\{ \lambda \mid \begin{array}{l} 0 \leq \lambda \leq N-1 \\ \hat{c}_{\lambda} = 0 \text{ all } (c_k) \in \mathcal{C} \end{array} \right\}$$

is called the null spectrum of  $\mathcal{C}$ .

Fig

$$q = 2$$

$$H = 15$$

$HS_R$

Preview  
of transform  
domain view  
of cyclic  
codes

*	0	1	2	4	8
*	0	3	6	12	9
*		5	10		
*		7	14	13	11

Lec 41

{ Estimating the  
parameters of a cyclic  
code.

## Recap

### \* finite field transform

- expanding  $\mathbb{F}_\Sigma \rightarrow \mathbb{F}_{\Sigma^m}$
- defn
- properties

### \* cyclic codes

- closed set
- cyclotomic cosets
- dual spectrum.

Lemma 1 The null spectrum  $NS_C$  of a cyclic code  $C$  is a closed set

$$\text{Pf } \lambda \in NS_C \Leftrightarrow \hat{c}_\lambda = 0 \quad \forall (c_t) \in C$$

$$\hat{c}_\lambda = 0 \Rightarrow [\hat{c}_\lambda]^2 = 0 \Rightarrow \hat{c}_{\lambda^2} = 0$$

since  $(c_t) \in \mathbb{F}_2^H$  (conjugacy)



$$\Rightarrow \lambda^q \in (NS)_R$$

Hence  $(NS)_R$  is a closed set.

Lemma 2 Let  $S \subseteq \{0, 1, \dots, N-1\}$

be a closed set of frequencies.

Then

$$C \triangleq \left\{ (c_k) \in \mathbb{F}_2^N \mid \hat{c}_\lambda = 0, \text{ all } \lambda \in S \right\}$$

is a linear cyclic code.

Pf. Let  $(a_t), (b_t) \in \mathcal{R}$

$$\Rightarrow \hat{a}_\lambda = 0 \quad \hat{b}_\lambda = 0 \quad \text{all } \lambda \in S$$

$$\Rightarrow \hat{a}_\lambda + \theta \hat{b}_\lambda = 0 \quad \text{all } \lambda \in S, \\ \theta \in \mathbb{F}_2$$

$$\Rightarrow (a_t + \theta b_t)_{t=0}^{N-1} \in \mathcal{R}.$$

This proves that  $\mathcal{C}$  is linear.

Next, if  $(c_t) \in \mathcal{R}$

and  $a_t = c_t - \tau$  all  $t$

$$\Rightarrow \hat{a}_\lambda = \alpha^{\lambda \tau} \hat{c}_\lambda$$

$$\Rightarrow \hat{a}_\lambda = 0 \quad \text{all } \lambda \in S$$

$$\Rightarrow (a_t)_{t=0}^{N-1} \in \mathcal{C}.$$

$\therefore \mathcal{P}$  is cyclic

---

## REMINDER: COMMON SETTING

$$q = p^e \quad (q, N) = 1$$

$m$  is the order of  $q \pmod{N}$

$\mathcal{C}$  is a cyclic code of block

length  $N$  over  $\mathbb{F}_q$ .

Defn A basic sequence  $(B_t)$

of frequency  $\lambda_0$ ,  $0 \leq \lambda_0 \leq N-1$  is

a sequence satisfying:

$$\hat{B}_\lambda = \begin{cases} 1 & \lambda = \lambda_0 \\ 0 & \text{else} \end{cases} \quad \lambda \geq 0$$

Fig

$$l = 2$$

$$H = 15$$

$$m = 4$$

$$\lambda_0 = 3 \Rightarrow$$

No  $t_c$

$$\beta_3 = 1 \Rightarrow$$

$$\beta_6 = \beta_{12} = \beta_9$$

$$= 1$$

as well !

$\beta_{\lambda}$

0
0
1
0
0

0			
1	2	4	8
3	6	12	9
5	10		
7	14	13	11

Lemma 3 Let  $\mathcal{C}$  be a cyclic code having null spectrum  $NS_{\mathcal{C}}$ .

Then

$$\mathcal{C} = \left\{ (a_t) \in \mathbb{F}_q^N \mid \hat{a}_{\lambda} = 0 \ \forall \lambda \in NS_{\mathcal{C}} \right\}$$

In particular, the basic sequence  $(\beta_t)$  of every frequency  $\lambda \notin NS_{\mathcal{C}}$  belongs to  $\mathcal{C}$ .

(proof is in the Appendix).



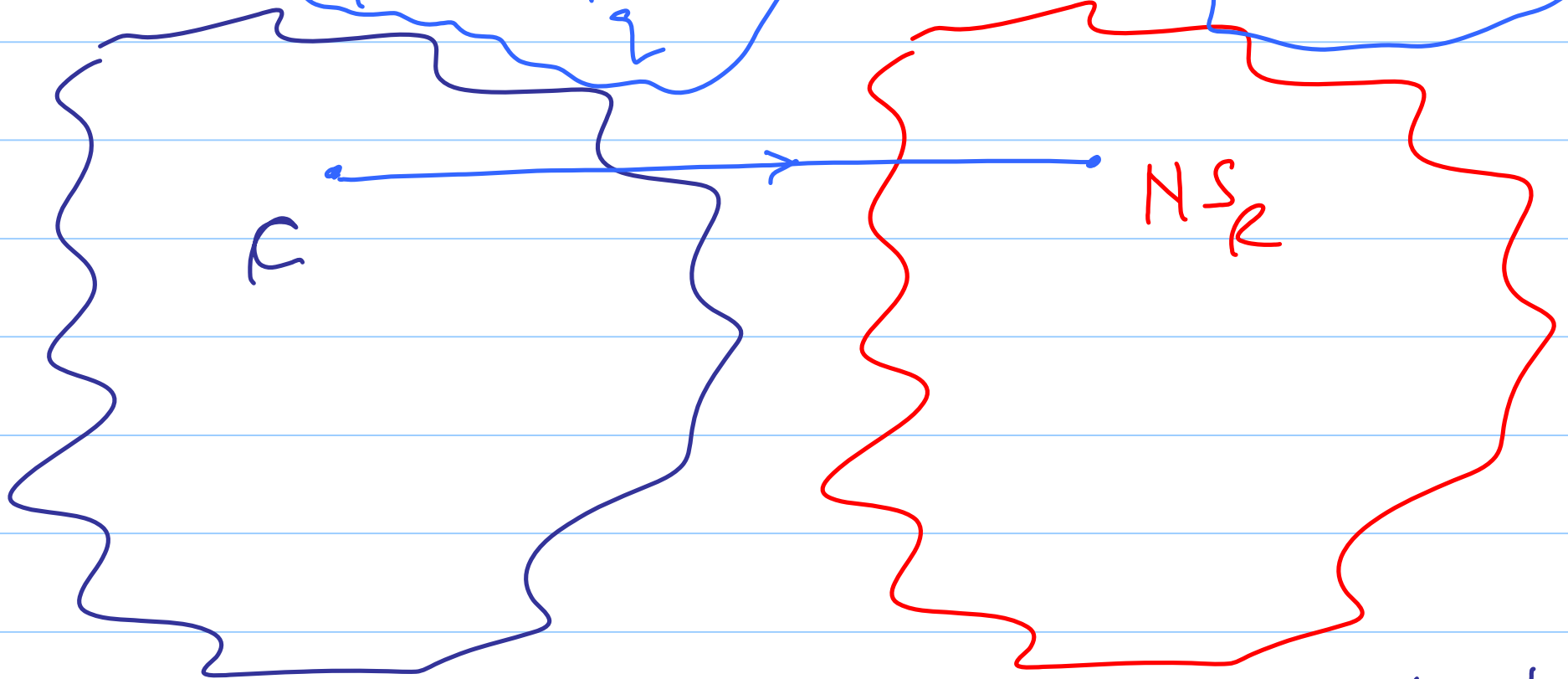
## Thm (Null Spectrum Theorem)

$\exists$  a 1-1 correspondence between  
cyclic codes over  $\mathbb{F}_q$  of block  
length  $N$  and closed sets of  
frequencies  $\subseteq \{0, 1, \dots, N-1\}$

pf.

all cyclic codes  
 $\mathcal{C} \subseteq \mathbb{F}^N$

all closed  
sets  $S$



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.,

$$K = |NS_R^c| = N - |NS_R|$$

Pf (presented in the Appendix).

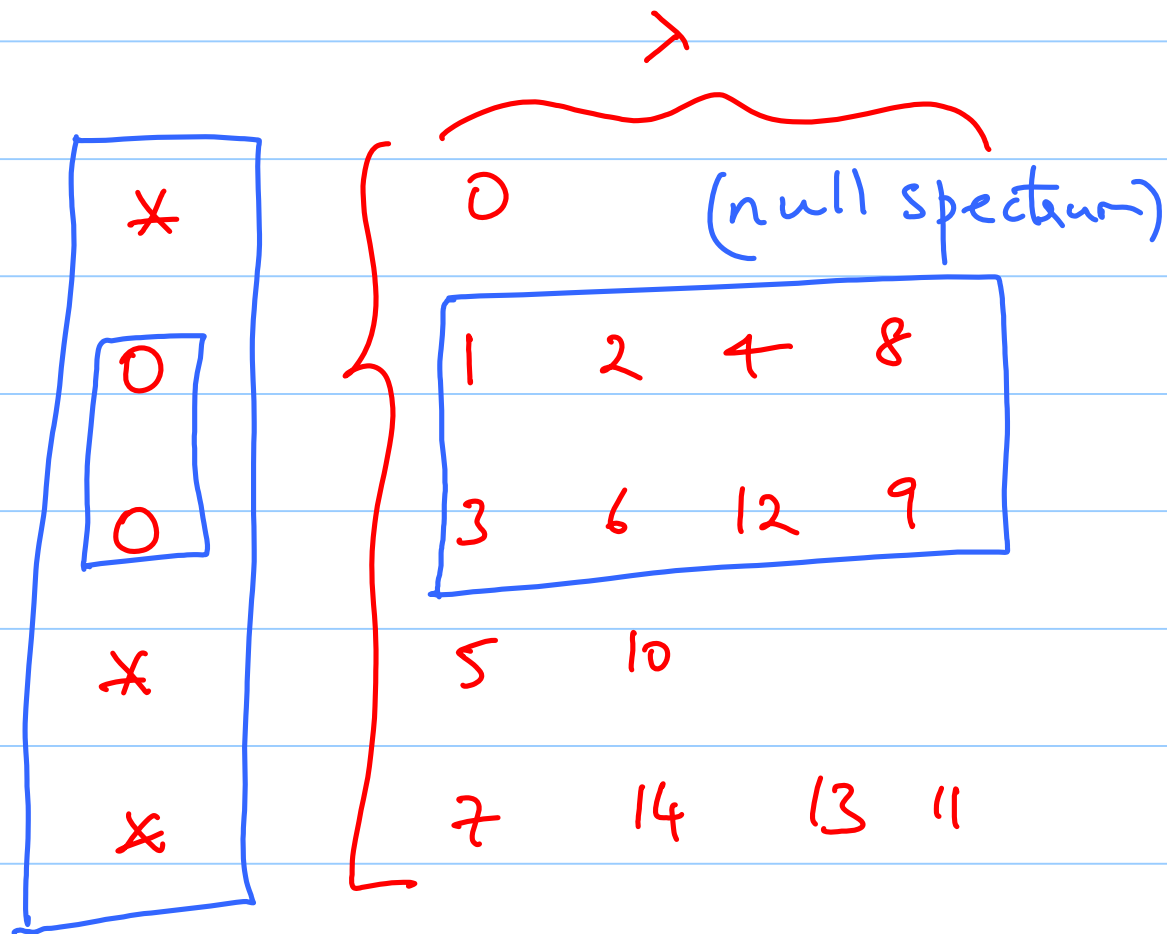
---

$$\underline{\mathbb{F}_9} \quad q = 2 \quad N = 15$$

$$m = 4$$

$$NS_{\mathcal{R}} = \{1, 2, 4, 8, 3, 6, 12, 9\}$$

$$\Rightarrow \dim(\mathcal{R}) = 15 - 8 = 7.$$



Thm (weight theorem) Let  $(a_t) \in \mathbb{F}_m^H$

have Hamming weight  $w_H((a_t)) = w$ .

Then  $(\hat{a}_\lambda)$  satisfies a linear recursion  
of degree  $= w$  i.e.;

$$\hat{a}_\lambda = \sum_{i=1}^w u_i \hat{a}_{\lambda-i} \quad \forall \lambda$$

where  $u_i \in \mathbb{F}_m$ .

Pf. ( please see the Appendix)

---



Thm Let  $\mathcal{C}$

be a cyclic code over  $\mathbb{F}_q$  of

length  $N$ ,  $(N, q) = 1$ , whose

null spectrum  $NS_{\mathcal{C}}$  contains

the consecutive set of frequencies

$$\left\{ m_0, m_0+1, m_0+2, \dots, m_0+d-2 \right\} \\ \subseteq \left\{ 0, 1, 2, \dots, N-1 \right\}$$

for some integers  $m_0, d$  with  $d \geq 2$ .

Then,

$$d_{\min}(C) \geq d$$

$d$  is called the  
designed  
distance of the  
code.

and hence  $C$  is a

$[N, N - |NS_C|, \geq d]$  code  
over  $\mathbb{F}_q$ .

Pf. Since  $\mathcal{C}$  is a linear code,  
it suffices to prove that  
the min Hamming weight of the  
code  $\geq d$ .

Suppose  $(c) \in \mathcal{C}$  has  
Hamming weight  $w > 0$ .

Then  $(\hat{C}_\lambda)$  satisfies a linear recursion of the form:

$$\hat{C}_\lambda = \sum_{i=1}^W \hat{C}_{\lambda - \alpha_i} u_i \quad \forall \lambda$$

so in particular

$$\bigwedge_{m_0+d-1} = \sum_{i=1}^W \bigwedge_{m_0+d-1-i} u_i$$

$\bigwedge_{L \rightarrow}$

$$\begin{array}{ccccccc} \times & \times & . & \boxed{0} & 0 & & 0 \\ & & & m_0 & m_0+1 & . & m_0+d-2 \end{array} \quad \begin{array}{ccc} * & * & * \\ m_0+d-1 & \dots & (N-1) \end{array}$$

$$\subseteq (Ns)_R$$

If  $w < d$ , then

$$m_0 + d - 1 - i \geq m_0 + d - 1 - (d - 1) \\ = m_0.$$

$$\therefore \hat{c}_{m_0 + d - 1 - i} = 0$$

one can then proceed to

show inductively that

$$\hat{c}_\lambda = 0 \quad \forall \lambda \Rightarrow (c_k) = \underline{0}$$

which contradicts our initial assumption that  $w \geq 0$ .

Hence  $w \geq d$  and

$$i. \quad d_{\min}(C) \geq d.$$

---

Eg Binary, (primitive) BCH codes

Here  $q = 2$ ,  $N = 2^m - 1$ ,

( $\Rightarrow$  the multiplicative order of  
 $q \pmod{N} = m$  CHECK!)

$$NS_R \geq \{m_0, m_0+1, \dots, m_0+d-2\}$$

$\therefore$  binary, primitive BCH codes

have parameters



$$\left[ N = 2^m - 1, \quad N - |NS_c| > d_{\min} \geq d \right]$$

A popular choice is  $m_0 = 1$ .

$$\begin{aligned} \therefore NS_R &\geq \{1, 2, \dots, m_0 + d - 2 = d - 1\} \\ &= \{1, 2, \dots, 2t\} \text{ if } d = 2t + 1 \end{aligned}$$

Note that in the  $\mathbb{Q}$ -cyclotomic  
cosets mod  $N$  (i.e., the

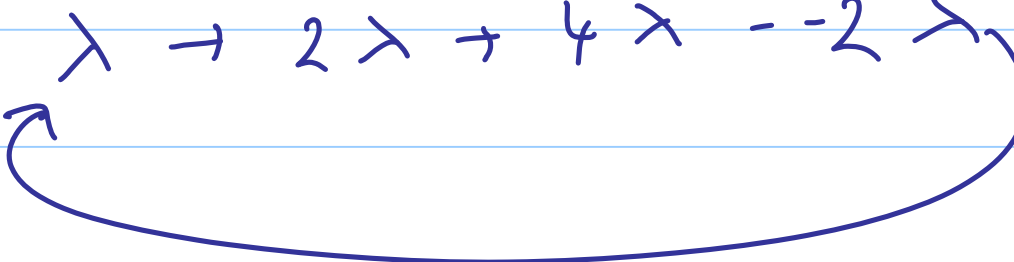
2-cyclotomic cosets  $(\text{mod } 2^m - 1)$

(i) the size of a cyclotomic coset is at most  $m$  since

$$\lambda 2^m (\text{mod } n)$$

$$= \lambda 2^m (\text{mod } 2^m - 1)$$

$$= \lambda !$$

$$\lambda \rightarrow 2\lambda \rightarrow 4\lambda \rightarrow \dots \rightarrow 2^{m-1}\lambda$$


(ii) if  $k \in \{1, 2, \dots, 2t\}$  is even,

$k = 2\ell$ , then  $\ell \in \{1, 2, \dots, 2t\}$ .

and  $\{k, \ell\}$  belong to the same

cyclotomic coset.

$\therefore$  it suffices to ensure that

$$\{1, 3, 5, \dots, 2t-1\} \subseteq NS_{\mathbb{Z}}$$

$\Downarrow$

set is of size  $t$

$\therefore$  the null spectrum need be no larger than  $mt$ .

$$\therefore \dim(R) = |Hs_K^c| = N - |Hs_K| \\ \geq 2^m - 1 - mt.$$

BCH code parameters:

$$\left[ 2^m - 1, 2^m - 1 - mt, d_{\min} \geq 2t + 1 \right]$$

Ex

$$l = 2$$

$$H = 15$$

$$m = 4$$

$$\lambda_0 = 3 \Rightarrow$$

No  $t_c$

$$\beta_3 = 1 \Rightarrow$$

$$\beta_6 = \beta_{12} = \beta_9$$

$$= 1$$

as well !

$\beta_3$

0
0
1
0
0

0			
1	2	4	8
3	6	12	9
5	10		
7	14	13	11

[

APPENDIX

Lemma 3 Let  $C$  be a cyclic code having null spectrum  $NS_C$ .

Then  $C$  contains the basic sequence of every frequency  $\lambda \notin NS_C$ .

# Lec 42 Decoding Cyclic Codes

## Recap

\* The null spectrum theorem

$$* \dim(C) = N - |N_C^s|$$

\* BCH codes

$$- d_{\min} \geq d$$

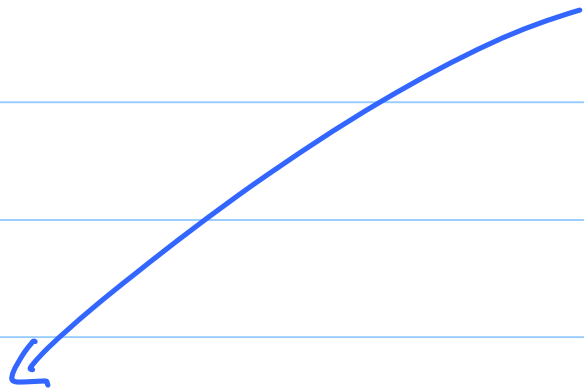
— proof via the



{ weight theorem

— primitive binary  
BCH codes

— dimension  
estimate



$$[2^m - 1, \geq 2^m - 1 - mt, d_{\min} \geq 2t + 1]$$

(parameters of a  $t$ -error-correcting  
BCH code).

Thm (BCH codes) Let  $\mathcal{C}$

be a cyclic code over  $\mathbb{F}_2$  of

length  $N$ ,  $(N, 2) = 1$ , whose

null spectrum  $NS_{\mathcal{C}}$  contains

the consecutive set of frequencies

$$\left\{ m_0, m_0+1, m_0+2, \dots, m_0+d-2 \right\} \\ \subseteq \left\{ 0, 1, 2, \dots, N-1 \right\}$$

for some integers  $m_0, d$  with  $d \geq 2$ .

Then,

$$d_{\min}(C) \geq d$$

$d$  is called the  
designed  
distance of the  
BCH code.

and hence  $C$  is a

$[N, N - tns_r, \geq d]$  code  
over  $\mathbb{F}_q$ .

Eg (Reed - Solomon codes)

consider the case  $q = p^e$ ,

$N \mid q - 1$ . Then

$$q \equiv 1 \pmod{N} \Rightarrow$$

the multiplicative order of  $q \pmod{N}$

$$= 1.$$

$$\underline{\mathbb{E}_g} \quad q = 2^4$$

$$N = q - 1 = 15$$

$$m_0 = 2 \quad d = 5$$

$$\therefore NS_{\mathcal{C}} \supseteq \{m_0, m_0 + 1, \dots, m_0 + d - 2\}$$

$$= \{2, 3, \dots, 5\}$$

$$\therefore \boxed{d_{\min} \geq 5}$$

$q$  cyclotomic cosets (mod  $n$ )

$$a \sim b \text{ iff } a = q^i b \pmod{n} \\ = b \pmod{n}$$

$$\text{Since } q = 1 \pmod{n} !$$

$$\therefore a \sim b \Leftrightarrow a = b.$$

Thus each equivalence class contains

only a single element:

$$\therefore \dim(\mathcal{C}) = N - (d-1)$$

$$\therefore k = N - d + 1 \geq N - d_{\min} + 1$$

hence the code is MDS

and  $d_{\min} = d$ .

*	0
*	1
0	2
0	3
0	4
0	5
*	6
*	7
*	8
:	:
:	:
*	14

General parameters of a  $R-S$  code:

$$[N, N-d+1, d]$$



DECODING

BCH

CODES

## BCH code Setting

$$q = p^e \quad (q, N) = 1$$

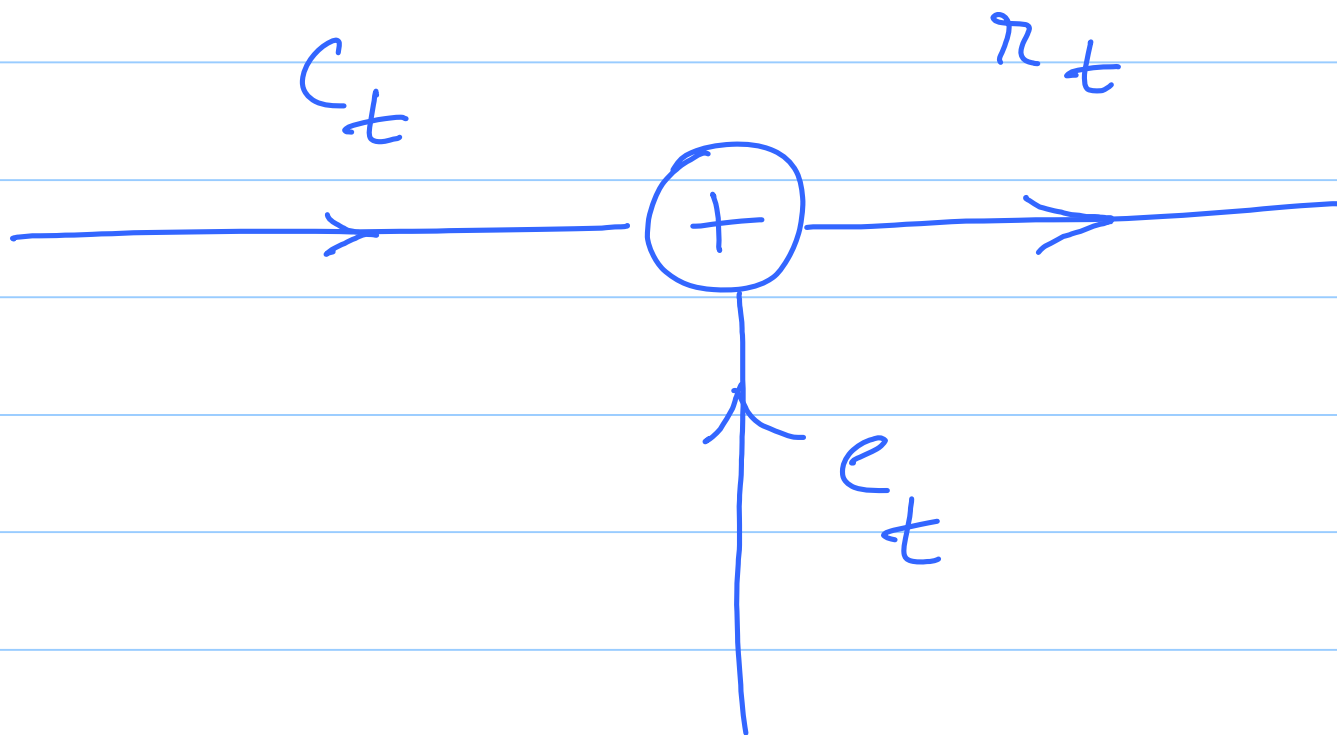
$m$  is the order of  $\{ \cdot \} \pmod{N}$

$C$  - cyclic code  $\subseteq \mathbb{F}_q^N$

$\alpha \in \mathbb{F}_m$  has order  $N$

$$NS_C = \{m_0, m_0+1, \dots, m_0+d-2\}$$

Channel Model:



$c_t, e_t, x_t \in \mathbb{F}_2$ , all  
 $0 \leq t \leq N-1$

$$\hat{r}_\lambda = \sum_{t=0}^{H-1} r_t \alpha^{\lambda t}$$

$$= \sum_t c_t \alpha^{\lambda t} + \sum_t e_t \alpha^{\lambda t}$$

$$= \hat{c}_\lambda + \hat{e}_\lambda$$

$$\therefore \boxed{\hat{r}_\lambda = \hat{c}_\lambda \quad m_0 \leq \lambda \leq m_0 + d-2}$$

# "SYNDROME" $S(z)$

$$S(z) \triangleq \sum_{\lambda=0}^{d-2} \hat{r}_{\lambda+m_0} z^{\lambda}$$

$$= \sum_{\lambda=0}^{d-2} \hat{e}_{\lambda+m_0} z^{\lambda}$$

Set  $r \triangleq d-1$  and define:

$$S_{\infty}(z) = \sum_{\lambda=0}^{\infty} \hat{e}_{\lambda+m_0} z^{\lambda}$$

$$= \sum_{\lambda=0}^{\infty} \sum_{t=0}^{N-1} e_t \alpha^{(x+m_0)t} z^{\lambda}$$

$$= \sum_{i=1}^W e_{t_i} \alpha^{m_0 t_i} \sum_{\lambda=0}^{\infty} (z \alpha^{t_i})^{\lambda}$$

$$= \sum_{i=1}^W \frac{e_{t_i} \alpha^{m_0 t_i}}{(1 - z \alpha^{t_i})}$$

where  
 $W = \#$  of  
 errors  
 in locations  
 $\{t_i\}$

$$= \frac{w(z)}{\sigma(z)} \quad (\text{say})$$

where:

$$\sigma(z) \triangleq \prod_{i=1}^n (1 - \alpha^{t_i} z)$$

{ ERROR  
LOCATOR  
POLYNOMIAL }

$$\omega(z) \triangleq \sum_{i=1}^w e_{t_i} z^{m_{0,t_i}} = \prod_{\substack{j=1 \\ j \neq i}}^w (1 - \alpha^{t_j} z)$$

ERROR EVALUATOR  
POLYNOMIAL

Note:

$$\deg(\omega(z)) = w \leq \left\lfloor \frac{d-1}{2} \right\rfloor = \left\lfloor \frac{n}{2} \right\rfloor$$



$$\deg(w(z)) = w-1 \leq \left\lfloor \frac{n}{2} \right\rfloor - 1$$

Replacing  $s_{\infty}(z)$  by  $S(z)$

---

$$S(z) = \sum_{\lambda=0}^{n-1} e_{\lambda+m_0} z^{\lambda}$$

$$= s_{\infty}(z) \pmod{z^n}$$

$$\therefore S(z) = \frac{w(z)}{v(z)} \pmod{z^r}$$

$$\therefore S(z) = \frac{w(z)}{v(z)} + \underbrace{A(z)}_{\downarrow} z^r$$

power series in  
 $z$  with no neg.  
 exponents

$$\therefore \sigma(z) s(z) - w(z) = \underbrace{A(z) \sigma(z)}_{\text{polynomial } B(z)} z^n$$

$$\therefore \boxed{w(z) = \sigma(z) s(z) + B(z) z^n}$$

$$\deg \leq \left\lfloor \frac{n}{2} \right\rfloor - 1$$

$$\deg \leq \left\lfloor \frac{n}{2} \right\rfloor$$

this is reminiscent of gcd  
computation via the  
Euclidean division algorithm.

# DECODING EXAMPLE

---

$$q = 2 \quad N = 15 \quad m = 4 \quad d = 5$$

$$\{m_0, m_0 + 1, \dots, m_0 + d - 2\}$$
$$= \{1, 2, 3, 4, 5, 6\}$$

(Triple-error correcting BCH code)

consequent choice of null spectrum:

---

null spectrum

*	0			
0	1	2	4	8
0	3	6	12	9
0	5	10		
*	7	14	13	11

$$x(t) = \begin{cases} 1 & t \in \{0, 3, 4, 5, 6, 7, 8, 12, 13\} \\ 0 & \text{else} \end{cases}$$

STEP 1

compute  $\hat{\pi}_\lambda$

$$m_0 \leq \lambda \leq m_0 + d - 2$$

$$\therefore \hat{\pi}_\lambda = \sum_{t=0}^{14} \pi_t \alpha^{\lambda t}$$

$$= \alpha^0 + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6 + \alpha^7 + \alpha^8 + \alpha^{12} + \alpha^{13}$$

[illegible]

$$= 2^6$$

$$\hat{x}_2 = \begin{bmatrix} \hat{x}_1 \end{bmatrix} = \alpha$$



$$\hat{\pi}_4 = [\hat{\pi}_1]^4 = 2^9$$

$$\hat{\pi}_3 = 2^{13} \quad (\text{direct computation})$$

$$\hat{\pi}_5 = 2^5 \quad ( \quad " \quad )$$

$$\hat{\pi}_6 = [\hat{\pi}_3]^2 = 2^{11}$$

STEP 2:

$$S(z) = \sum_{\lambda=0}^{\infty} z^{\lambda+m_0}$$

$$= z^{11} z^5 + z^5 z^4 + z^9 z^3 + z^{13} z^2 + z^{12} z + z^6$$

$$= (z^{11} z^5 z^9 z^{13} z^{12} z^6)$$

SHORTHAND NOTATION

	$z^n$	$s(z)$	Quotient
$z^7 = z^6$	1	0	
$\begin{array}{cccccc} 11 & 5 & 9 & 13 & 12 & 6 \\ \alpha & \alpha & \alpha & \alpha & \alpha & \alpha \end{array}$	0	1	$\begin{array}{cc} \alpha^4 & \alpha^{13} \\ \alpha & \alpha \end{array}$
$\begin{array}{cccccc} 8 & 12 & 6 & & & \\ \alpha & \alpha & \alpha & 0 & \alpha^4 & \end{array}$		$\alpha^4 \quad \alpha^{13}$	$\alpha^3 \quad \alpha^2$
$\begin{array}{cccccc} \deg & 14 & 3 & 2 & & \\ 7 \mid \frac{7}{2} \mid -1 & \alpha & \alpha & \alpha & 0 & \end{array}$		$\alpha^7 \quad \alpha^{11} \quad 0$	$\alpha^9 \quad 0$
$\begin{array}{cccc} & & & \\ & & & \\ & & & \\ \alpha & 0 & \alpha^4 & \end{array}$		$\alpha \alpha^5 \quad \alpha^4 \quad \alpha^{13}$	
$= K w(z)$		$= K s(z)$	

$2^4 \ 2^{13} \Rightarrow$  Quotient

$$\begin{array}{r} 2^{11} \ 2^5 \ 2^9 \ 2^{13} \ 2^{12} \ 2^6 \\ 2 \ 2 \ 2 \ 2 \ 2 \ 2 \end{array} \Bigg| \begin{array}{cccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$1 \ 2^9 \ 2^{13} \ 2^2 \ 2 \ 2^{10}$$

$$\hline 2^9 \ 2^{13} \ 2^2 \ 2 \ 2^{10} \ 0$$

$$2^9 \ 2^3 \ 2^7 \ 2^{11} \ 2^{10} \ 2^4$$

$$\hline \text{Rem} = 2^8 \ 2^{12} \ 2^6 \ 0 \ 2^4$$

---

$2^3 \quad 2^2 = \text{Quotient}$

$$\begin{array}{r} 8 \quad 12 \quad 6 \quad 4 \\ 2 \quad 2 \quad 2 \quad 0 \quad 2 \end{array} \left| \begin{array}{r} 11 \quad 5 \quad 9 \quad 13 \quad 12 \quad 6 \\ 2 \quad 2 \quad 2 \quad 2 \quad 2 \quad 2 \end{array} \right.$$

$$2^{11} \quad 1 \quad 2^9 \quad 0 \quad 2^7$$

$$0 \quad 2^{10} \quad 0 \quad 2^{13} \quad 2^2 \quad 2^6$$

$$2^{10} \quad 2^{14} \quad 2^8 \quad 0 \quad 2^6$$

$$\text{Rem} = 2^{17} \quad 2^3 \quad 2^2 \quad 0$$

$$\begin{array}{r}
 2^{14} \ 2^3 \ 2^2 \ 0 \mid 2^8 \ 2^{12} \ 2^6 \ 0 \ 2^9 \\
 \phantom{2^{14} \ 2^3 \ 2^2 \ 0 \mid} 2^8 \ 2^{12} \ 2^{11} \ 0
 \end{array}$$

$$\text{Rem} = 2 \ 0 \ 2^9$$

$$1 + (2^4 z + 2^{13}) (2^3 z + 2^2)$$

$$= 1 + 2^2 2^7 + z (2 + 2^6) + 1$$

$$= 2^2 2^7 + z 2^{11}$$


---

$$(z^4 z + z^{13}) + (z^9 z) (z^7 z^2 + z^4 z)$$

$$= z^3 z + z^2 z^5 + z^4 z + z^{13}$$

$$\therefore \prod_{n=1}^{\infty} (1 - z^n) = K \left( \begin{array}{c} \rightarrow \end{array} \right)$$

$$\therefore K = z^2.$$

$$\therefore \sigma(z) = z^3 z^3 + z^2 z^7 + z^6 z + 1$$



## Chien search

$$\begin{aligned} z=1 &\Rightarrow z^3 + z^7 + z^6 + 1 \\ &= 1 + z^3 \left( 1 + z^3 \underbrace{(1+z)}_{z^9} \right) \\ &\quad \underbrace{\hspace{1.5cm}}_{z^9} \\ &= z^9 \neq 0 \end{aligned}$$

$$z = z^2 \Rightarrow$$

$$z^3 z^6 + z^7 z^9 + z^6 z^2 + 1$$

$$= 1 + z^8 \left( 1 + z \underbrace{(1 + z^2)}_{z^8} \right)$$

$\underbrace{\hspace{10em}}_{z^7}$

$$= 0$$

$$\therefore \boxed{z = z^2 \text{ is a zero}}$$

Similarly it turns out that

$\prod_{i=1}^W (1 - z^{t_i})$  has  
 $z = z^2, z^{11}, z^{14}$  as the  
3 zeros.

$\therefore$  error locations are the

reciprocals:

$$z^{-2} \quad z^{-11} \quad z^{-14} \Rightarrow z^{13}, z^9, z$$

$$\therefore t_1 = 1$$

$$t_2 = 4$$

$$t_3 = 13$$



# NPTEL: Error-Correcting Codes

## Exercises

P. Vijay Kumar  
Indian Institute of Science

### 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  $\mathcal{C}$  is a code whose symbol alphabet is the set  $\{0, 1, 2\}$ , i.e.,  $\mathcal{C}$  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

$$d_{\min} \geq t_d + t_c + 1. \quad (1)$$

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

$$a + b \in H \text{ whenever } a \in H \text{ and } b \in H.$$

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  $l_1 a = l_2 a$  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

$$G = Z_2^4 = \{ \text{all binary 4 tuples} \}$$

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

$$\sum_{i=0}^2 a_i z^i \sum_{i=0}^2 b_i z^i = \sum_{i,j=0}^2 a_i b_j z^{i \oplus j}$$

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  $d_{min} = 7$
- (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  $\mathcal{C}$  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  $\mathcal{C}$  such that  $\mathcal{C}^\perp$  is  $\mathcal{C}$  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  $\mathcal{C}$  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 ( $c_0, c_1, \dots, c_{k-1}$ ) are precisely the  $k$  message symbols, ( $m_0, m_1, \dots, m_{k-1}$ ).

Under this definition, is the linear block code  $\mathcal{C}$  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  $\mathcal{C}$  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  $\mathcal{C}$  ? Explain how you obtained your answer.
- (b) What is the minimum distance of the *dual* code  $\mathcal{C}^\perp$  ? Again, explain how you obtained your answer.

13. Determine the minimum distance  $d_{min}$  of the  $[7, 3]$  linear block code  $\mathcal{C}$  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  $\mathcal{C}$  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  $\mathcal{C}$ .

16. A linear block code  $\mathcal{C}$  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  $P_{we}$  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  $d_{min} = 7$ .
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

$$\underline{r} = [1 \ 1 \ 1 \ 0 \ 1]^t,$$

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

$$\underline{r} = [1 \ 1 \ 0 \ 0 \ 0]^t.$$

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  $\mathcal{C}$  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  $\mathcal{C}$  is the smallest integer  $\rho$  such that for any  $\underline{x} \in \mathbb{F}_2^n$ , there exists a codeword  $\underline{c} \in \mathcal{C}$  such that
- $$d_H(\underline{c}, \underline{x}) \leq \rho.$$
- (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  $F_2[[D]]$ , 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)  $\frac{D^2}{1+D^2+D^5}$

23. Determine whether the convolutional codes encoded using the  $G(D)$  below given 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

$$G(D) = [1 + D + D^2 \quad 1 + D^2].$$

- (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  $d_{free}$  of the code.
- (c) If the received sequence (across a BSC) is

$$\underline{r} = (01 \ 00 \ 01 \ 00 \ 00 \ 00 \dots\dots)$$

find (the information sequences associated to) all survivors at node level 6.

- (d) If the received sequence (across an AWGN channel) is

$$\underline{r} = (4 \ -1 \ -3 \ 2 \ 6 \ -5 \ 2 \ 4 \ 5 \ 3 \ 5 \ 5 \dots\dots)$$

find (the information sequences associated to) all survivors at node level 6.

25. Consider a rate 1/3 convolutional code with

$$G(D) = [1 + D \quad 1 + D^2 \quad 1 + D + D^2].$$

- (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  $\mathcal{C}$  to exhibit catastrophic error propagation ? Explain fully your answer.

27. This question pertains 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  $((-\infty, \infty], \min^*, +)$  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  $\underline{v}^T = [v_1, v_2, v_3]$  is a codeword if and only if  $H\underline{v} = 0$ . 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

$$\underline{y}^T = [0 \ 1 \ 0].$$

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

(d) the result of decoding

30. Consider the rate  $\frac{1}{2}$  convolutional code having polynomial generator matrix

$$G(D) = [1, \quad 1 + D].$$

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

$$\underline{y}^T = [01 \quad 01].$$

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.

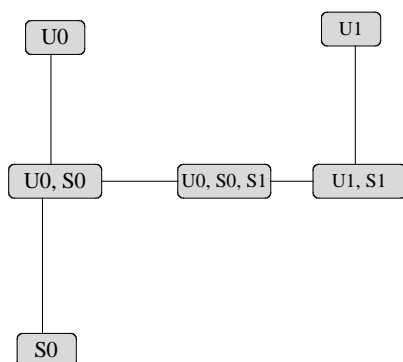


Figure 1: Junction Tree associated to the convolutional code.

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

$$G(D) = [1 + D + D^2 \quad 1 + D^2].$$

If the received sequence (across a discrete memoryless AWGN channel) is

$$\underline{r} = (4 \quad -1 \quad -3 \quad 2 \quad 6 \quad -5),$$

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  $\epsilon \ll 1$ , is the vector

$$\underline{y} = [1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T.$$

Use the GDL to make decisions based on maximizing the a posteriori probabilities

$$p(u_i/\underline{y})$$

of the code symbols  $u_i, i = 1, 2, 3, 4, 5, 6, 7$ .

35. Verify that the  $\min^*$  sum semi-ring is in fact a semi-ring, starting from the definition of the  $\min^*$  operation:

$$\min^*(x, y) = \min\{x, y\} - \ln(1 + e^{-|y-x|}) = -\ln(e^{-x} + e^{-y}).$$

Identify the underlying set and the identity element under the  $\min^*$  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

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

Thus all codewords  $\underline{c} = [c_1, c_2, c_3, c_4, c_5, c_6, c_7]^T$ , satisfy  $H\underline{c} = 0$ . 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  $Pr(c_4 = 0/\underline{r})$  and  $Pr(c_4 = 1/\underline{r})$  where the received vector  $\underline{r}$  is given by  $\underline{r} = [1111111]^T$ . You may assume that the channel is a binary symmetric channel (BSC) having crossover probability  $0 < \epsilon \ll 1$ .

**A suggested approximation** In your computations, you will run into expressions of the form

$$a_i \epsilon^i + a_{i+1} \epsilon^{i+1} + a_{i+2} \epsilon^{i+2} + \dots + a_{i+k} \epsilon^{i+k},$$

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

$$a_i \epsilon^i + a_{i+1} \epsilon^{i+1} + a_{i+2} \epsilon^{i+2} + \dots + a_{i+k} \epsilon^{i+k} \approx a_i \epsilon^i.$$

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!

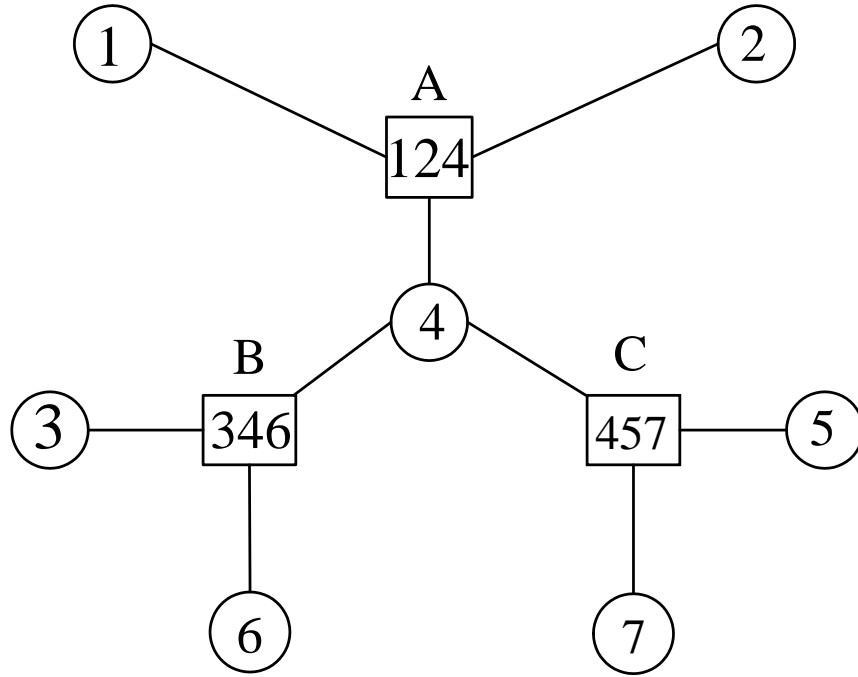


Figure 2: Junction Tree associated to the  $[7, 4, 2]$  code..

## 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  $\mathcal{C}$ . Thus the channel is a BSC with cross-over probability  $\epsilon \ll 1$  and all messages passed are either 1 or  $-1$ . You may assume that the neighborhood of every node in the Tanner graph of  $\mathcal{C}$  is tree-like to depth 8. What is the probability  $p_{-1}^{(1)}$  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  $\epsilon \ll 1$  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

$$\psi_v^{(0)}(l_0), \psi_v^{(l)}(l_0, l_1, l_2, \dots, l_{d_v-1}), \psi_c^{(l)}(l_1, l_2, \dots, l_{d_c-1}).$$

- (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 ?

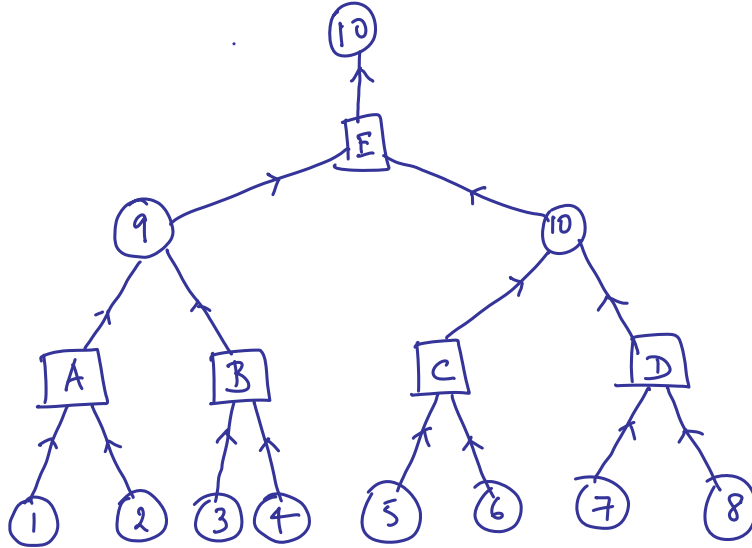


Figure 3: Computational tree associated to node 10 in Tanner Graph.

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  $\gcd$  of 6711 and 831. Express the  $\gcd$  as a linear combination  $u * 6711 + v * 831$  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  $\checkmark$ 
  - $\alpha + \alpha^4 \in GF(4)$
  - $\alpha + \alpha^8 \in GF(8)$
  - none of the above
57. The polynomials over  $GF(2)$  given below are all irreducible. Identify with a  $\checkmark$ , 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

$$(s(t), t = 0, 1, 2, \dots, 14) = 000110000101101 .$$

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

$$\hat{a}_\lambda \in \{0, 1\}, \text{ all } \lambda, \quad 0 \leq \lambda \leq 14.$$

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  $d_{\min} = 6$ .
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  $\mathcal{C}$  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  $(1, 1, \dots, 1, 1)$  belong to the cyclic code  $\mathcal{C}$  ? Explain your answer.
69. Use the irreducible polynomial (irreducible over  $\mathbb{F}_3$ )  $x^2 + x + 2$  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  $\mathbb{F}_9$ .  
  - (a) Set up an add-1 table for  $\mathbb{F}_9$ .
  - (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  $\mathbb{F}_2$ .
71. If  $\alpha, \beta$  in  $\mathbb{F}_{16}$  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.