ERROR-CORRECTING CODES

P. VIJAY KUMAR

Course Outline:
Lectures 1-15

Basics of Block Codes
(2)

Mathematical Preliminaries: groups, rings and fields
(3)

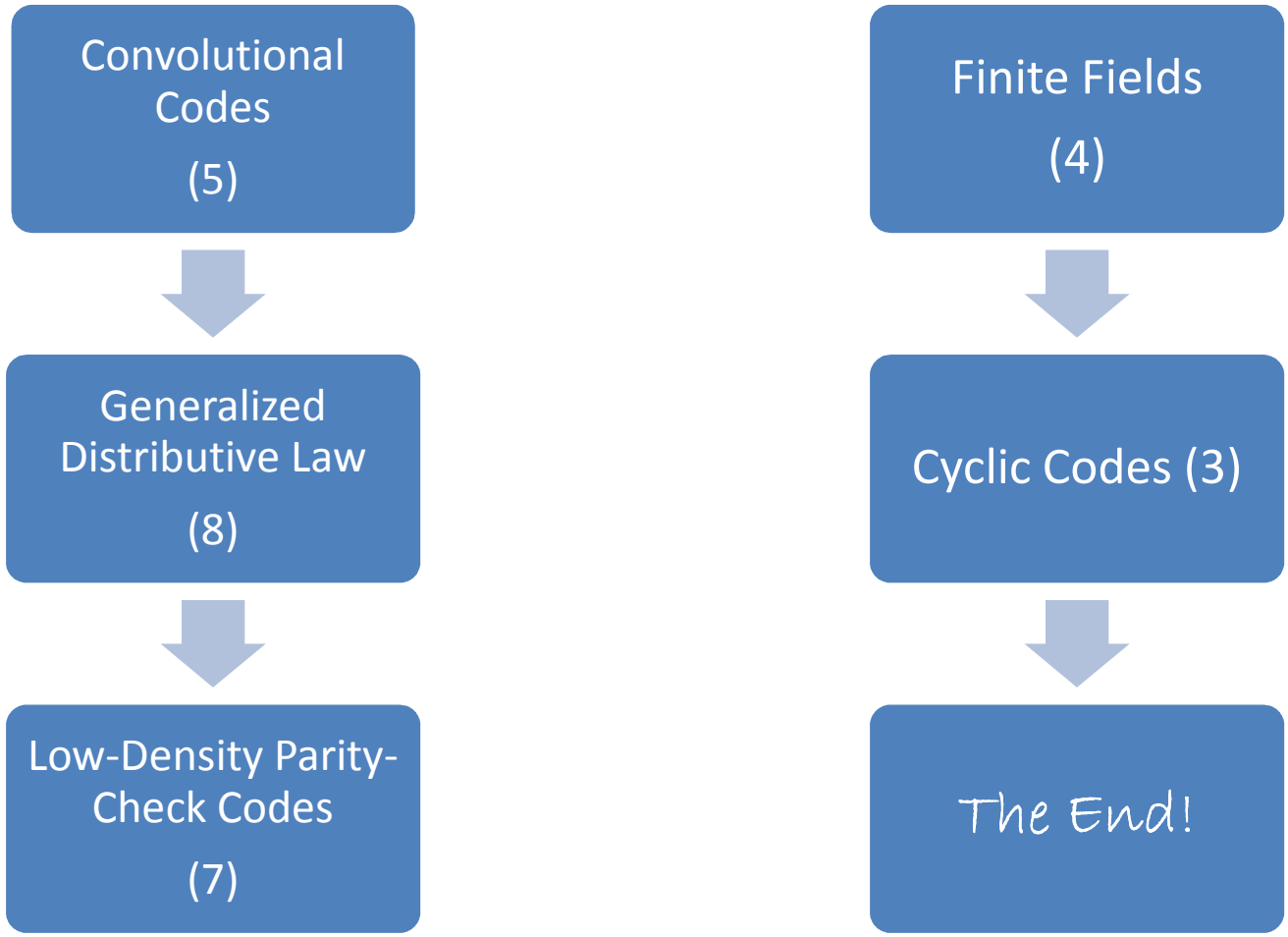
Mathematical Preliminaries: vector spaces
(3)

Linear Codes
(3)

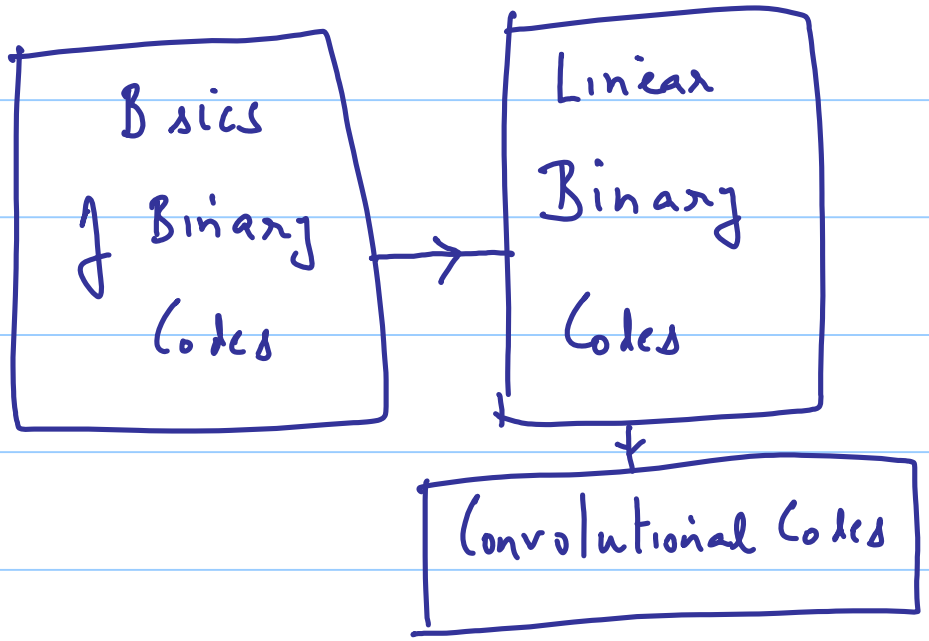
Bounds on Code Size
(2)

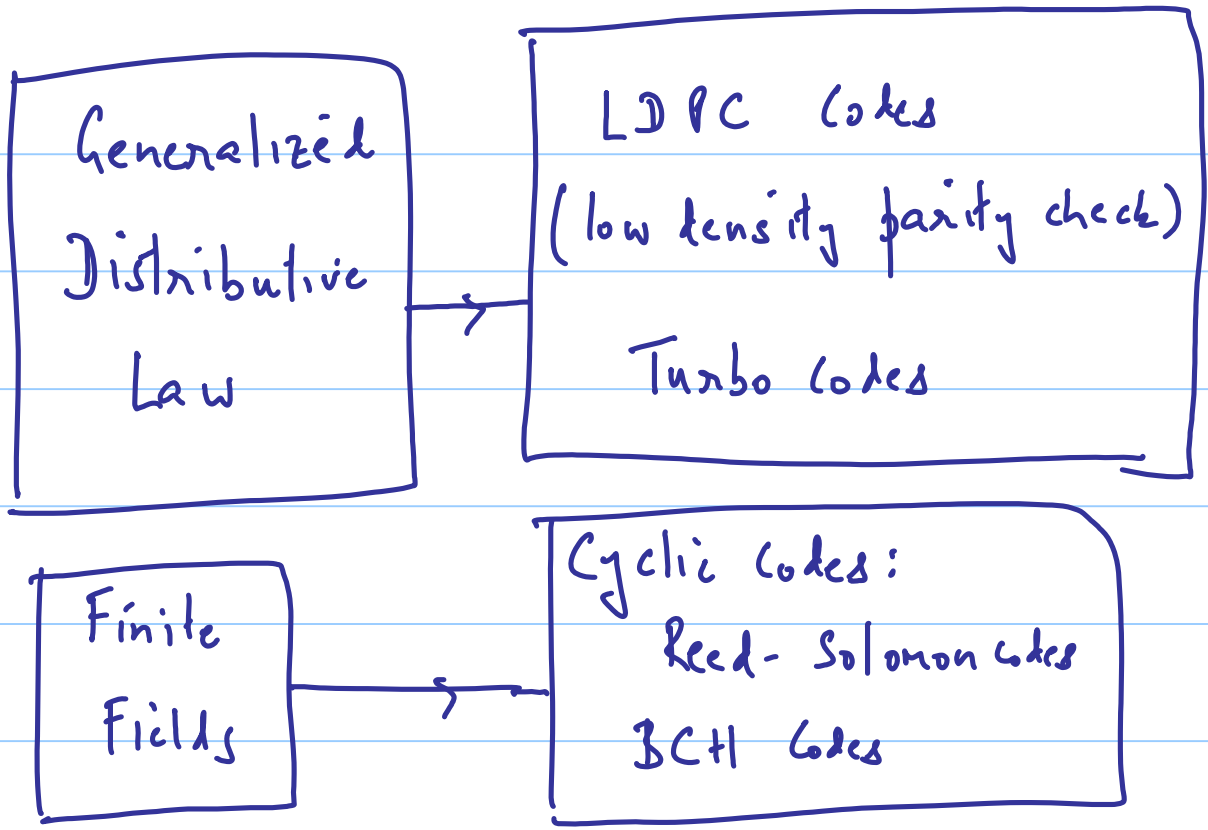
Standard Array Decoding
(2)

Course Outline:
Lectures 16-42

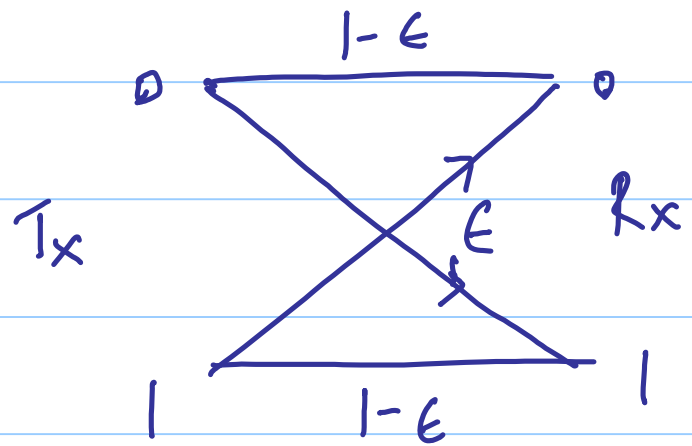


Lec 1: Course Overview § Basics

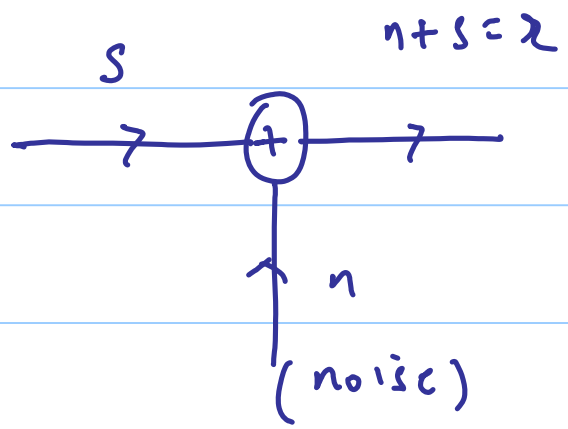



Generalized
Distributive
Law
LDPC Codes
(low density parity check)
Turbo Codes
Finite
Fields
Cyclic Codes:
Reed-Solomon codes
BCH Codes

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)

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

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

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

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 $\underline{x}$.

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

Properties

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

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

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

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

$n = 5$

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

$= 4 + 2 = 6.$

Define

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

(Schur or Componentwise product)

$$w_H(\underline{x} + \underline{y}) = w_H(\underline{x}) + w_H(\underline{y}) - 2 w_H(\underline{x} \odot \underline{y})$$

Eg

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

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

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:

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

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

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

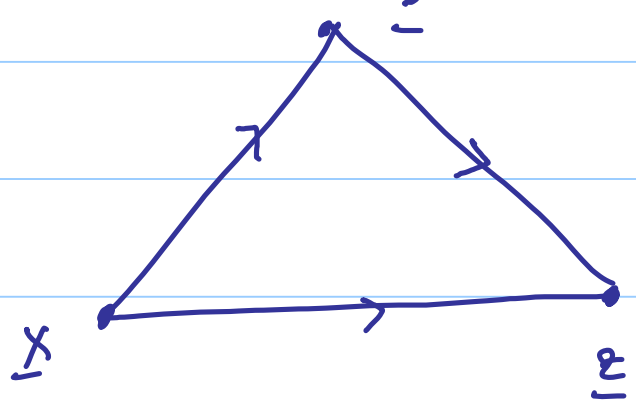
Properties

(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

$$d_H(\underline{X}, \underline{Z}) \leq d_H(\underline{X}, \underline{Y}) + d_H(\underline{Y}, \underline{Z})$$

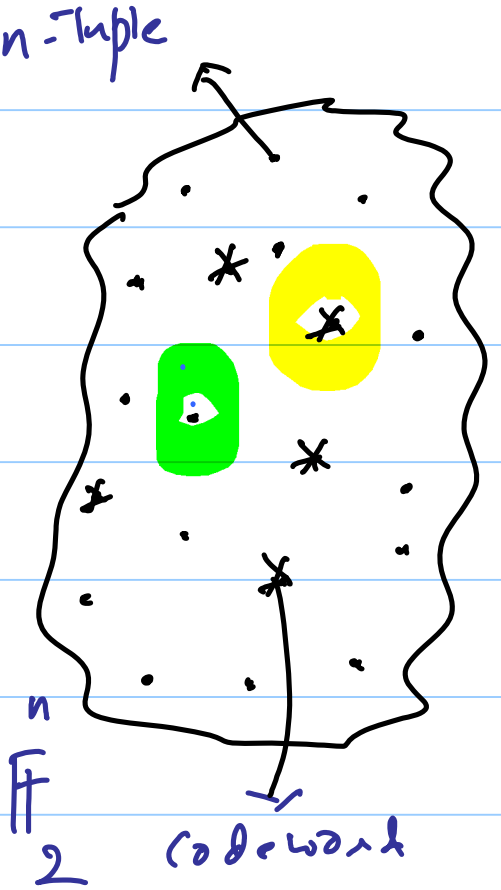


1x
4
1
1
3
1

Binary (Block) Codes

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 $\mathcal{C}$ = $|\mathcal{C}|$ = # of codewords in the code

2. rate R of $\mathcal{C}$ = $\frac{\log_2 |\mathcal{C}|}{n}$

Where n is the block length of the code

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

(3) block length n of the code itself

(4) the minimum distance

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

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

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 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 Publishsing 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.

(all of our example codes will have block length $n = 7$)

Eg 1 The repetition code.

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

Parameters:

(i) size $= 2$

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

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

Eg 2. Single Parity Check Code (spc code)

$$\mathbb{C} = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_7 \end{bmatrix} \;\middle|\; \sum_{i=1}^{7} x_i = 0 \right\}$$

(modulo 2 sum = 0)

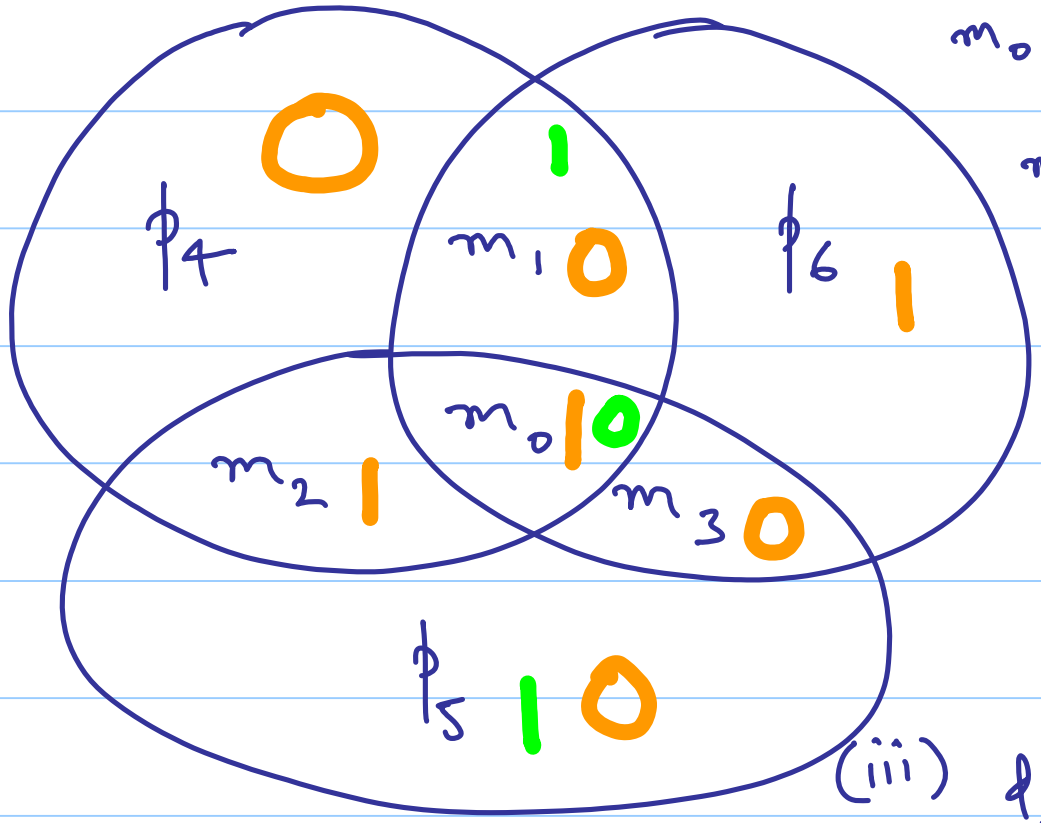
Code Parameters

(i) size = 2^6

(ii) rate = $\frac{6}{7}$

(iii) $d_{min}(\mathbb{C}) = 2$ Eg $d_H \left\{ \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right\}$

Eg 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) $d_{min}(C) = ?$

can show that $d_{min} = 3$ (Exercise)

Code parameters:

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

(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 detected and corrected and any combination of t errors, $t_c < t \leq t_d$ can be detected as an unconnectable error.

(note: in the pair, we will always assume that $t_d \geq t_c$).

Theorem 1 A binary block code $\mathcal{C}$ is a (t_c, t_d) code iff:

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

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

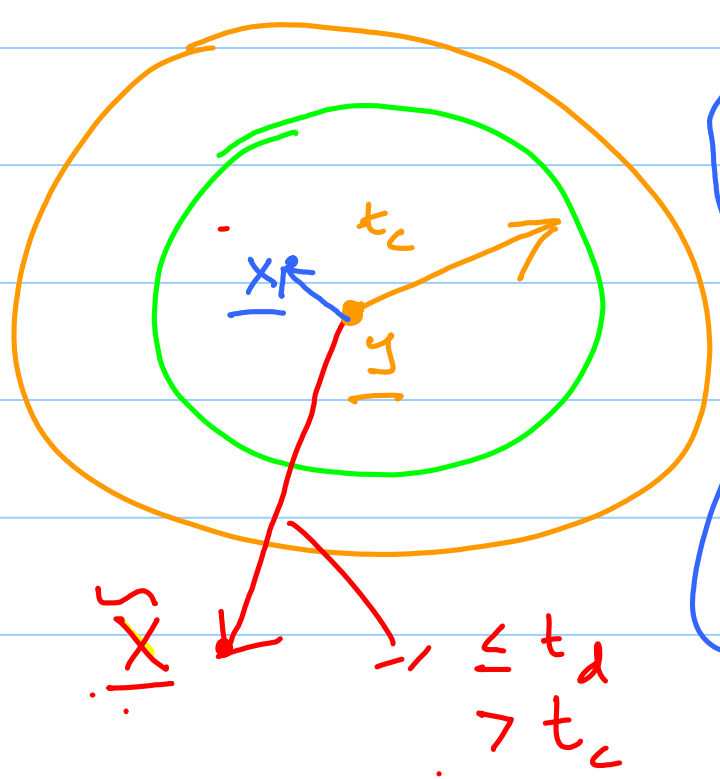
Adopt the following decoding algorithm:

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

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

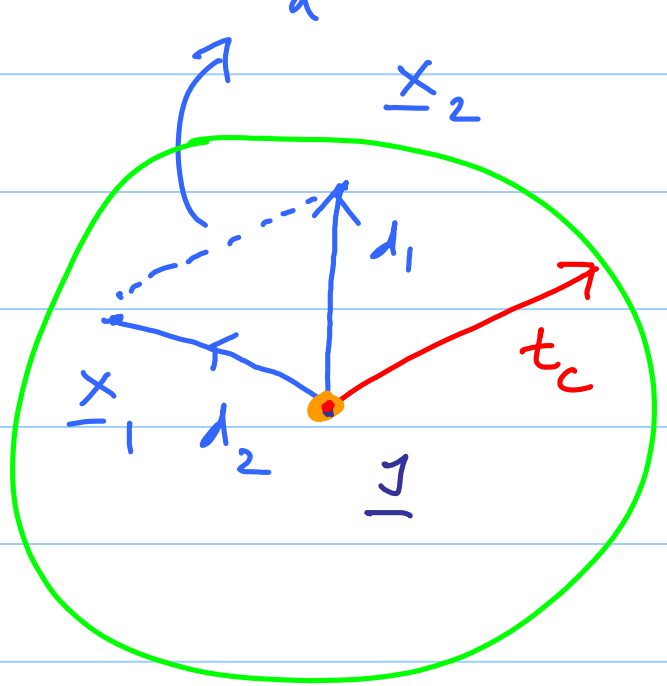
define

$$B(\underline{a}, r) = \left\{ \underline{z} \in \mathbb{F}_2^n \;\middle|\; d_H(\underline{a}, \underline{z}) \leq r \right\}$$



— If $B(\underline{y}, t_c)$ 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 $B(\underline{y}, t_c)$ to contain more than one codeword.

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

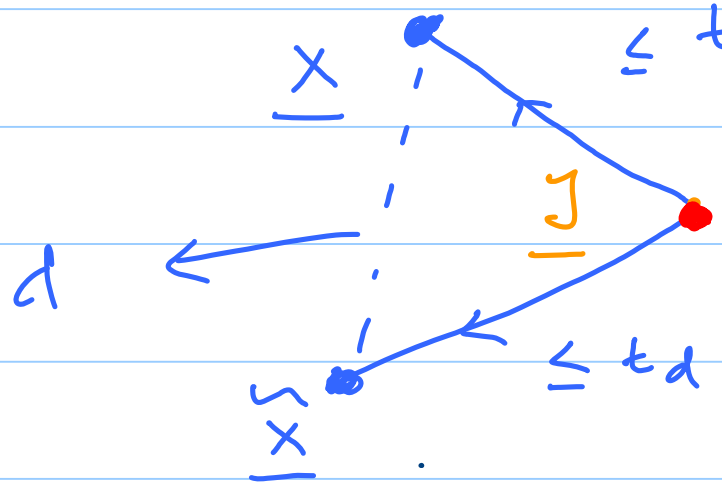
and $d_H(\underline{y}, \underline{x}_2) \leq t_c$

$\Rightarrow$ (by t_c Δ inequality) that

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

$$< t_c + t_d + 1$$

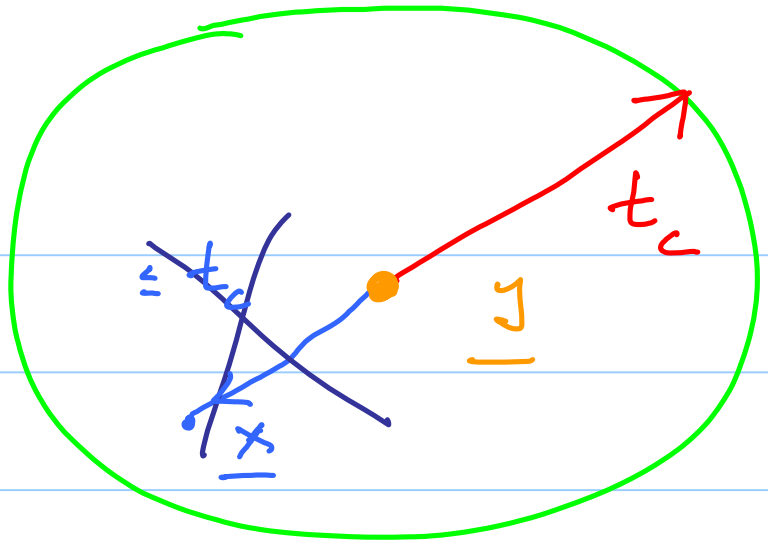
(contradiction)



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

$< t_c + t_d + 1$

$= d_{min}(C)$.



suppose there is no codeword 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 (if the only if part)

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

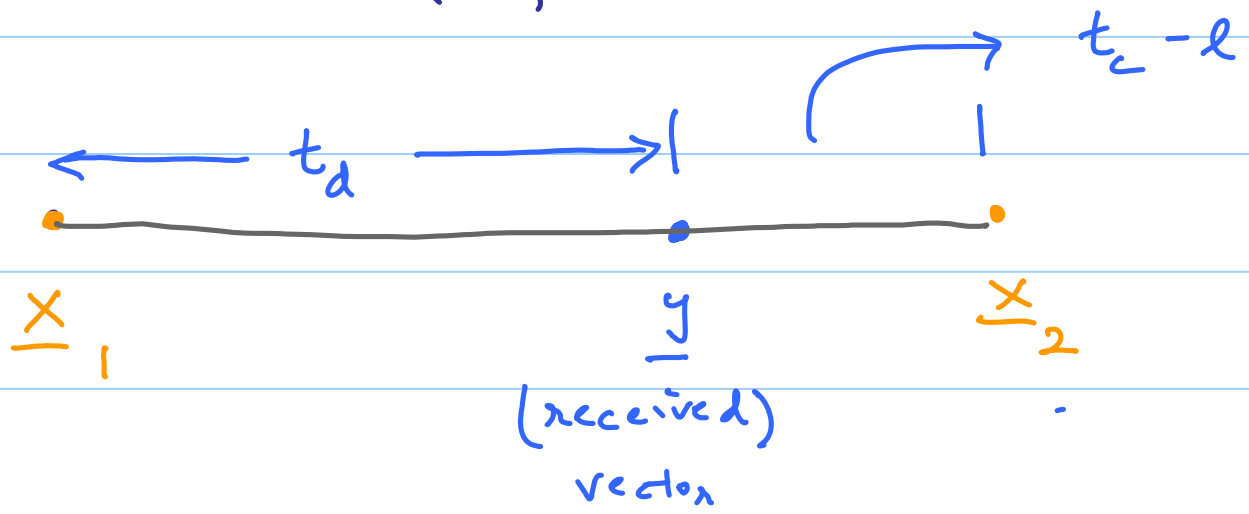
Suppose not $\Rightarrow$ $d_{min} < t_c + t_d + 1$

$$\boxed{d_{min} = t_c + t_d - l} \quad (\text{say}) \quad l \geq 0$$

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

(there exists)

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



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

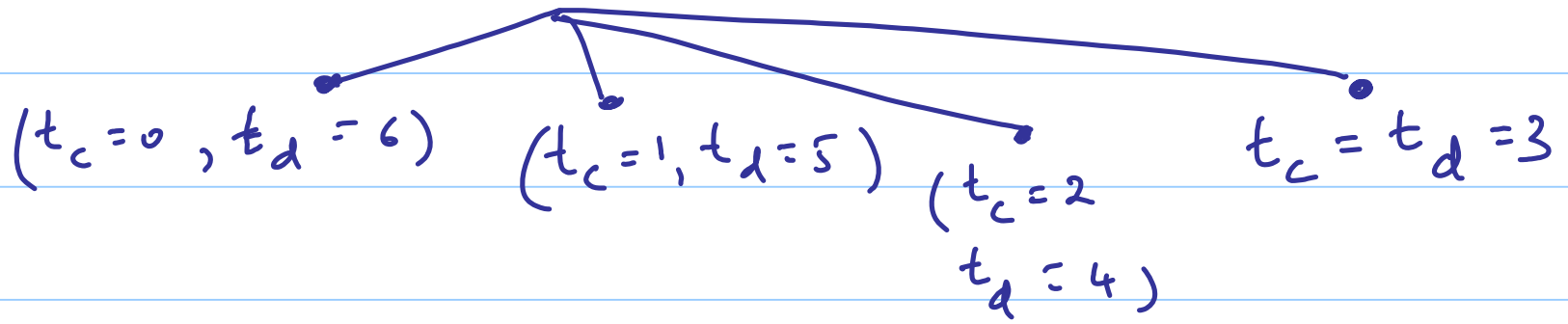
Thus when $d_{min} < t_c + t_d + 1$,

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

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

pronounced "a inverse"

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

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

Note: Abelian groups are also called commutative groups.

Eg. $(\mathbb{F}_2^n, +)$ → modulo 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.

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

this is an example of an Abelian group.

Eg Consider $(G, \cdot) = (Z_n, +)$ modulo-n addition.

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

$Z_n = \{0, 1, 2, \ldots, n-1\}$

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

$\Rightarrow$ $e_1 + e_2 = e_1$

$\Rightarrow$ $e_1 = e_2$

(ii) every element has a unique inverse.

Suppose both c and b are inverses of a $\Rightarrow$ $cab = c(ab) = c(e) = c$

$(ca)b = (e)b = b$

$\therefore b = c$

(iii) $(ab)^{-1} = b^{-1}a^{-1}$ (Exercise)

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

(v) cancellation law holds:

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

If $c^{-1}(ca) = c^{-1}(cb)$

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

$\Rightarrow ea = cb$

$\Rightarrow a = b$

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

Lec 4: Subgroups and Equivalence Relations

(a further example of a group)

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

$\mathbb{Z}_n^*$: nonzero elements in the set of integers modulo n

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

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

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!

Eg $(\mathbb{Z}_p^*, \cdot)$ p = prime

- CL ✓
- ASSOC ✓
- I.E. ✓ $(=1)$
- INVERSE

COMMUT. ✓

Eg $p = 5$ $(\mathbb{Z}_5^*, \cdot)$

$a \cdot b = 0 \pmod p$

$\Rightarrow a \cdot b \overset{\Delta}{=} \text{Rem}\left\{\frac{a \times b}{p}\right\}$

$\Rightarrow p \mid a \cdot b \Rightarrow p \mid a$ or $p \mid b$

$\Rightarrow a = 0 \mod p$ or else $b = 0 \mod p$

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

$p = 5$

$(2)^{-1} = ?$ $\qquad \mathbb{Z}_p^* = \{1, 2, 3, 4\}$

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

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

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

by cancellation:

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

(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.

Eg 1 $H = G$ clear ✓

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

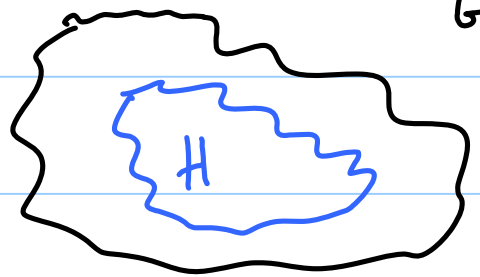
easy to check that $(\{e\}, \cdot)$ is a subgroup.

(called trivial examples).

Testing for a subgroup

Brute force:

- CL
- ASSOC
- I . E
- INVERSE



Better:

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

Setting $a = b \Rightarrow a b^{-1} = e$

$\Rightarrow e \in H$

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

CLOSURE: given $a, b \in H$,

we know that $b^{-1} \in H$

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

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

---

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

$a, a^2, a^3 \ldots \ldots$

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 A that are related to b via R)

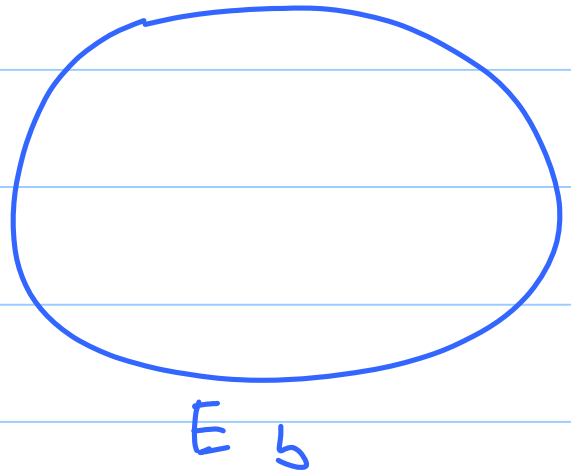
DEFINITION A relation R is said to be an equivalence relation on A provided:

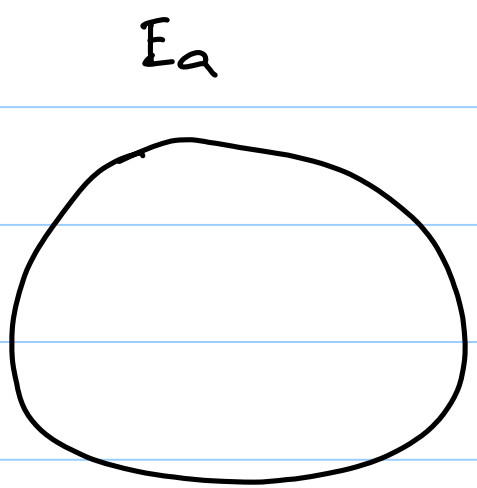
(i) $a \sim a$ REFLEXIVE

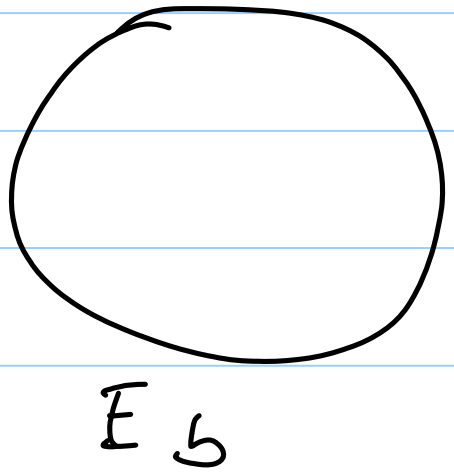
(ii) $a \sim b \Rightarrow b \sim a$ SYMMETRY

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

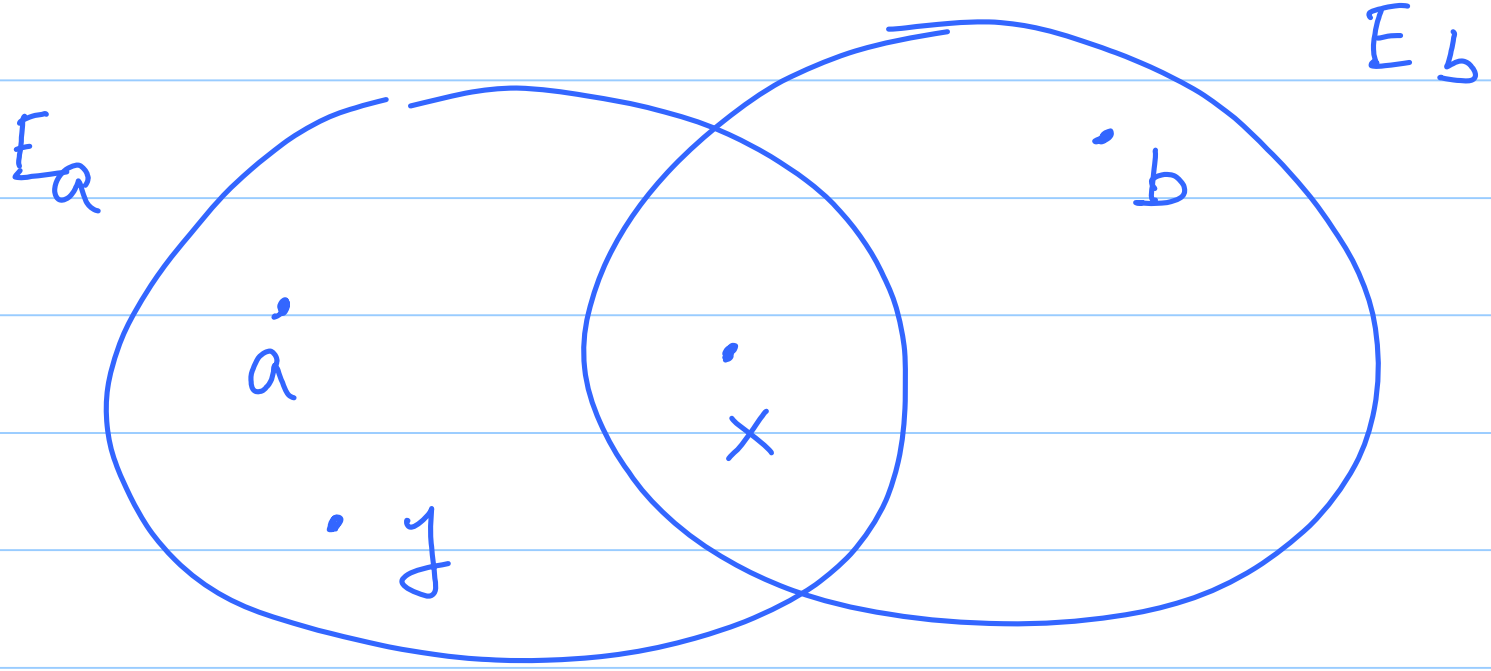
CLAIM: If R is an equivalence relation then if $a, b \in A$, then either $E_a = E_b$ or else $E_a \cap E_b = \phi$ (the empty set)







(we will sketch the proof using a figure:)



Suppose $y \in E_a \setminus E_b$

Suppose $x \in E_a \cap E_b$

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

$\Rightarrow \quad y \in E_b$

a contradiction!

Cosets of a subgroup.

Let $(a, \cdot)$ be a group and

$(H, \cdot)$ be a subgroup of a.

let us define

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

i.e.,

$$R = \{ (a,b) \in G \times G \mid a \cdot b^{-1} \in H \}$$

CLAIM: This is an equivalence relation. ✓

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

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

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

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

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

$\therefore$ Yes!

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

$\Downarrow \qquad \Downarrow$

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

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

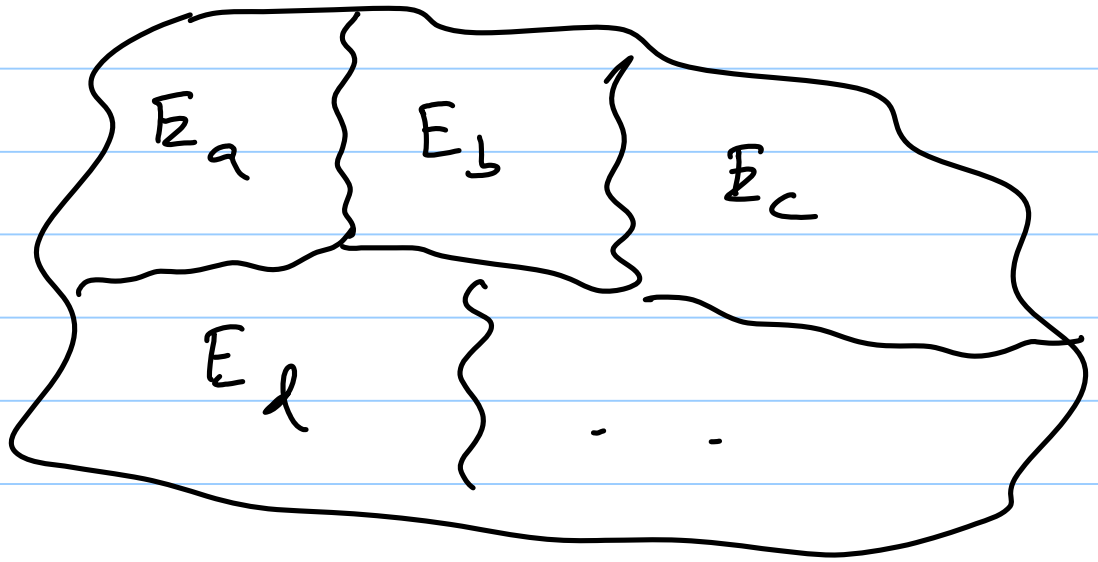
$\Rightarrow a b^{-1} b c^{-1} \in H$

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

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

Lec 5: Cosets, Rings & Fields

We saw last time that an equivalence relation partitions the set. S $\sim$

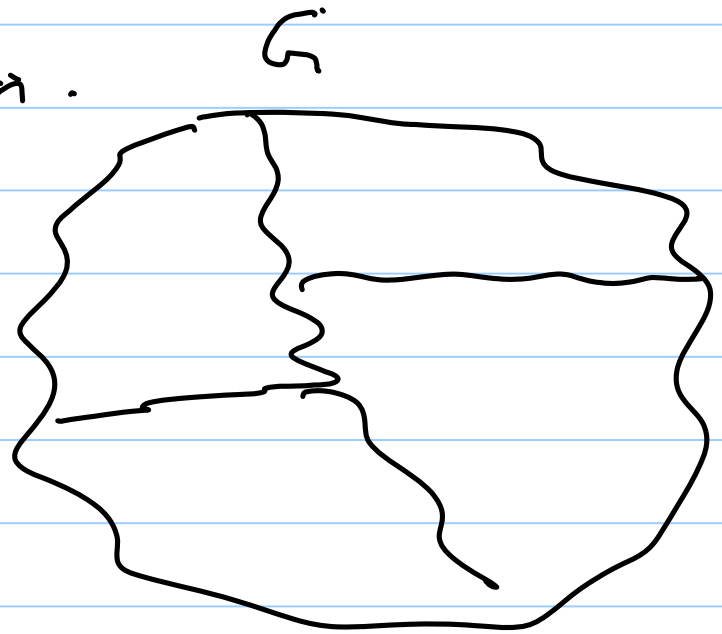


$(G, \cdot)$ group

$(H, \cdot)$ subgroup

$a, b \in G$ $\quad a \sim b$ if $ab^{-1} \in H$

saw last time that this is an equivalence relation.



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

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

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

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

$\therefore$ $\boxed{E_b = Hb}$

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!)

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

the equivalence classes are all of the form:

$$Hb \iff H + b$$

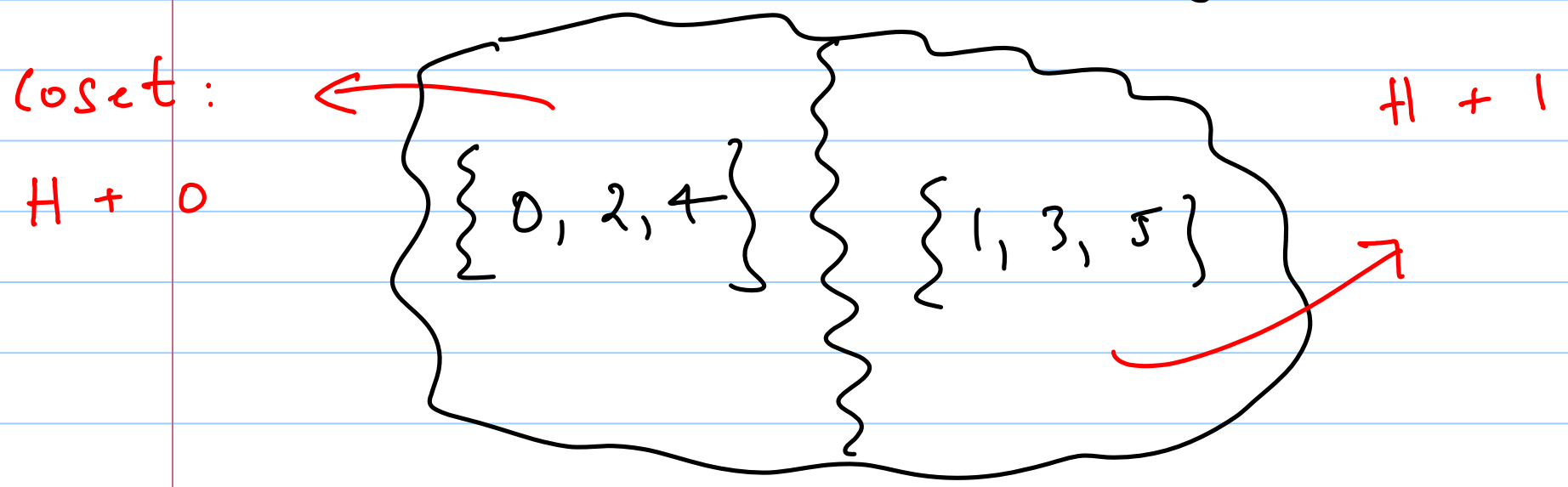
$H = \{0, 2, 4\}$

$b = 0 \Rightarrow H + b = H$ itself

$b = 1 \Rightarrow H + 1$

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

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

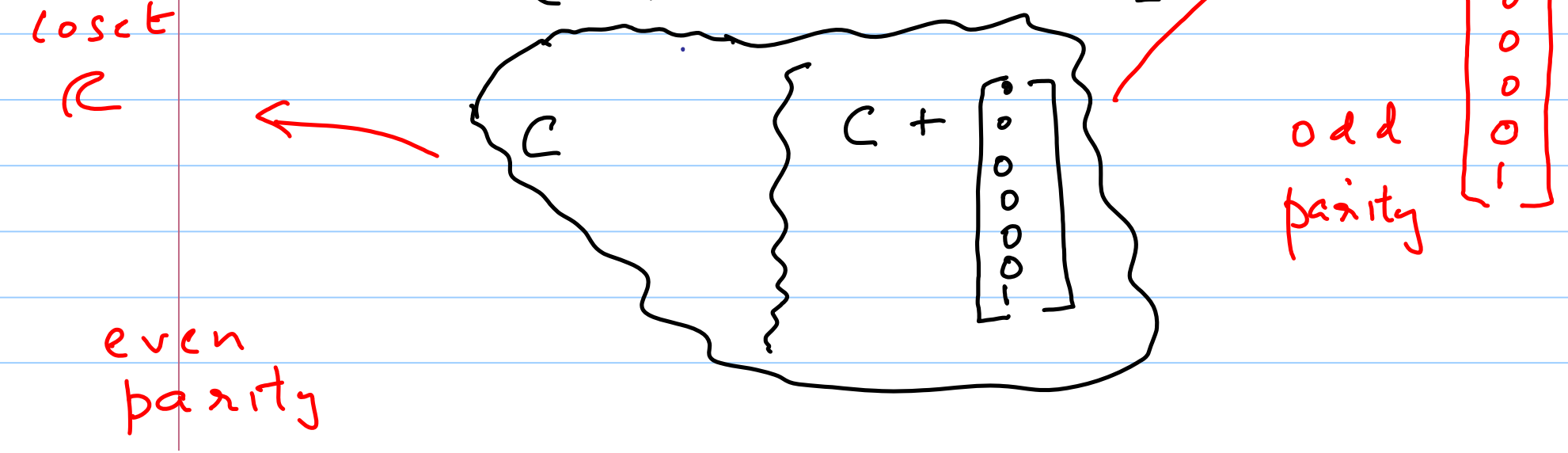


Eg 2 (of partitioning into equivalence classes)

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

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

where C is the even parity code (spc code).



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 $ab^{-1} \in H$

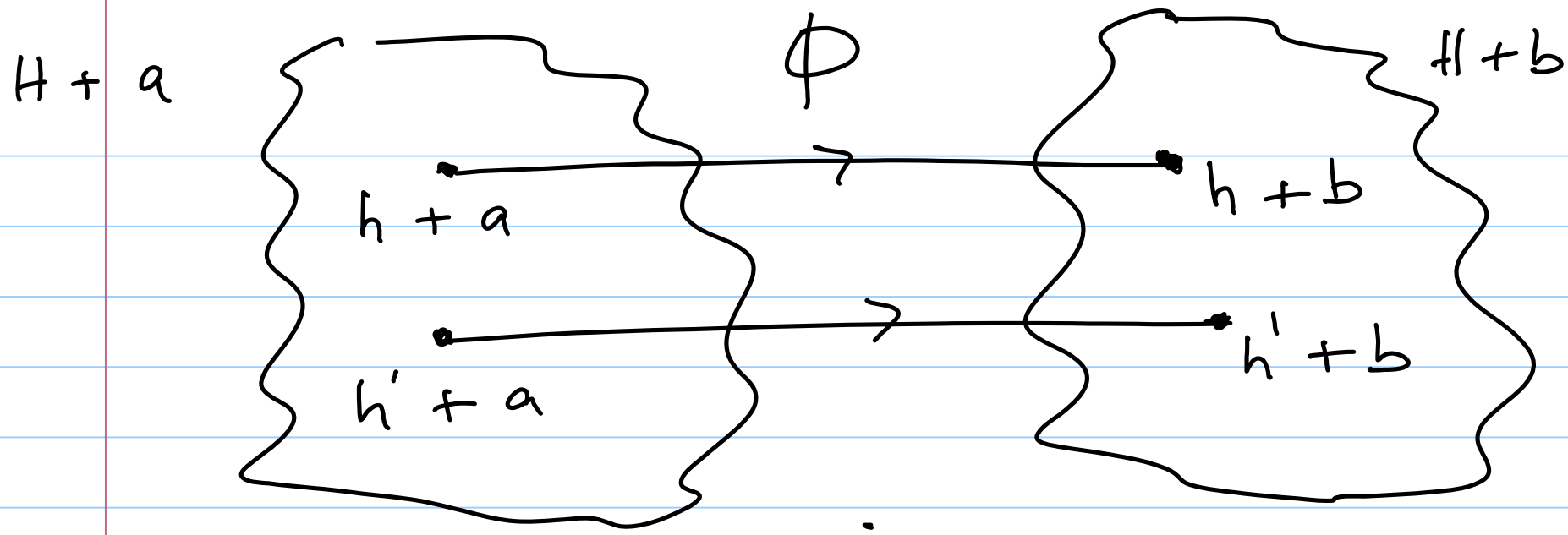
Then there is a 1-1 correspondence between the elements of different equivalence classes.

(i.e.) there is a 1-1 correspondence between the {cosets of H in G.}

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 Hb, $b \in G$.

Let Ha, Hb be two distinct equivalence classes.



we define ϕ via:

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

clearly ϕ is onto

To show 1-1, assume to the contrary that

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

i.e., $h_1 + b = h_2 + b$

$\Rightarrow$ by cancellation

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) $a, b \in R \Rightarrow a \cdot b \in R$

(CLOSURE UNDER MULTIPLICATION)

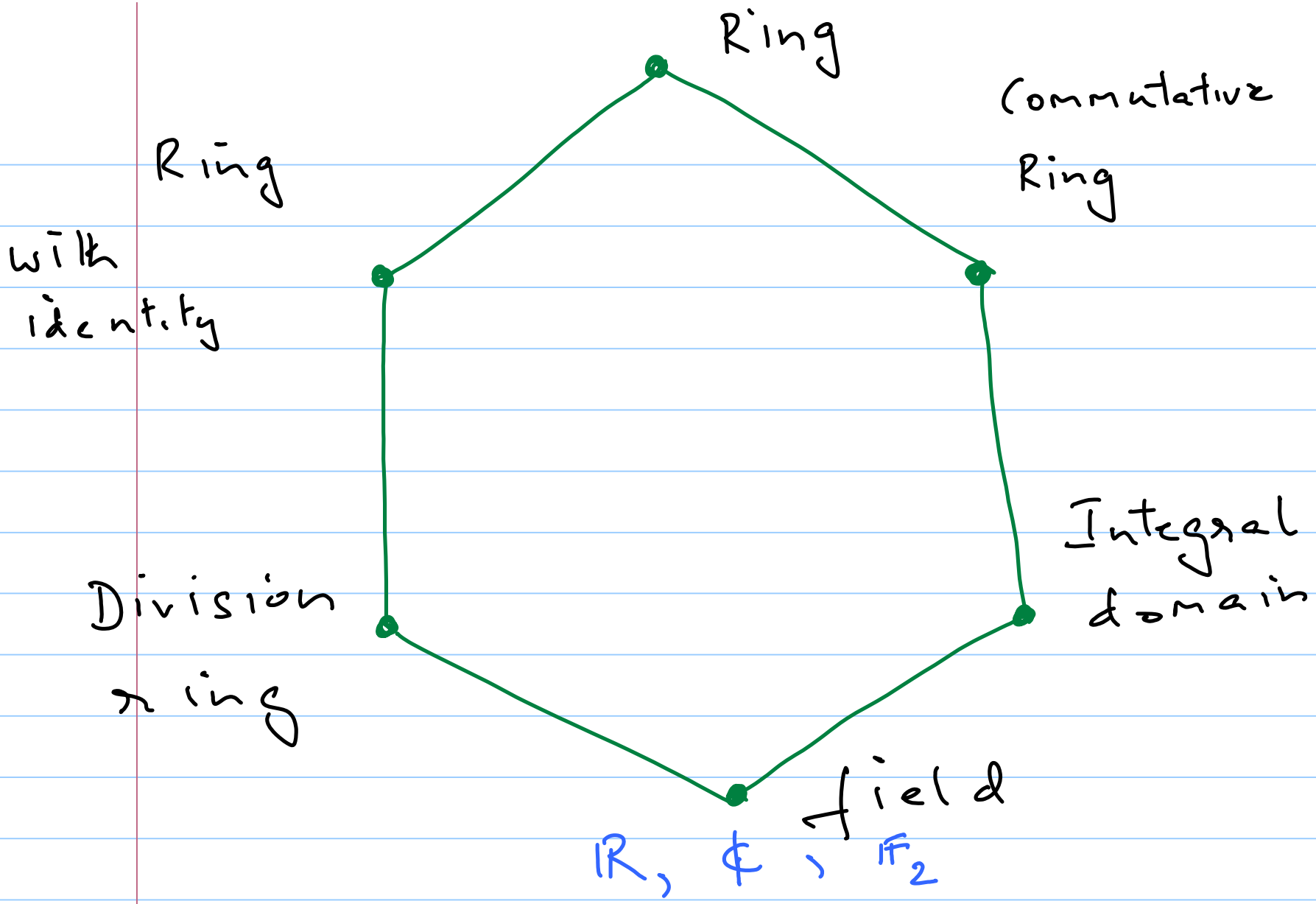
(iii) $a \cdot (b + c) = a \cdot b + a \cdot c$

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

(DISTRIBUTIVE LAWS)

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

ASSOCIATIVITY UNDER MULTPLN.


Ring
Commutative Ring
Integral domain
field
$\mathbb{R}$, $\mathbb{C}$, $\mathbb{F}_2$
Division ring
Ring with identity

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,

$$a \cdot b = 0 \quad \text{for } a, b \in R$$

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

- A division ring is a ring with identity in which every nonzero 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
  - non zero elements have inverses

$\Rightarrow$ in a field F,

$(F, +)$ is an Abelian group

$(F^*, \cdot)$ is also an Abelian group

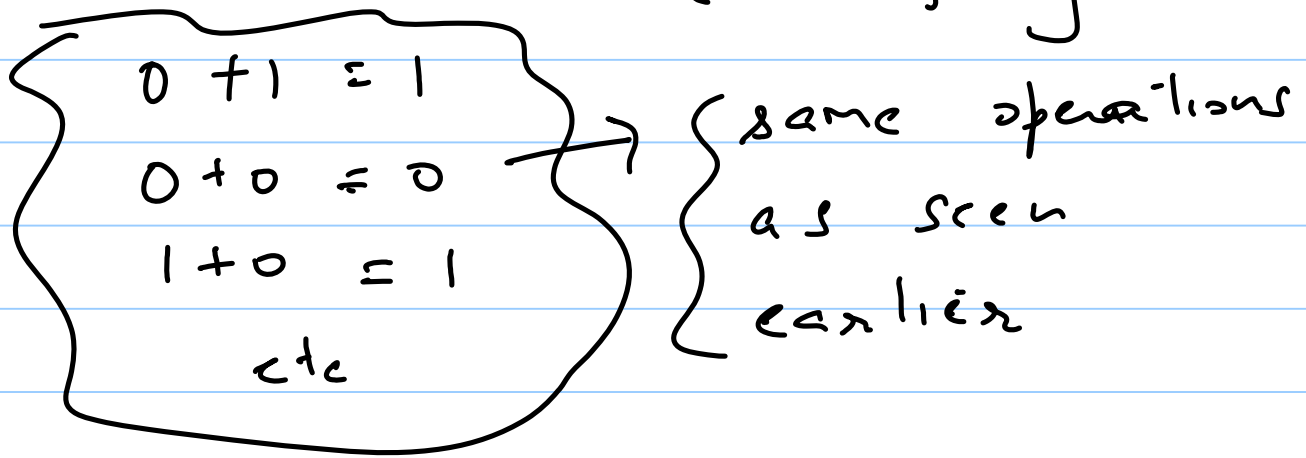
Examples of rings

— the real numbers $\mathbb{R}$

the complex numbers $\mathbb{C}$

the finite field $\{0, 1\}$

} fields

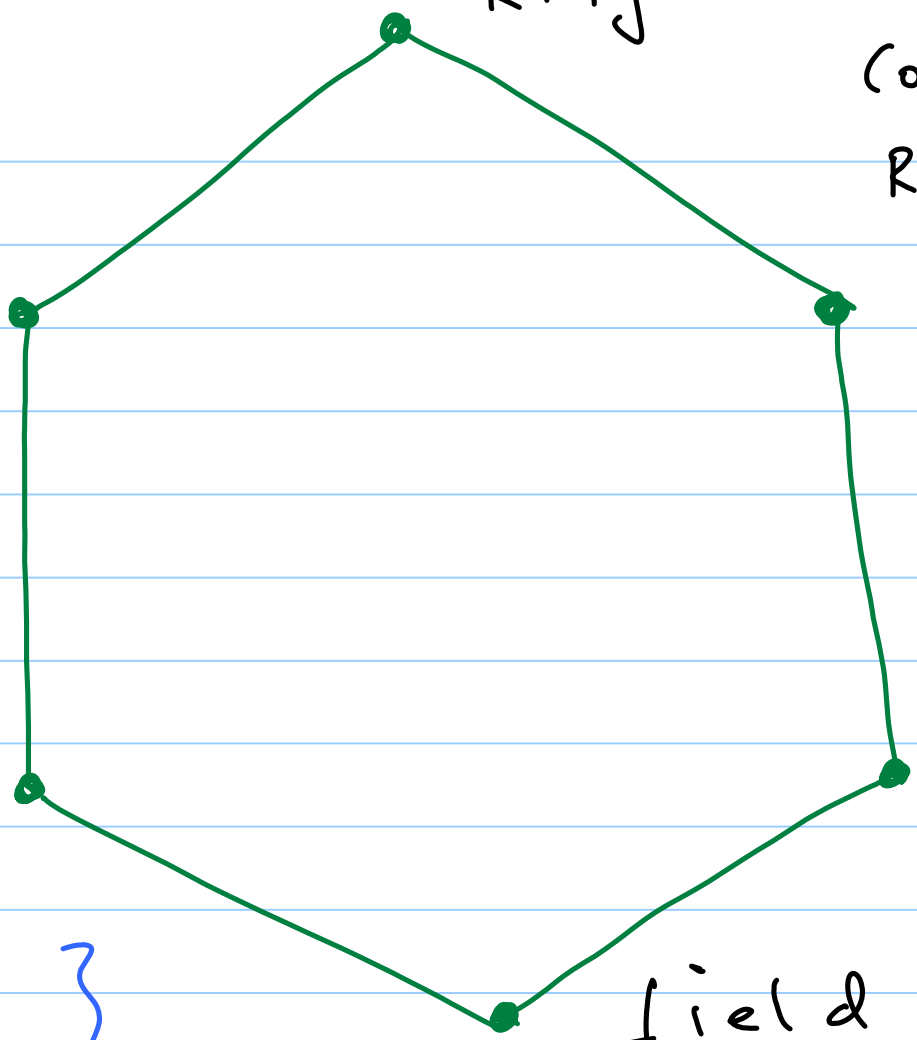


Summary of Lecture 5:

- 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 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
Commutative Ring
$\mathbb{Z}_6$
Integral domain
$\mathbb{Z}$, $2\mathbb{Z}$
$\mathbb{F}[x]$
Field
$\mathbb{R}$, $\mathbb{C}$, $\mathbb{F}_2$
Division ring
Hamilton's quaternions
Ring with identity
$\mathbb{Z}$ $\mathbb{R}^{m\times n}$
$\mathbb{F}[x]$

$\mathbb{F}_2 = \{0, 1\}$ 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 — integral domain

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

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

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

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

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

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

(degree)

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

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

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

Vector Spaces

Defn A vector space $(V, +, \mathbb{F}, \cdot)$ 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) $c \underline{v} \in V$, $c \in \mathbb{F}$, $\underline{v} \in V$ — CLOSURE UNDER SCALAR MULTPLN.

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

(iv) $(c_1 + c_2) \underline{v} = c_1 \underline{v} + c_2 \underline{v}$

$c(\underline{v}_1 + \underline{v}_2) = c\underline{v}_1 + c\underline{v}_2$

DISTRIBUTIVE LAWS

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

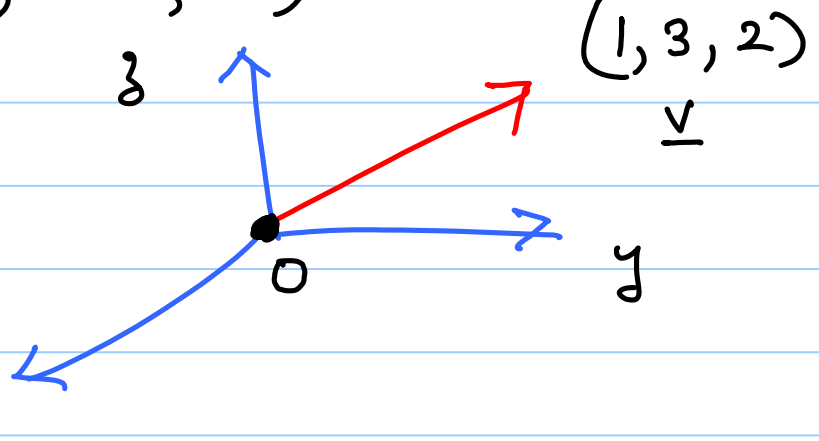
1 ↓ multiplicative identity in $\mathbb{F}$

Examples

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

Eg $n = 3$

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



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

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

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

Derived Properties

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

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

$\underline{0}$: additive identity in the group $(V, +)$

Pf

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

$$\| \quad \text{adding } -\underline{v} \text{ to}$$

$$1\underline{v} \quad \text{both } \underline{v},\ \underline{v} + 0\underline{v}$$

$$\| \quad \text{we see that}$$

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

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

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

$\|$

$$c(\underline{v})$$

$$\therefore \quad c(\underline{v}) = c\,\underline{0} + c\,\underline{v}$$

$$\therefore \quad c(\underline{v}) + -\big(c(\underline{v})\big) = c\,\underline{0} + c\,\underline{v} + \big(-c(\underline{v})\big)$$

$\uparrow\!\!\uparrow$

(additive inverse of $c\,\underline{v}$)

$$\therefore \quad \underline{0} = c\,\underline{0} \qquad \text{///}$$

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

Pf if $c = 0$ done

if $c \neq 0$, consider

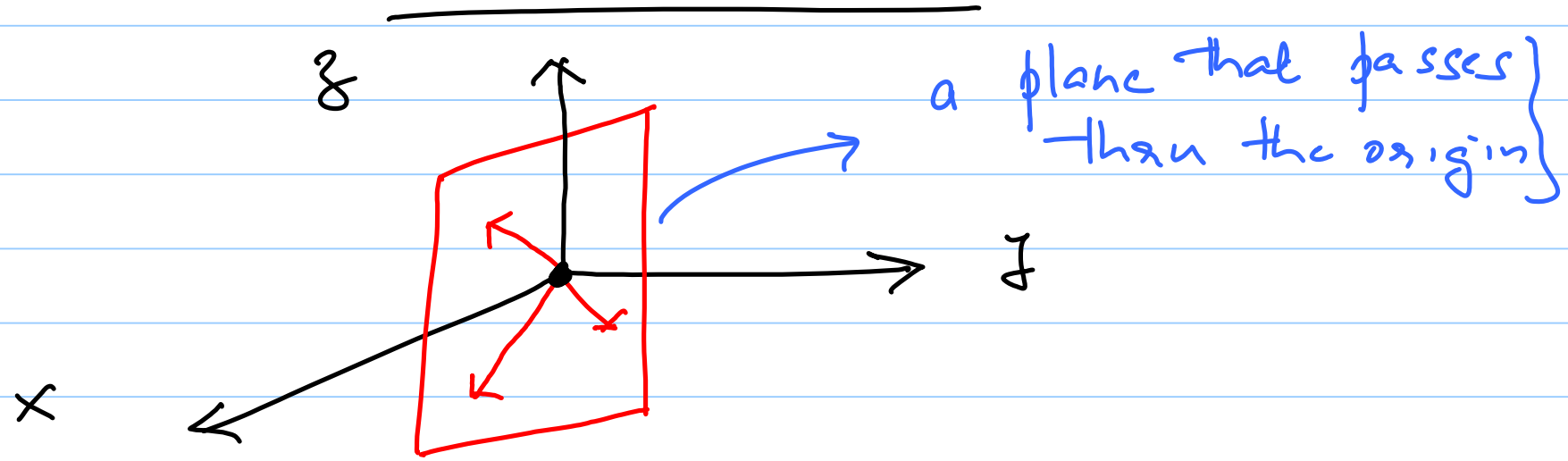
$$c^{-1}(c\,\underline{v}) = c^{-1}(\underline{0}) = \underline{0}$$

$$
\begin{aligned}
&= \\
&(c^{-1} c)\, \underline{v} \\
&= \\
&1\, \underline{v} \\
&= \\
&\underline{v}
\end{aligned}
$$

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

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

(proof left as an exercise!)



Subspaces

Defn A subspace of a vector space $(V, +, \mathbb{F}, \cdot)$ 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:

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

Test for the presence of a subspace?

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

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

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

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

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

how does this

Lec 7: Linear Codes, { linear independence

Summary

- { example rings and their classification
- vector spaces
  - 5 examples
  - derived properties
- subspaces
  - test for a subspace

Test for the presence of a subspace?

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

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

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

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

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

how does this

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

— setting $c = 1 \Rightarrow$ { closure under vector addition

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

$$\underline{x} + (-1)\underline{x} \in W$$

$$\Rightarrow \underline{0} \in W$$

identity element

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

ensures that

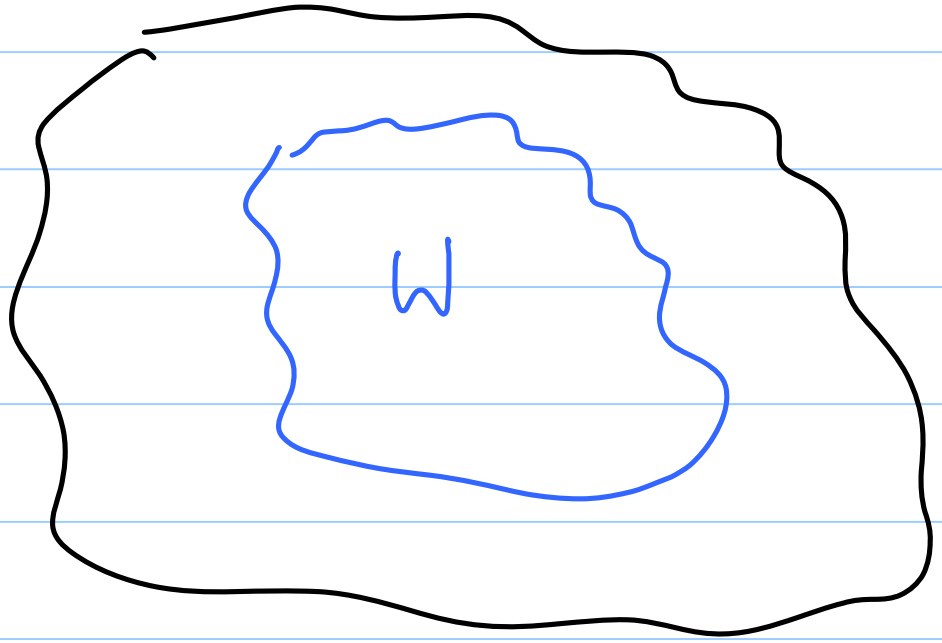
$$(-1)\underline{y} \in W$$

{ inverse is present

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

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

closure under scalar multiplication



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

///

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

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

$\Downarrow$

even parity or spc code.

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

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

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

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

$$= (\underline{1}^t \ \underline{x}) + c(\underline{1}^t \ \underline{y})$$

$$= 0 + c \cdot 0 = 0 \qquad ///$$

Thus R is a subspace of $\mathbb{F}_2^7$.

Note: that for the particular case when $\mathbb{F} = \mathbb{F}_2$ the only nonzero scalar $= 1$

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

reduces to

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

It follows from this that there is no distinction between subspaces of $(\mathbb{F}_2^n, +, \mathbb{F}_2, \cdot)$ and of subgroups of $(\mathbb{F}_2^n, +)$

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

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

linear codes are also called group codes.

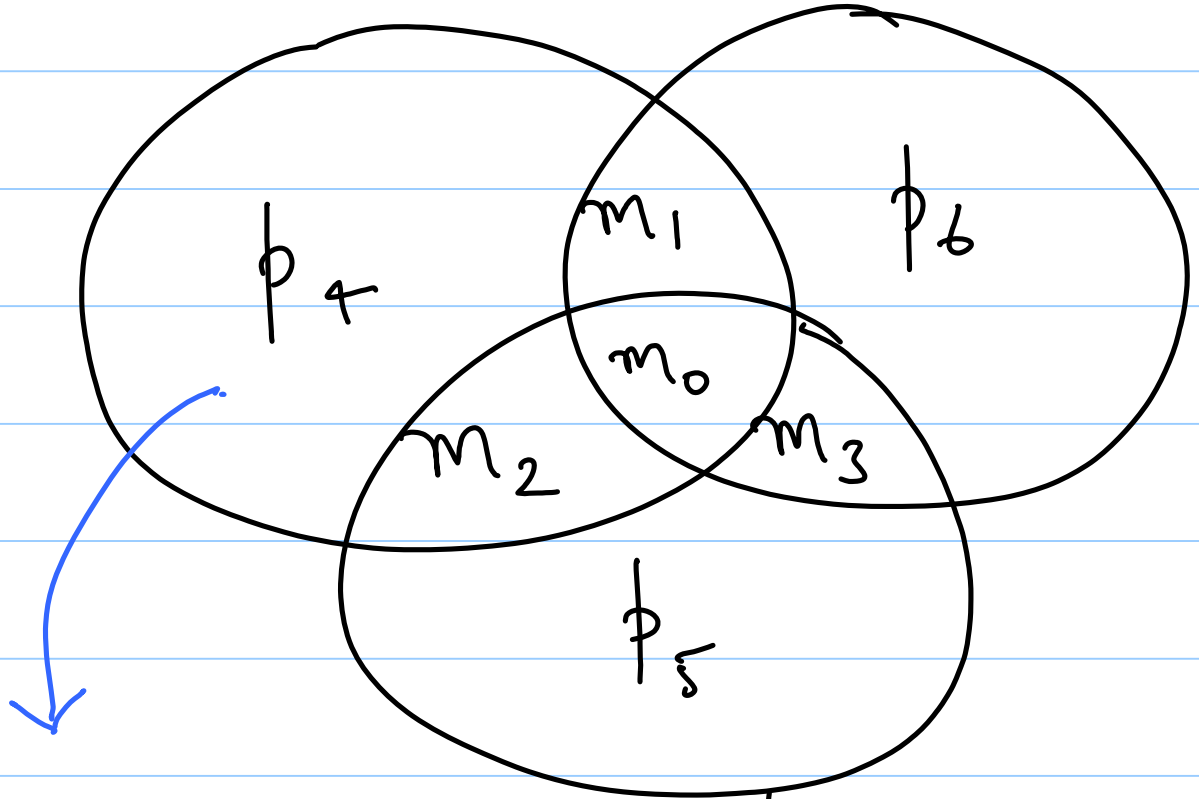
Eg 1 We have already seen that the spc code is a linear code.

Eg 2 the repetition code:

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

clearly this is a linear code.

Eg 3 Hamming Code



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

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

Thus the Hamming code is precisely the set of all codewords $\underline{c}$ satisfying

$$H\,\underline{c} = \underline{0}$$

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

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

$$= \underline{0}$$

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

///

note: Given a $m \times n$, the collection of all vectors $\underline{x}$ s.t. $H\underline{x} = \underline{0}$ is called the nullspace of H:

$$\eta(H) = \left\{ \underline{x} \;\middle|\; H\underline{x} = \underline{0} \right\}.$$

null space of H

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

LINEAR INDEPENCE

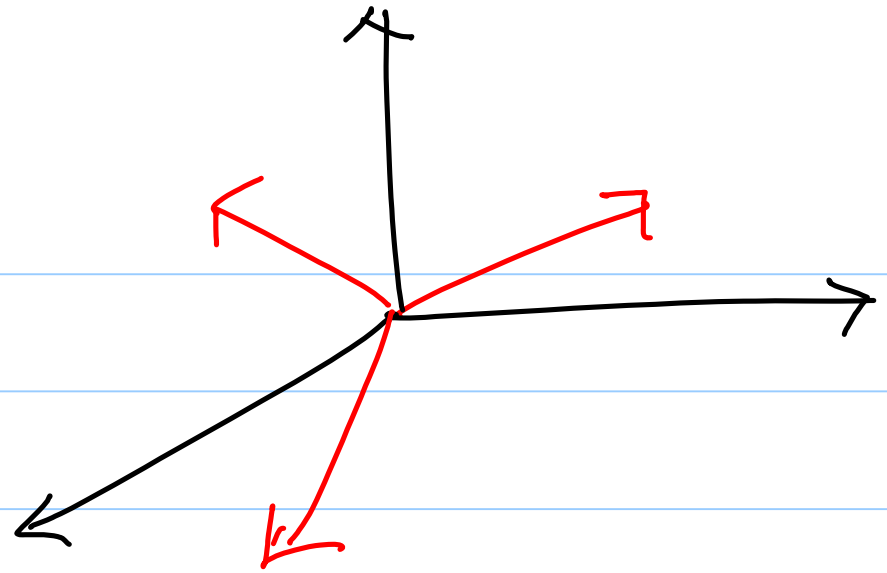
Defn. The vectors $\{\underline{\alpha}_1, \underline{\alpha}_2, \ldots \underline{\alpha}_n, \ldots\}$ are said to be linearly independent if

$$\sum_{j=1}^{r} c_{i_j} \underline{\alpha}_{i_j} = \underline{0}$$

is possible iff

$$c_{i_j} = 0 \quad \text{and} \quad j = 1, \ldots, r.$$

Eg. 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.

Eg 2

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

row-reduced echelon matrix

— { the nonzero rows are linearly independent
3

— the 1, 2, 3 columns containing the pivots are linearly independent.

Eg. vector space:

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

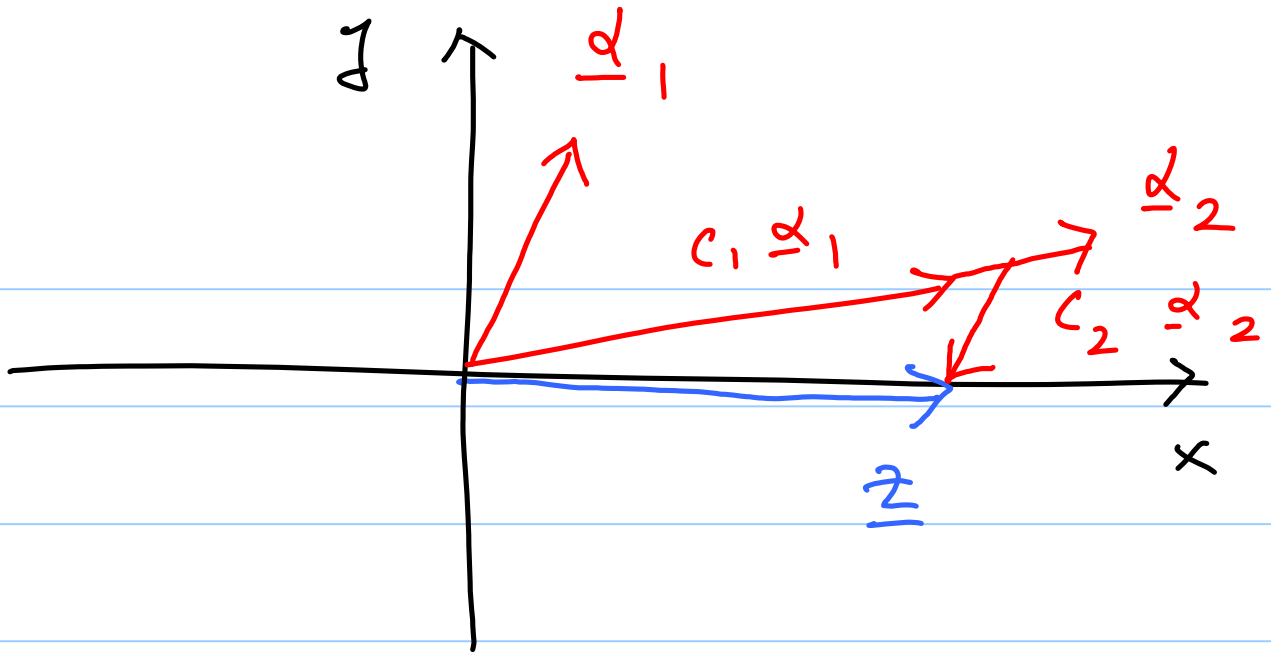
$$\begin{cases} \alpha_1 = x + a & a \in \mathbb{F} \\ \alpha_2 = x^2 + bx + c & b, c \in \mathbb{F} \\ \alpha_3 = x^5 \end{cases}$$

can be verified that these 3 polynomials are linearly independent.

Spanning

Defn A set $\{ \underline{\alpha}_1, \underline{\alpha}_2, \ldots \underline{\alpha}_n, \ldots \}$ is said to **span** a vector space V if every vector $\underline{z} \in V$ can be expressed in the form:

$$\underline{z} = \sum_{j=1}^{n} c_j \underline{\alpha}_j \quad , \quad c_j \in \mathbb{F}$$



$r \geq 1$ is an integer

$$\underline{z} = \sum_{j=1}^{r} c_j \underline{a}_j$$

a linear combination of $\{ \underline{a}_j \}$.

Lec 8: Spanning & Basis

Recap

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

Defn The space spanned by a set $\{\alpha_1, \alpha_2 \ldots \alpha_n \ldots\}$ is the set of all (finite) linear combinations of the form:

$$\sum_{j=1}^{n} c_{i_j} \alpha_{i_j}$$

$n \geq 1$ is an integer,

$c_{i_j} \in$ If $\{$ underlying field of the vector space.

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

notation:

$$W = \langle \alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \cdots \rangle$$

space spanned by the $\{\alpha_i\}$.

Qn: What is the space spanned

by

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

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

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

$$\{ \alpha_1 \quad \alpha_2 \quad \cdots \quad \}$$

of vectors such that:

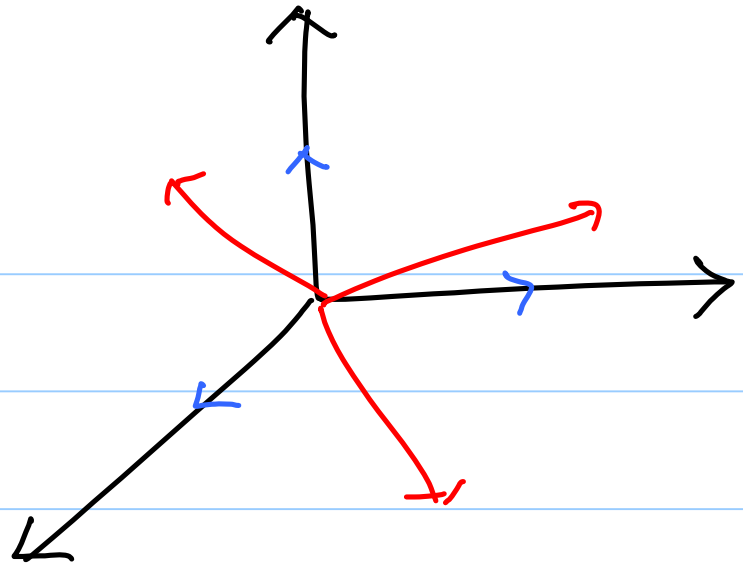
(i) {the set is a linearly independent set

(ii) {the set spans the vector space V

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

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

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

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

$$\{\alpha_1\ \alpha_2\ \cdots\ \alpha_m\} \text{ AND } \{\beta_1\ \beta_2\ \cdots\ \beta_n\}$$

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 $\boxed{n \leq m}$

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

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

V, it follows that $\boxed{n \geq m}$

$\therefore \boxed{n = m}$ ///

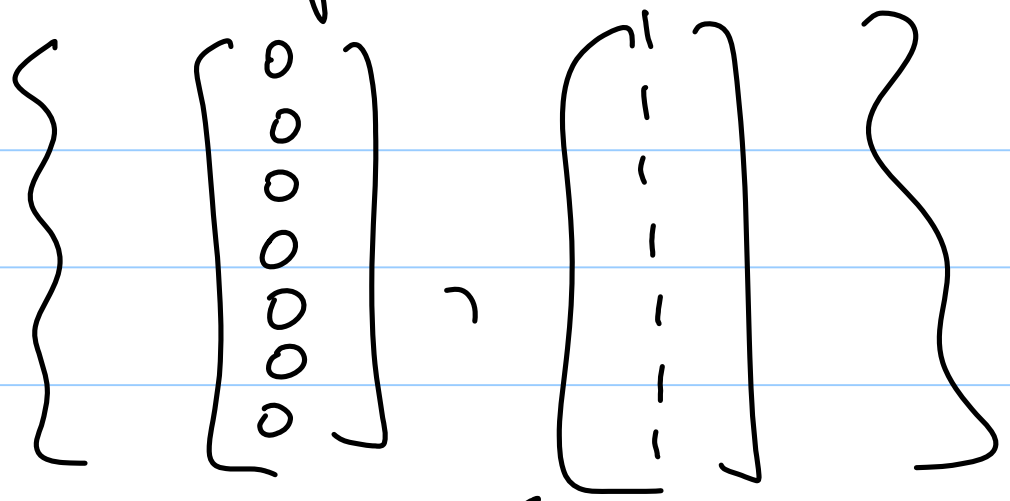
DIMENSION

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.

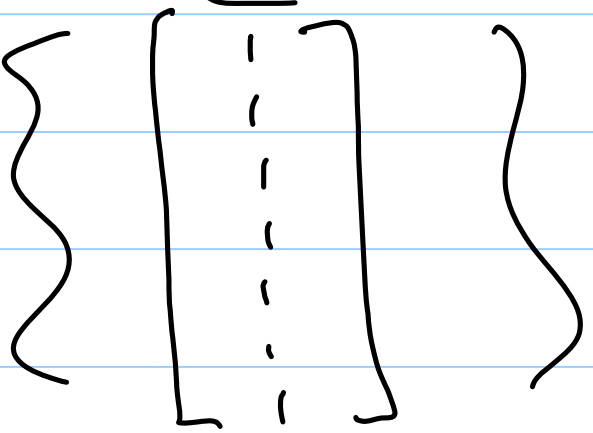
DIMENSION OF A LINEAR CODE

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

Eg The repetition code:



basis $\Rightarrow$



$\therefore$ the dimension $= 1$

$[7, 1, 7]$ code.

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
  - of a vector space
  - of a linear code
    - examples

Note ⓐ 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

(b). Given a basis $\{\underline{\alpha}_1 \; \underline{\alpha}_2 \; \cdots \; \underline{\alpha}_n \; \cdots\}$ for a vector space V, every vector has a unique expansion as a linear combination of elements of the basis.

Pf Suppose

$$\underline{x} = \sum_{j=1}^{r} c_{i_j} \underline{\alpha}_{i_j} = \sum_{k=1}^{s} a_{l_k} \underline{\alpha}_{l_k}$$

$$\Rightarrow \sum_{j=1}^{r} c_{i_j} \underline{\alpha}_{i_j} - \sum_{k=1}^{s} a_{l_k} \underline{\alpha}_{l_k} = 0$$

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

$r = s$, $\{\underline{\alpha}_i\}$ and $\{\underline{\ell}_k\}$ are the same and the coefficients are the same. ///

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

standard basis:

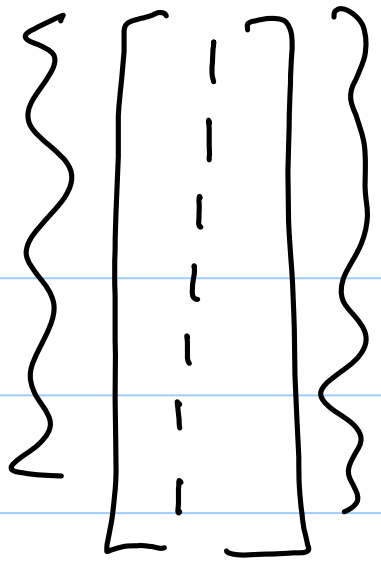
$$\left\{ \underline{\alpha}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \underline{\alpha}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \underline{\alpha}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$

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

this coefficient set is unique.

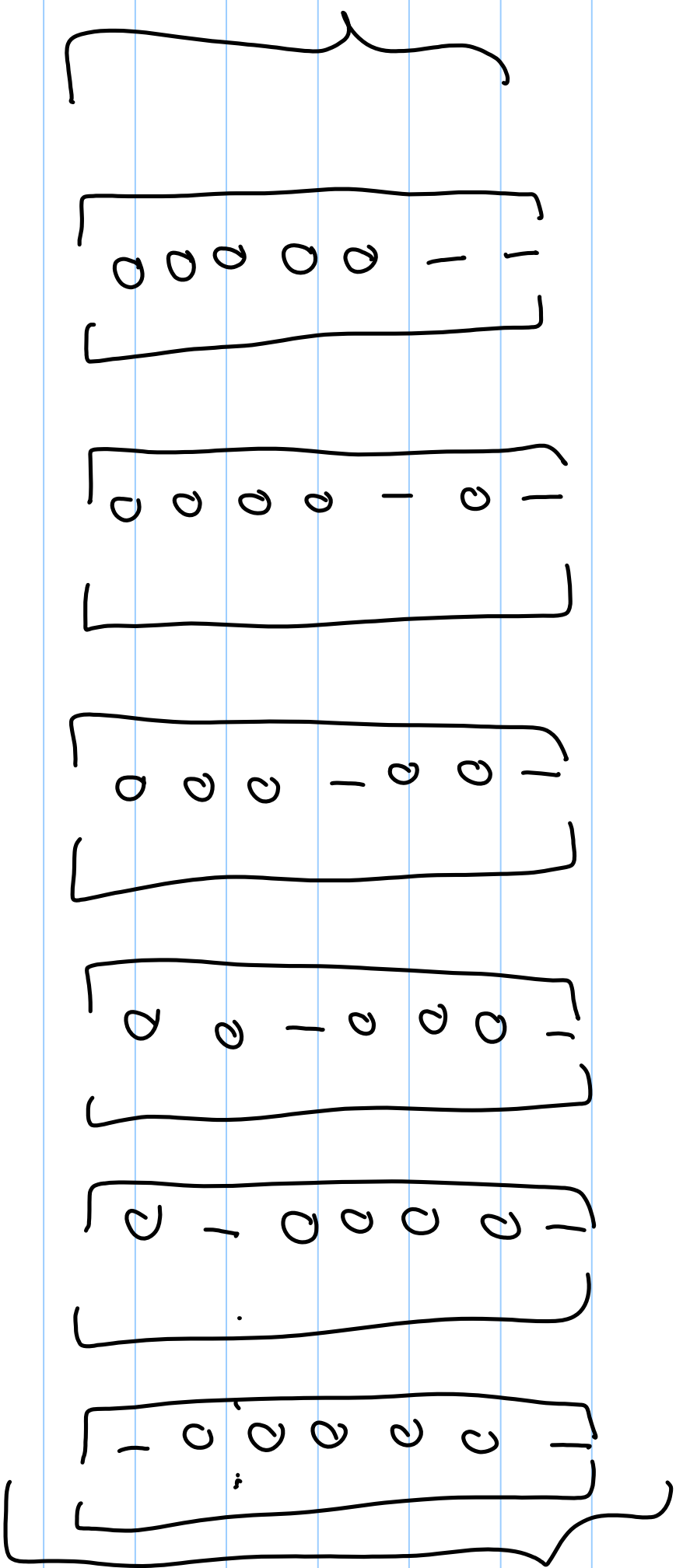
Basis for the 3 example codes

Eg 1 (the repetition code)

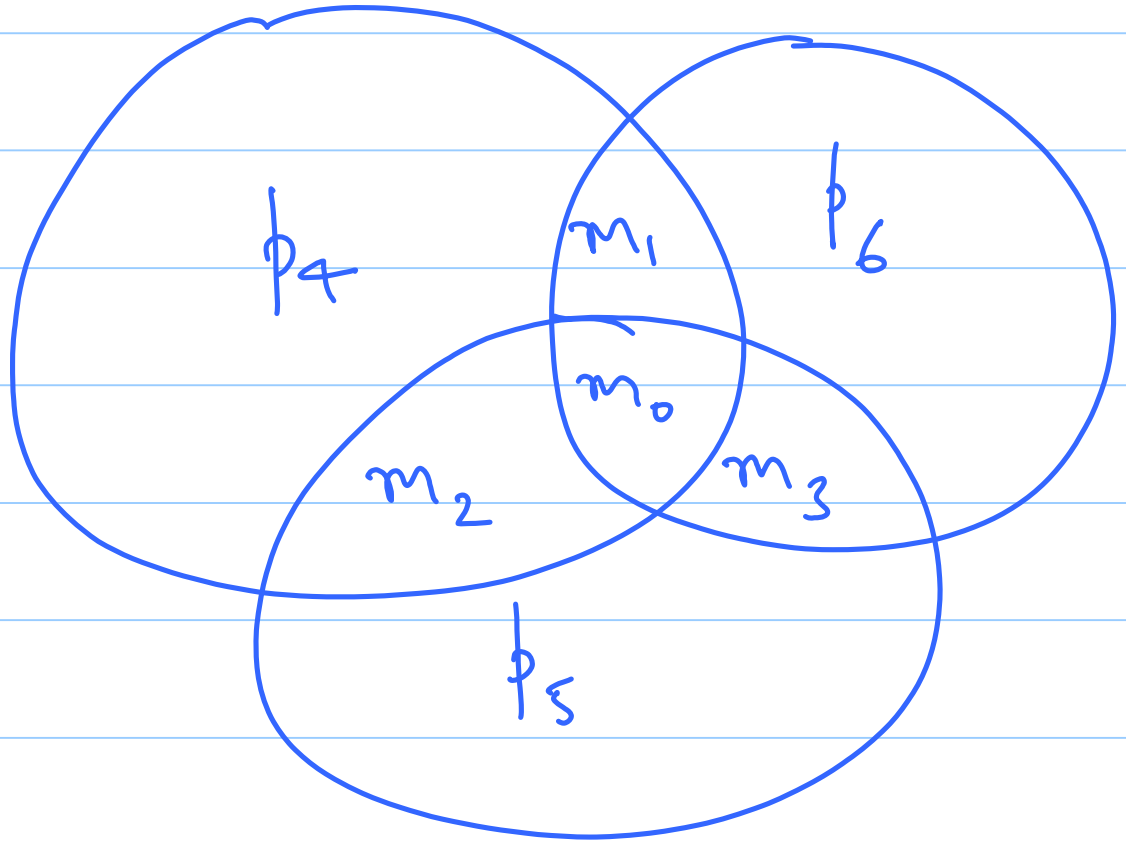
basis = 

Eg 2 Single parity-check code

basis = { the collection of six vectors below:



Eg 3 The Hamming code.



$$
\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}
\begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}
=
\begin{bmatrix} m_3 \\ m_2 \\ m_1 \\ m_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}
$$

The rows of this matrix are the basis of the Hamming code.

The rows of this matrix are the basis of the Hamming code.

This is a generator matrix 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:

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

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

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

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

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

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

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

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 mx for $\mathcal{C}$.

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

Eg 1. The Hamming code
(provided earlier)

Eg 2 (rep code)

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

Eg Spc Code $[7, 6]$

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

(6×7)

$(k \times n)$

Lemma The nullspace of a generator mx of the code $\mathcal{C}$ is precisely the dual code

Pf Let $\underline{x}$ be in the null space of G.

(**note:** the null space of an $(m \times n)$ mx A is precisely the set

$$\eta(A) = \left\{ \underline{x} \;/\; A\underline{x} = \underline{0} \right\}$$

null space.)

$$\vdots \quad \begin{bmatrix} \underline{g}_1^t \\ \underline{g}_2^t \\ \vdots \\ \underline{g}_k^t \end{bmatrix} \begin{bmatrix} \\ \underline{x} \\ \\ \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

G

clearly from the definition of the dual code it follows that if $\underline{y} \in \mathcal{C}^\perp$, then $\underline{y} \in \mathcal{N}(G)$.

On the other hand if $\underline{x} \in \mathcal{N}(G)$, then

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

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

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

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

$$= 0$$

$$\Rightarrow \underline{x} \in \mathcal{C}^{\perp} .$$

///

PARITY-CHECK MATRIX

H

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

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

The dual code $\mathcal{C}^\perp$ is the repetition code having generator mx

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

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

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

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

Thm $(R^\perp)^\perp = R$.

(Sketch) Let H be a gen $m \times n$ of $R^\perp$.

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

Lec 10 { Systematic Generator Matrix

Recap :

(i) Uniqueness of representation wrt a basis

(ii) examples of the 3 Gds

(iii) Dual Code definition

— example

(iv) the generator matrix

(v) $\eta(G) = R^{\perp}$

(vi) the parity-check mx

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

Eg.

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

row space of G is the spc code.

Thm (fundamental theorem of linear algebra)

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

then

$$\text{rank}(A) + \dim(\text{null space}(A)) = n \quad (1)$$

Also for any given matrix A,

$$\text{rank}(A) = \dim(\text{rowspace of } A) \quad \cdots (2)$$

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

$$\dim(\text{rowspace}(A)) + \dim(\text{nullspace}(A)) = n \quad \cdots \text{③}$$

Thm $((\mathcal{C}^\perp)^\perp = \mathcal{C}$

Pf. Let H be a generator matrix for the dual code.

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

Consider the equation:

$$\begin{bmatrix} \underline{h}_1^t \\ \underline{h}_2^t \\ \vdots \\ \underline{h}_{n-k}^t \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \underline{0}$$

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.,

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

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

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

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

It follows that

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

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.

$$\begin{bmatrix} \underline{h}_1^t \\ \underline{h}_2^t \\ \vdots \\ \underline{h}_{n-k}^t \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \underline{0} \quad \cdots (4)$$

H

(the proof follows from noting that the rows $\underline{h}_i^t$ of H belong to the dual code from (4) and form a basis for the code $R^{\perp}$ 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 mx for the $[n, k]$ code C, then H is a valid p.c. mx for the code C.

Pf.

$$\begin{bmatrix} \underline{h}_1^t \\ \vdots \\ \underline{h}_{n-k}^t \end{bmatrix} \begin{bmatrix} \underline{g}_1 & \cdots & \underline{g}_k \end{bmatrix}^T = [0]$$

H^t

It follows that

$$\underline{h}_i^t \, \underline{g}_j = 0 \quad \forall \; i, j$$

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

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

$\Rightarrow R = \eta(H)$

and hence by Lemma 1, H is a valid p.c. mx of R.

Eg If $G = \left[\underbrace{I_k}_{(k \times k)\ \text{identity mx}} \;\vdots\; P\right]$

then $H = \left[P^T \;\vdots\; I_{n-k}\right]$

is a valid p.c. mx of R.

This is because:

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

$$= \underbrace{\begin{bmatrix} P^T & I_{n-k} \end{bmatrix}}_{(n-k \times k)} \begin{bmatrix} I_k \\ \hline P^T \end{bmatrix} = P^T + P^T$$

$$= [0] .$$

Also it is clear that rank (H) $= (n-k)$.

Eg repetition code.

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

I_k $\quad$ P

$$\therefore H = \left[\begin{array}{cc} P^T & I_{n-k} \end{array}\right] = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and this is clearly a valid pc

m x d the repetition code.

SYSTEMATIC GENERATOR MATRIX

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

$$G = \left[\begin{array}{c:c} I_k & P \end{array}\right] .$$

Qn Does every code possess a systematic generator matrix ?

Ans No. Let G be any generator mx of the code R.

$$G = \left[\begin{array}{ccc|ccc} \underline{a}_1 & \underline{a}_2 & \cdots \underline{a}_k & \underline{a}_{k+1} & \cdots & \underline{a}_n \end{array}\right]$$

$$= \left[\begin{array}{c|c} G_1 & G_2 \end{array}\right].$$

then since any other gen mx for R can be obtained through taking linear combinations of G above, it follows that R has a systematic gen mx iff

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

Eg

P is the spc sol^{n} of

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

(7×9)

(4×4)

y_1

y_2

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

A second reason for being interested
in a systematic gen mx:
Consider the map between message
vector $\underline{m}$ and codewords $\underline{c}$ given
by:

$$\begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_t \end{bmatrix} \rightarrow \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_t \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_t \end{bmatrix}$$

$$\boxed{\begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_t \end{bmatrix}}$$

m_1

m_2

$\vdots$

m_t

the message symbols

are split up

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 $\phi : R_1 \longrightarrow R_2$ in 1-1 and onto fashion with the further property that the mapping ϕ corresponds to a coordinate permutation.

Lec 11 { Minimum Distance of a Linear Code

Recap

- defined nowspace
- FTLA
- $(C^{\perp})^{\perp} = C$
- { 2 Lemmas helpful in finding a p.c. mx H

- Systematic generator matrix
  - 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 non-zero codeword.

Pf. Let $\underline{c}$ have $w_H(\underline{c}) = w_{min}$.

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

$$\therefore \boxed{d_{min} \leq w_{min}} \quad \text{----- (1)}$$

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

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

$$\Rightarrow \quad W_H\left(\underbrace{\underline{c}_1 + \underline{c}_2}\right) = \phi_{min}$$

$\in R$ since R is linear!

$$\Rightarrow \quad \boxed{W_{min} \leq \phi_{min}} \qquad \cdots\cdots (2)$$

$\therefore$ from (1) and (2),

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

Eg $\mathcal{C}$ repetition code

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

$\therefore$ $w_{min} = 7 = d_{min}$

Eg $\mathcal{C}$ = spc Gdc.

$$\mathcal{C} = \left\{ \underline{c} \;\middle|\; \sum_{i=1}^{n} c_i = 0 \right\}$$

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

1 1 0 0 0 1 0

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

H

$\underline{h}_1 \ \underline{h}_2 \ \underline{h}_3 \qquad \underline{h}_7$

$$H\underline{c} = \underline{0} \iff \sum_{i=1}^{7} c_i \underline{h}_i = \underline{0}$$

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

Defn Given an $(n-k \times n)$ p.c. matrix H of an $[n, k]$ linear block code R, we define the parameters to

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

Thm $d_{min} = s + 1$

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

$\Rightarrow$ the existence of a linear dependence relation amongst w corresponding columns of the p.c. mx H.

Eg $S = 2$ 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 length n is any code possessing a p.c. mx 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} $d_{min} = n - k + 1$ MDS

Eg 2 (general) spc code :

$$[n, (n-1), 2]$$

n k d_{min}

$d_{min} = n - k + 1$ ∴ 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 $t = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor$.

Ex 1 $\int_C \vec{F} \cdot d\vec{r}$

$\vec{F} = \langle x^2, y^2 \rangle$

C_1: $x^2 + y^2 = 1$

C_2: $(x-2)^2 + y^2 = 1$

Lec 12 : Bounds on the size of a code

Recap

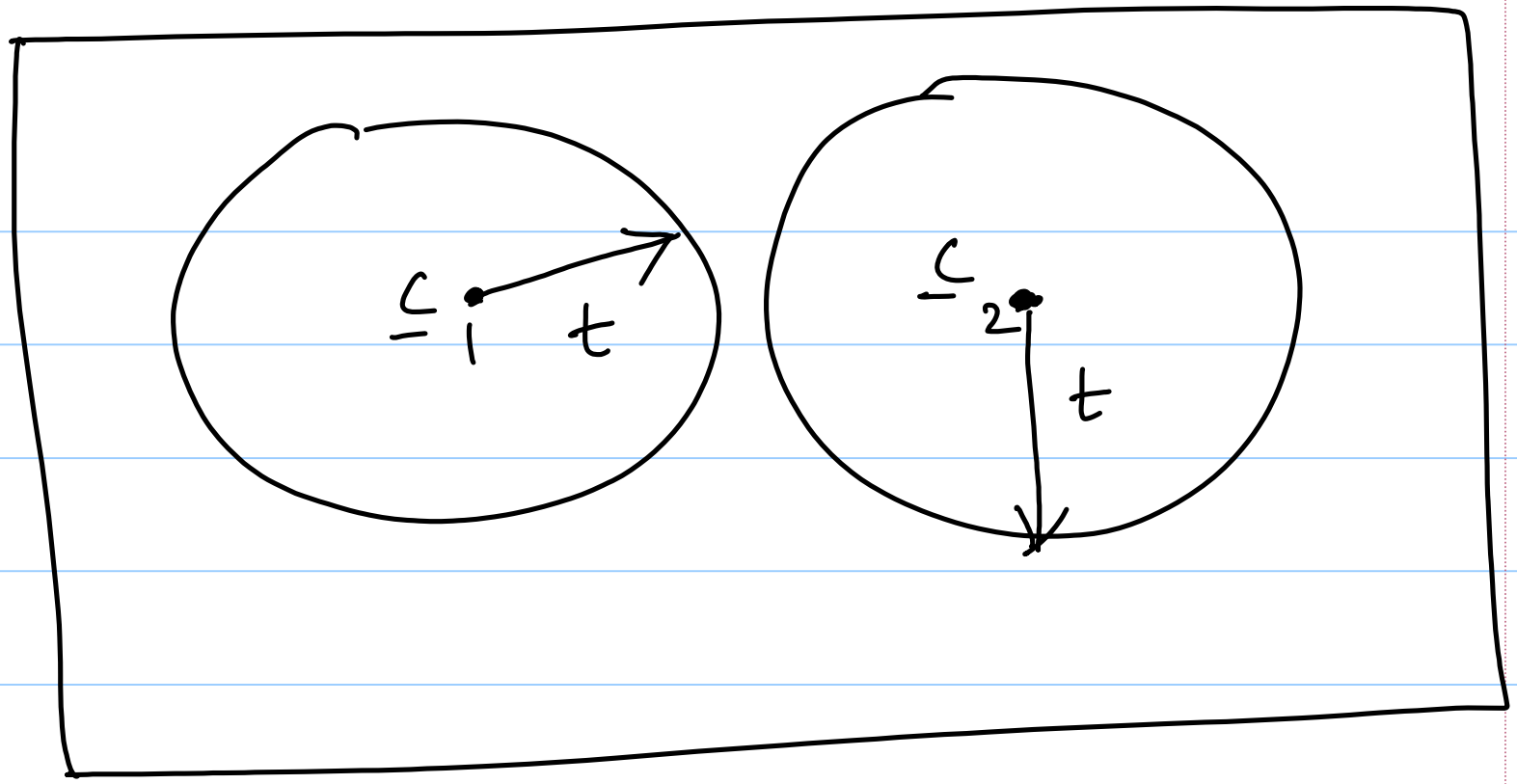
- $d_{min} = w_{min}$
- $d_{min} = s + 1$
- examples
- General Hamming code
- Singleton bound & MDS codes
- Proof of the Hamming Bound

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. $\Leftarrow$



$\mathbb{F}_2^n \supseteq \left\{ B(\underline{c}, t) \;\middle|\; \underline{c} \in C \right\}$

But these balls are disjoint:

$$B(c, t) \cap B(c', t) = \phi$$

It follows that

$$2^n \geq M \left| B(c, t) \right|$$

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

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

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

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

Eg Consider the repetition code for odd values of block length n:

parameters: $[n, 1, n]$

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

$$\sum_{i=0}^{n} \binom{n}{i} = 2^n \qquad \left(\begin{array}{c}\text{well}\\ \text{known}\end{array}\right)$$

When n is odd, since

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

that

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

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

$\therefore$ all such repetition codes are perfect!

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

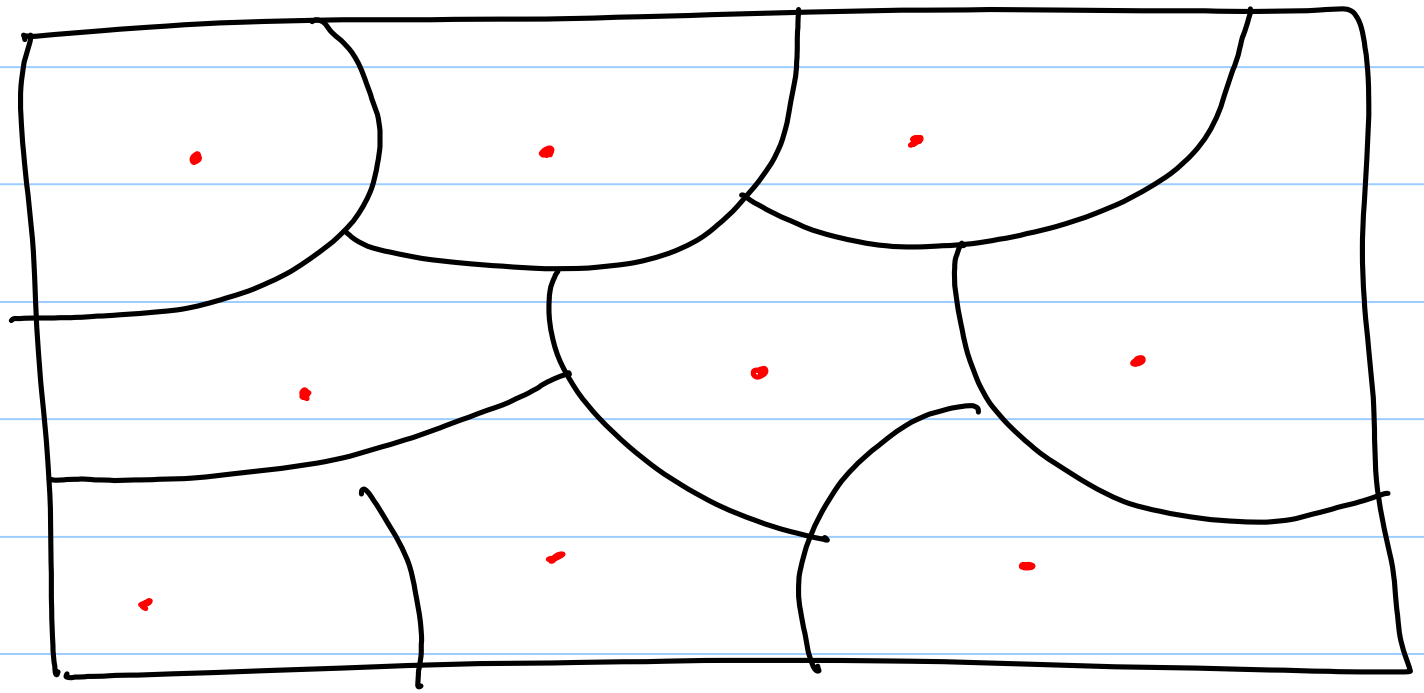
Eg 2 The general Hamming code is always perfect!

Pf. parameters:

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

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

$$= \frac{2^n}{2^{n-r}} \quad \therefore \text{perfect.}$$



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

Golay's observation:

If a linear code is perfect.

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

$$\Leftrightarrow \boxed{\sum_{i=0}^{t} \binom{n}{i} = 2^{n-k}} \quad \cdots\cdots (1)$$

Eg 3 · The Golay code

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

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

$$= 1 + 23 + \frac{23 \cdot \overset{11}{\cancel{22}}}{2} + \frac{23 \cdot \overset{11}{\cancel{22}} \cdot \cancel{21}7}{1 \cdot 2 \cdot \cancel{3}}$$

$$\begin{array}{r} 23 \\ 77 \\ \hline 161 \\ 161 \end{array}$$

$$= 1 + 23 + 253 + (23)(77)^{\frac{1771}{3}}$$

$$= \begin{array}{r} 1771 \\ 253 \\ 23 \\ 1 \\ \hline 2048 \end{array} = 2^{11} = 2!!$$

This numerical calculation led Golay 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 GRC construction).

As long as $\delta \leq$

$$M \left[\sum_{i=0}^{\delta-1} \binom{n}{i} \right]$$

n

2

$\vdots$

2

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

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

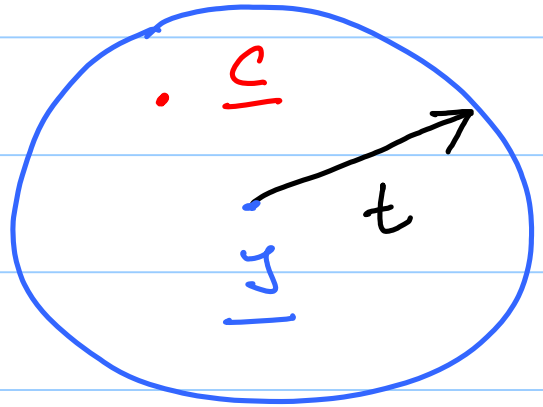
features:

(i) long codes are employed

(ii) bounded-distance decoding (BDD).

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

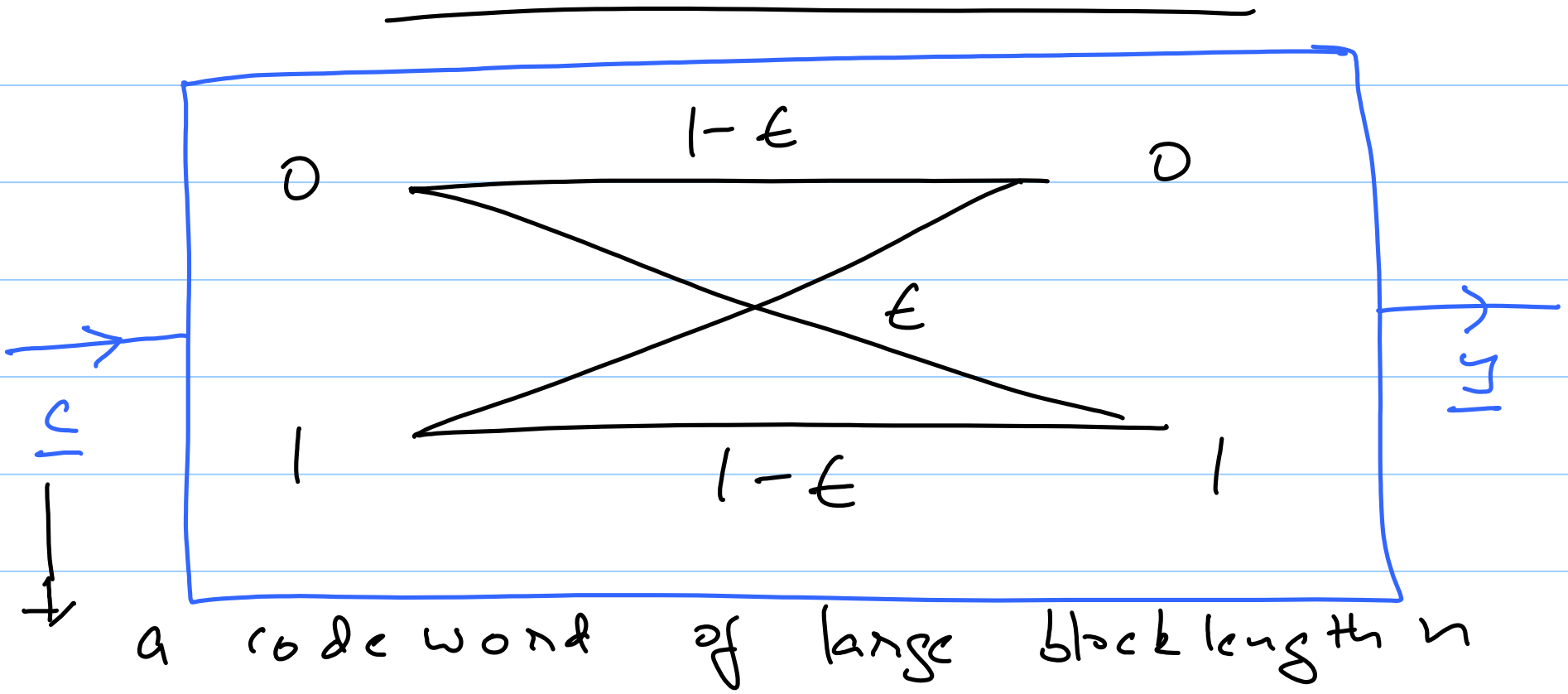
$$t = \left\lfloor \frac{d-1}{2} \right\rfloor$$



$\underline{y}$ examines the ball $B(\underline{y}, t)$ and declares $\hat{\underline{c}}$ (the decoded codeword) to equal $\underline{c}$ if $\underline{c}$ is the only codeword in the ball. (Else, gives up).

$$2^{F} \quad S$$

note: by reliable communication we mean 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)

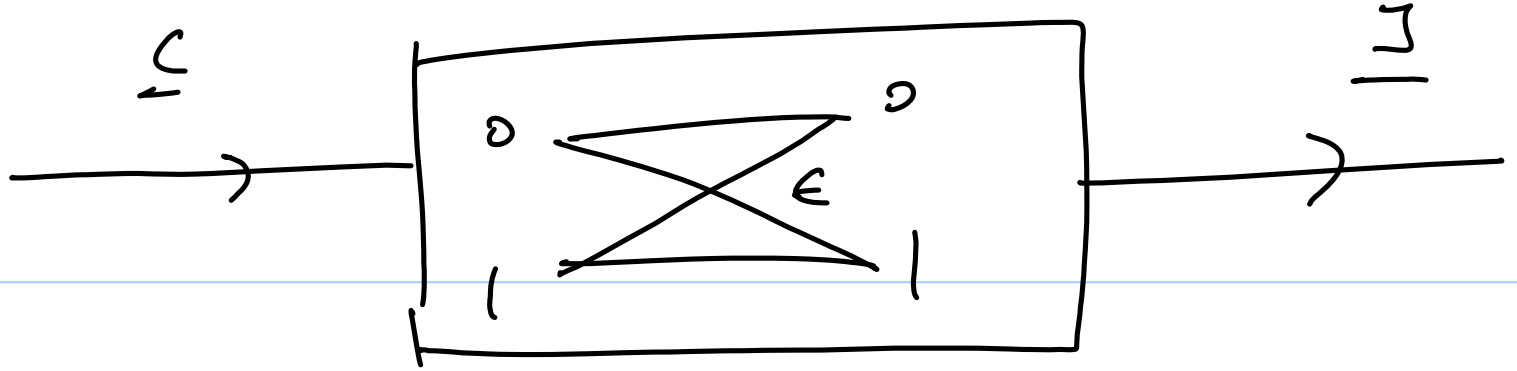
when n is large this distribution tends to become Gaussian with parameters:

mean = $n\epsilon$, 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.



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

$\longrightarrow$ Gaussian distribution

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

variance.

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

$\frac{1}{\sqrt{\pi}} \frac{\eta}{(\omega-\omega_0)^2+\eta^2}$
ω_0

ω_0
η

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

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

$$d = 2n\epsilon$$

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:

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

(rate)

δ = fractional minimum distance

Qn: How does $R(\delta)$ vary with δ?

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

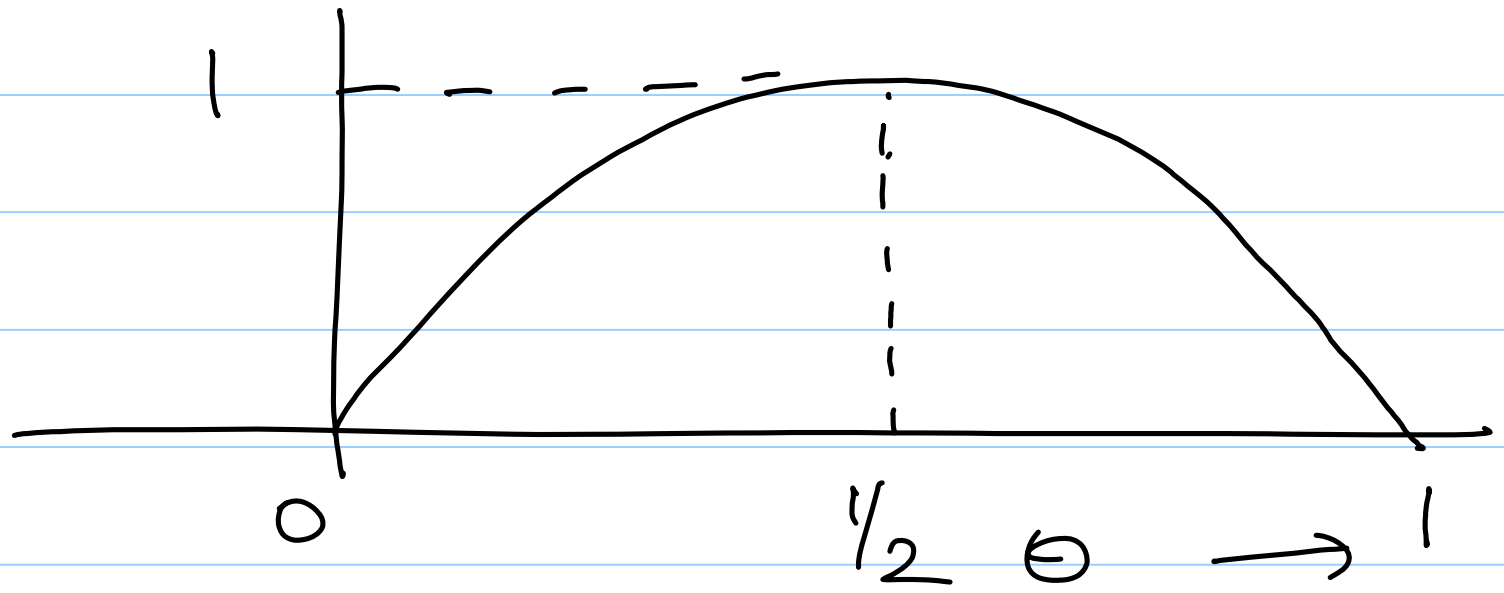
$$\boxed{1 - H_2(\delta) \leq R(\delta) \leq 1 - H_2\left(\frac{\delta}{2}\right)}$$

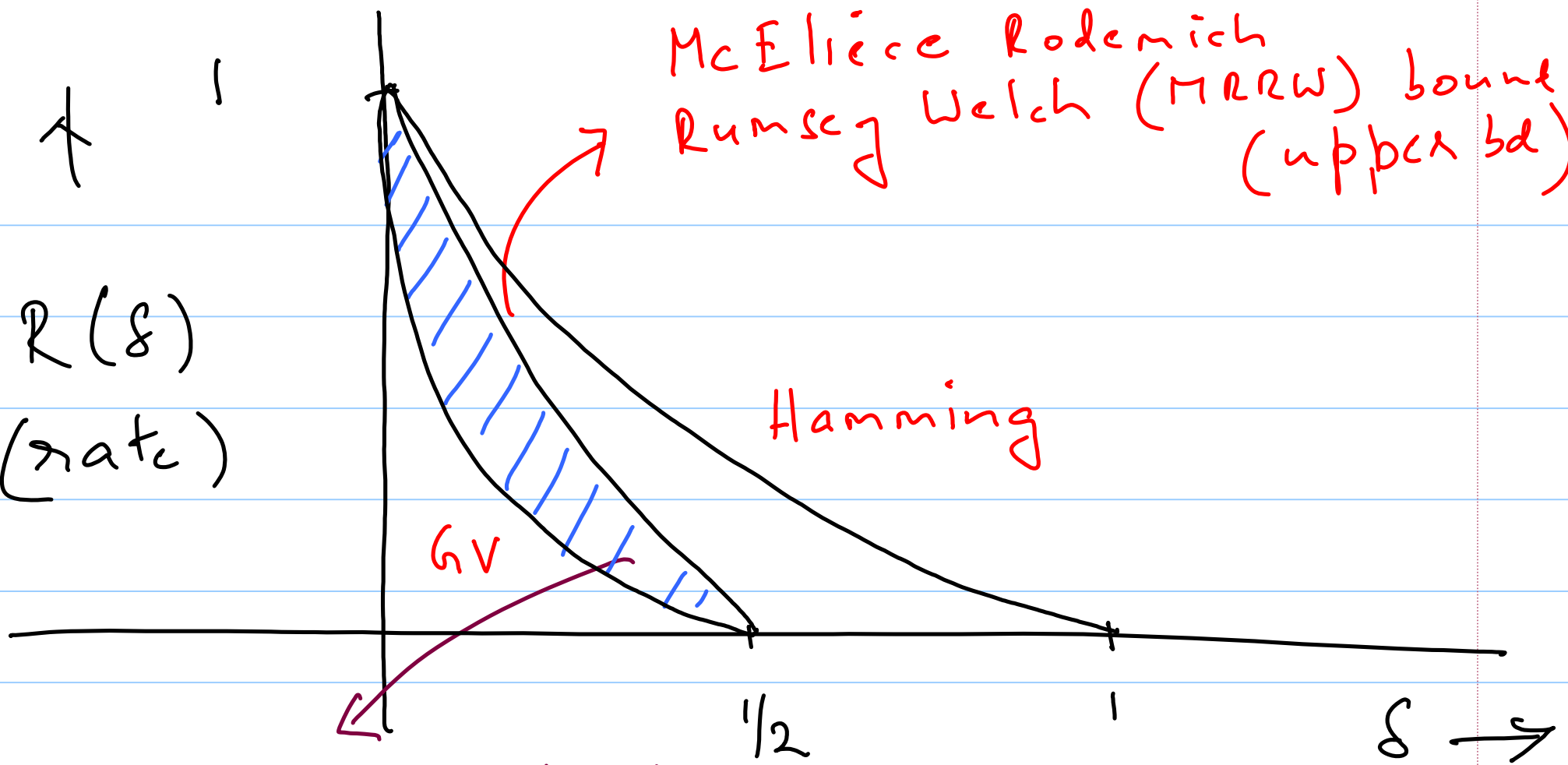
GV Hamming

For $0 \le \theta \le 1$,

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

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

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

)

$$1 - H_2(2\epsilon) \underset{\text{GV}}{\leq} R(\delta) \underset{\text{Hamming}}{\leq} 1 - H_2(\epsilon) \quad \cdots\cdots (1)$$

On the other hand Shannon tells us that

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

Conclusion: Our combination of long block length and BP, causes us to require the use of long block codes that achieve the Hamming bound to achieve channel capacity.

Minimum Probability of Error Decoder

$\Sigma = \begin{cases} \text{event that a codeword is erroneously} \\ \text{decoded} \end{cases}$

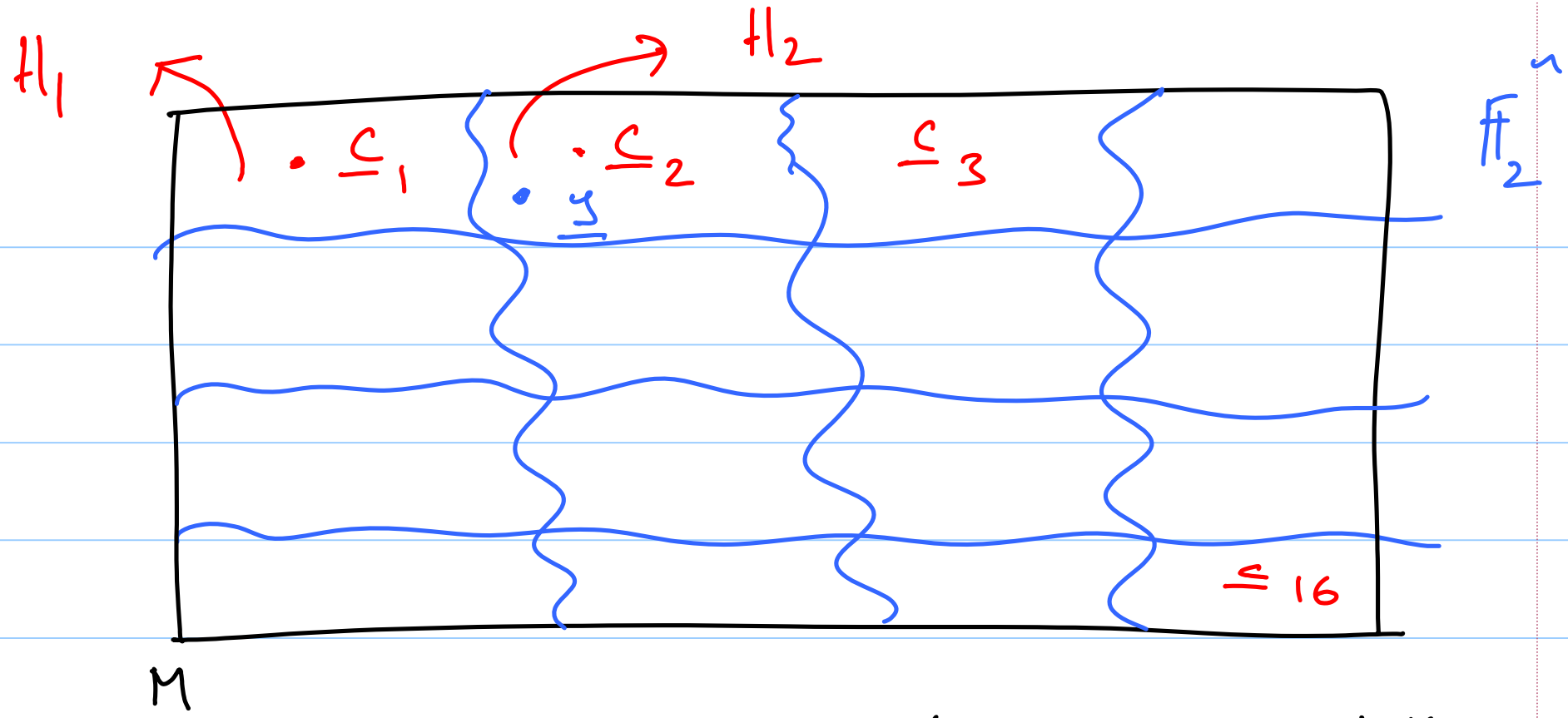
Σ^c = event that is correctly decoded.

$M = |\mathcal{C}|$

$$P_r(\Sigma^c) = \sum_{i=1}^{M} P_r(\underline{c}_i) P_r(\Sigma^c | \underline{c}_i)$$

$$= \sum_{i=1}^{M} P_r(\underline{c}_i) P_r(\underline{y} \in \mathcal{H}_i | \underline{c}_i)$$

$\underline{y}$: received vector



$$= \sum_{i=1} P_1(c_i) \sum_{y \in F_2} P_2(y \mid c_i) \underbrace{I_{H_i}(y)}$$

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

$$= \sum_{\underline{y} \in \mathbb{F}_2^n} \sum_{i=1}^{M} P_r(c_i)\, P_r(\underline{y} \mid c_i)\, I_{H_i}(\underline{y})$$

to maximize the probability of correct decisions, the decoder assigns

$\underline{y}$ to H_i $\Leftrightarrow$ $P_r(c_i)\, P_r(\underline{y} \mid c_i)$

$$\geq P_r(c_j)\, P_r(\underline{y} \mid c_j)$$

$$j \neq i$$

$$1 \leq i, j \leq M$$

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

in which case the probability of correct decision is maximized by selecting $\underline{y} \in H_i$ iff

$$P_r(\underline{y} \mid \underline{c}_i) \geq P_r(\underline{y} \mid \underline{c}_j)$$

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 $d_H(\underline{y}, \underline{c}_i) = d$

$$p_{\underline{Y}}(\underline{y} \mid \underline{c}_i) = \epsilon^{d}(1-\epsilon)^{n-d}$$

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

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

$\Rightarrow$ $P_R(\underline{r} \mid \underline{c}_i)$ is maximized

by minimizing

$$d_H(\underline{r}, \underline{c}_i)$$

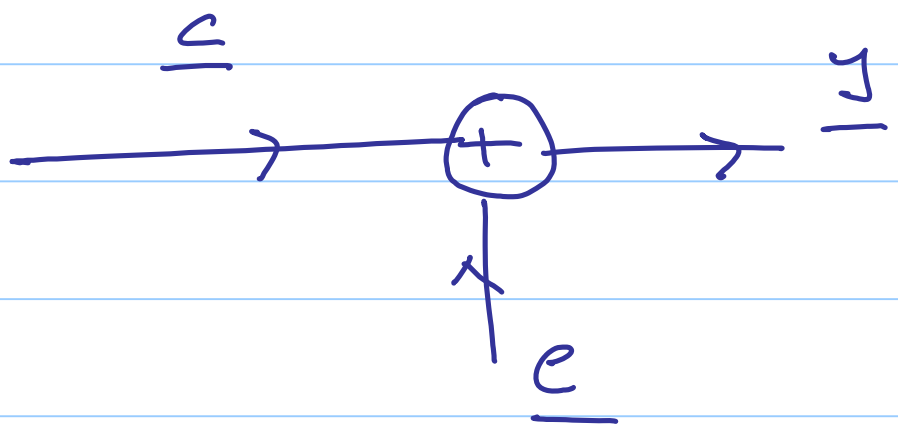
Lec 14 Standard Array Decoding

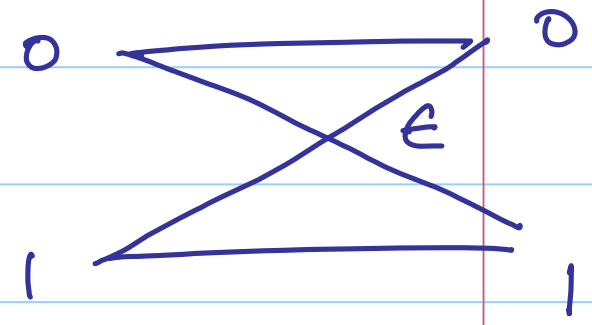
Recap

- asymptotic bounds
- min prob of error decoder
  - maximum likelihood decoder → MLD
  - minimum Hamming distance decoder → MDD

MDD

Channel Model:



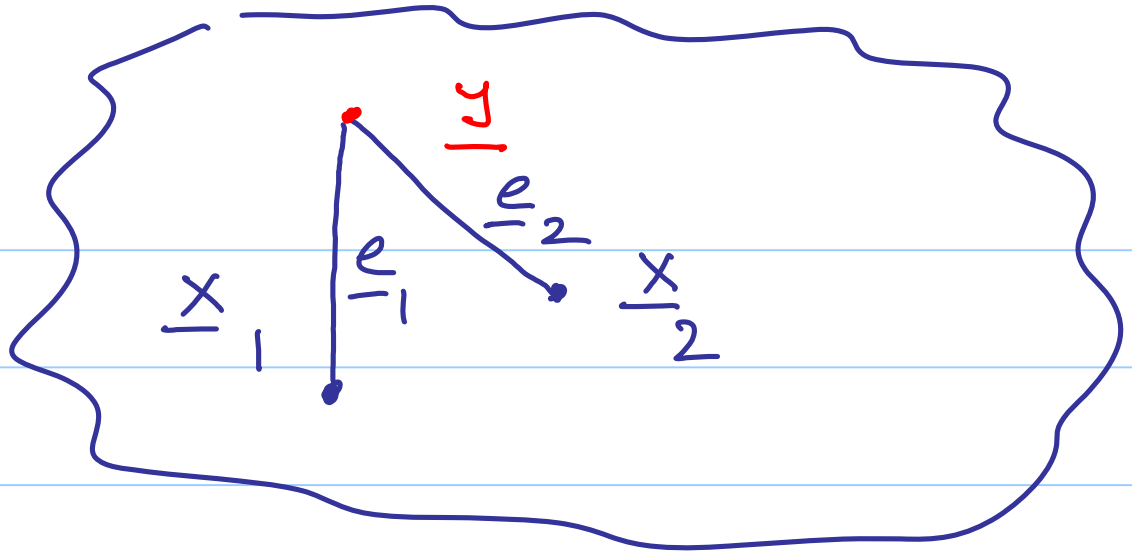


The MDD chooses $\hat{c}$ (the decoded codeword) such that (s.t.)

$d_H(y, \hat{c})$ is a minimum

with $\hat{c} \in$ CODE

$\Rightarrow$



$\mathbb{F}_2^n$

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

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

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

Step 1 form the set

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

Step 2 Let $\hat{\underline{e}}$ be the element in $\underline{y} + \mathcal{C}$ having least Hamming weight.

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

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

Note that (i) $\underline{y} + \mathcal{C}$ is a coset of the subgroup $\mathcal{C}$ of $\mathbb{F}_2^n$.

(ii) and hence the decoder action is only a fn of the coset of $\mathcal{C}$ to which $\underline{y}$ belongs and not to $\underline{y}$ itself.

Eg

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

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

$$G = \left[\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{array}\right] \qquad H = \left[\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{array}\right]$$

I_2 $\quad$ P $\quad$ H_2

3 other cosets of C

coset leaders ⇑

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

$n - k = 2$

S

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

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

MLD algorithm ⇒ $\underline{y} + \hat{\underline{e}}$

$= 0111 + 0010 = 0101$ ✓

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

$$\underline{s} = \underset{(n-k \times n)}{H}\,\underline{y}$$

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

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

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

Pf. $\phi : \underline{y} + \mathcal{C} \to \underline{y}\underline{H}^T = \underline{s}$

Is ϕ well-defined?

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

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

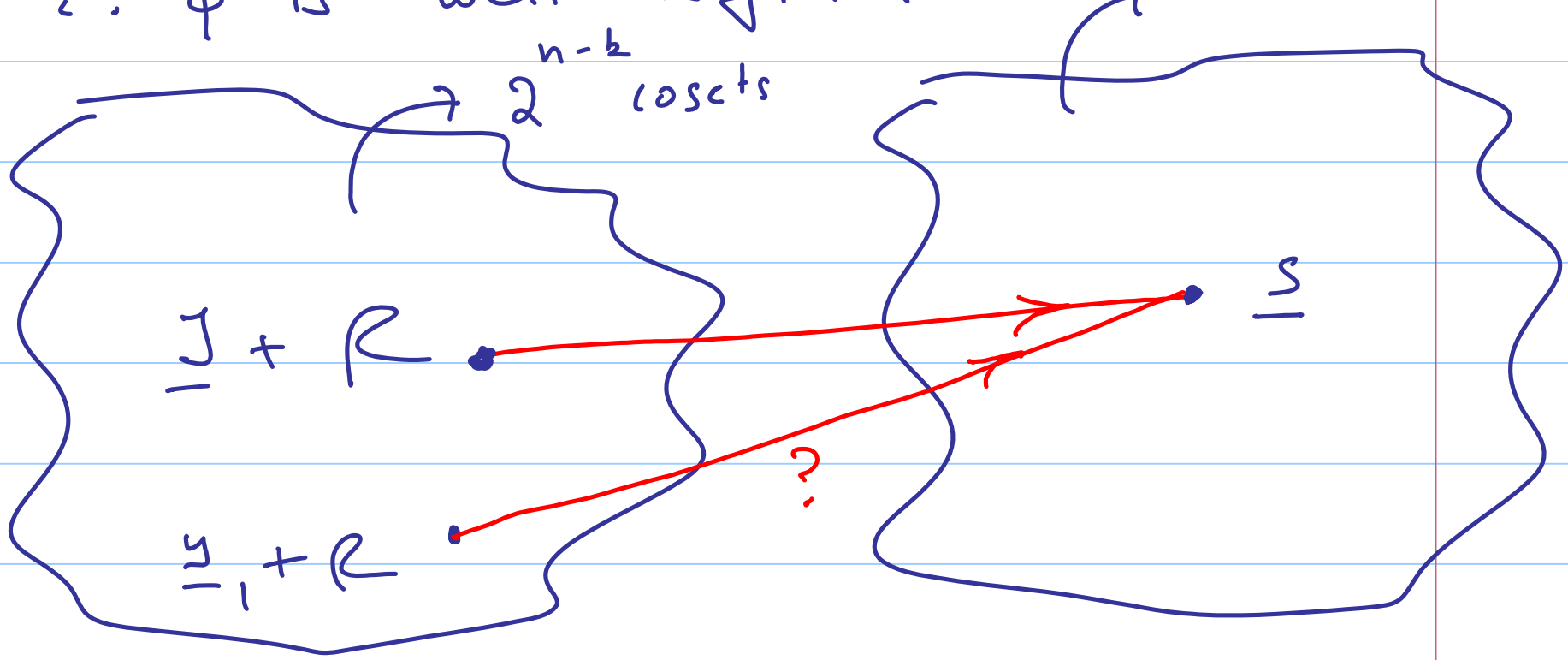
$$\therefore H\underline{y}' = H\underline{y} + H\underline{e}$$

$$\underline{e} \in \underline{0}$$

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

$\therefore \phi$ is well-defined.

size $= 2^{n-k}$



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

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

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

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

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

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

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

Cost: 111

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

Step 1 Compute syndrome

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

Step 2 Use table lookup to determine $\hat{e}$ (the coset leader associated to syndrome $\underline{s}$)

Step 3 decode to :

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

///

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

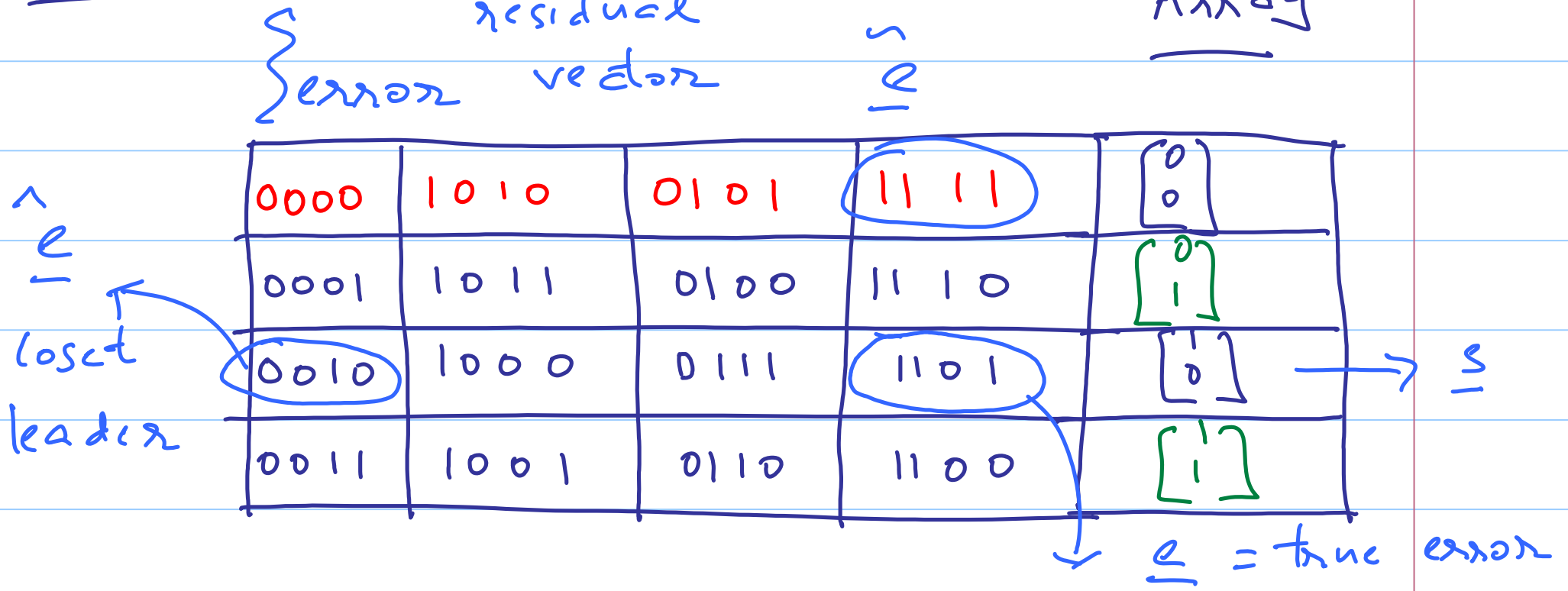
associated to the code.

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 Array



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 $\underline{y} = \underline{c} + \underline{e}$, $\underline{c} \in R$

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

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

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

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

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

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

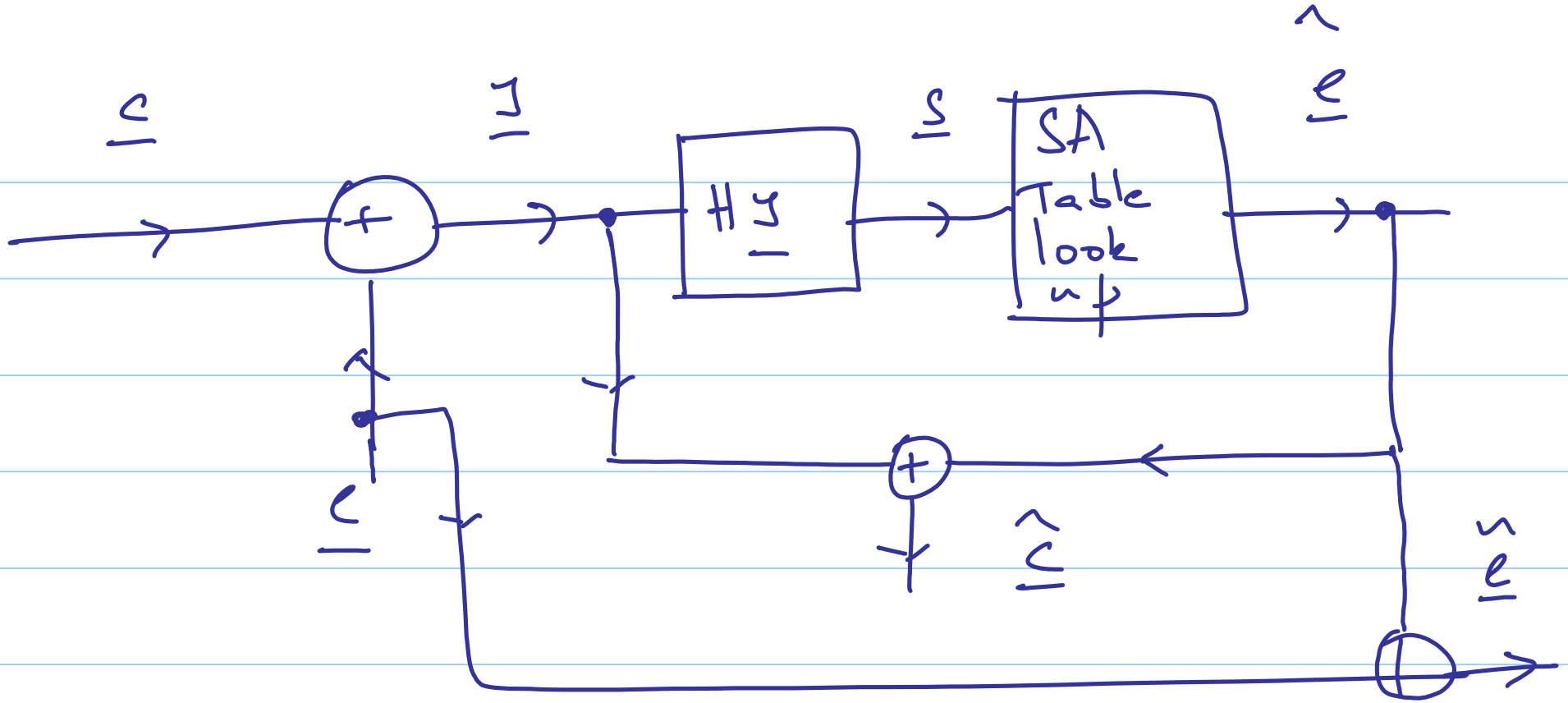
Decoder computes:

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

$\Downarrow$

residual error pattern

It follows then that in the standard array, the residual error vector $\hat{\underline{e}}$ is the vector in the table at the head $\downarrow$ 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 coset 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:

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

$\underline{m}^t$

00	10	01	11
$\underbrace{0000}$	$\underbrace{1010}$	$\underbrace{0101}$	$\underbrace{1111}$
0001	1011	0100	1110
0010	1000	0111	1101
0011	1001	0110	1100

$\underline{C}$

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

residual message - error patterns!

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

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

$1010 \rightarrow m_1$ m_2

$\Rightarrow m_1$ is decoded erroneously

m_2 is decoded correctly

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

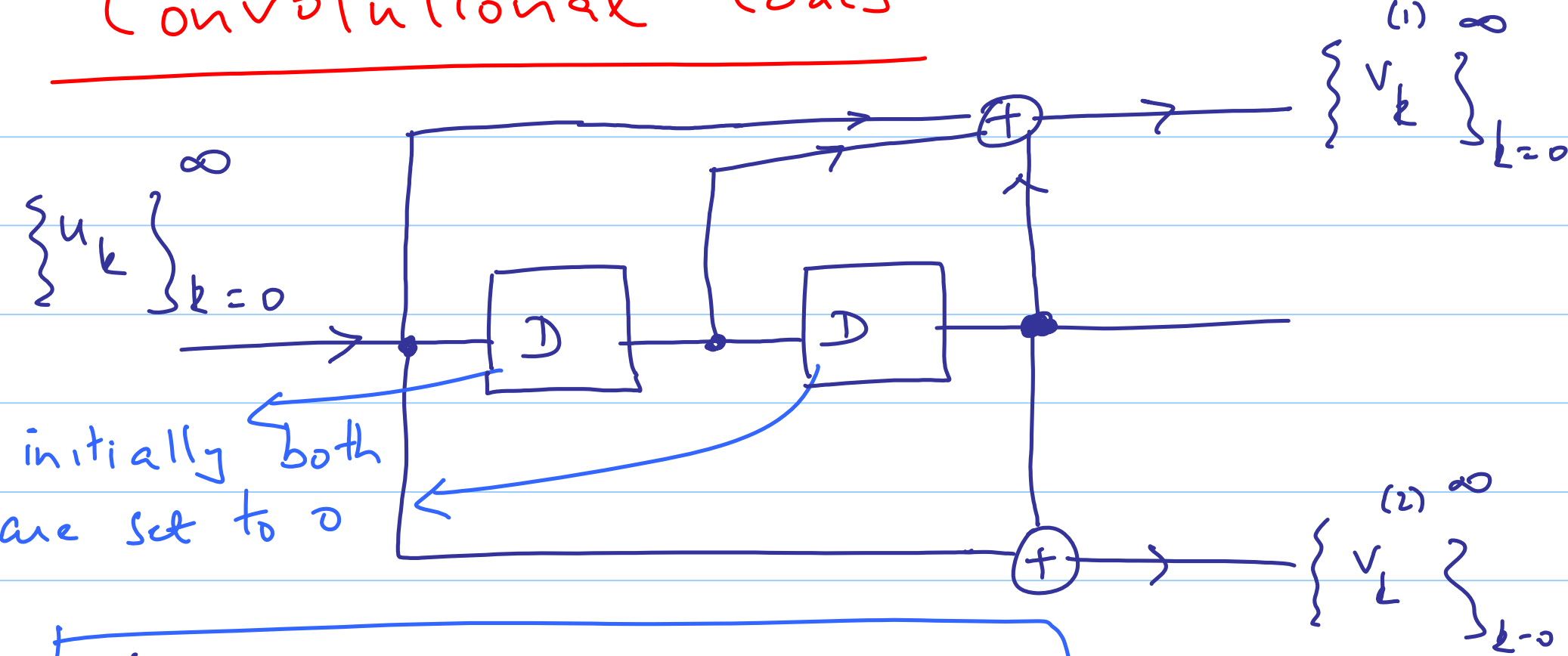
0000	1010	0101	1111
0001	1011	0100	1110
0010	1000	0111	1101
0011	1001	0110	1100

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 the analysis to be carried out.

Convolutional Codes

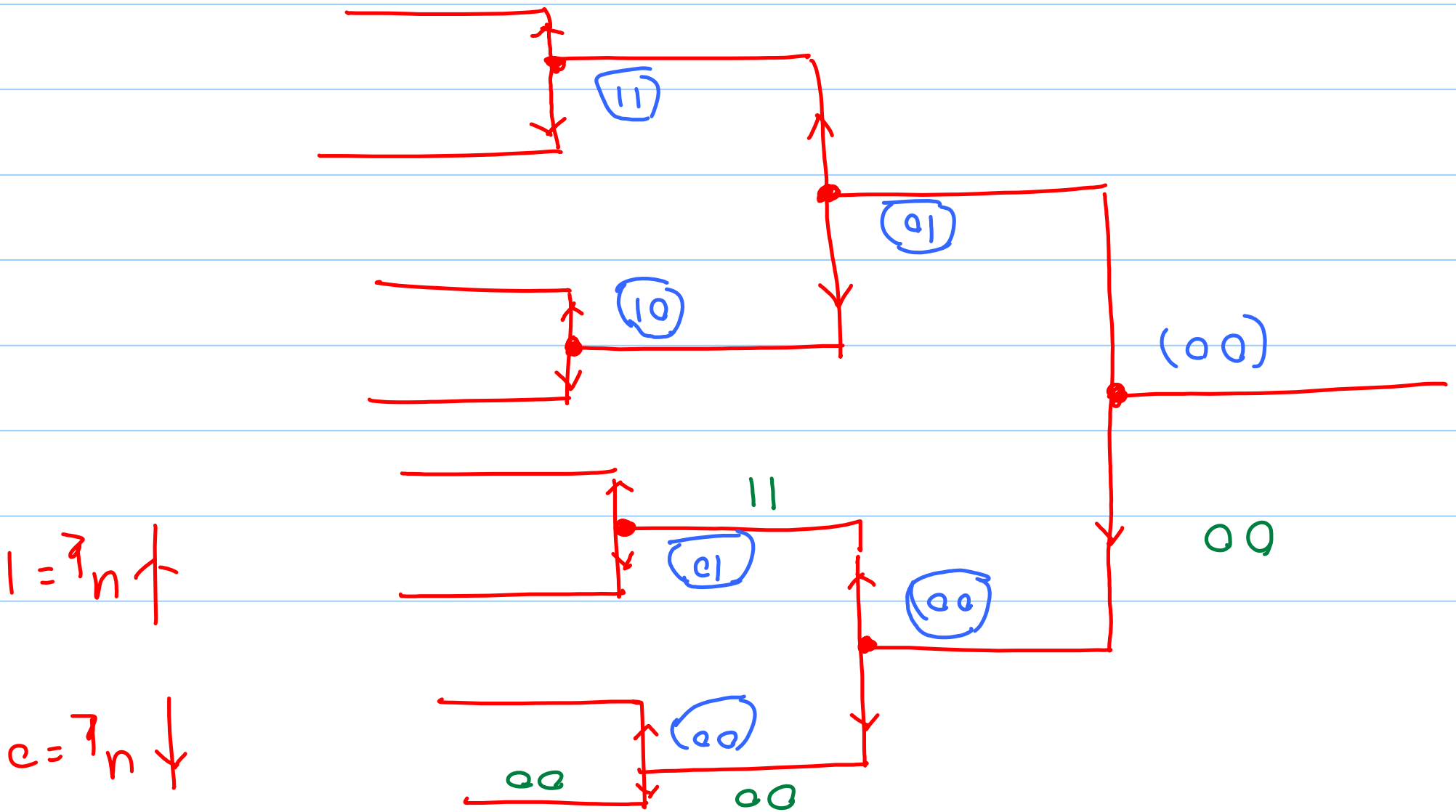


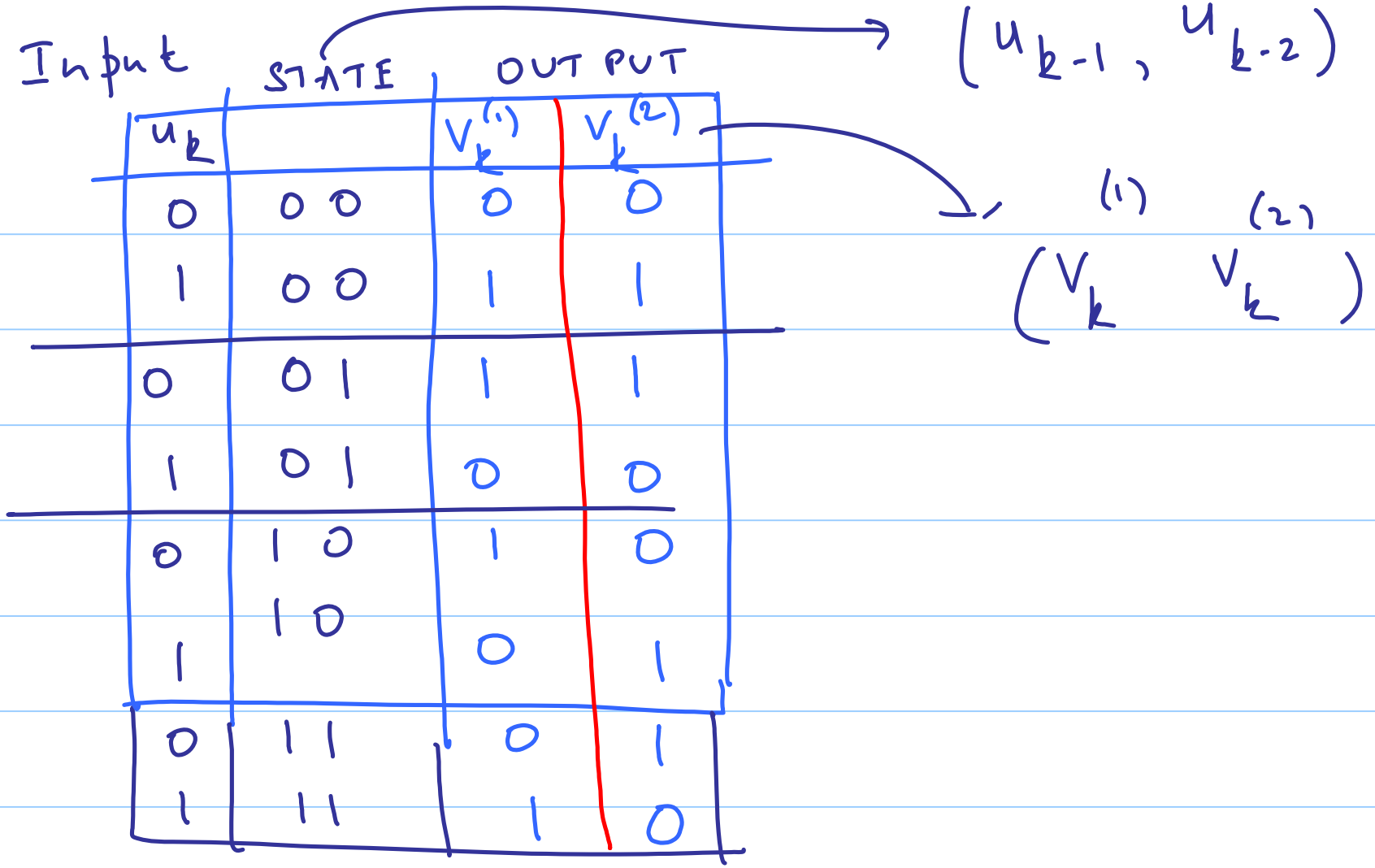
(1)

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

(2)

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


11
01
10
00
11
01
00
00
00
00
1 = ↑n
0 = ↓n


Input
STATE
OUT PUT
u_k
$V_k^{(1)}$
$V_k^{(2)}$
(u_{k-1}, u_{k-2})
$(V_k^{(1)} \quad V_k^{(2)})$
0 0 0 0
1 0 0 1 1
0 0 1 1 1
1 0 1 0 0
0 1 0 1 0
1 1 0 0 1
0 1 1 0 1
1 1 1 1 0

Convolutional codes belong to the class of tree codes that are:

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

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

$$\begin{bmatrix} \cdots \end{bmatrix}$$

{ Semi - infinite generator mx.

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

$$\mathbb{F}((x)) = \left\{ \sum_{k=-d}^{\infty} a_k x^k \;\middle|\; a_k \in \mathbb{F},\ d \geq 0 \right\}$$

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

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

Lec 16 State and Trellis

Recap

- completed performance analysis of the SAD
- convolutional codes
  - encoder
  - {semi- infinite generator matrix

— formal power series

$$\mathbb{F}((D)) = \left\{ \sum_{k=-d}^{\infty} a_k D^k \;\middle|\; a_k \in \mathbb{F} \right\}$$

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

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

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

$$+ \quad \left(D^4 + D^5 + D^6 + D^7 + D^8 \right)$$

$$= \quad 1 + D + D^3 + D^4 + D^5 + D^6 + D^8 + D^{19} + \cdots$$

$(\mathbb{F}((D)), \cdot) \Rightarrow$ closure

associative

$\lambda \cdot c = 1$

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

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

Pf.

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

//

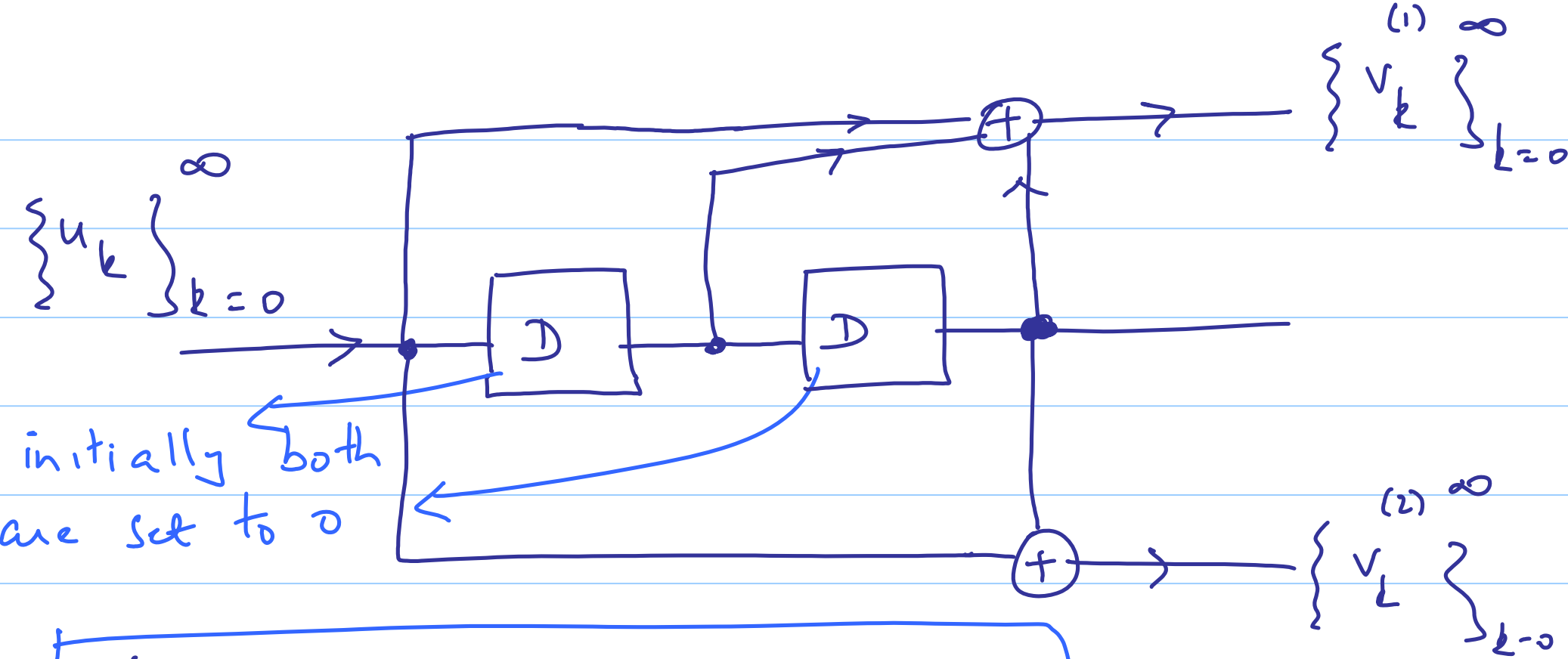
$$\therefore \frac{D^{-1}}{1+D^2} = D^{-1}\left(1 + D^2 + D^4 + D^6 + \cdots\right)$$

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

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)

$$v_k = u_k + u_{k-1} + u_{k-2} \quad \ldots\ldots (1)$$

(2)

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

Define:

$$U(D) \triangleq \sum_{k=0}^{\infty} u_k D^k$$

} input power series

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

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

---- (2)

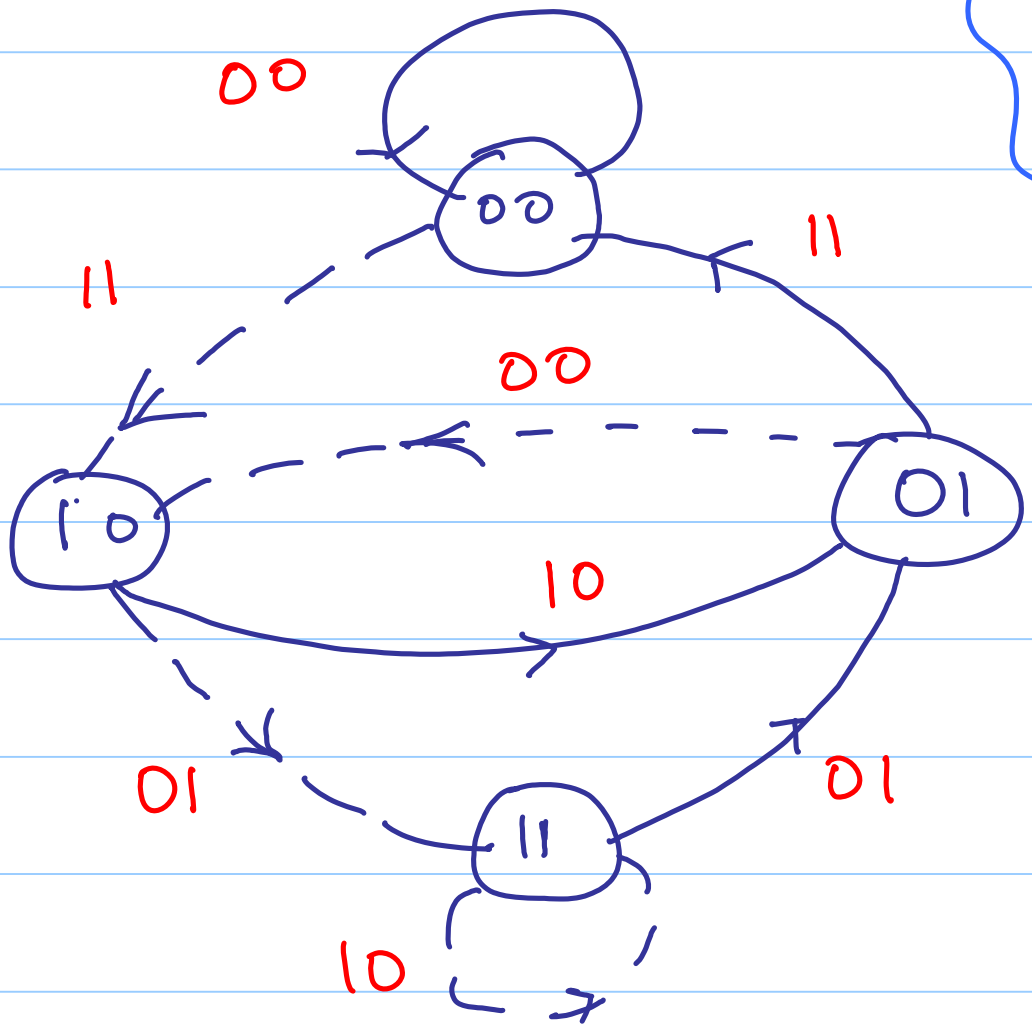
Converting the time-domain input-output relation given in (1) to the D-transform

domain, we get:

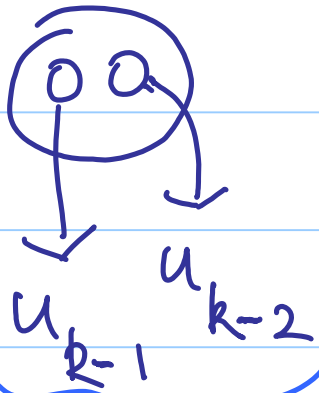
$$\begin{bmatrix} V^{(1)}(D) & V^{(2)}(D) \end{bmatrix}$$

$$= U(D) \underbrace{\begin{bmatrix} 1+D+D^2 & 1+D^2 \end{bmatrix}}_{\substack{G(D) \\ (1\times 2)}}$$

Finite - State Machine Description



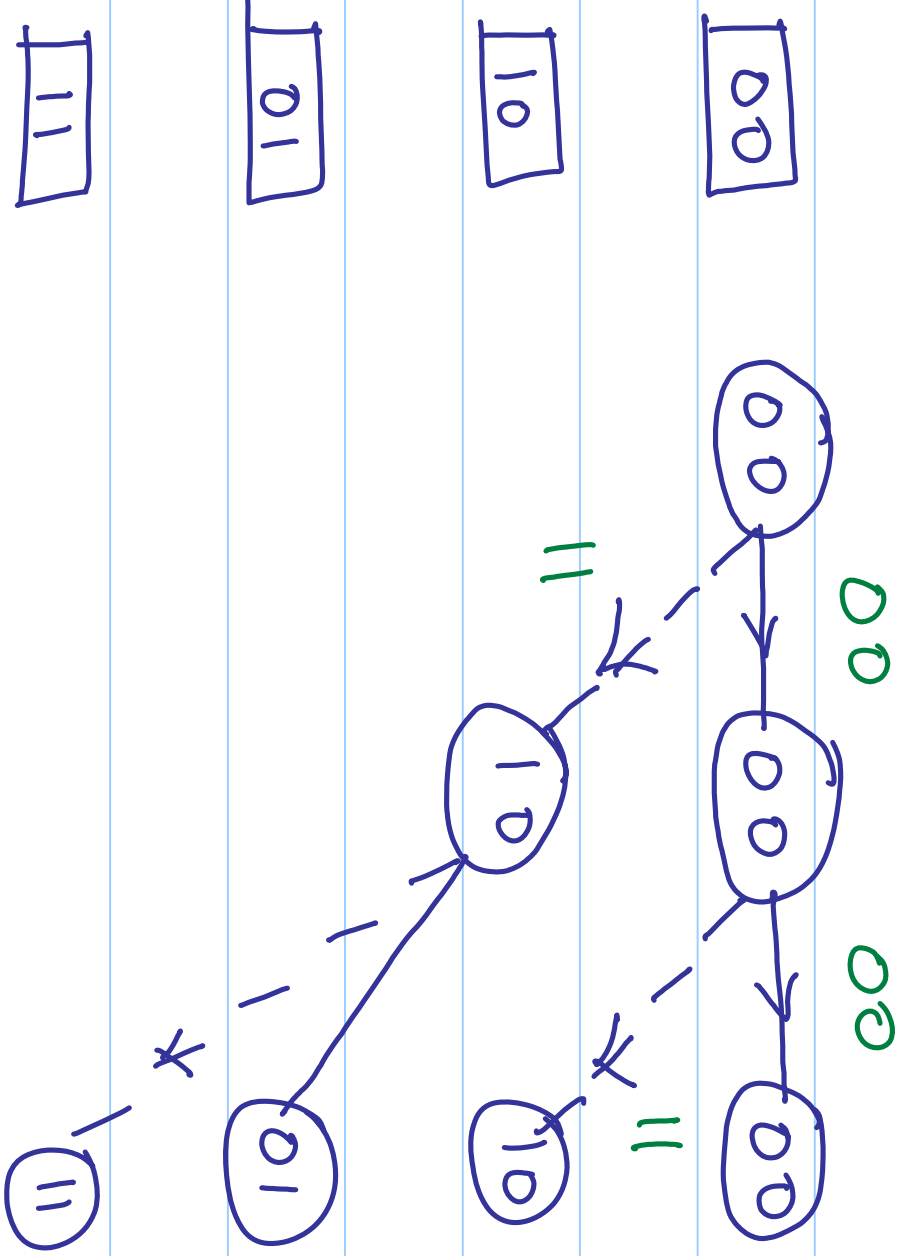
past 2 symbols

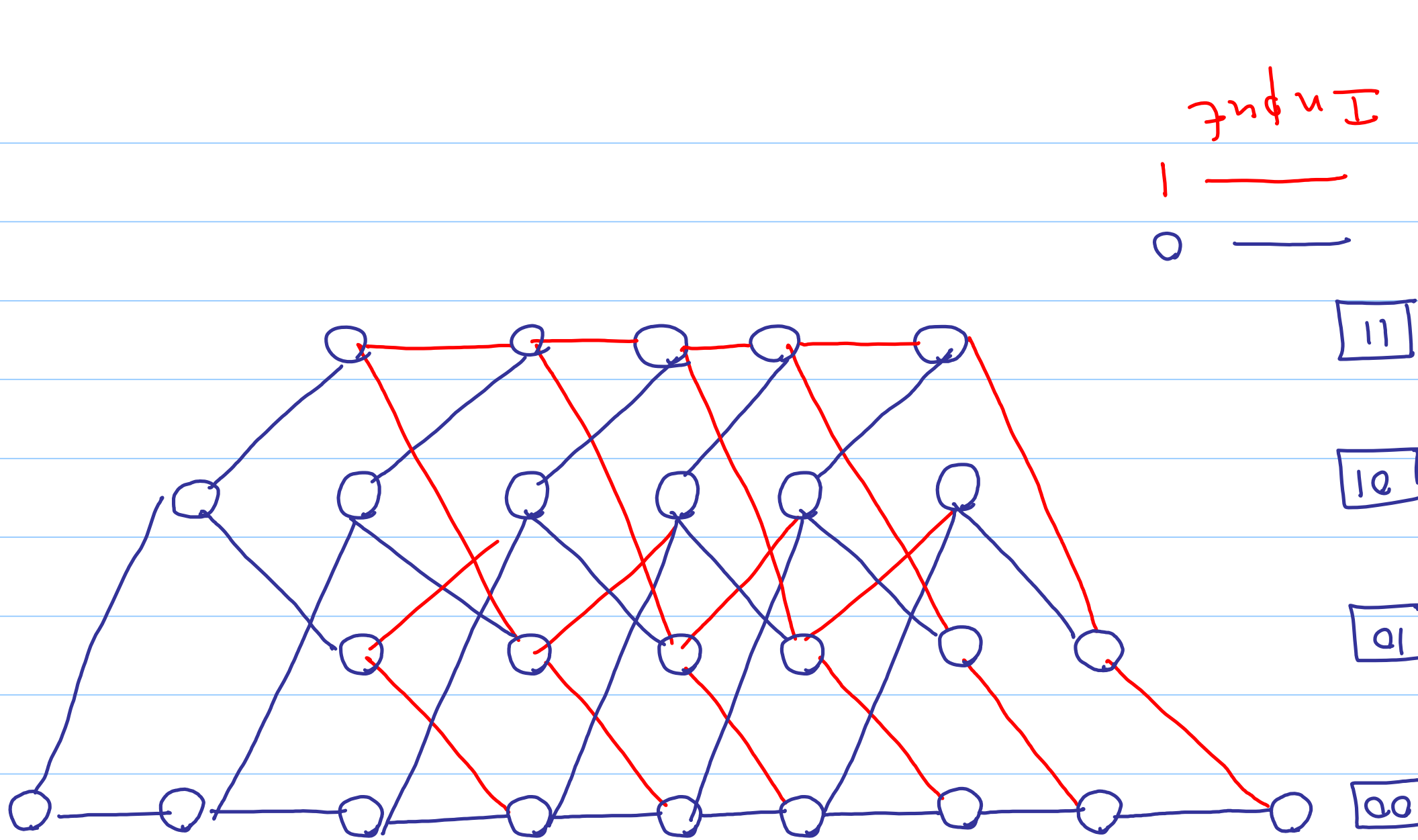


input = 0

input = 1

In		State		Output	
				$V_{OUTPUT}^{(1)}$	$V_{OUTPUT}^{(2)}$
1	1	0	0	0	0
0	1	0	0	1	1
1	0	0	1	1	0
0	1	0	1	0	1
1	0	1	0	0	1
0	1	1	0	1	0
1	0	1	1	0	1
0	1	1	1	1	0


=
10
1
00
=
00
*
1
*
00
=
10
1
=
00


Input
1
0
11
10
01
00

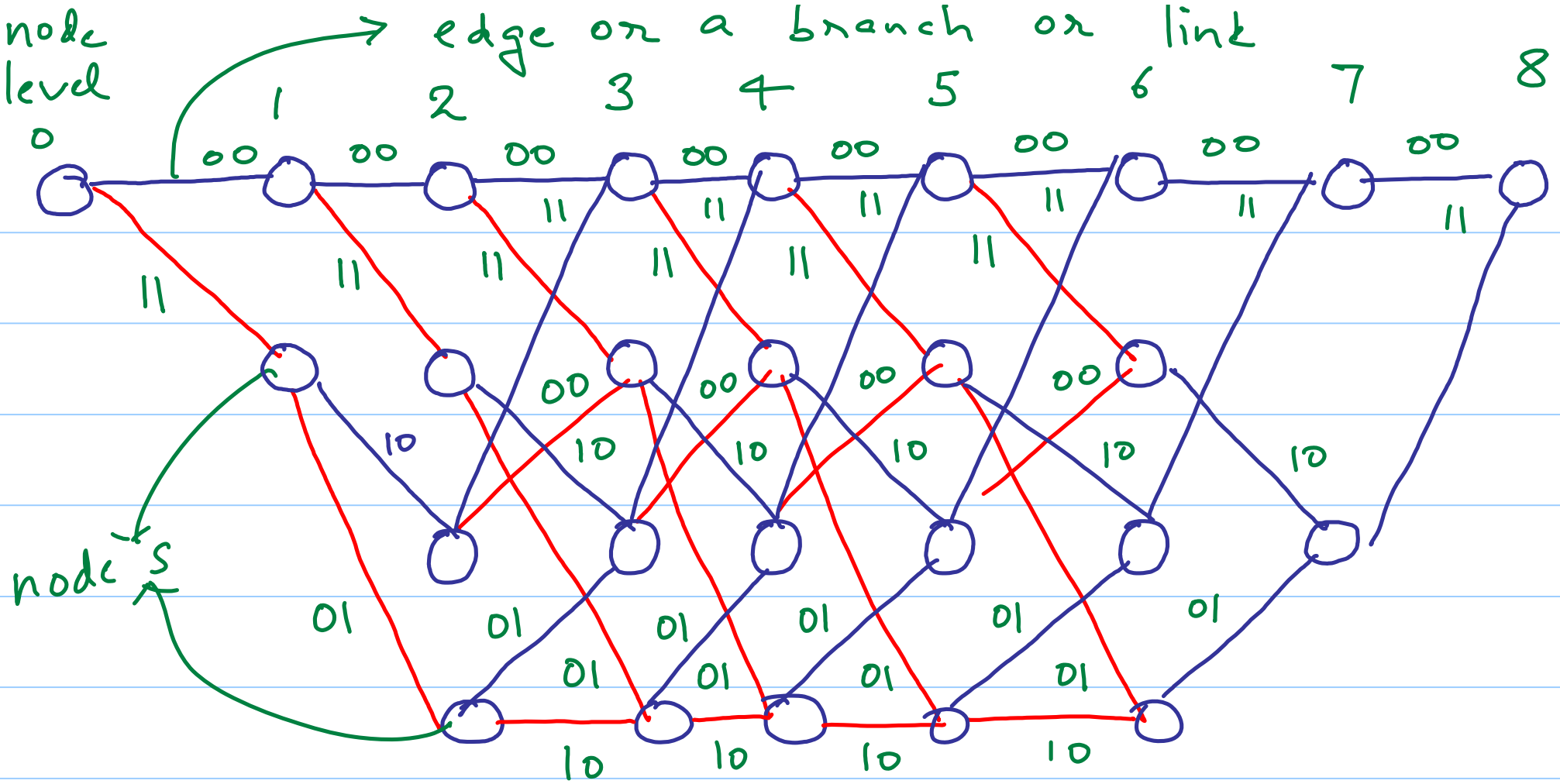
Input
1
0
11
10
01
00
START
t = 4

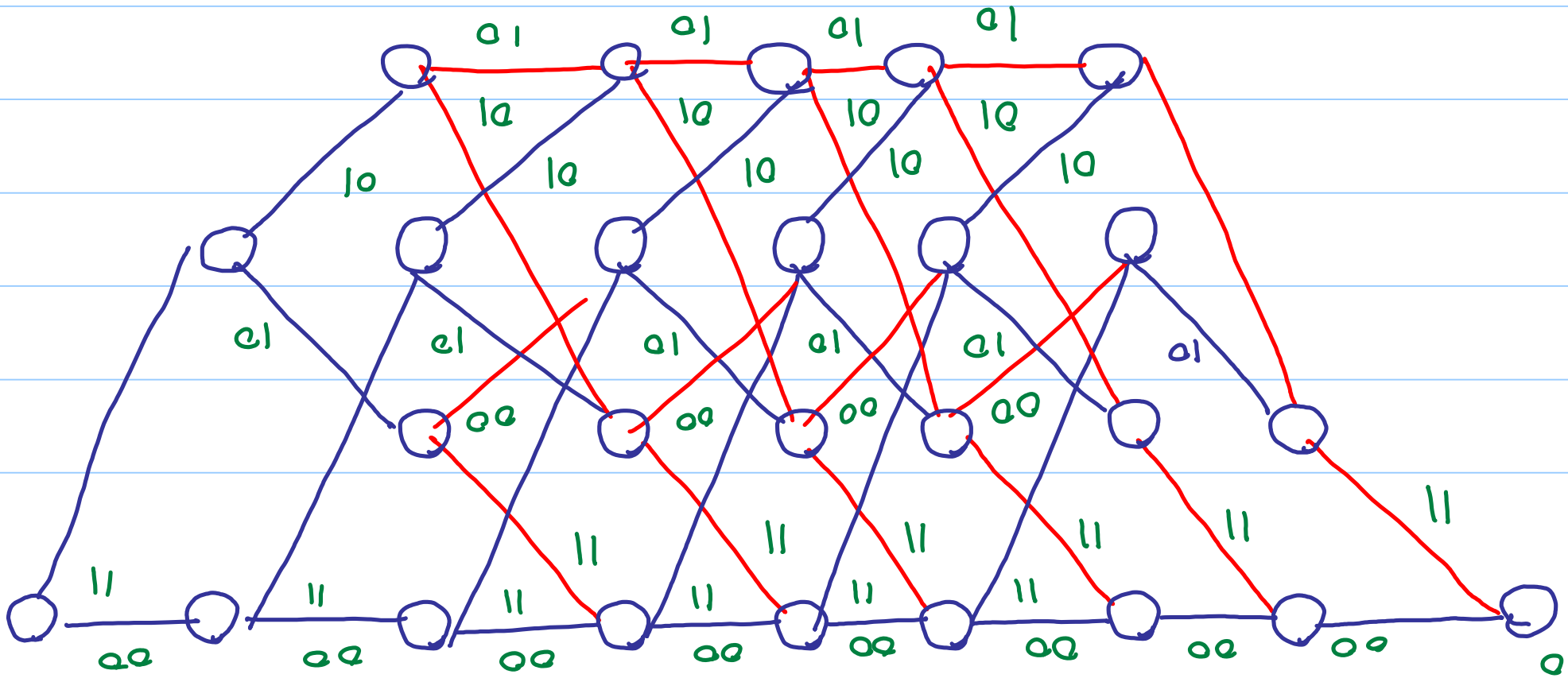
Lec 17: The Viterbi Decoder

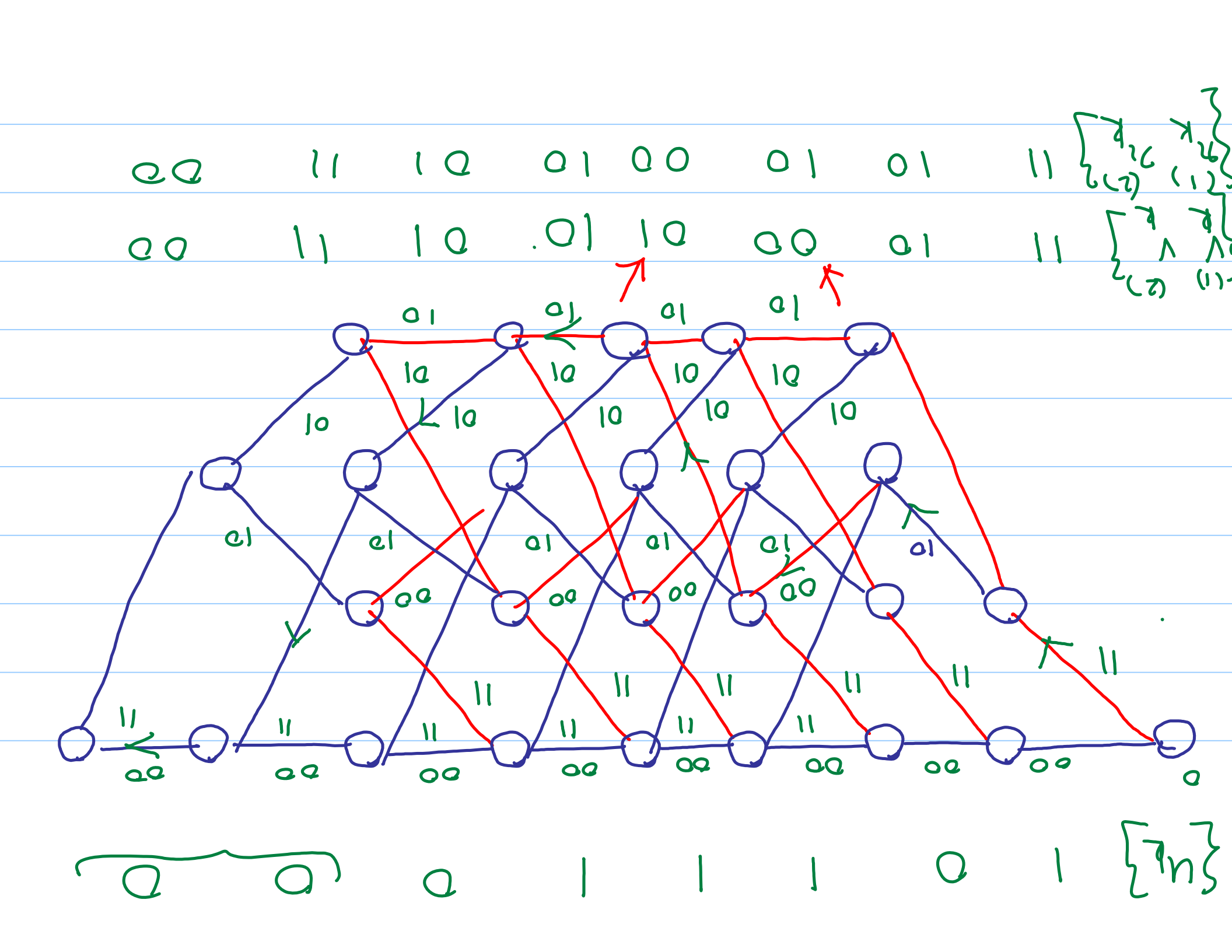
Recap:

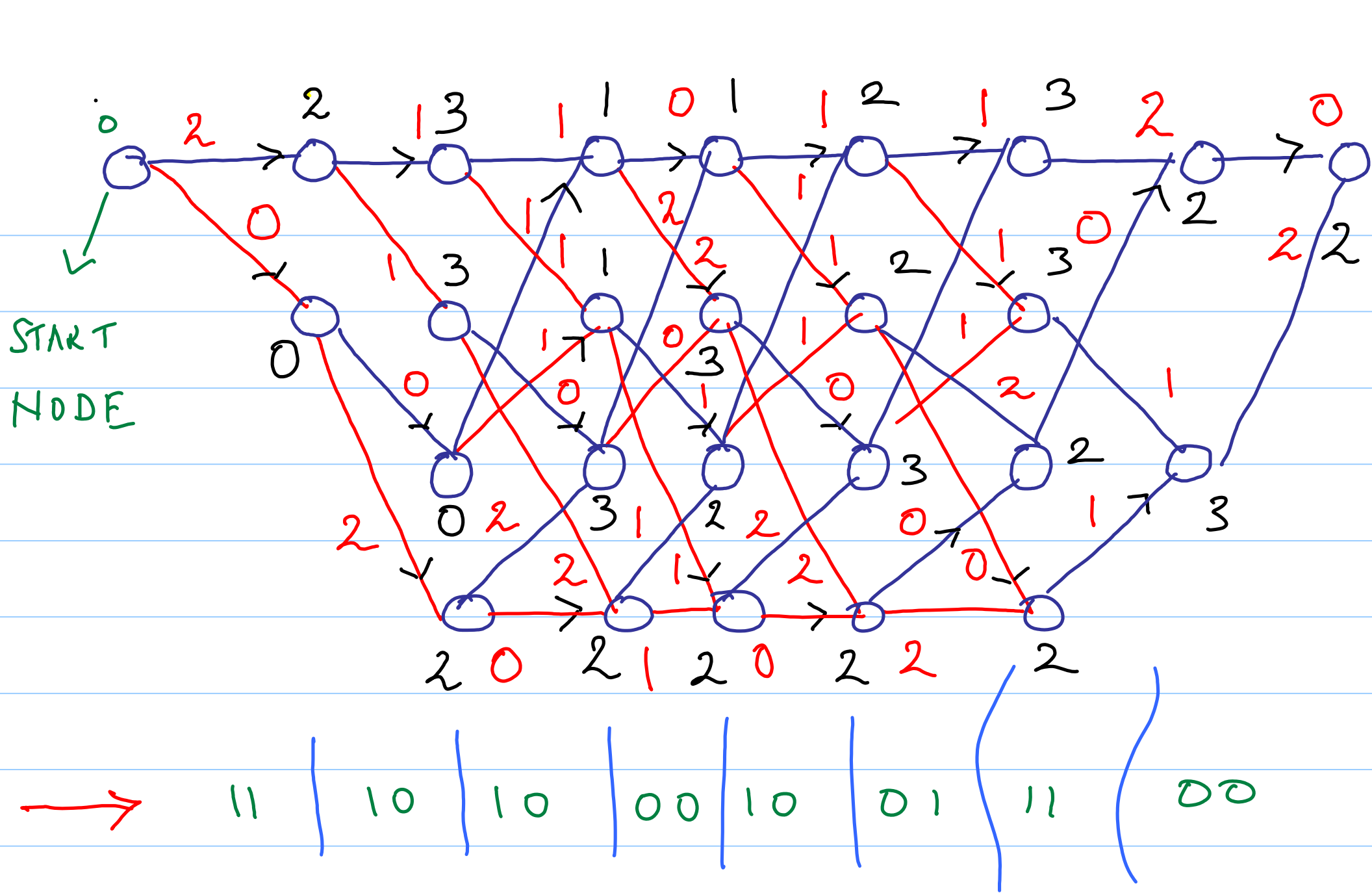
* completed our discussion on formal power series and polynomials leading to the polynomial generator matrix of the convln. Code
* state diagram of the encoder viewed as a FSM
* trellis diagram

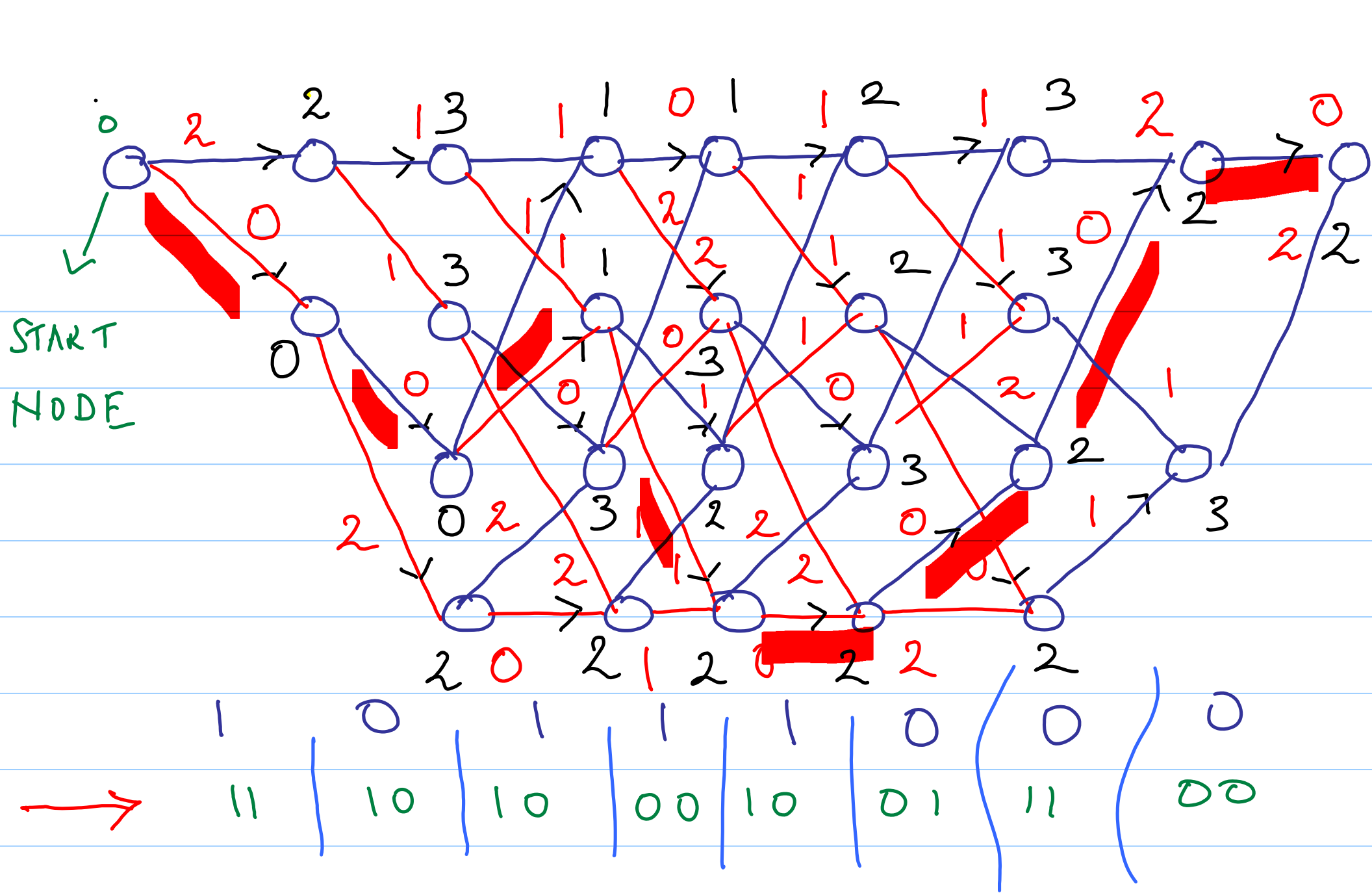
Input
1
0
11
10
01
00
START
t = 4


node level
edge or a branch or link
0
1
2
3
4
5
6
7
8
node S
00
11
10
01


01
01
01
01
10
10
10
10
10
10
10
10
01
01
01
01
01
01
00
00
00
00
00
11
11
11
11
11
11
11
11
11
00
00
00
00
00
00
00
0




START
NODE
11 | 10 | 10 | 00 | 10 | 01 | 11 | 00


START
NODE
1 0 1 1 1 0 0 0
11 10 10 00 10 01 11 00

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

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

$$d_{11}\left(\left\{ v_k^{(1)} \; v_k^{(2)} \right\}_{k=0}^{M-1}, \left\{ r_k^{(1)} \; r_k^{(2)} \right\}_{k=0}^{M-1} \right)$$

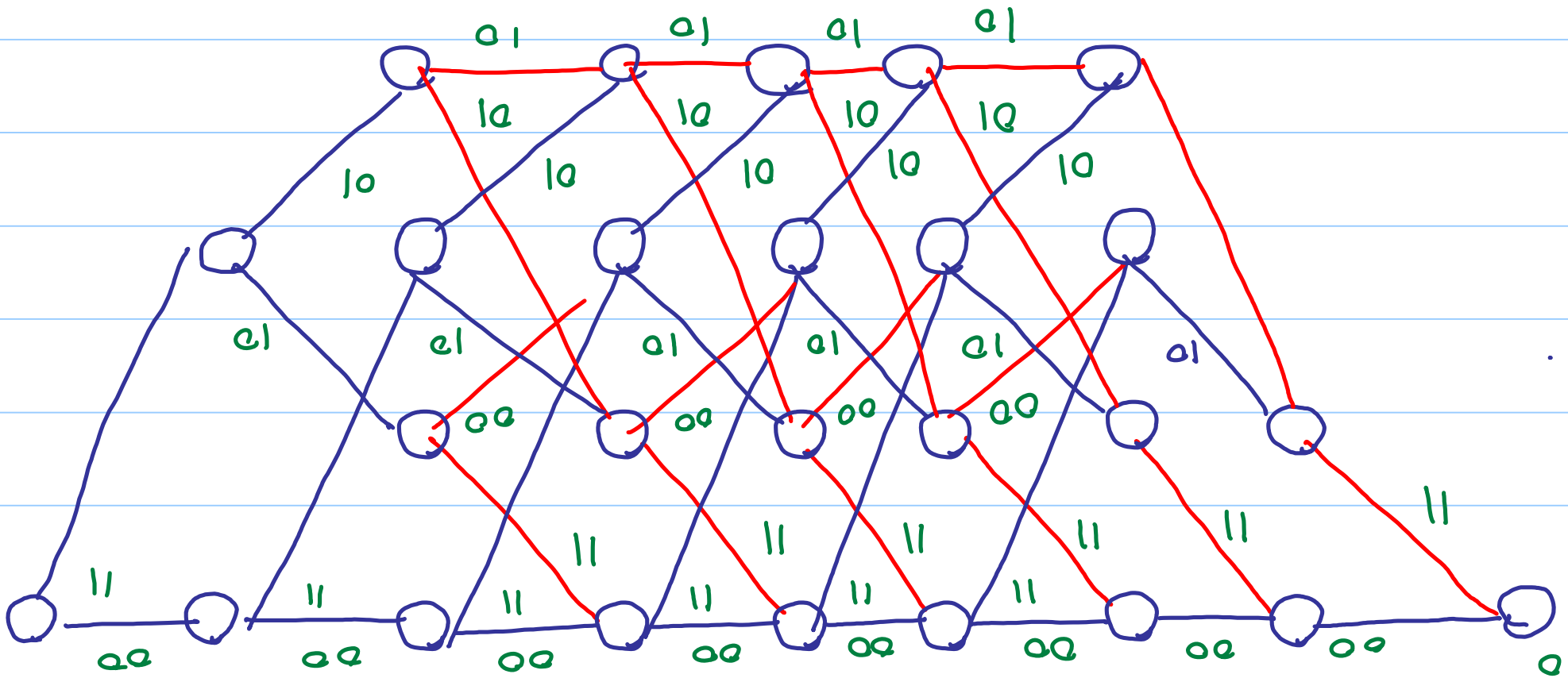
Similarly, an edge (or branch) in the trellis associated to code symbols

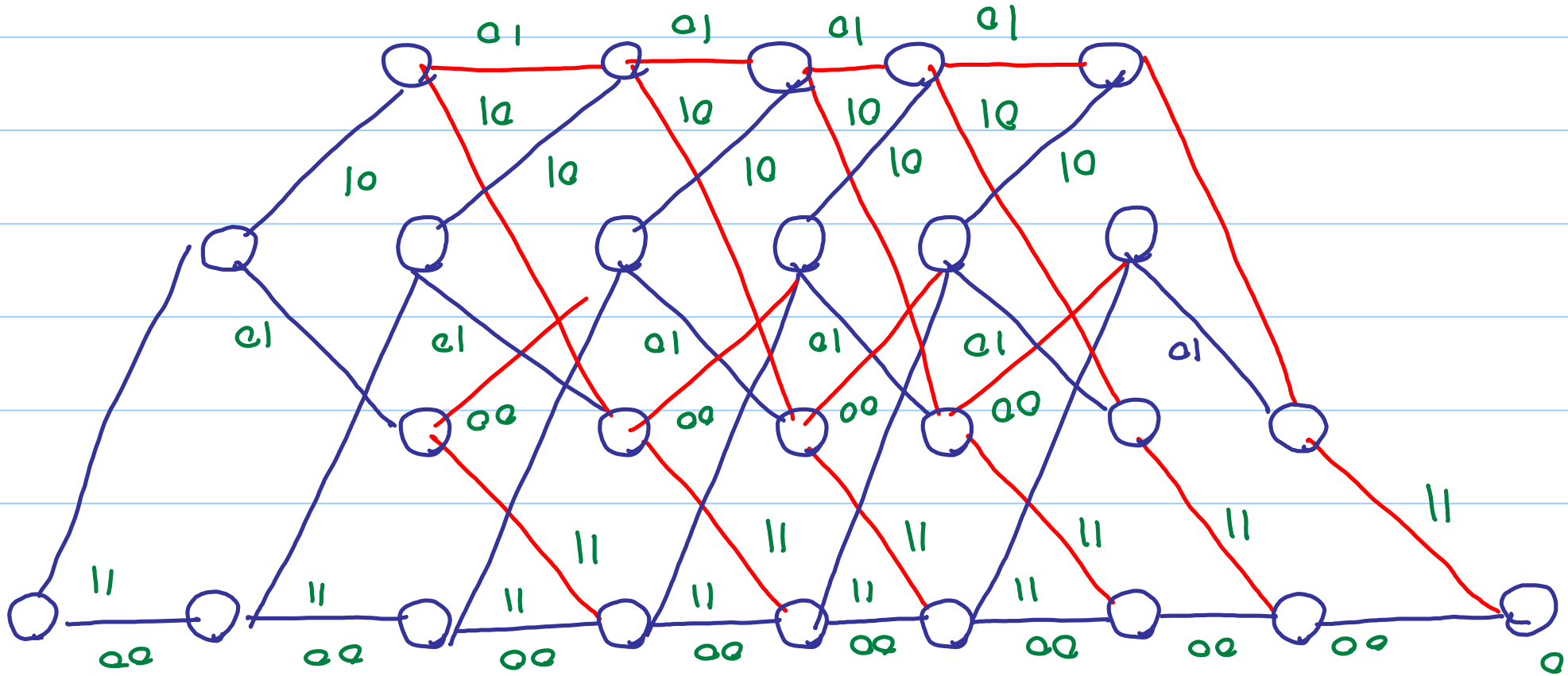
$$\left(V_k^{(1)} \; V_k^{(2)}\right) = d_H\left(\left\{V_k^{(1)} \; V_k^{(2)}\right\}, \left\{r_k^{(1)} \; r_k^{(2)}\right\}\right).$$

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

survivor at a node,

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


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


01
01
01
01
10
10
10
10
10
10
10
10
01
01
01
01
01
01
01
00
00
00
00
00
00
11
11
11
11
11
11
11
11
11
11
00
00
00
00
00
00
00
00
0

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

per message sequence of length N is given by:

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

$$\underbrace{\qquad\qquad\qquad}_{\text{loss in rate}}$$

Eg (of catastrophic error propagation) (CEP)

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

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

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

$$\Leftrightarrow \{u_k\}_{k=0}^{\infty} = 1\,1\,1\,1\,1 - - - - - - -$$

Then

$$G(D)$$

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

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

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

$\{V_k^{(1)}\} \Leftrightarrow 1\,0\,0\,0 - - - 0 - - - -$

channel corrupted

$\{V_k^{(2)}\} \Leftrightarrow 1\,1\,0\,0\,0 - - - 0 - - - -$

ASIDE

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

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

$$D+1$$

$$D^2+D$$

$$D+1$$

$$D+1$$

$$0$$

An encoder mx $G(D)$ for a convolutional code is said to have catastrophic

error propagation if there is some input $U(D)$ with ∞ Hamming weight that generates an output

$$\left[V^{(1)}(D) \quad V^{(2)}(D) \quad \cdots \quad V^{(n)}(D) \right]$$

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 ∞ # of message symbol errors.

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

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

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

$$\gcd\left(g_1(D)\ \ g_2(D)\ \ \cdots\ \ g_n(D)\right) = D^l \qquad l \geq 0$$

where $G(D) = \left[g_1(D)\ \ g_2(D)\ \ \cdots\ \ g_n(D)\right]$

is the PGM of the code.

GCD - quick review

gcd (27, 63) = ? 9

$63 \div 27 \Rightarrow$

$$27 \overline{)63} \quad 2, \quad 54, \quad 9 \text{ Remainder}$$

$27 \div 9 \Rightarrow$

$$9 \overline{)27} \quad 3, \quad 27, \quad 0$$

gcd

Remainder		Quotient
63		
27	1	2
9	0	3
0		

63	1	
27	0	
-2	1	2
7	-3	3

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

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
1	D	$D+1$	D
0			

gcd → 1

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

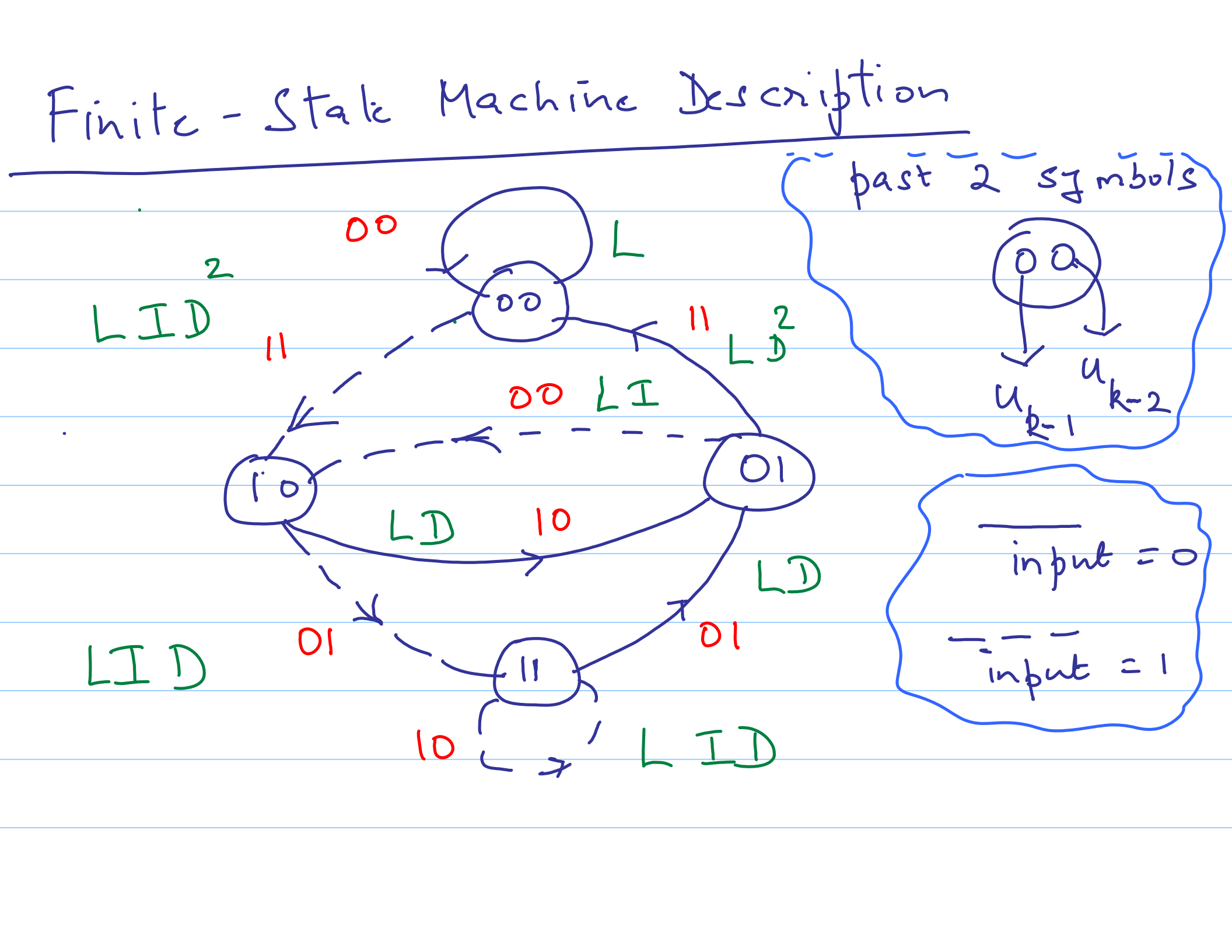
there is no CEP (by the theorem).

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

Remainder	D^2+1	$D+1$	Quotient
D^2+1	1	0	
$D+1$	0	1	D
$D+1$	1	D	1
0			

Note: $\gcd \neq D^l$, $l \geq 1$ and hence, this code has CEP.

Finite-State Machine Description

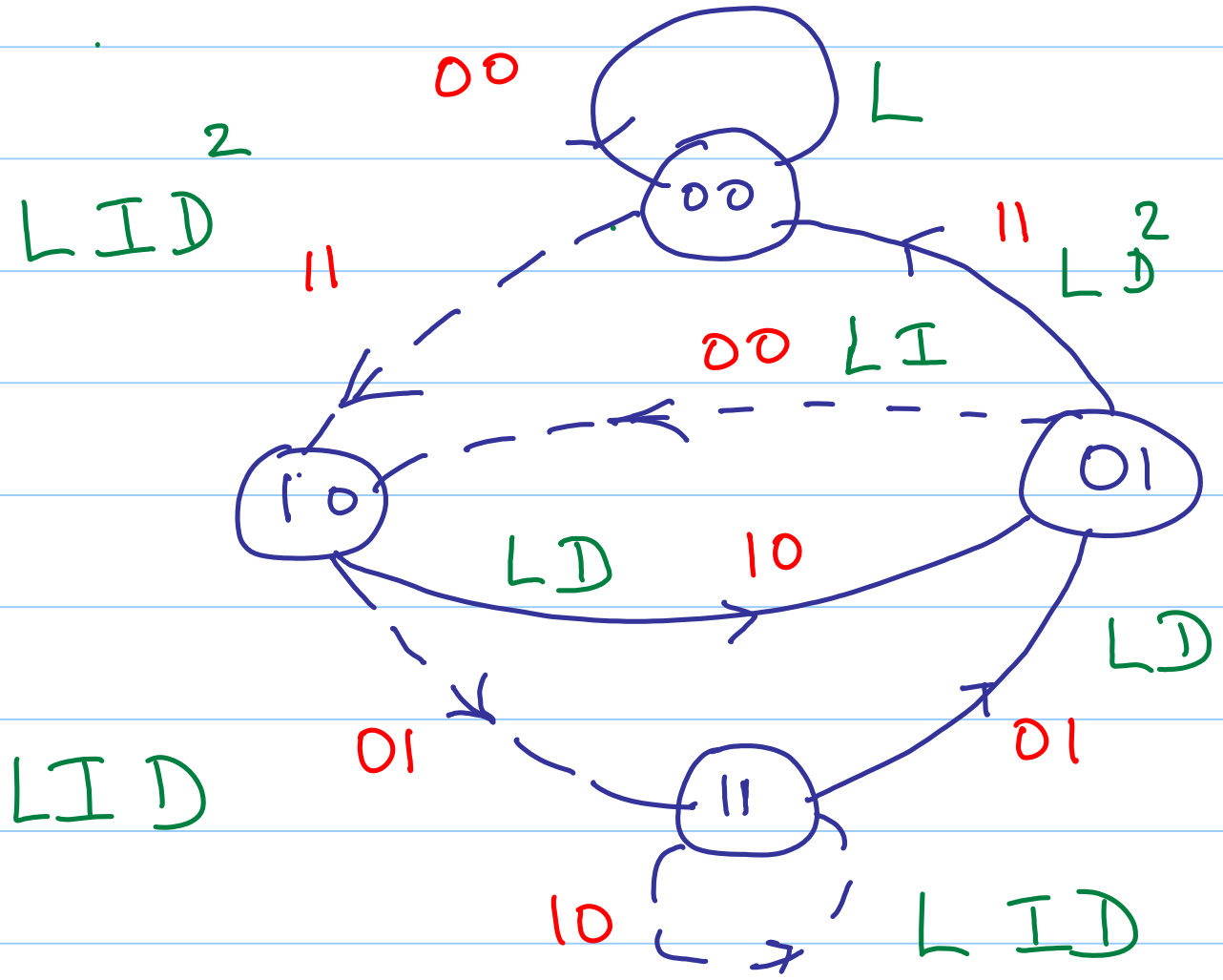


Lec 19 Path Enumeration

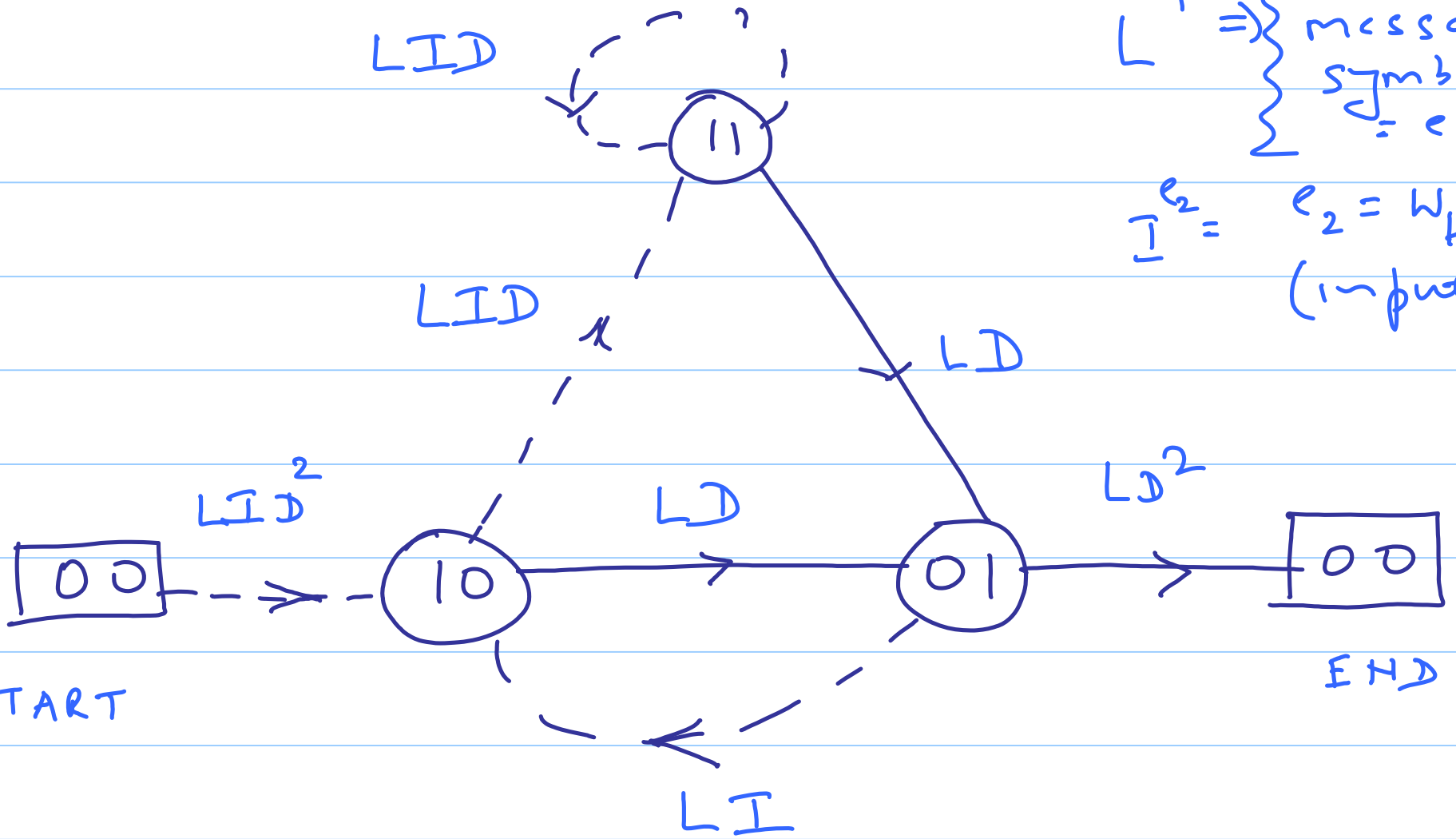
Recap:

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

Finite - State Machine Description



Modified State Diagram



L^{e_1} ⇒ { # of message symbols = e_1

I^{e_2} = $e_2 = W_{H1}$ (input)

D^{e_3} $e_3 = W_H$ (output)

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

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

a_{ijk} = # of paths in the modified state diagram of path length i, whose associated input (message) sequence has Hamming weight j and whose associate 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: $A_{START}(L, I, D) = 1$

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

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

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

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

Solving using Cramer's rule yields:

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

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

Note: Any expression of the form

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

can be expanded as:

{with no constant term

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

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

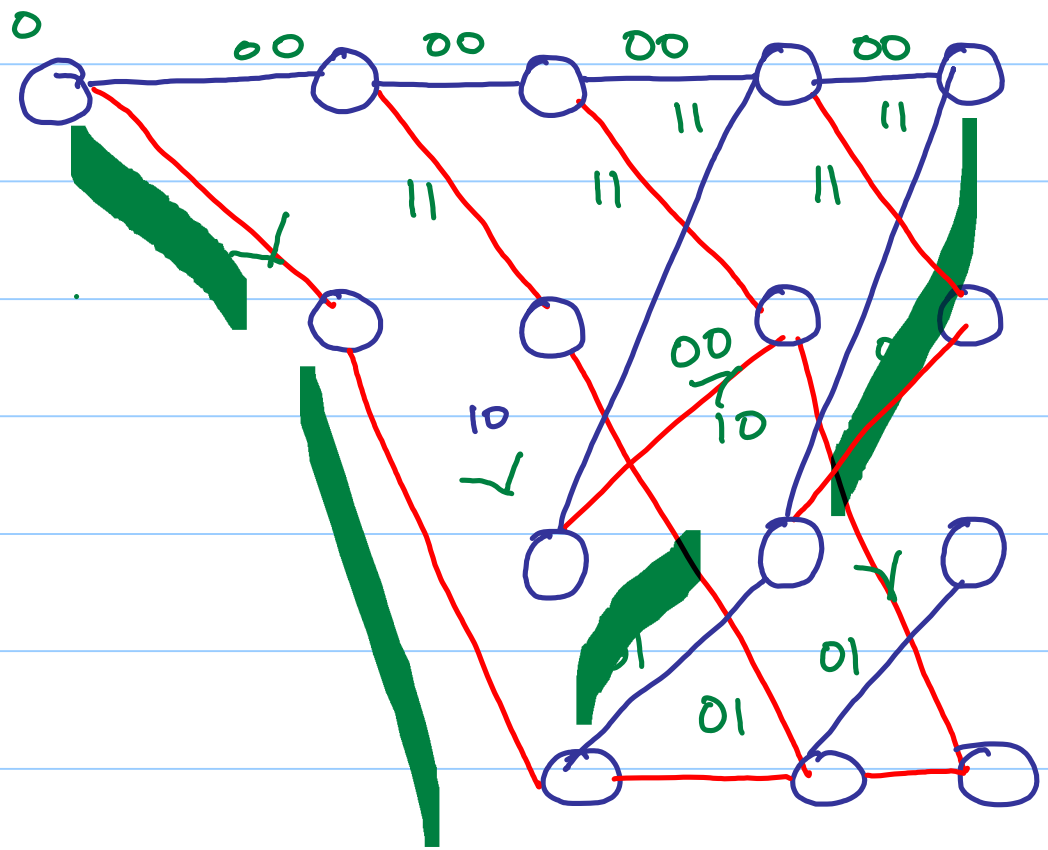
$$= L^3 I D^5 \left\{ 1 + LID(1+L) + L^2 I^2 D^2 (1+L)^2 + \cdots \right\}$$

How many paths have length 4?

(Our interest is in L^4 terms)

$$(L^3 I D^5)\, LID = L^4 I^2 D^6 .$$

⇒



L 4 I 2 D 6

General rate (k/n) convolutional code

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

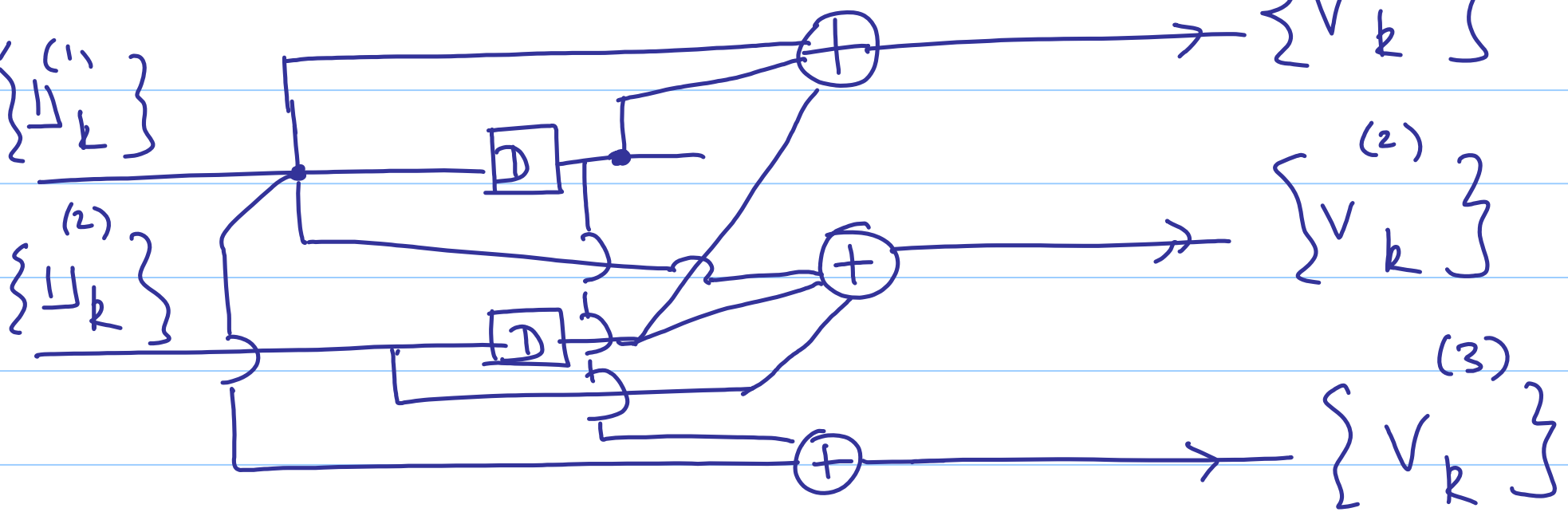
PGM

(2×3)

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

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

$$\nu = \sum_{i=1}^{k} \max_{1 \le j \le n} \{ \nu_{ij} \}$$

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

In the example: $\begin{bmatrix} 1+D & D & 1+D \\ D & 1+D & 0 \end{bmatrix}$

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

$$\therefore v = \max\{1,1,1\} + \max\{1,1,-\infty\}$$

$$= 1+1 = 2.$$

General I/O relation:

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

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

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

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

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

$$= \sum_{i=1}^{k} g_{ij}(D) \, U^{(i)}(D)$$

$$\Rightarrow \left[V^{(1)}(D) \; \cdots \; V^{(k)}(D) \right] = \left[U^{(1)}(D) \; \cdots \; U^{(k)}(D) \right] G(D)$$

Lec 20 { Viterbi decoder over the AWGN Channel

Recap

* path information enumeration using generating function techniques

* { General rate k/n convolutional code – PGM derivation

Thm (CEP) A n.a.s.c on the PGM of a general, rate $\frac{k}{n}$ convolutional code to not have CEP is that:

$$\gcd\left(\Delta_1(D), \cdots \Delta_{\binom{n}{k}}^{(D)}\right) = D^{\ell}, \quad \ell \geq 0$$

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 $G(D)$

Eg

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

(2×3)

$k = 2$
$n = 3$

$\therefore \binom{n}{k} = \binom{3}{2} = 3$

$$\Delta_1(D) = \begin{vmatrix} 1+D & 1 \\ D & 1+D \end{vmatrix} = 1+D^2+D$$

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

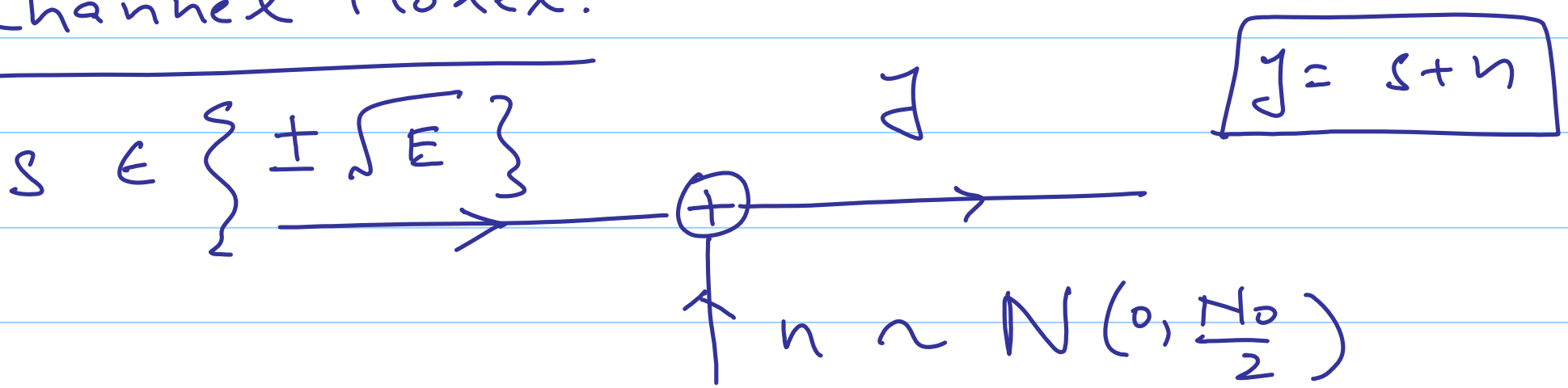
$$\Delta_3(D) = \begin{vmatrix} 1 & 1+D \\ 1+D & 0 \end{vmatrix} = (1+D)^2$$

Can be verified that

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

Viterbi decoding over the AWGN channel

Channel Model:



($\frac{E}{N_0}$ is a measure of the SNR on the channel)

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

— {choose that codeword that

maximizes $p(\underline{y} \mid \underline{c})$ → code symbol c_i

where $s_i = (-1)\sqrt{E}$

s_i → symbol transmitted over the AWGN channel

In the convolutional code:

$$\underline{c} = \left(\left(v_t^{(1)} \; v_t^{(2)} \; \cdots \; v_t^{(n)} \right), \; t = 0, 1, \ldots N-1 \right)$$

$$p(\underline{y} \mid \underline{c}) = \prod_{t=0}^{N-1} \prod_{j=1}^{n} p\left(y_t^{(j)} \mid v_t^{(j)}\right)$$

where

$$p\left(y_t^{(j)} \mid v_t^{(j)}\right) = \frac{1}{\sqrt{2\pi (N_0/2)}} \exp\left(\frac{-1}{2\frac{N_0}{2}} \left(y_t^{(j)} - s_t^{(j)}\right)^2\right)$$

$$\left(s_t^{(j)} = \sqrt{E}\,(-1)^{v_t^{(j)}}\right)$$

Clear that MLD 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:

$$\sqrt{E} \sum_{t=0}^{N-1} \sum_{j=1}^{n} y_t^{(j)} (-1)^{o_t^{(j)}}$$

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 process 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 $A_{EHD}(L, I, D)$ derived can be used to provide an upper bound on the probability of bit error incurred while

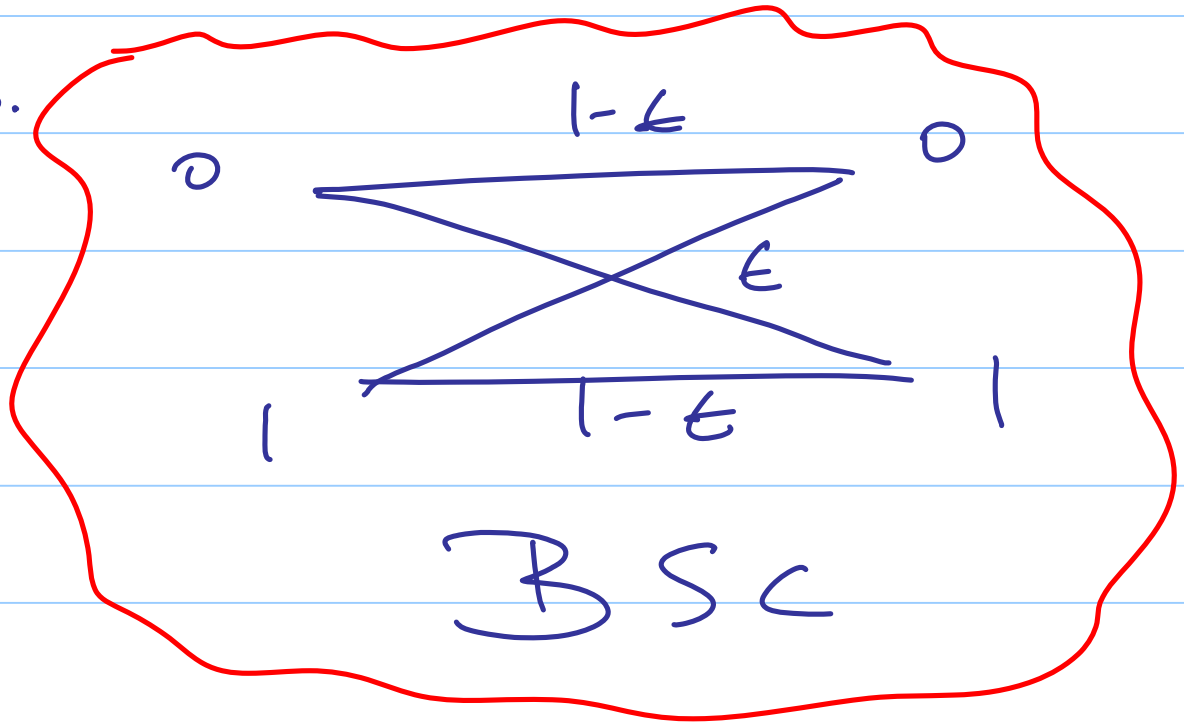
employing the Viterbi decoder:

Case (i) BSC channel:

$$P_{bc} \leq \frac{1}{k} \frac{\partial}{\partial I} A_{END}(L, I, D) \Big|_{\substack{L=1 \\ I=1}}$$

BIT ERROR PROB.

$$D = 2\sqrt{\epsilon(1-\epsilon)}$$



Case (ii) AWGN channel:

$$P_{bc} \leq Q\left(\sqrt{\frac{2E\, d_{free}}{N_0}}\right) \exp\left(-\frac{E\, d_{free}}{N_0}\right)$$

$$\times \frac{1}{k} \frac{\partial}{\partial I} A_{END}(L, I, D) \Big|_{\substack{L=1 \\ I=1}}$$

$$D = \exp\left(-E/N_0\right)$$

Lec 21 The Generalized Distributive Law

Recap

* CEP in general, rate k/n convolutional code.
* Viterbi decoding over the AWGN channel
* Expressions to compute bit error probability of the Viterbi decoder.

Clarification : $d_{free} = ?$

d_{free} is the minimum distance between a pair of distinct codewords (of ∞ length) in the convolutional code.

Can be computed from the power series.

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

Eg $$A_{END}(L, I, D) = \frac{L^3 I D^5}{1 - LID(1+L)}$$

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

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

$$\therefore \boxed{d_{free} = 5}$$ ///

GDL

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

$a(b+c)$: 1 addition, 1 multiplication.

$ab + ac$: 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 $|A| = \ell$

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:

$$\alpha(x, w) = \sum_{y z} f(x y w)\, g(x, z)$$

$$= \left(\sum_y f(x,y,w)\right)\left(\sum_z g(x,z)\right)$$

$$\alpha(x,w) = f'(xw) \cdot g'(x)$$

$f' \Rightarrow q^2(q-1) \qquad g' \Rightarrow q(q-1)$

$\alpha \Rightarrow q^2$

total # of computations

$= q^3 - q^2 + q^2 - q + q^2$

$= q^3 + q^2 - q$

(versus $2q^4 - q^2$)

$$\beta(y) = \sum_{xwz} f(xyw)\, g(xz)$$

$$= \sum_{x} \left(\sum_{w} f(xyw) \right) \left(\sum_{z} g(xz) \right)$$

$f'(xy)$ $\qquad$ $g'(x)$

total # of computations

$$= q^2(q-1) + q(q-1) + q(q+q-1)$$

$$= q^3 - q^2 + q^2 - q + 2q^2 - q$$

$$= q^3 + 2q^2 - 2q$$

$///$ versus $2q^4 - q$

DETOUR: Semirings

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:

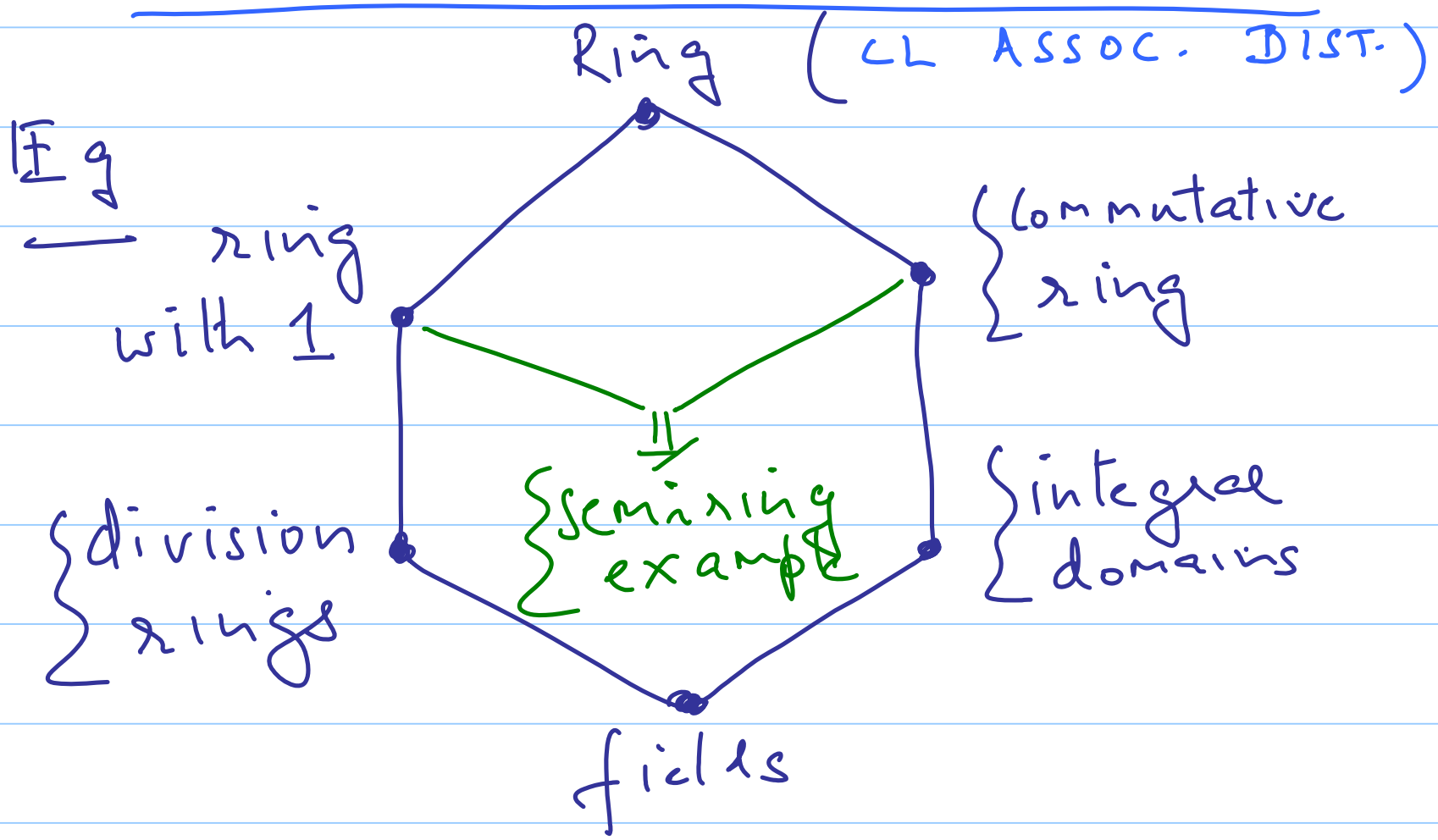
- CLOSURE
- IDENTITY EL
- ASSOCIATIVE
- COMMUTATIVE

(note that multiplicative inverses need not exist)

(iii) $(R, +, \cdot)$ must satisfy the

distributive law:

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



every commutative ring with identity is a semiring

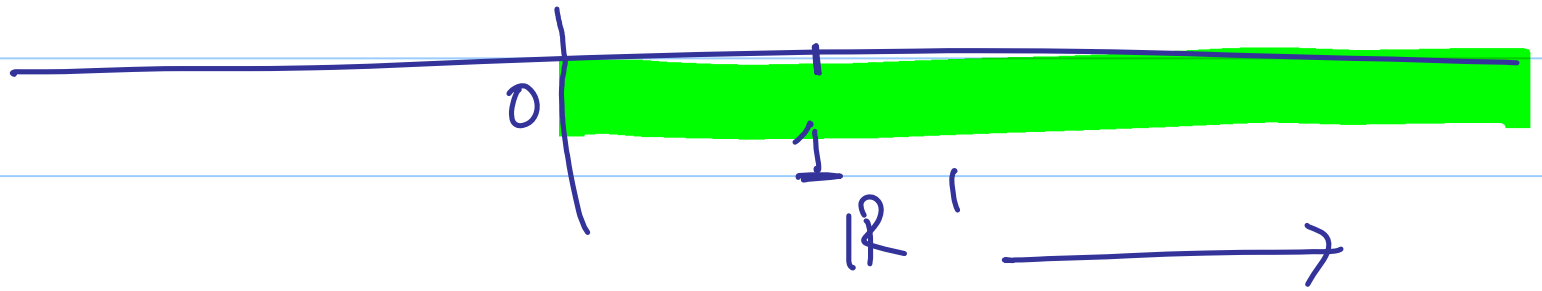
in particular, every field is a semiring

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

$\mathbb{F}[x]$

$\mathbb{Z}$

Eg $(\mathbb{R}, +, \cdot) = ([0, \infty), +, \cdot)$



$(\mathbb{R}, +)$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

$(\mathbb{R}, \cdot)$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

— DIST LAW ✓

Lec 22 The MPF Problem

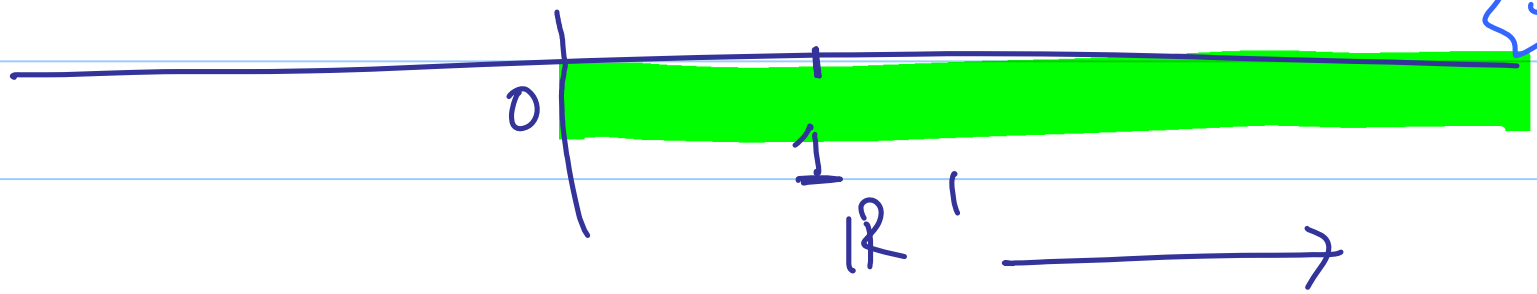
Recap

- d_{free}
- { distributive law }
  { # of computations
- { semiring
  - examples.

Further examples of semiring

Eg 3

Eg 3 $(\mathbb{R}, \text{MAX}, \cdot) = ([0, \infty), \text{MAX}, \cdot)$ } MAX PRODUCT SEMIRING



$(\mathbb{R}, \text{MAX})$

- 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?

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

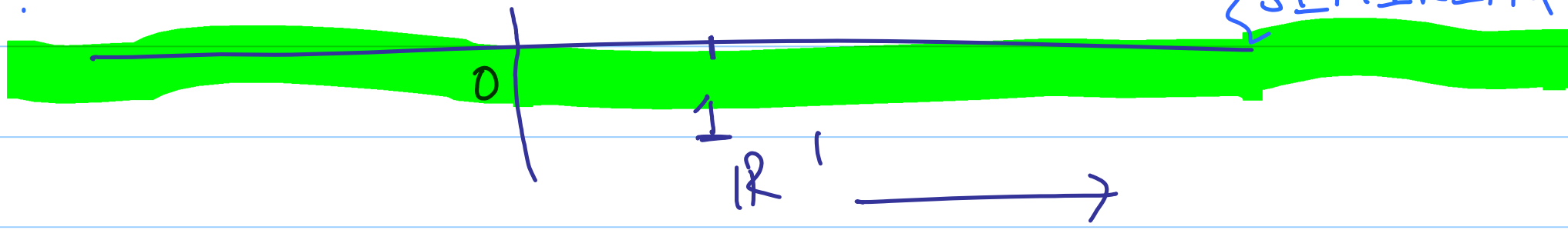
{ SUM, PRODUCT, SR }

$$a(\text{MAX}\{b, c\}) = \text{MAX}\{ab, ac\}$$

✓ ?

Eg: $(\mathbb{R}, \text{MIN}, +) = ((-\infty, \infty], \text{MIN}, +)$ } MIN SUM

SEMIRING



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

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

$(\mathbb{R}, +)$

- CL ✓
- ASSOC ✓
- ID. EL ✓
- COMM ✓

$$\min\{e, a\} = a \qquad e = +\infty !$$

DISTRIBUTIVE LAW?

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

$$a + \min\{b, c\} = \min\{a+b, a+c\}$$

✓

?

Yes!

THE MPF PROBLEM

(marginalize a product function)

Setting:

$$S = \{1, 2, \ldots, n\}$$

$$X_S = \{x_1 \; x_2 \; \ldots \; x_n\}$$

$X_i \in A_i$ alphabet

$$|A_{:i}| = q_i$$

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

$$S_j = \{ i_1 \; i_2 \; \cdots \; i_{n_j} \} \subseteq S$$

$$X_{S_j} \triangleq \{ X_{i_1} \; X_{i_2} \; \cdots \; X_{i_{n_j}} \}$$

→ LOCAL DOMAINS

A_{s_j} = alphabet from which x_{s_j} is drawn

$|A_{s_j}| = q_{s_j}$

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:

$$\beta(x_S) = \prod_{j=1}^{M} \alpha_j(x_{S_j})$$

j^{th} objective function:

$1 \leq j \leq M$

$$P_{d_{S_i}}(x_{S_i}) = M \prod_{c} \phi_c(x_c)$$

marginalization

product function

$$d_{S_i} = \sum_{S \setminus S_i} d_S$$

$$d_{S_i} = \sum_{S \setminus S_i} \prod_{c} \phi_c(x_c)$$

ASIDE:

$$p(x_1) = \sum_{x_2} p(x_1, x_2)$$

Eg 1 (The 8-dimensional Walsh – Hadamard Transform)

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

Alphabets A_i

$A_i = \{0, 1\}$ Same for all $1 \le i \le M$

$|A_i| = 2$

This is thus an example of the MPF problem.

$S = \{1, 2, 3, 4, 5, 6\}$ $X_S = \{x_1 x_2 x_3 x_4 x_5 x_6\}$

R = field of all real numbers $= \mathbb{R}$

$S_1 = \{1\ 2\ 3\}$ $X_{S_1} = \{x_1\ x_2\ x_3\}$

$S_2 = \{1\ 4\}$ $X_{S_2} = \{x_2\ x_4\}$ $x_1\ x_4$

$\alpha_1(X_{S_1}) = f(x_1 x_2 x_3)$ $\alpha_2(X_{S_2}) = (-1)$

$$S_3 = \{ 2\ 5 \} \quad x_{S_3} = \{ x_2\ \ x_5 \}$$

$$S_4 = \{ 3\ 6 \} \quad x_{S_4} = \{ x_3\ \ x_6 \}$$

$$S_5 = \{ 4\ 5\ 6 \} \quad x_{S_5} = \{ x_4\ x_5\ x_6 \}$$

$$\alpha_3(x_{S_3}) = (-1)^{x_2 x_5}$$

$$\alpha_4(x_{S_4}) = (-1)^{x_3 x_6}$$

$$\alpha_5(x_{S_5}) = 1 \quad !$$

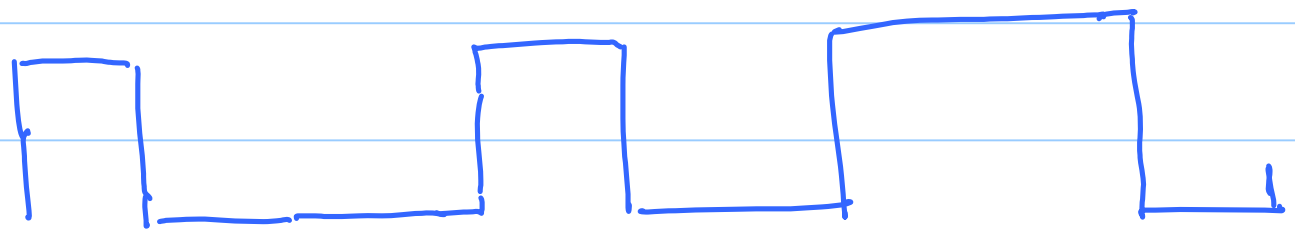
$$
\begin{bmatrix} F(000) \\ F(001) \\ F(111) \end{bmatrix}
$$

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

Walsh Hadamard matrix

$$
\begin{bmatrix} f(000) \\ f(001) \\ f(010) \\ f(011) \\ \vdots \\ f(111) \end{bmatrix}
$$

1 -1 -1 1 -1 1 1 -1



Eg 2 $[7, 4, 2]$ linear block code

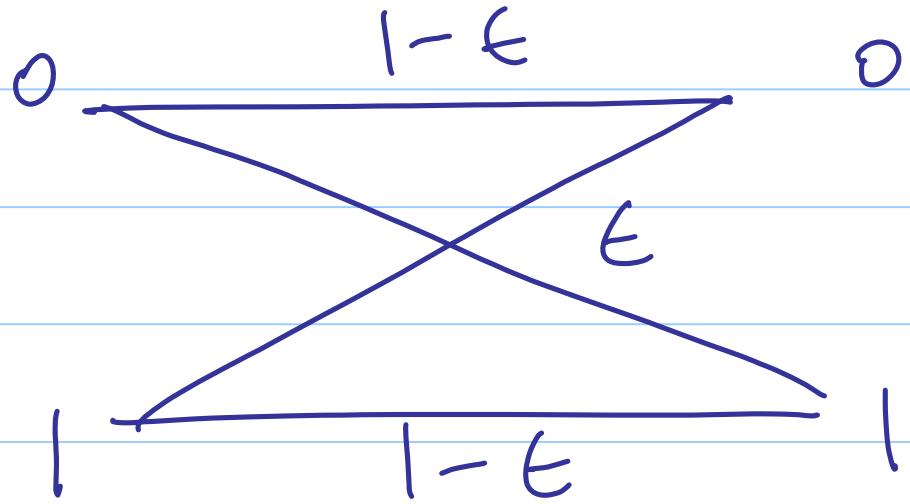
$n \quad k \quad d$

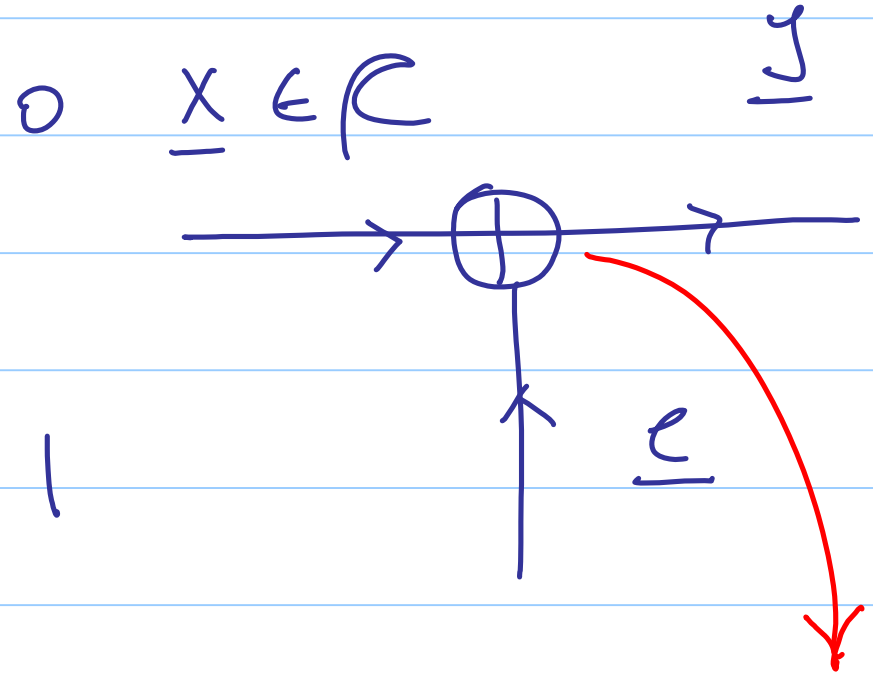
parity - check matrix

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

(3×7)

BSC :





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

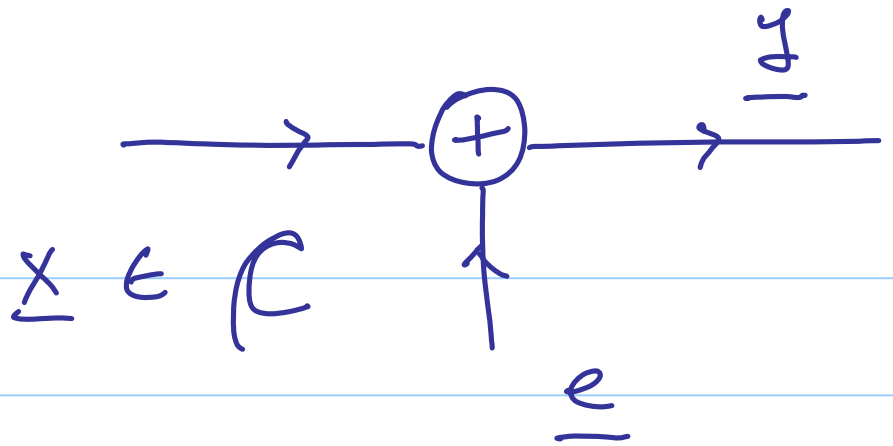
$n \quad k \quad d$

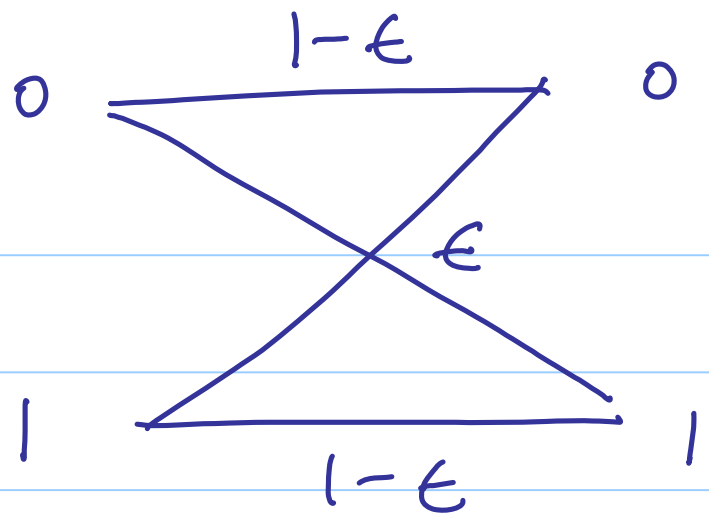
parity - check matrix

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

(3×7)

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





Define

$$F_i(x_i) = \max_{x_1 \cdots x_{i-1}\, x_{i+1} \cdots x_7} p(\underline{y}/\underline{x})$$

$$\underline{x} \in C$$

We will compute $F_i(x_i)$ for $i = 1, 2, \ldots, 7$

for $x_i \in \{0, 1\}$

example:

$F_1(0)$	$F_1(1)$
$F_2(0)$	$F_2(1)$
$F_3(0)$	$F_3(1)$
$F_4(0)$	$F_4(1)$
$F_5(0)$	$F_5(1)$
$F_6(0)$	$F_6(1)$
$F_7(0)$	$F_7(1)$

$\Longleftrightarrow$

0	1	
2	8	1
1	8	1
8	6	0
2	8	1
-1	8	1
8	4	0
6	8	1

decoded codeword = $[1\ 1\ 0\ 1\ 1\ 0\ 1]$

$$F_i(x_i) = \max_{\substack{x_1 \cdots x_{i-1} \\ x_{i+1} \cdots x_7}} p(\underline{y} \mid \underline{x})$$

$$\underline{x} \in C$$

$$= \max_{\substack{x_1 \cdots x_{i-1} \\ x_{i+1} \cdots x_7}} p(\underline{y} \mid \underline{x}) \chi_C(\underline{x})$$

$$\chi_R(\underline{x}) = \begin{cases} 1 & \underline{x} \in R \\ 0 & \text{else} \end{cases}$$

$$\chi_{\mathcal{C}}(\underline{x}) = \chi_{124}(\underline{x})\,\chi_{346}(\underline{x})\,\chi_{457}(\underline{x})$$

$$H = \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{matrix} \\ \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \end{array}$$

$$\chi_{124}(\underline{x}) = \chi_{124}(x_1 x_2 x_4) = \begin{cases} 1 & x_1 + x_2 + x_4 = 0 \\ 0 & \text{else} \end{cases}$$

$$\chi_{346}(\underline{x}) = \chi_{346}(x_3 x_4 x_6) = \begin{cases} 1 & x_3 + x_4 + x_6 = 0 \\ 0 & \text{else} \end{cases}$$

$$\chi_{457}(\underline{x}) = \chi_{457}(x_4 x_5 x_7) = \begin{cases} 1 & x_4 + x_5 + x_7 = 0 \\ 0 & \text{else.} \end{cases}$$

$$F(x_{\cdot 2}) = \max \prod_{i=1}^{2} \phi(x_{i\cdot}, x_{i-1\cdot})$$

$$x_{i+1\cdot} \dots x_{2\cdot}$$

$$x_{1\cdot} \dots x_{i-1\cdot}$$

$$\max_{x_{2\cdot}} \chi_{124}(x_1, x_2, x_4) \cdot \chi_{347}(x_3, x_4, x_7) \cdot \chi_{45t}(x_4, x_5, x_t)$$

Local Domains

$\{x_i\}$ $1 \le i \le 7$

$\{x_1\ x_2\ x_4\}$

$\{x_3\ x_4\ x_6\}$

$\{x_4\ x_5\ x_7\}$

Local Kernels

$\phi(\vec{x}_i \mid x_i)$

$\chi_{124}(x_1\ x_2\ x_4)$

$\chi_{346}(x_3\ x_4\ x_6)$

$\chi_{457}(x_4\ x_5\ x_7)$

Eg ML code-symbol decoding of the

$[7, 4, 2]$ code.

$$p(x_i \mid \underline{y}) = \sum_{\sim x_i} p(\underline{x} \mid \underline{y}) \quad \text{proportional to} \quad \alpha \sum_{\sim x_i} p(\underline{x}, \underline{y})$$

$\underline{x} \in \mathcal{C}$ $\qquad$ $\underline{x} \in \mathcal{C}$

$$= \sum_{\sim x_i} p(\underline{x})\, p(\underline{y} \mid \underline{x})$$

$\underline{x} \in \mathcal{C}$ $\qquad$ $\frac{1}{|\mathcal{C}|}$

$$\propto \sum_{\sim x_i} \prod_{j=1}^{7} p(y_j \mid x_j)\, \chi_{124}(x_1, x_2, x_4)$$

$$\chi_{346}(x_3, x_4, x_6)$$

$$\chi_{457}(x_4, x_5, x_7)$$

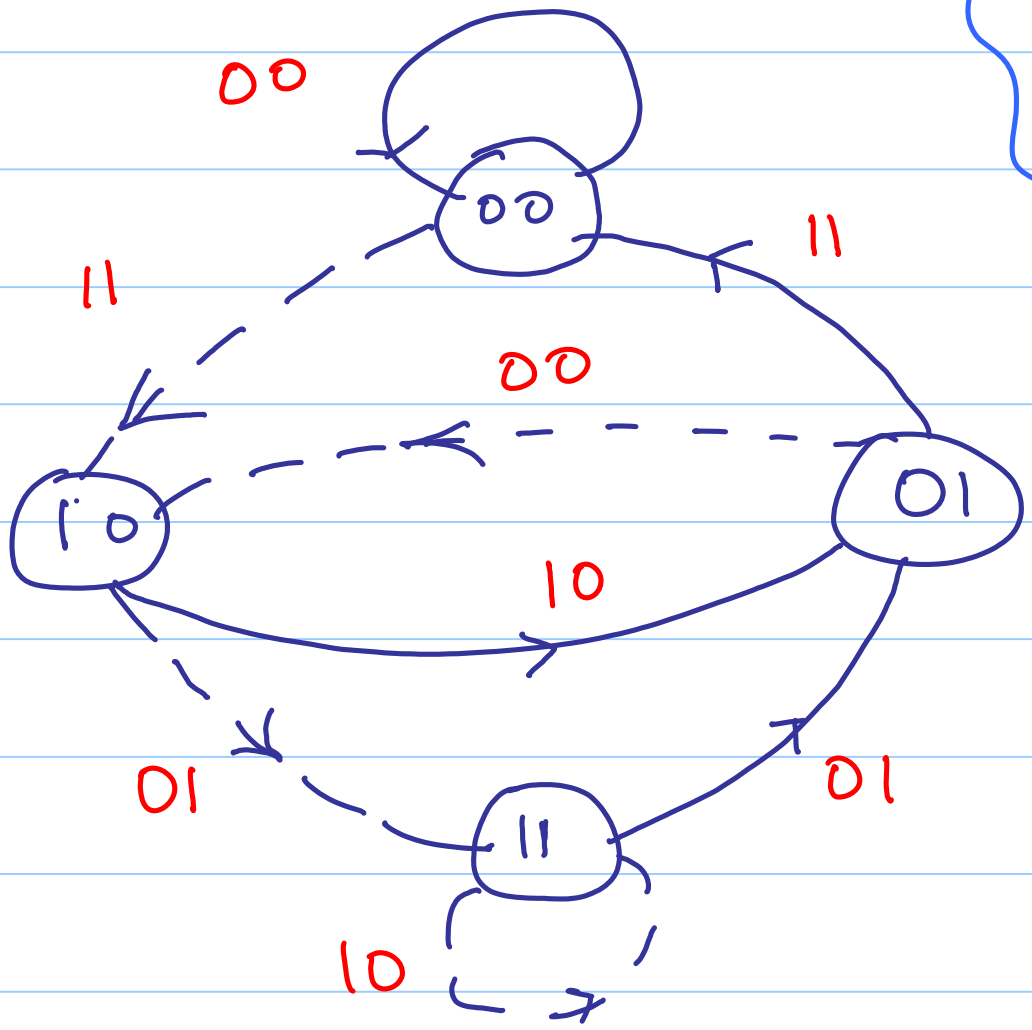
clear that this is once again an instance of the MPF problem.

The computation this time however, is carried out in the sum product

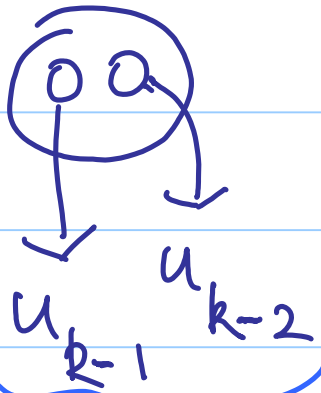
semiring. The local domains as well as local kernels remain exactly the same.

Eg { ML codeword decoding of convolutional codes.

Finite - State Machine Description

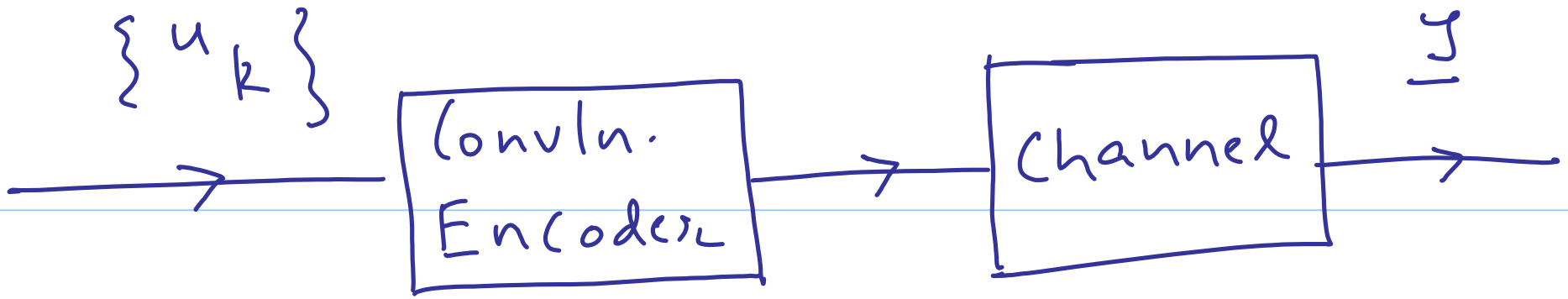


past 2 symbols



input = 0

input = 1



assume a rate $1/n$ convln code for simplicity

$$\{(V_k^{(1)} \; V_k^{(2)} \; \cdots \; V_k^{(n)})\}$$

Goal: ML codeword decoding, i.e.,

identifying the codeword $\underline{v}$ such that $\phi(\underline{r}|\underline{v})$ is a maximum.

which is equivalent to identifying the message vector $\underline{u}$ which is such that $\phi(\underline{r}|\underline{u})$ 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

$y_k = (y_{k1} \; \cdots \; y_{kn})$

(In the example below, $N=4$)

s_0
u_0
y_0
s_1
u_1
y_1
s_2
u_2
y_2
s_3
u_3
y_3
s_4

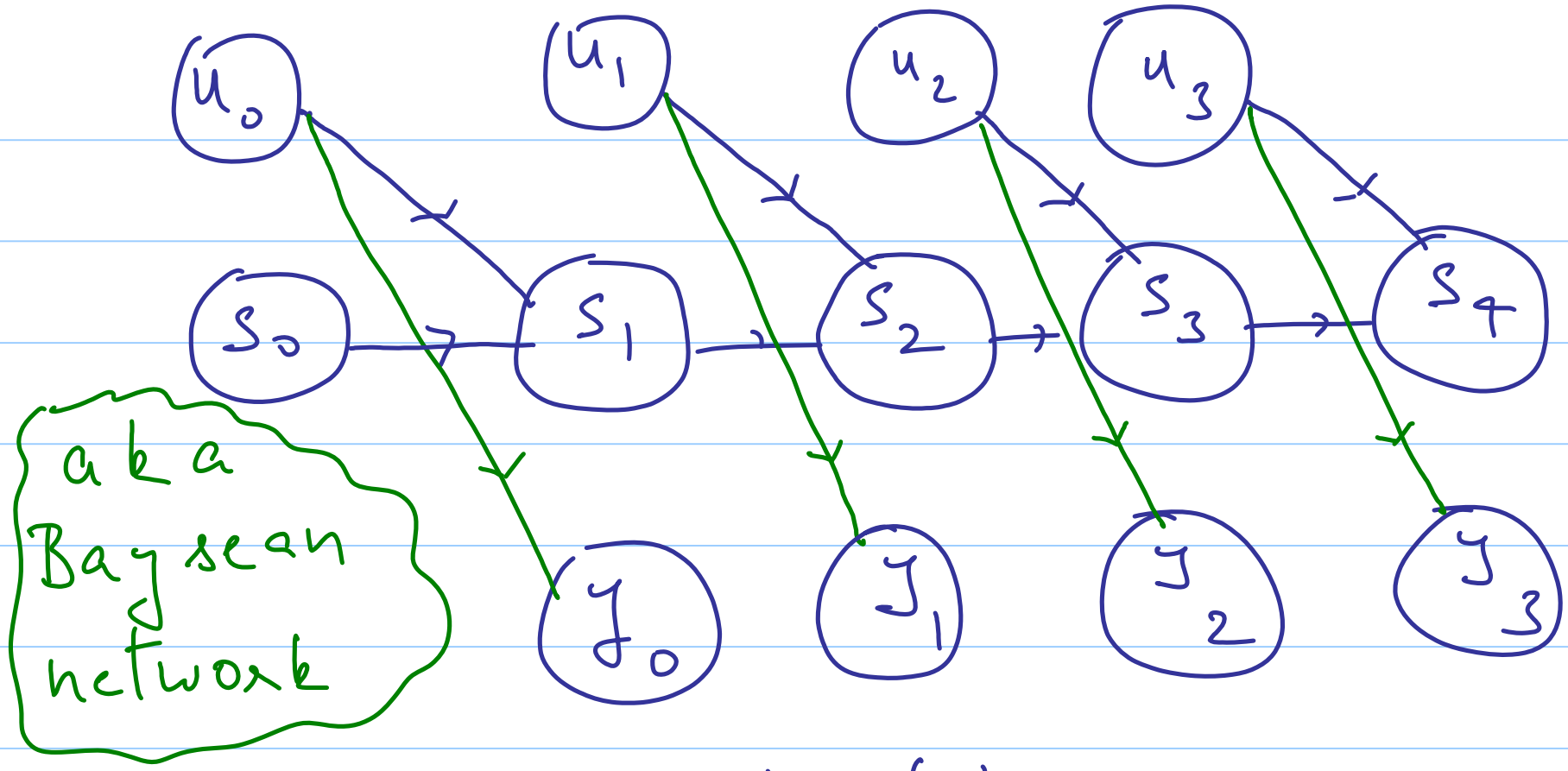
Lec 24 Junction Trees

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. distbn fn:

$$p\left(\{u_i\}_{i=0}^{3}, \{s_i\}_{i=0}^{4}, \{y_i\}_{i=0}^{3}\right)$$

$$= p(s_0) \prod_{i=0}^{3} p(u_i)\, p(s_{i+1} \mid s_i, u_i)$$

$$p(y_i \mid s_i, u_i)$$

$$F_i(u_i) = \max_{\sim u_i} p(\underline{y} \mid \underline{u}) \qquad 0 \le i \le 3$$

$$\propto \max_{\sim u_i} p(\underline{y}, \underline{u})$$

since all message vectors $\underline{u}$ are equally likely

$$= \max_{\sim u_i} p(\underline{y}, \underline{u}, \underline{s})$$

$\underline{s}$

$$= \max_{\sim u_i} \; p(s_0) \prod_{i=0}^{3} p(u_i)\, p(s_{i+1} \mid s_i\, u_i)$$

$$\leq \; p(j_i \mid s_i\, u_i)$$

LD

$\{u_i\}$

$\{s_0\}$

$\{s_{i+1} \;\; s_i \;\; u_i\}$

$\{s_i \;\; u_i\}$

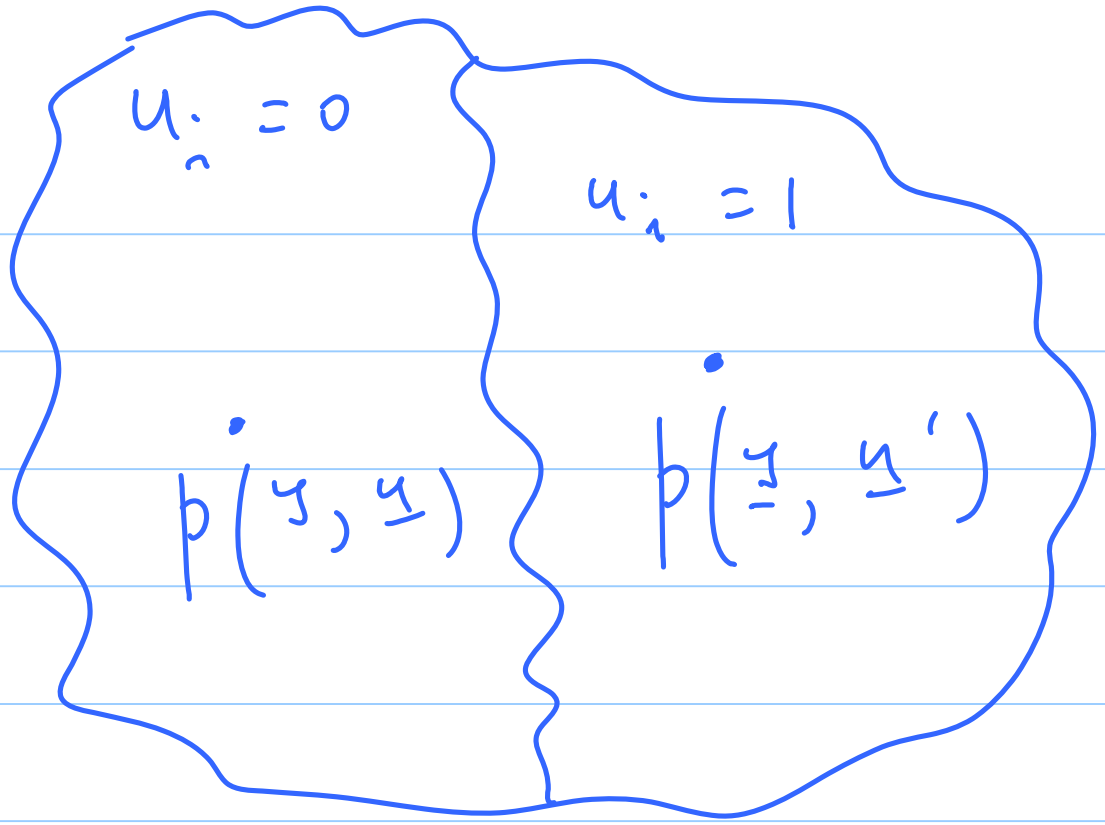
LK

$p(u_i)$ $\qquad 0 \leq i \leq 3$

$p(s_0)$

$p(s_{i+1} \mid s_i \;\; u_i)$

$p(j_i \mid s_i \;\; u_i)$

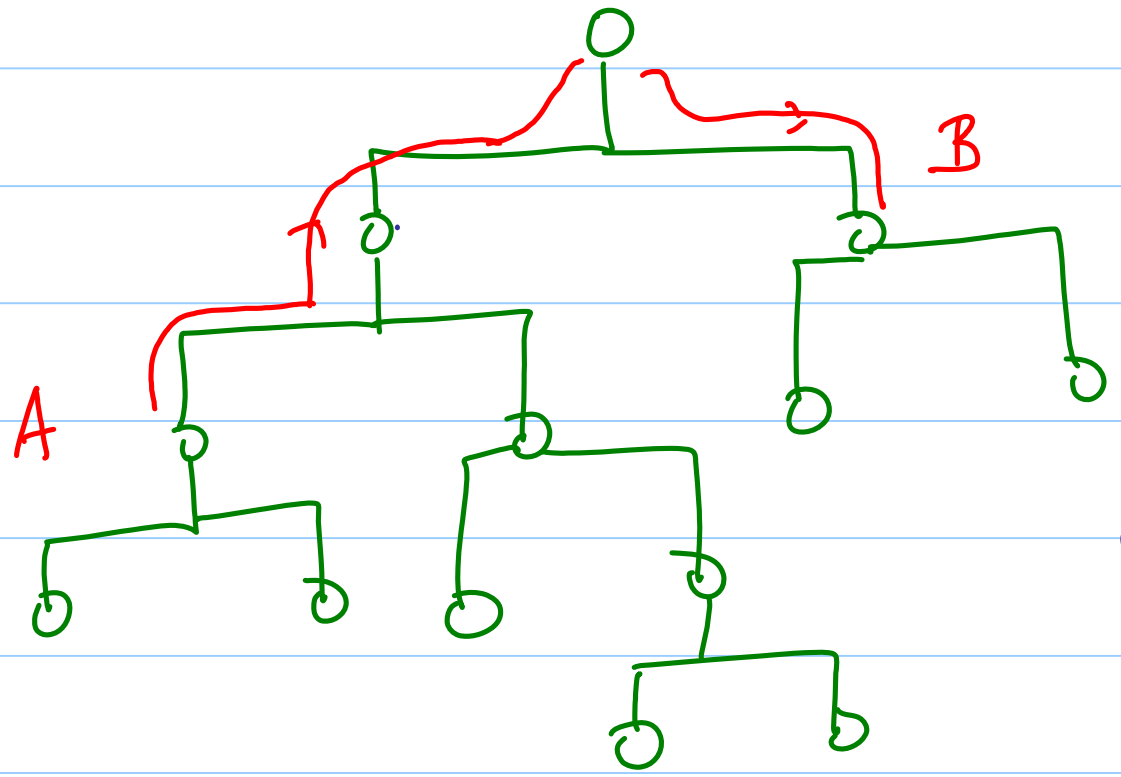


We have thus formulated the ML codeword decoding of a convln code as an MPF problem.

The first step in attempting to solve (using the GDL) 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:

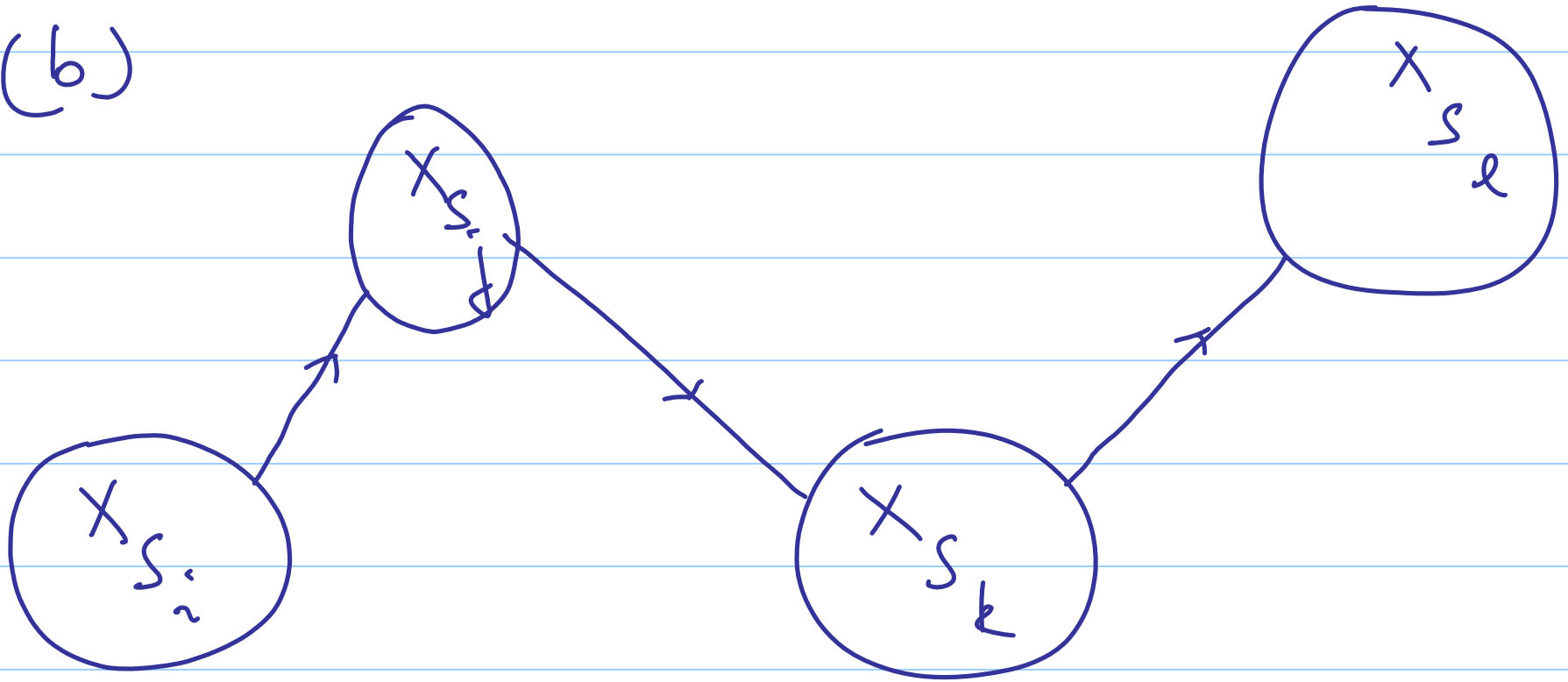
the # of nodes − # of edges = 1

* In any tree there is a unique path between any two distinct nodes.

Defn. In the setting of the MPF problem, a jn tree is a graph whose nodes are in 1-1 correspondence with the local domains $\{X_{S_i}\}_{i=1}^{M}$. and where edges are drawn between nodes in such a way that

(a) the graph is a tree

(b)



For every node X_{S_j} on the unique path lying between nodes X_{S_i} and X_{S_k}, it must be that

$$S_j \subseteq S_i \cap S_k$$

Eg 2 $[7, 4, 2]$ linear block code

parity - check matrix

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

(3×7)

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

Local Domains

$\{x_i\}$ $1 \le i \le 7$

$\{x_1\ x_2\ x_4\}$

$\{x_3\ x_4\ x_6\}$

$\{x_4\ x_5\ x_7\}$

Local Kernels

$\phi(\vec{x}_i \mid x_i)$

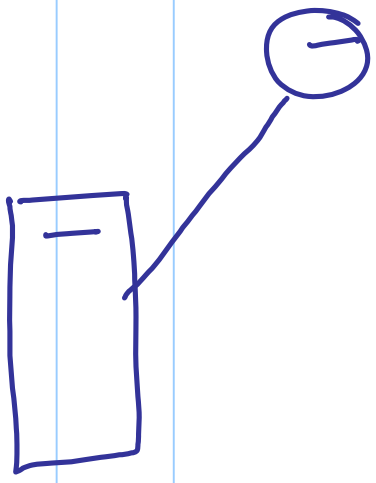
$\chi_{124}(x_1\ x_2\ x_4)$

$\chi_{346}(x_3\ x_4\ x_6)$

$\chi_{457}(x_4\ x_5\ x_7)$

$x_1 x_2 x_4 \equiv 124$

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

f
f
f
f
projection onto X4

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 MPF problem, and which

is also a tree, then it must be that

$$\text{edge weight of } g \leq \text{node weight of } g - n$$

where n = # 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

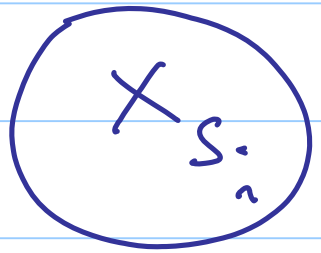
is also a trace, then it must be that

$$\text{edge weight of } y \leq \text{node weight of } y - n$$

where n = # of variables

$= |S|$.

Pf. The weight of a node associated to a local domain



$= |S_i|$

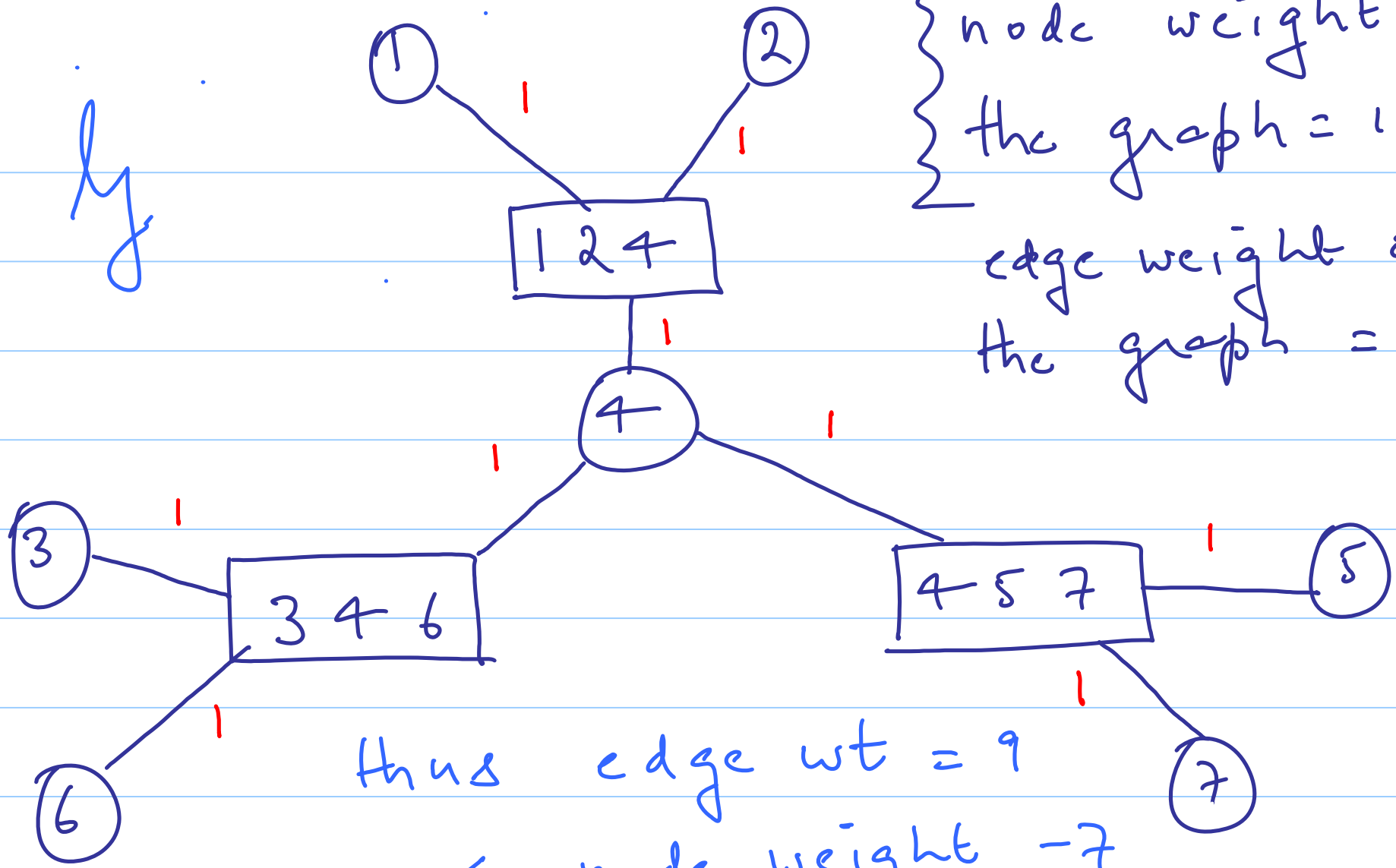
The edge weight of an edge connecting nodes associated to x_{S_i} and x_{S_j} $= |S_i \cap S_j|$.

node weight of the graph

= {Sum of the node weight of the nodes in the graph.

edge weight of the graph

= {Sum of the edge weights of edges in the graph.



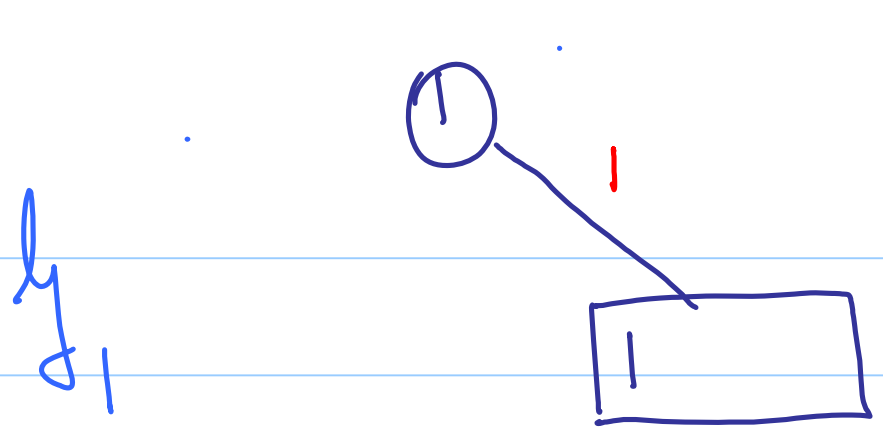
node weight of the graph = 16

edge weight of the graph = 9

thus edge wt = 9

$\leq$ node weight − 7

$= 16 - 7$



projection onto x_1

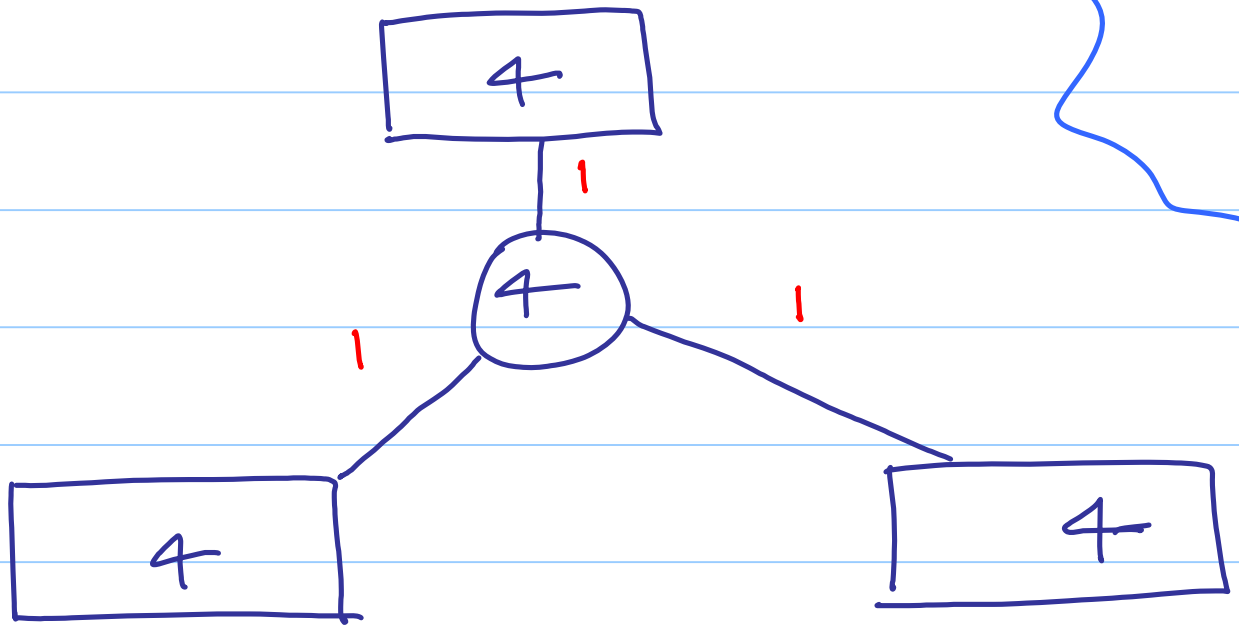
G_1

$$EW(G) = NW(G) - 1$$

$$\left\{\begin{array}{l}\text{edge}\\ \text{weight}\end{array}\right. \quad \Updownarrow \quad \left\{\begin{array}{l}\text{node}\\ \text{weight}\end{array}\right.$$

$$\Leftrightarrow \quad \#\text{ of edges in } G_1 = \left\{\begin{array}{l}\#\text{ of nodes}\\ \text{in } G_1 - 1\end{array}\right.$$

(this is true since G_1 is a tree)



$NW(y_4) = 4 = \#$ of nodes in y_4

$EW(y_4) = 3 = \#$ of edges in y_4

$\therefore$ $\boxed{EW(y_4) = NW(y_4) - 1}$

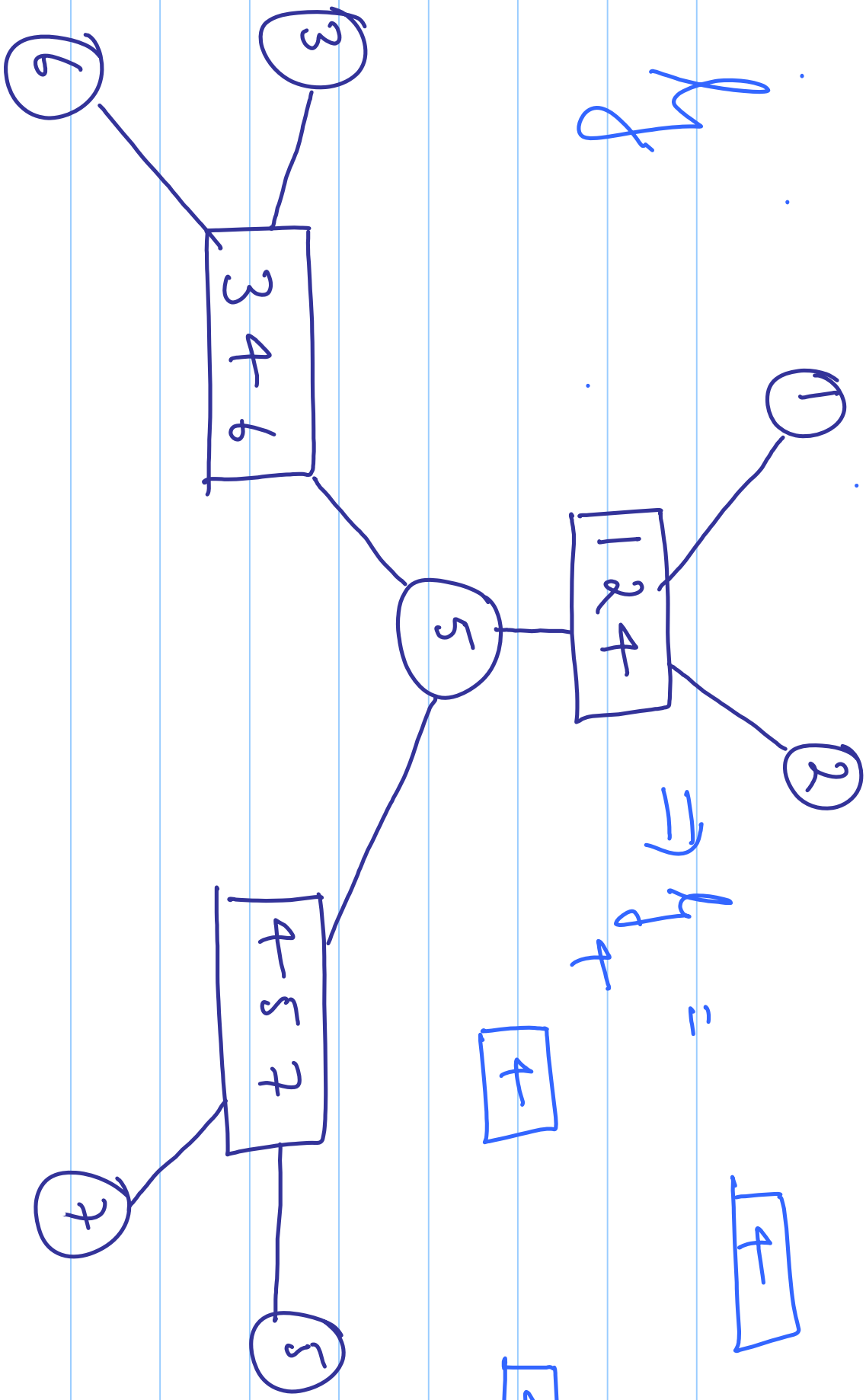
Pf. When G is a n jn tree we have

$$EW(G) = \sum_{i=1}^{n} EW(G_i)$$

$$= \sum_{i=1}^{n} \left(NW(G_i) - 1 \right)$$

$$= \sum_{i=1}^{n} NW(G_i) - n$$

$$\therefore \boxed{EW(G) = NW(G) - n} \quad (1)$$


6
3
3 4 6
1
1 2 4
5
2
4 5 7
7
5

When y is not a join tree, then at least one of the y_i will fail to be a tree (will be the union of ≥ 2 trees instead) and hence

$$EW(y_i) < NW(y_i) - 1$$

If we now proceeded to argue as when deriving (1) we would end up with

$$EW(f) < NW(f) - n$$

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

$$\alpha(x,w) = \sum_{y,z} f(x\,y\,w)\, g(x,z)$$

$$\beta(y) = \sum_{x,w,z} f(x,y,w)\, g(x,z)$$

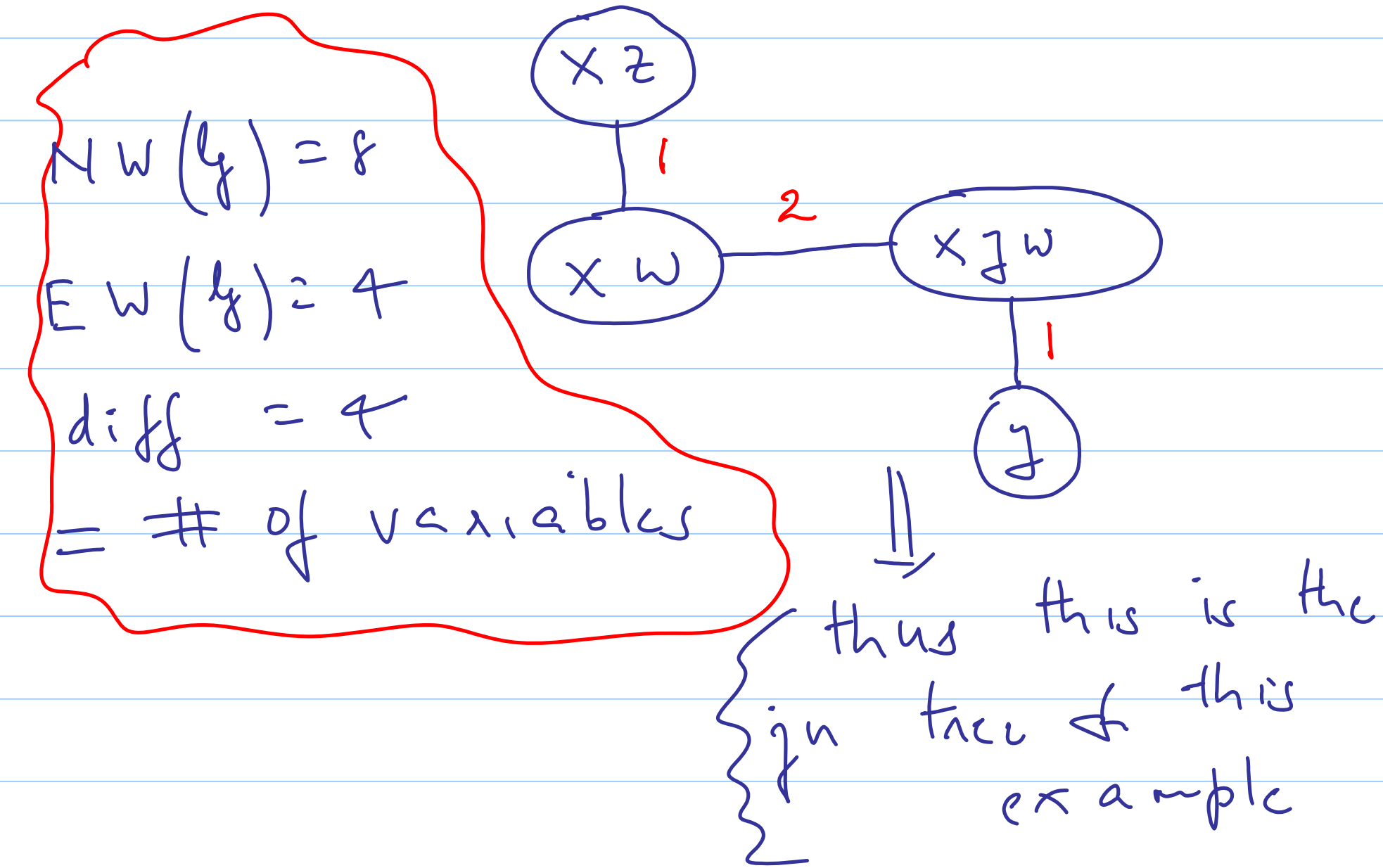
objective functions

LD	LK
$\{y\}$	1

$w\,x\,y\,z \in A$

$|A| = 2$

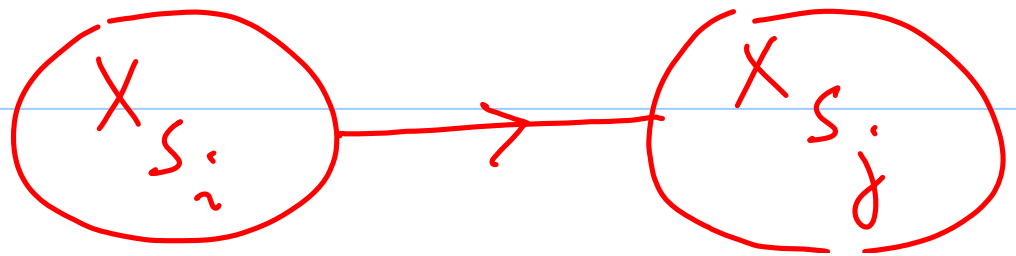
LD	LK
$\{x\,y\,w\}$	$f(x\,y\,w)$
$\{x\,z\}$	$g(x\,z)$
$\{x\,w\}$	1


NW(y) = 8
EW(y) = 4
diff = 4
= # of variables
xz
1
xw
2
xyw
1
y
thus this is the
same in face of this
example

ASIDE

It turns out that an edge connecting nodes X_{s_i} and X_{s_j} incurs a cost =

$$q_{s_i} + q_{s_j} - q_{s_i \cap s_j} \tag{2}$$



Eg 2 (The 8-dimensional Walsh – Hadamard Transform)

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

LD

$\{x_1\ x_2\ x_3\}$

$\{x_1\ x_4\}$

$\{x_2\ x_5\}$

$\{x_3\ x_6\}$

$\{x_4\ x_5\ x_6\}$

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

$\{x_1\ x_2\ x_3\}$

$\{x_1\ x_4\}$

$\{x_2\ x_5\}$

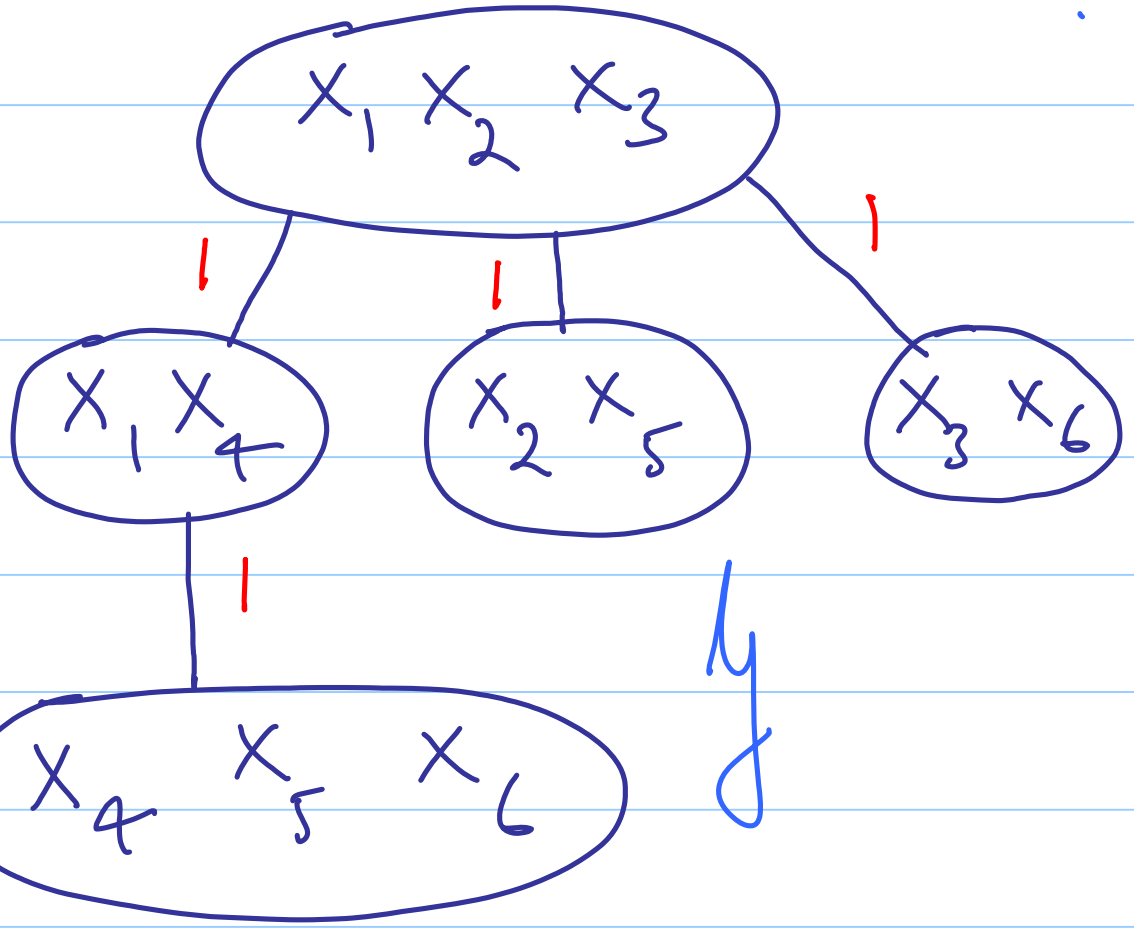
$\{x_3\ x_6\}$

$\{x_4\ x_5\ x_6\}$

$NW(y) = 12$

$EW(y) = 4$

$\Big\}$ diff = 8 ? # variables!



Lec 26 — Message passing on the Jn Tree

Recap

* completed proof showing that a jn tree (i.e. if 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

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

$$p(x_4 | \underline{y}) \propto \sum_{\sim x_4} p(\underline{y} | \underline{x}) \chi_R(\underline{x})$$

$$= \sum_{\sim x_4} \prod_{j=1}^{7} p(y_j | x_j) \chi_{124}(x_1 x_2 x_4)$$

$$\chi_{346}(x_3 x_4 x_6)$$

$$\chi_{457}(x_4 x_5 x_7)$$

LD

$\{x_j\}_{j=1}^{7}$

$\{x_1 x_2 x_4\}$

$\{x_3 x_4 x_6\}$

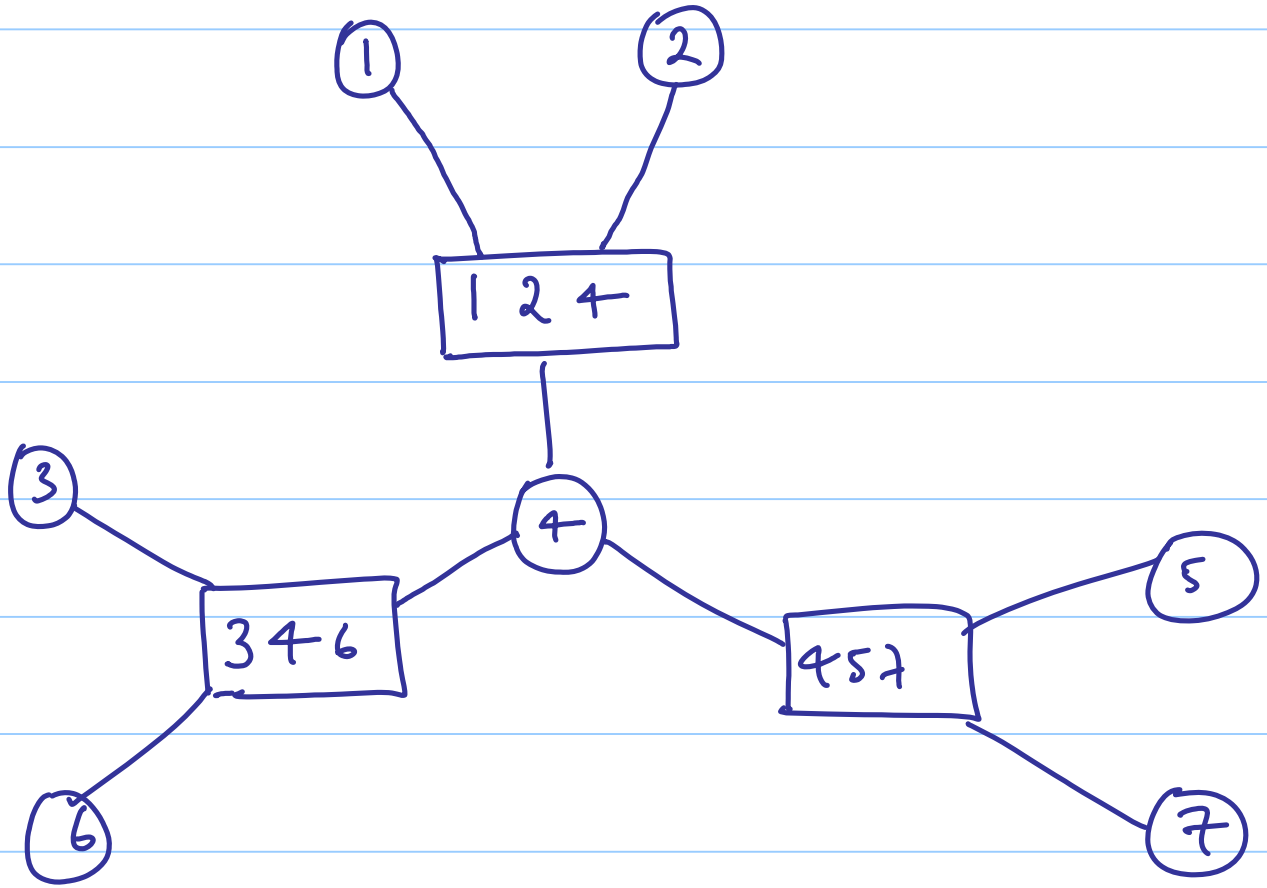
LK

$p(y_j | x_j)$

$\chi_{124}(x_1 x_2 x_4)$

$\chi_{346}(x_3 x_4 x_6)$

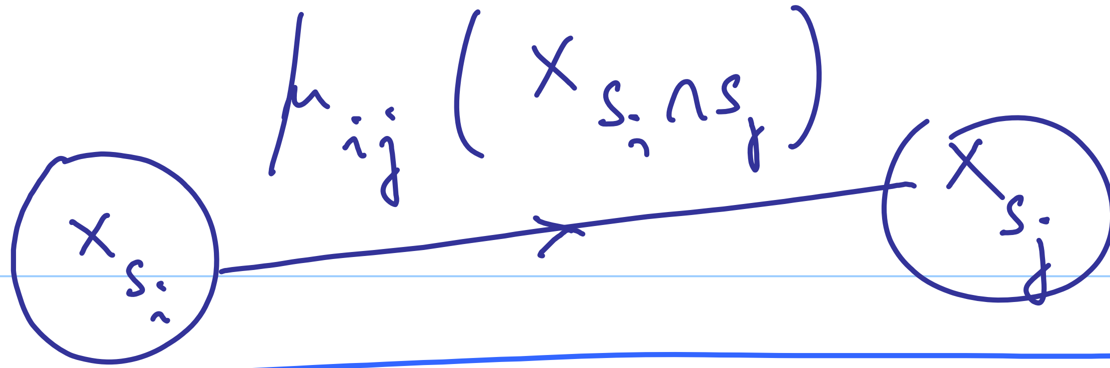
$\{x_4\ x_5\ x_7\}$ $\chi_{457}(x_4\ x_5\ x_7)$



346
124
457
6
3
1
4
2
7
5

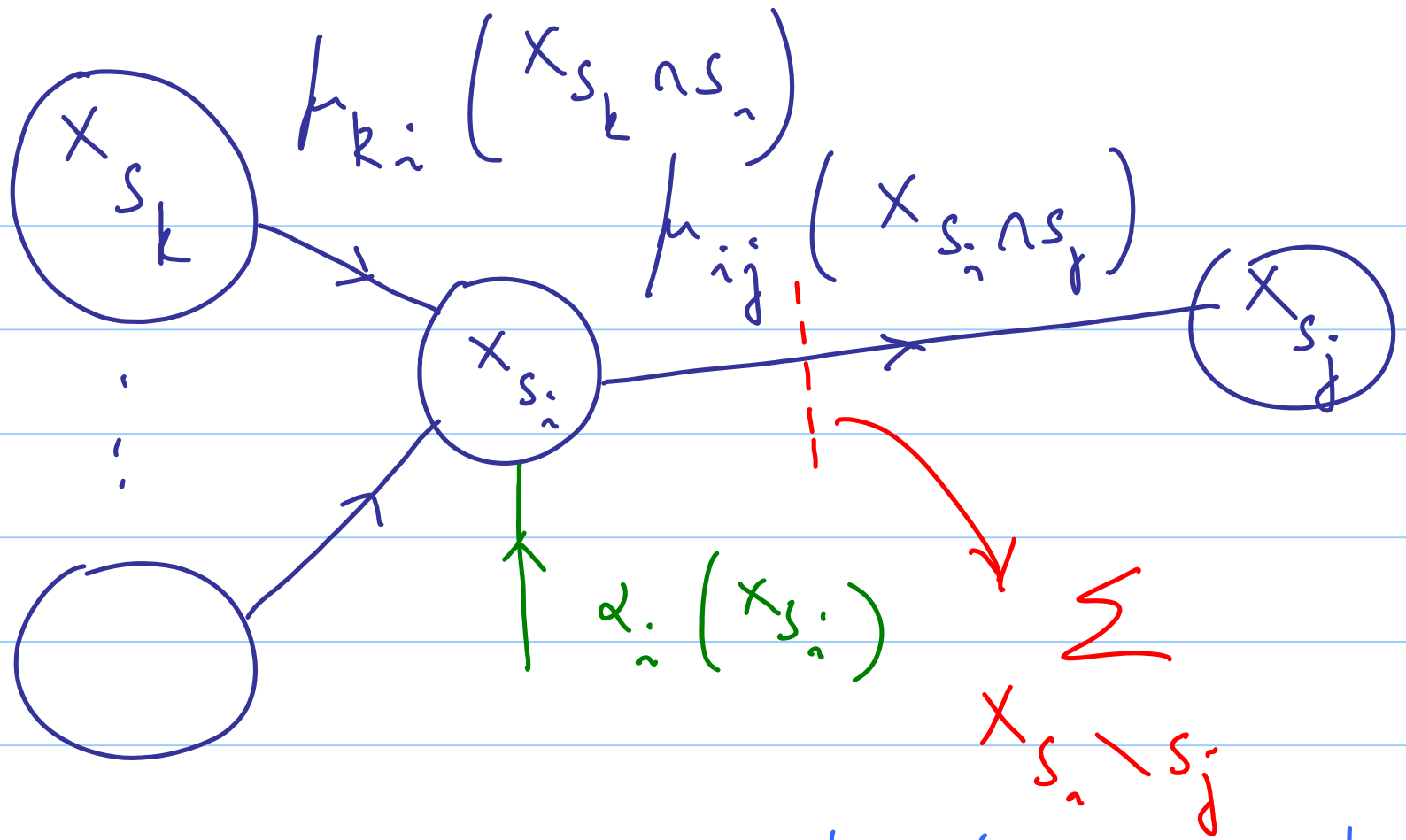
Step 1 — in solving (where possible) the MPF problem is to organize the local domains into a jn tree.

Step 2 — Pass messages along the edges of the jn tree in accordance with some schedule



$$\mu_{ij}(x_{s_i \cap s_j}) = \sum_{x_{s_i \setminus s_j}} \alpha_i(x_{s_i}) \prod_{\substack{k \in N_i \\ k \neq j}} \mu_{ki}(x_{s_k} \cap x_{s_i})$$

N_i = set of neighbors of x_{s_i}



Note: the message $\mu_{ij}(x_{s_i \cap s_j})$ passed on by a leaf node is simply the marginalization of its local kernel.

Eg 3 $[7, 4, 2]$ Code

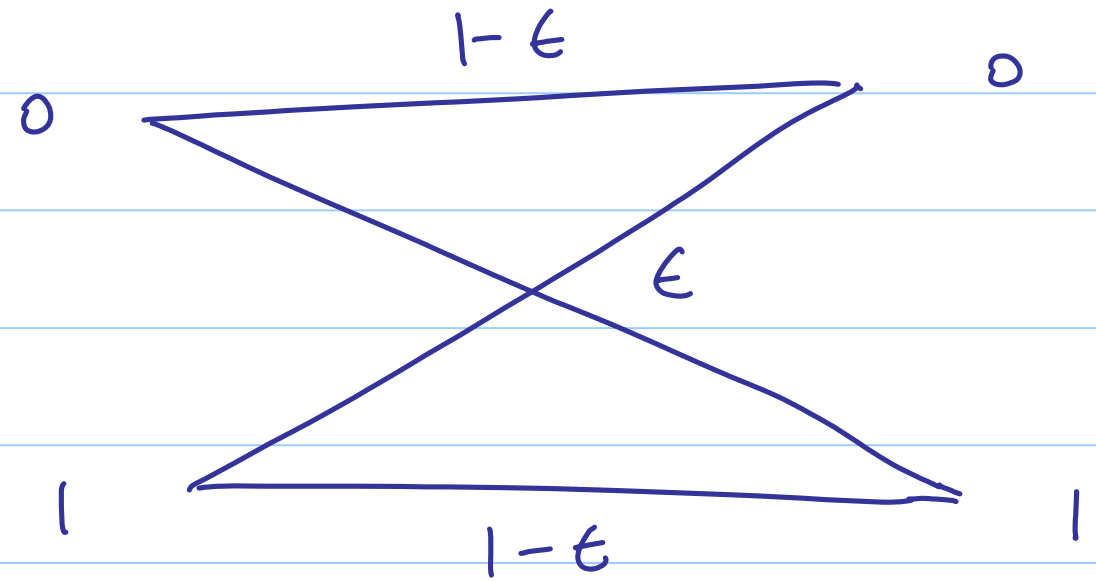
$$H = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

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



$$\underline{y} = \overset{1\ 2\ 3\ 4\ 5\ 6\ 7}{(0\ 1\ 1\ 0\ 1\ 0\ 0)}$$

$$p(y_1 | x_1) \Rightarrow \begin{bmatrix} p(y_1 | x_1 = 0) \\ p(y_1 | x_1 = 1) \end{bmatrix}$$

This is typical in that it is a vector representation of the fn.

$$\underline{y} = \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{matrix}$$

$$p(y_1 | x_1) \Rightarrow \begin{bmatrix} p(y_1 | x_1 = 0) \\ p(y_1 | x_1 = 1) \end{bmatrix} = \begin{bmatrix} 1-\epsilon \\ \epsilon \end{bmatrix}$$

$$\Downarrow$$

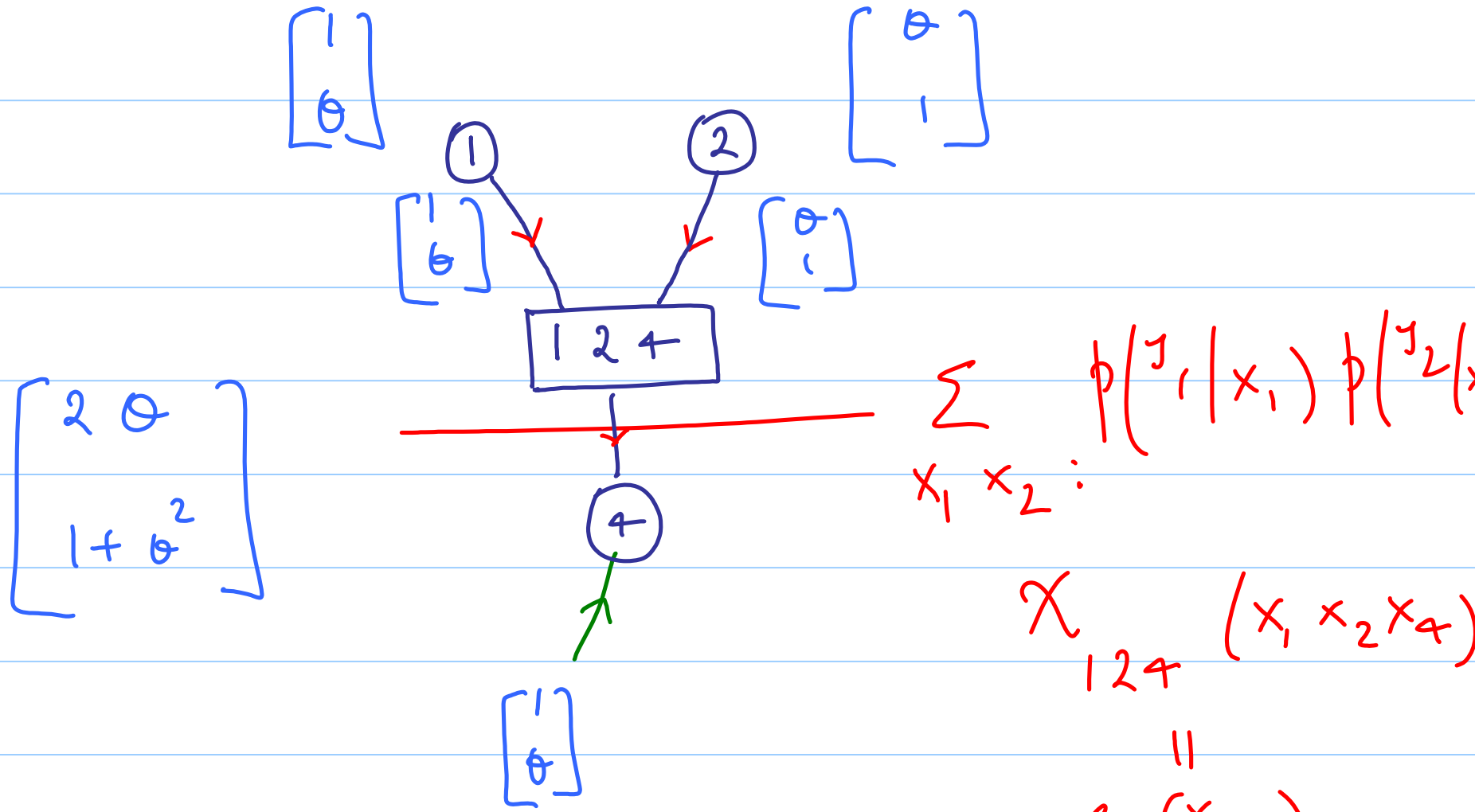
scaling by $(1-\epsilon)$ yields:

$$\begin{bmatrix} 1 \\ \theta \end{bmatrix}$$

where $\theta = \left(\frac{\epsilon}{1-\epsilon}\right)$

346
124
457
6
3
1
2
4
7
5

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 join tree towards the corresponding local domain.



$$\sum_{x_1 x_2} p(y_1|x_1)\, p(y_2|x_2)$$

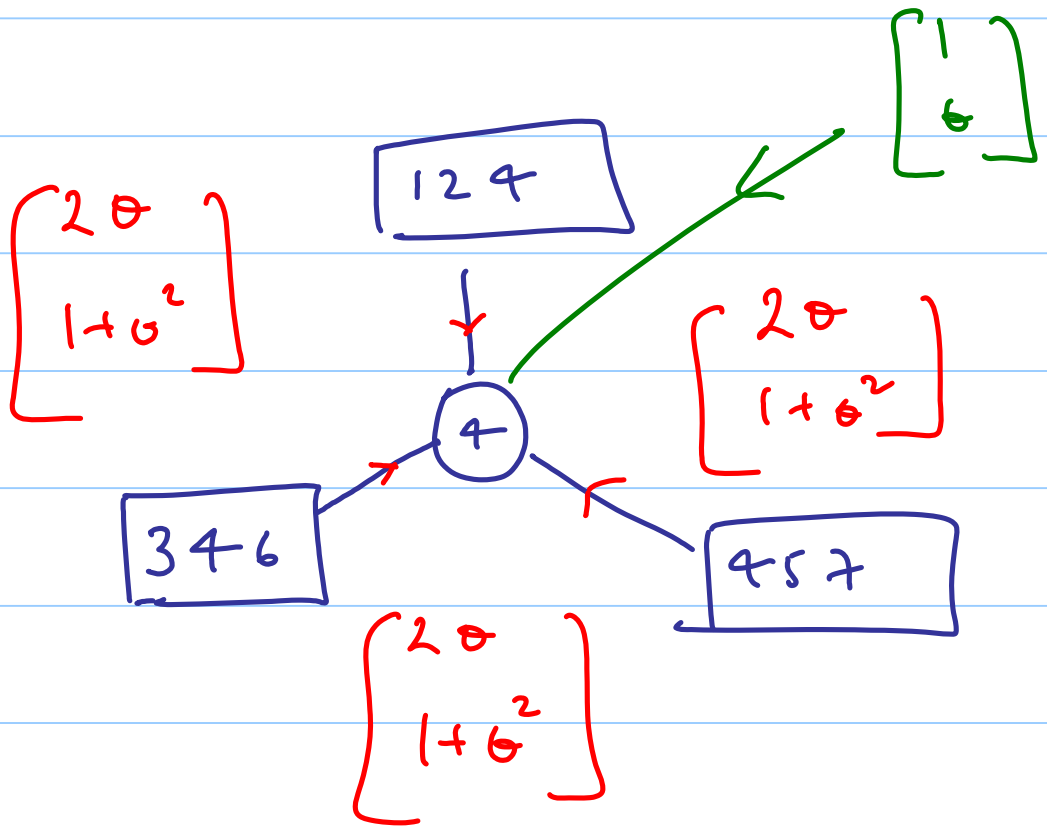
$$\chi_{124}(x_1 x_2 x_4)$$

$$\| $$

$$g(x_4)$$

$$\beta_4(x_4) = \begin{bmatrix} (2\theta)^3 \\ \theta(1+\theta^2)^3 \end{bmatrix} \cong \begin{bmatrix} 8\theta^3 \\ \theta \end{bmatrix}$$

$$\theta = \frac{\epsilon}{1-\epsilon}$$

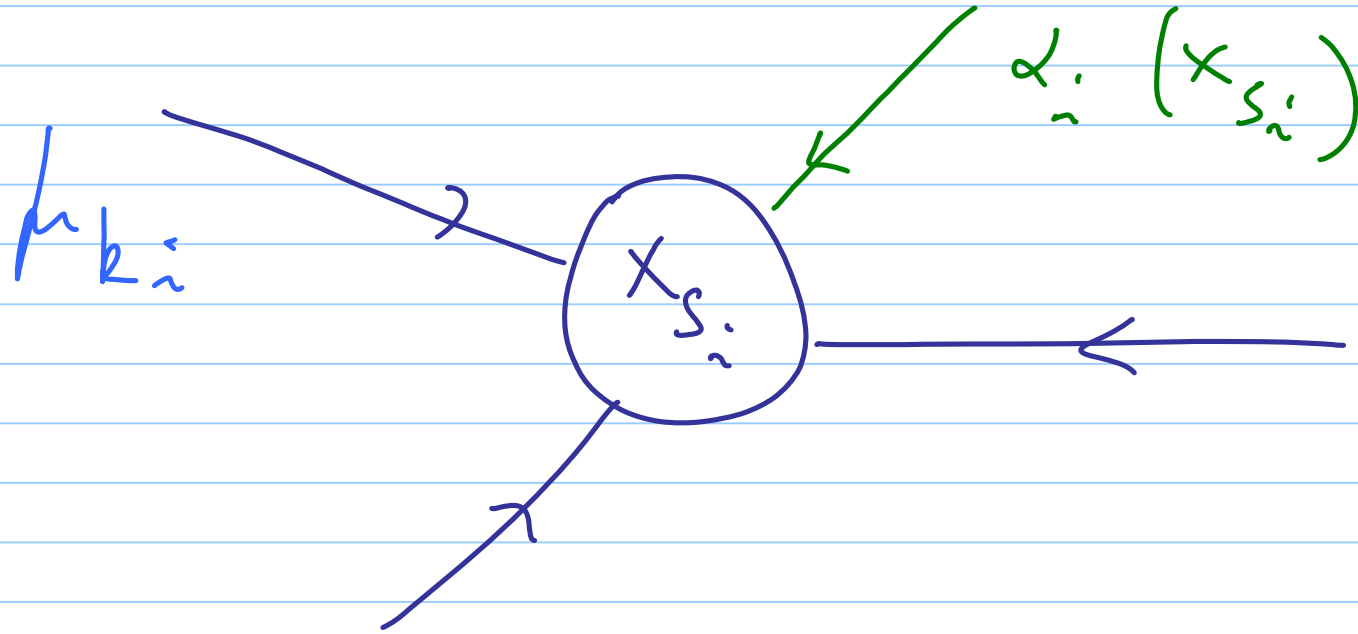


Lec 27 — GDL Approach to Decoding Convolutional Codes

Recap

- final example of in-the 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

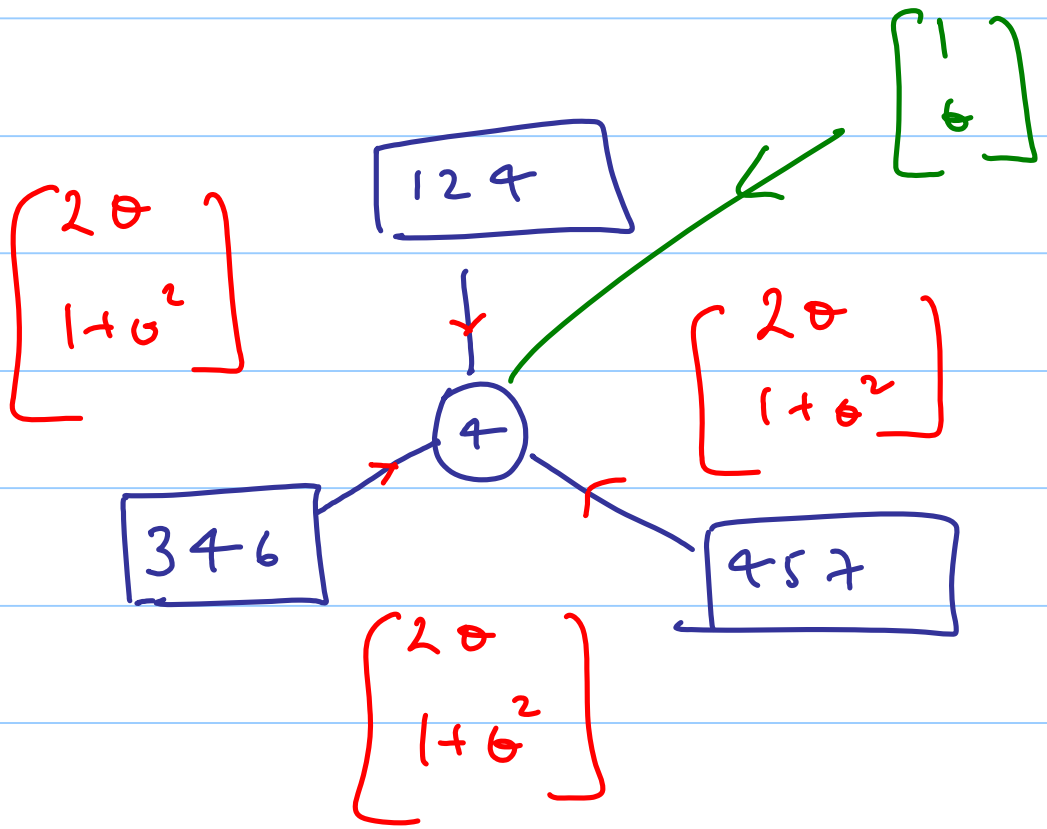


At this stage the node computes its objective fn. simply by computing the product of the incoming messages and the local kernel.

$$\beta_j(x_{s_j}) = \alpha_j(x_{s_j}) \prod_{k \in N_j} h_{kj}(x_{s_k \cap s_j})$$

$$\beta_4(x_4) = \begin{bmatrix} (2\theta)^3 \\ \theta(1+\theta^2)^3 \end{bmatrix} \approx \begin{bmatrix} 8\theta^3 \\ \theta \end{bmatrix}$$

$\theta = \frac{\epsilon}{1-\epsilon}$



$$\begin{bmatrix} 2\theta \\ 1+\theta^2 \end{bmatrix} \odot \begin{bmatrix} 2\theta \\ 1+\theta^2 \end{bmatrix} \odot \begin{bmatrix} 2\theta \\ 1+\theta^2 \end{bmatrix} \odot \begin{bmatrix} 1 \\ \theta \end{bmatrix}$$

Schur product

$$= \begin{bmatrix} 8\theta^3 \\ \theta(1+\theta^2)^3 \end{bmatrix} \propto \begin{bmatrix} 8\theta^3 \\ \theta \end{bmatrix} \begin{matrix} x_4 = 0 \\ x_4 = 1 \end{matrix}$$

$$\beta_4(x_4) \propto p(x_4 | \underline{y})$$

Since $\theta << 1$ (typically) the ML code-symbol decoder will decode x_t to $= 1$

From the example and the in 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

$$F_i(x_i) = \max_{\sim x_i} \prod_{i=1}^{7} \phi(J_i | u_i) \cdot$$

$\sim x_i$

max

$x_1 \; -- \; x_{i-1}$

$x_{i+1} \; -- \; x_7$

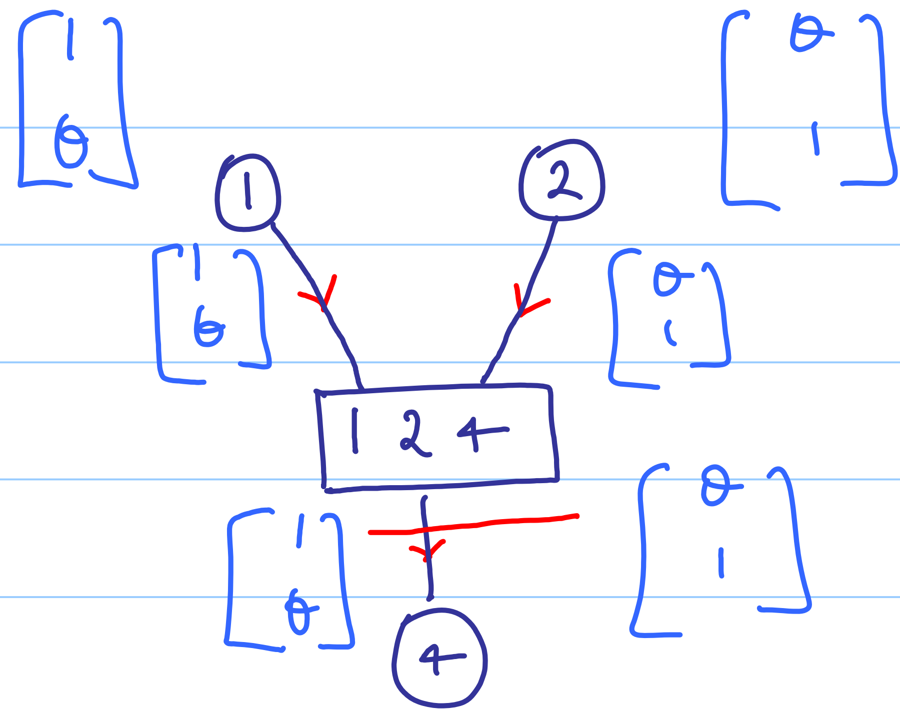
$$\chi_{124}(u_1 u_2 u_4) \cdot$$

$$\chi_{346}(u_3 u_4 u_6) \cdot$$

$$\chi_{457}(u_4 u_5 u_7)$$

Eg decode $[7, 4, 2]$ code $\underline{J} = (0\,1\,1\,0\,1\,0\,0)$

346
124
457
6
3
1
2
4
7
5



$$g(x_4) = \max_{x_1, x_2} \; p(y_1 | x_1)\, p(y_2 | x_2)\, \chi_{124}(x_1 x_2 x_4)$$

$$\begin{bmatrix} 1 \\ \theta \end{bmatrix}, \begin{bmatrix} \theta \\ 1 \end{bmatrix}$$

$$g(0) = \max\{1.0,\ 0.1\} = 0$$

$$g(1) = \max\{1.1,\ 0.0\} = 1$$

LD

$\{x_j\}_{j=1}^{7}$

$p(j \mid x_j)$

$\{x_1\ x_2\ x_4\}$ $\quad x_{124}\ (x_1\ x_2\ x_4)$

$\{x_3\ x_4\ x_6\}$ $\quad x_{346}\ (x_3\ x_4\ x_6)$

$\therefore$ even HHC ML code word decoder

$\therefore$ $F_4(X_4) =$

$X_4 = 0$: ϕ, ϕ

$X_4 = 1$: ϕ

$\{$ decodes $x_4 = 1$

346
3, 6,
346
124
1
2
1, 2,
124
457
4, 5
457
3
6
4
7
5

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

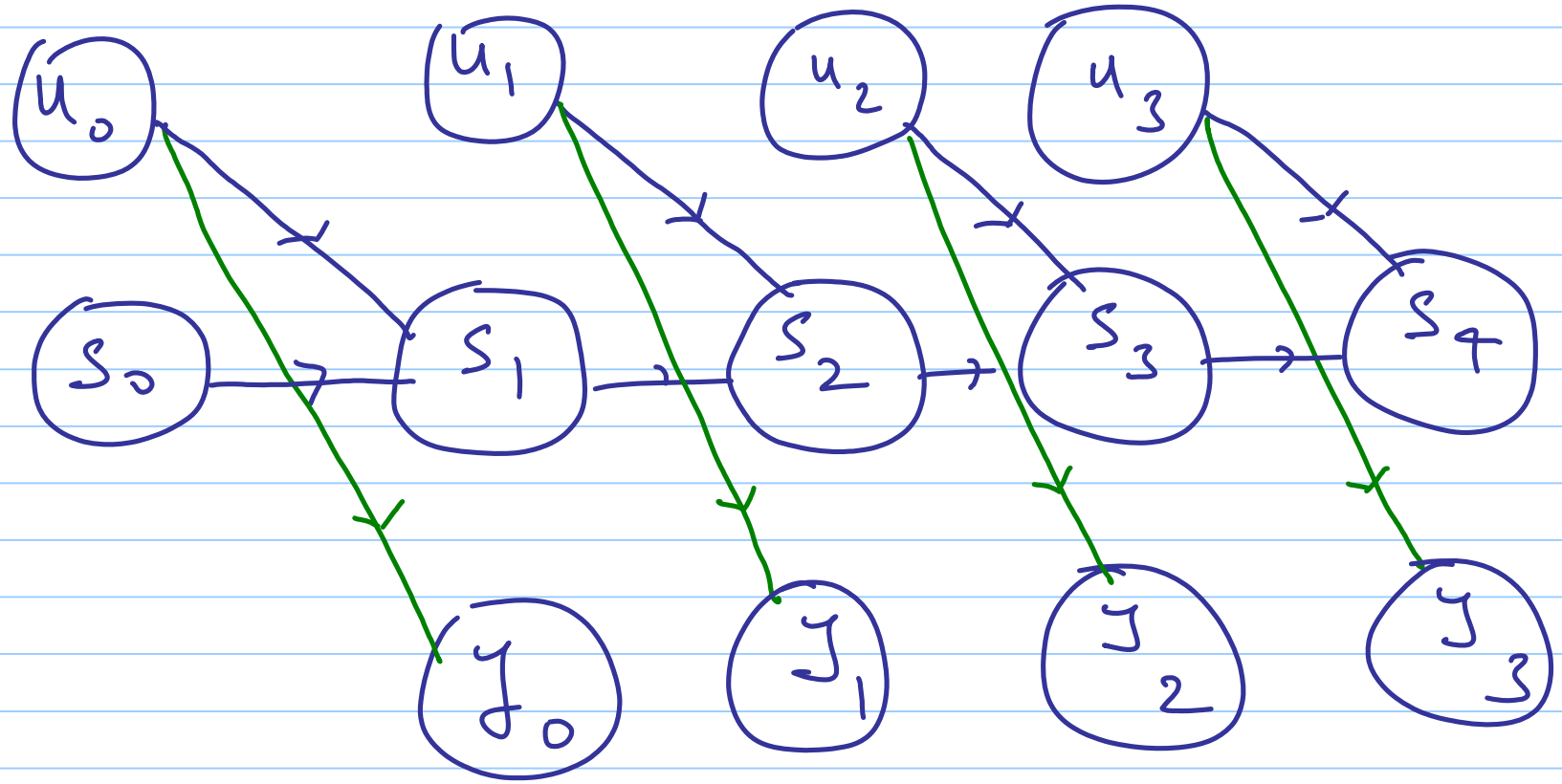
$$= \sum_{\substack{\text{edges} \\ E_{ij}}} \left(q_{S_i} + q_{S_j} - q_{S_i \cap S_j} \right)$$

(additions & multiplications)

Turns out that the complexity involved in computing the objective fn. at all nodes is bdd. above by

$$+ \left(\begin{array}{c} \text{single - vertex} \\ \text{complexity} \end{array} \right).$$

ML code-symbol decoding of a convolutional code



We consider the same convolutional as before with the difference that we are now interleaved in ML code-symbol decoding.

$$p(u_k \mid \underline{y}) \propto p(u_k, \underline{y}) = \sum_{\sim u_k} p\left(\{u_k\}_{k=0}^{3}, \underline{y}\right)$$

$$= \sum_{\sim u_k} \sum_{\underline{s}} p\left(\{s_k\}_{k=0}^{4}, \{u_k\}_{k=0}^{3}, \{y_k\}_{k=0}^{3}\right)$$

$$= \sum_{u_k} \sum_{s} p(s_0) \prod_{k=0}^{3} p(u_k)\, p(s_{k+1} \mid s_k, u_k)\, p(y_k \mid s_k, u_k)$$

Lec 28

ML Code - Symbol Decoding of the convolutional code (BCJR Algorithm)

Recap

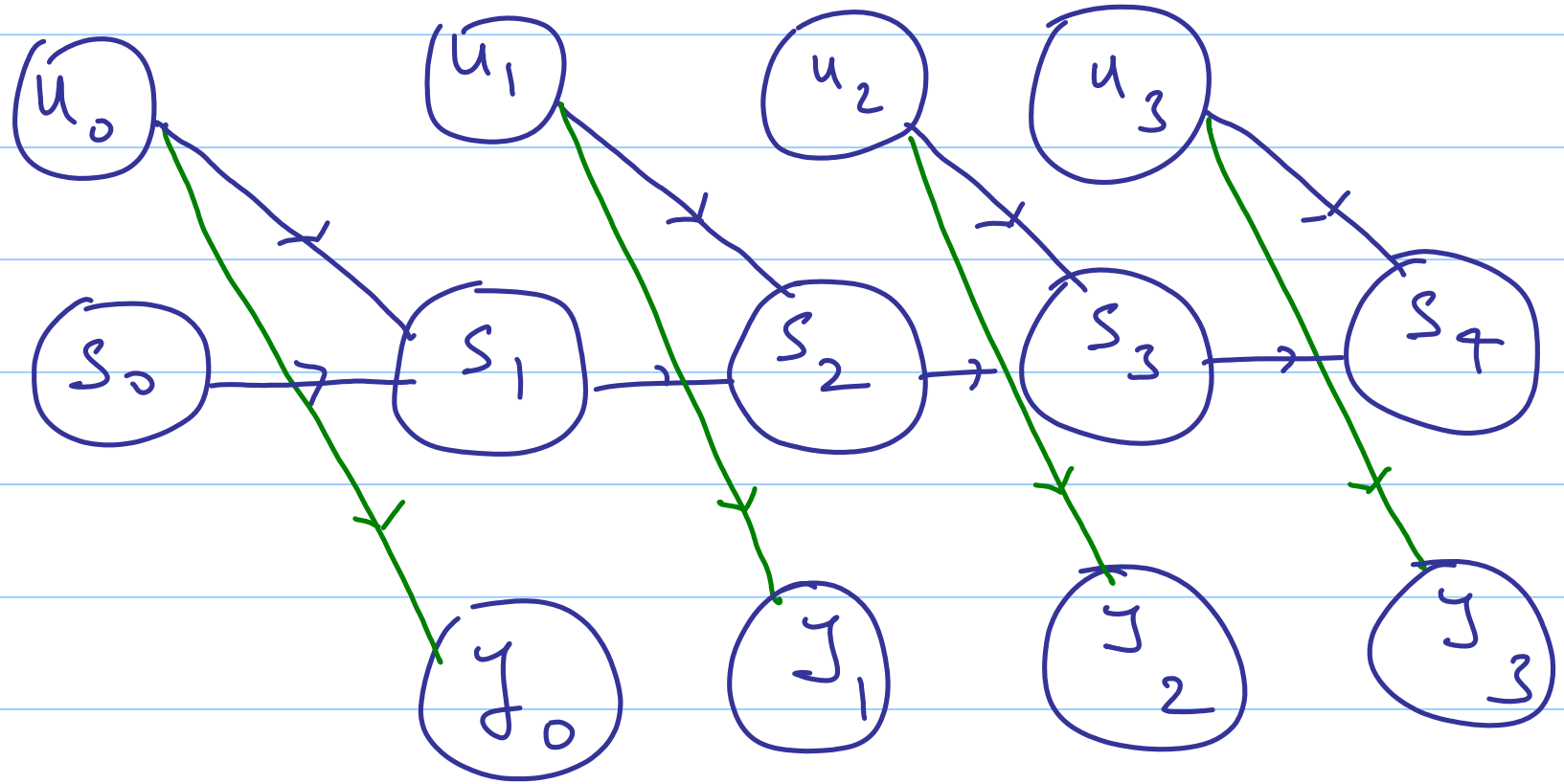
* formalizing the final step in the GDL - computing the objective function

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 GDL
  - single-vertex
  - bound for the all-vertex version of the GDL

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.

$$p(u_k \mid \underline{y}) \propto p(u_k, \underline{y}) = \sum_{\sim u_k} p\left(\{u_k\}_{k=0}^{3}, \underline{y}\right)$$

$$= \sum_{\sim u_k} \sum_{\underline{s}} p\left(\{s_k\}_{k=0}^{4}, \{u_k\}_{k=0}^{3}, \{y_k\}_{k=0}^{3}\right)$$

$$= \sum_{u_k} \sum_{s} p(s_0) \prod_{k=0}^{3} p(u_k)\, p(s_{k+1} \mid s_k\, u_k)\, p(j_k \mid s_k\, u_k)$$

LD	LK	
$\{u_i\}$	$p(u_i)$	$0 \le i \le 3$
$\{s_0\}$	$p(s_0)$	
$\{s_{i+1}\ s_i\ u_i\}$	$p(s_{i+1} \mid s_i\ u_i)$	$0 \le i \le 3$
$\{s_i\ u_i\}$	$p(j_i \mid s_i\ u_i)$	$0 \le i \le 3$

s_0
$s_0 u_0$
u_0
$s_1 s_0 u_0$
$\Sigma_{s_0 u_0}$
$f_1(s_1)$
$s_1 u_1$
u_1
$f_2(s_2)$
$s_2 s_1 u_1$
$s_2 u_2$
$\Sigma_{s_1 u_1}$
u_2
$s_3 s_2 u_2$
$s_4 s_3 u_3$
$s_3 u_3$
u_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

Eg $G(D) = \left[1 + D + D^2 \quad 1 + D^2 \right]$

$f_1(s_1)$
11
01
10
00
s_1
11
10
01
00
s_2
$f_2(s_2)$

$$f_2(s_2) = \sum_{s_1 u_1} f_1(s_1)\, p(u_1)\, p(y_1 \mid s_1 u_1)\, p(s_2 \mid s_1 u_1)$$

$$f_2(s_2) = \sum_{s_1} f_1(s_1) \left\{ \sum_{u_1} p(u_1) \cdot p(y_1 \mid s_1 u_1) \cdot \right.$$

$p(u_1) = 1/2$

$$\left. \cdot\, p(s_2 \mid s_1 u_1) \right\}$$

$p(s_2 \mid s_1 u_1)$ is $\{0,1\}$-valued fn.

$\uparrow (s_2, s_1)$

$\uparrow (s_2, s_1)$ $\quad -f_1(s_1)$

$f_2(s_2)$

$$\begin{bmatrix} f_2(00) \\ f_2(10) \\ f_2(01) \\ f_2(11) \end{bmatrix} = \begin{bmatrix} * & 0 & * & 0 \\ * & 0 & * & 0 \\ 0 & * & 0 & * \\ 0 & * & 0 & * \end{bmatrix} \begin{bmatrix} f_1(00) \\ f_1(10) \\ f_1(01) \\ f_1(11) \end{bmatrix}$$

s_0
$s_0 u_0$
u_0
$s_1 s_0 u_0$
$s_1 u_1$
u_1
$s_2 s_1 u_1$
$s_2 u_2$
u_2
$s_3 s_2 u_2$
$s_3 u_3$
u_3
$s_4 s_3 u_3$
s_4
u_3
$d_4(s_4)$
$= 1$
(henc)
$d_3(s_3)$

$$g_3(s_3) = \sum_{u_3} p(y_3 | s_3 u_3)\, p(u_3) \cdot \sum_{s_4} p(s_4 | s_3 u_3)\, g_4(s_4)$$

$$= \sum_{s_4} g_4(s_4) \left\{ \sum_{u_3} p(u_3) \cdot p(y_3 | s_3 u_3) \cdot p(s_4 | s_3 u_3) \right\}$$

$$\Delta(s_3, s_4)$$

$$g_3(s_3) \quad \Delta(s_3, s_4) \quad g_4(s_4)$$

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

Conclusion: { both forward and backward recursions correspond to mx multiplication

s_0
$s_0 u_0$
u_0
$s_1 s_0 u_0$
$s_1 u_1$
u_1
$s_2 s_1 u_1$
$s_2 u_2$
u_2
$s_3 s_2 u_2$
$s_3 u_3$
u_3
$s_4 s_3 u_3$
$q_3(s_3)$
$q_2(s_2)$
$\sum_{u_3}$
$\sum_{s_3}$
s_2

$$p(u_k \mid \underline{y}) = p(u_2) \sum_{s_2} p(y_2 \mid s_2 u_2) f_2(s_2) \sum_{s_3} g_3(s_3) p(s_3 \mid s_2 u_2)$$

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 GDL: 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 $\mathcal{C}$ be a $[n, k, d]$ linear block code. Typically when n is large the codes of practical interest have

$$\frac{k}{n} = R, \qquad 0 < R < 1$$

and

$$\frac{d_{min}}{n} = \delta, \qquad 0 < \delta < 1.$$

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

$(n-k \times n)$

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.

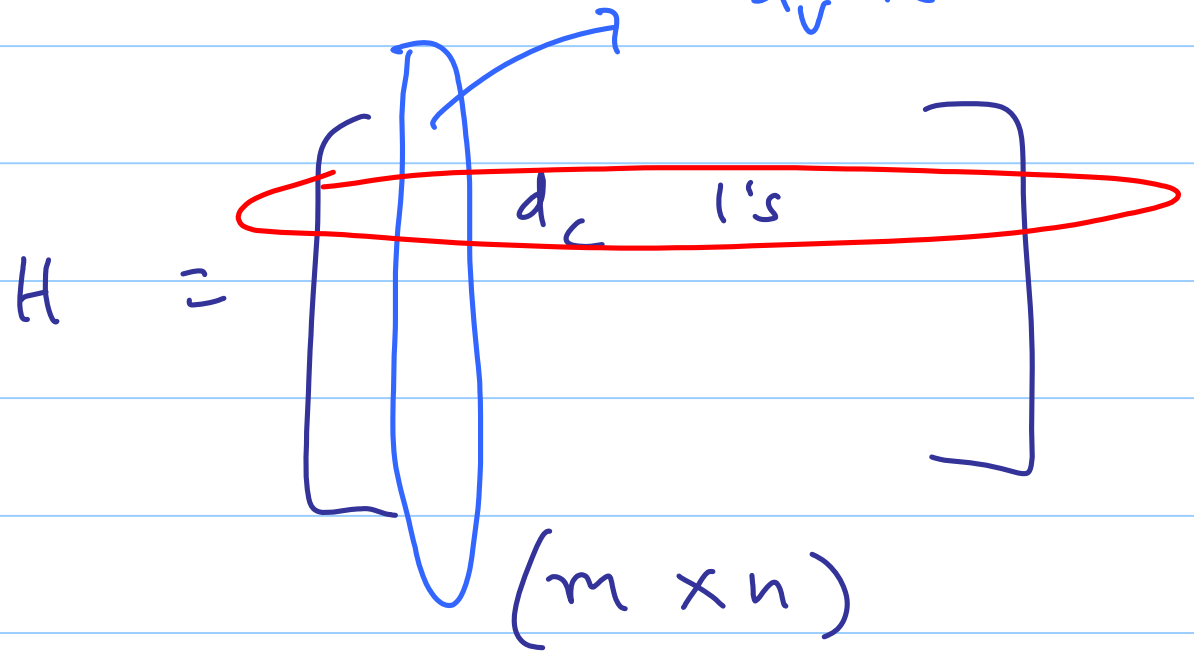
However in the case of an LDPC code the # of 1's in H is of order n.

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_v, 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.



$\therefore m d_c = n d_v$

$\therefore \boxed{\frac{m}{n} = \frac{d_v}{d_c}}$

$rk(H) \leq m$

(2)

$\therefore \dim\left(\eta(H)\right) \geq n-m$

$\therefore \dim(Q) \geq n-m$

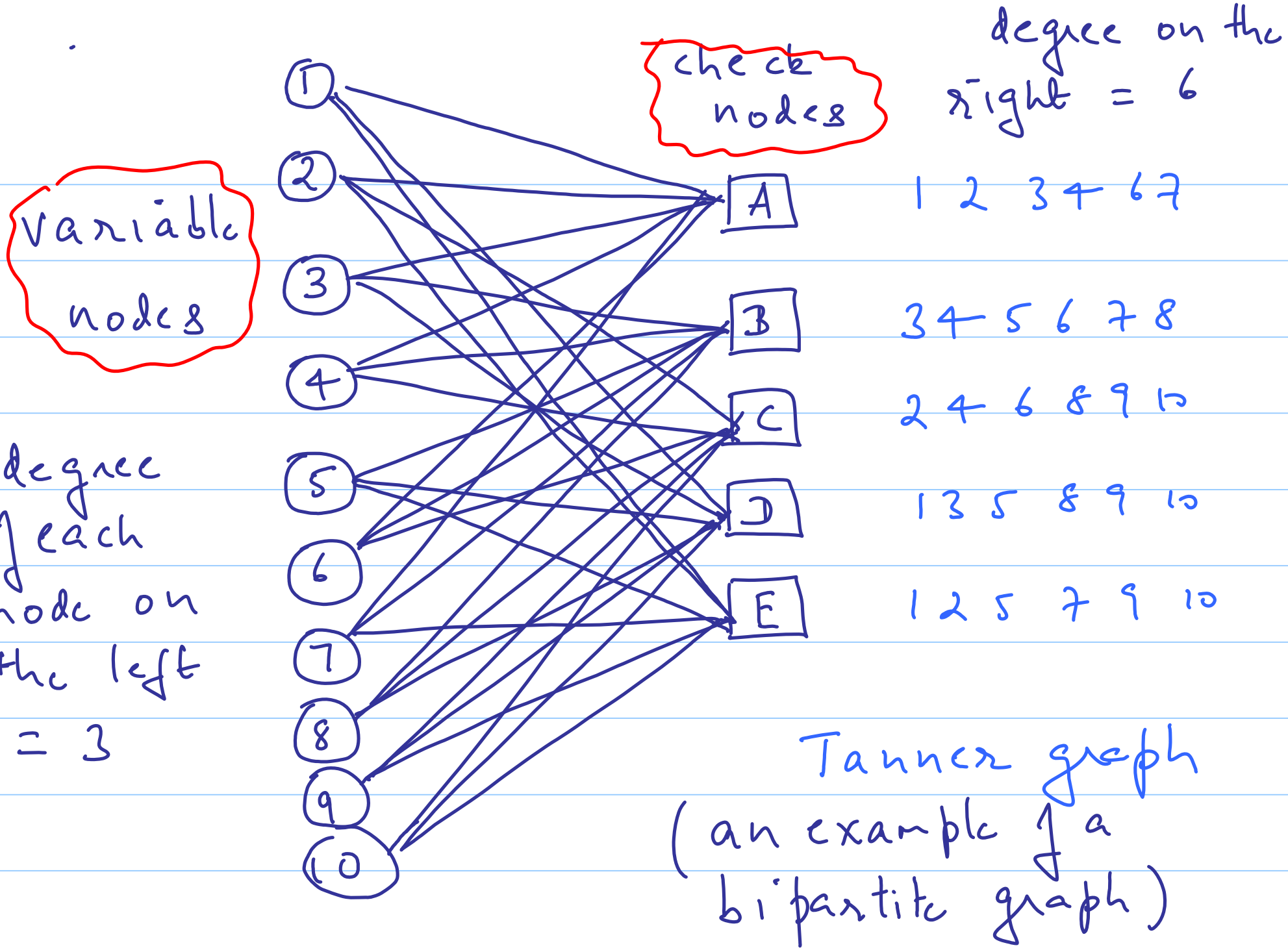
$\therefore \dim(P) \geq n-m$

$\therefore$ (1)

$\therefore$

$\therefore$ (3)

from (1) and (2)



degree on the right = 6

1 2 3 4 6 7

3 4 5 6 7 8

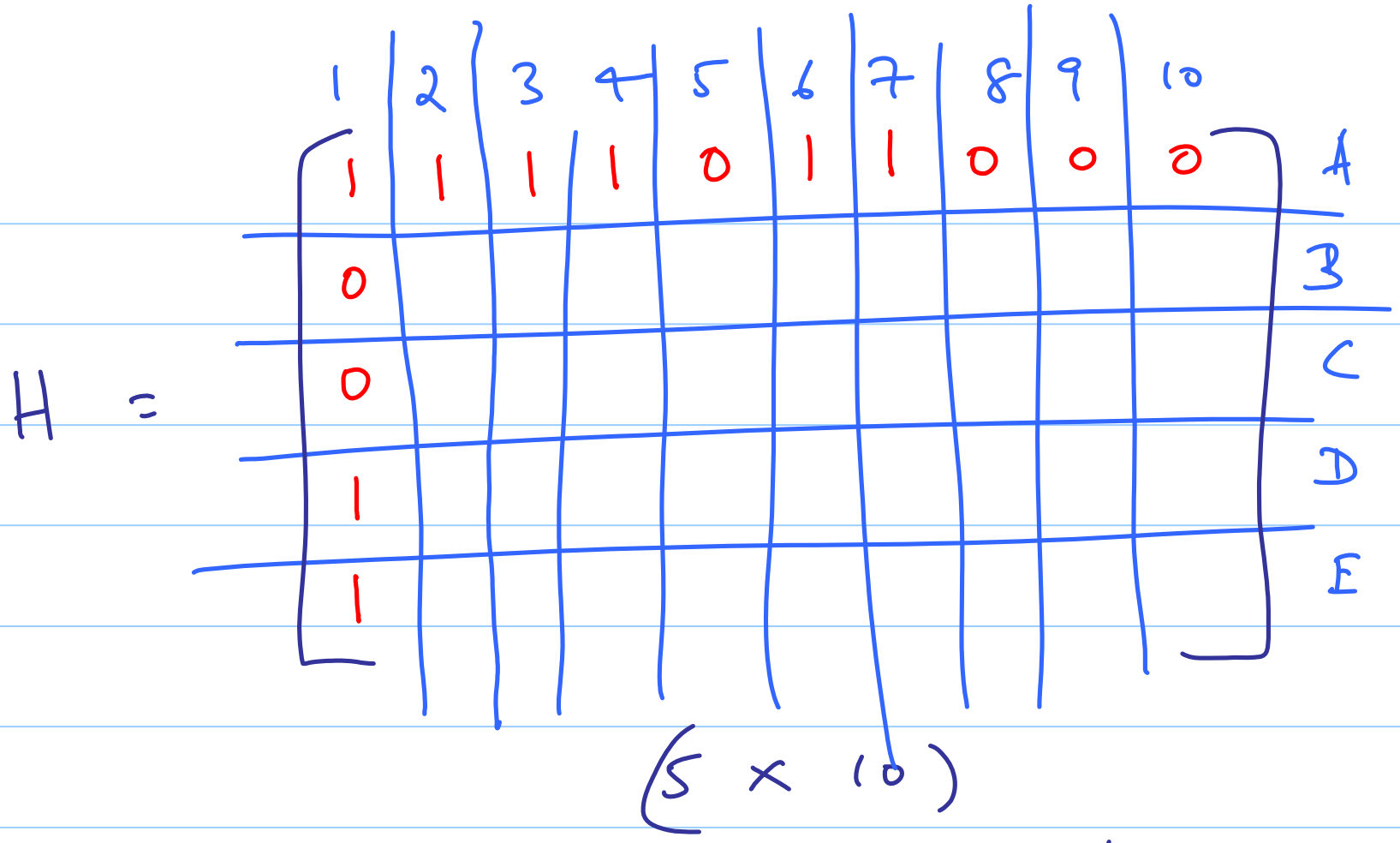
2 4 6 8 9 10

1 3 5 8 9 10

1 2 5 7 9 10

degree of each node on the left = 3

Tanner graph (an example of a bipartite graph)



this is an instance of a $(d_v = 3, d_c = 6)$ - regular code

$$\therefore \quad \frac{k}{n} \geq 1 - \frac{d_v}{d_c} = 1 - \frac{3}{6} = \frac{1}{2}$$

$$\therefore \quad \boxed{\text{rate}(R) \geq \frac{1}{2}}$$

designed rate

Note that the (d_v, d_c) constraint implies that the total # of 1's in the p.c. $mx = n d_v = m d_c$

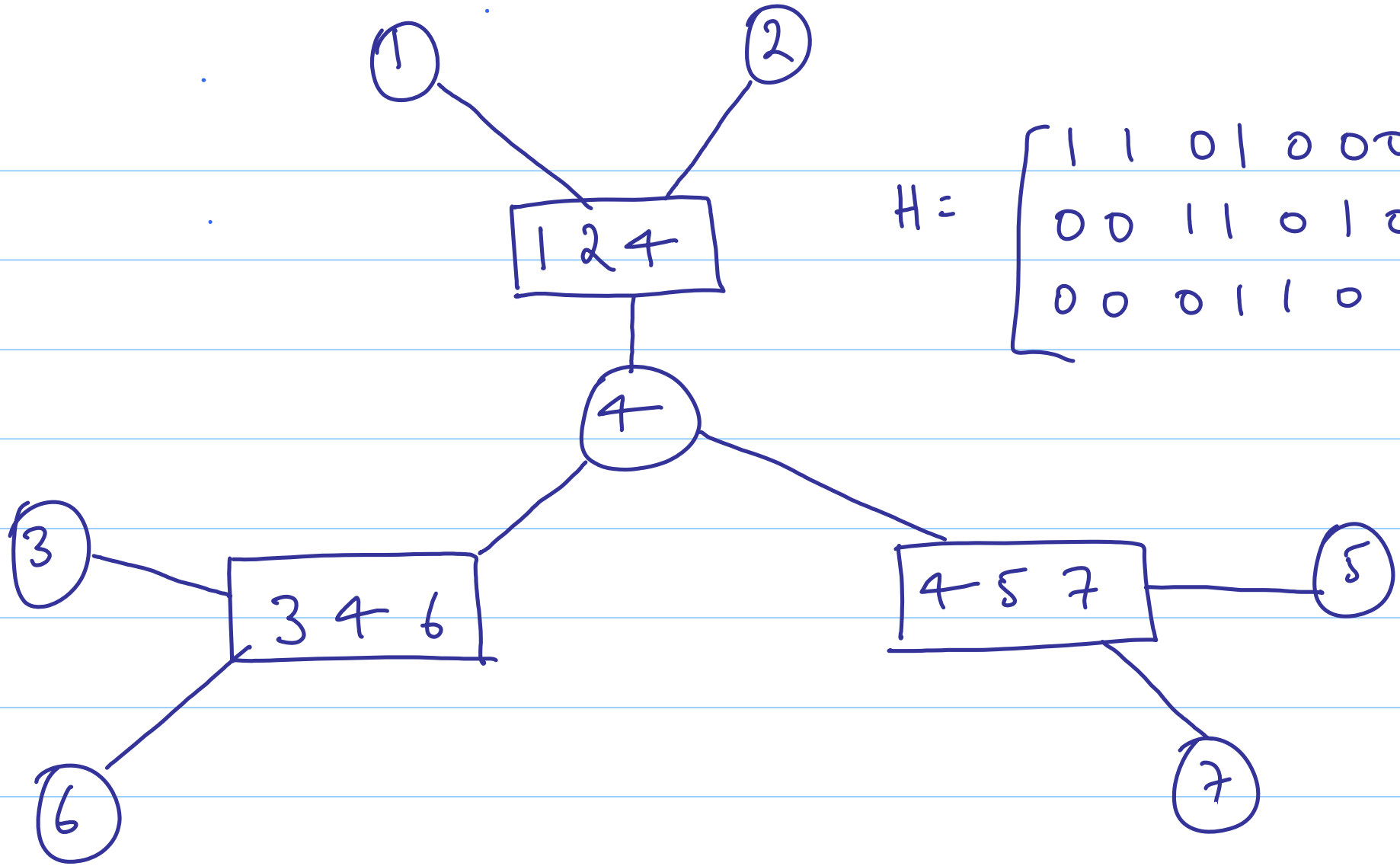
and hence is of order n

(since (d_v, d_c) are fixed and

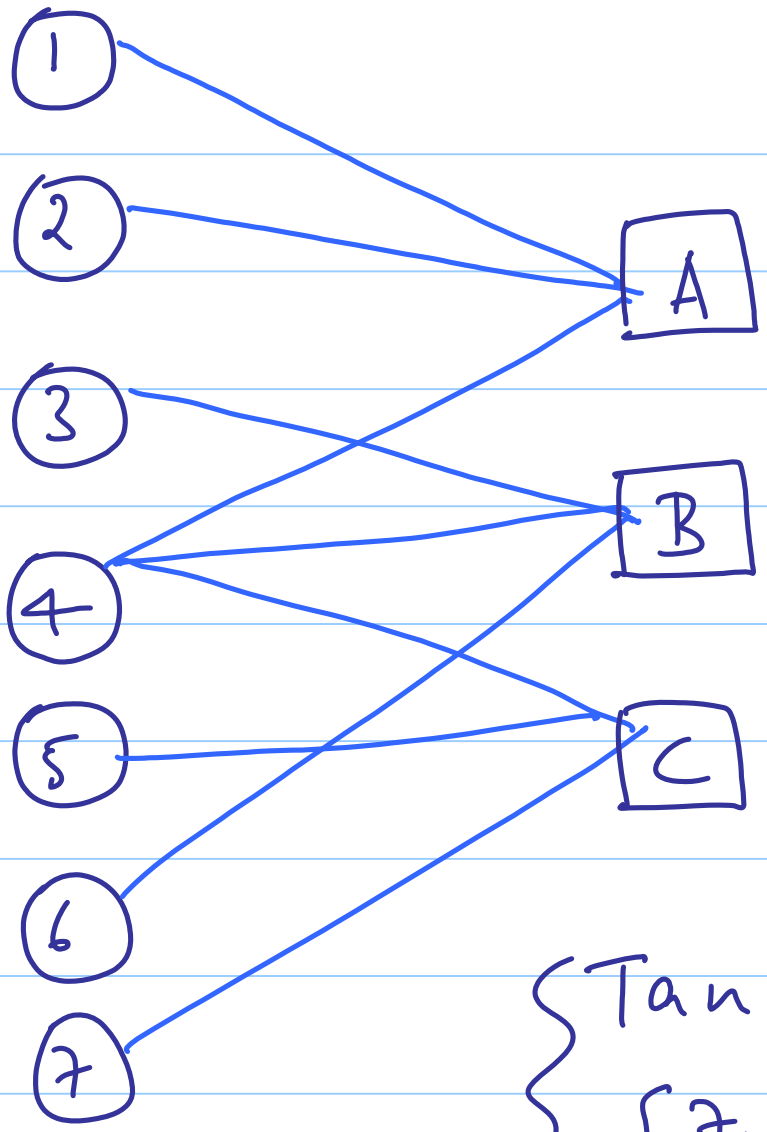
independent of n)

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.



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



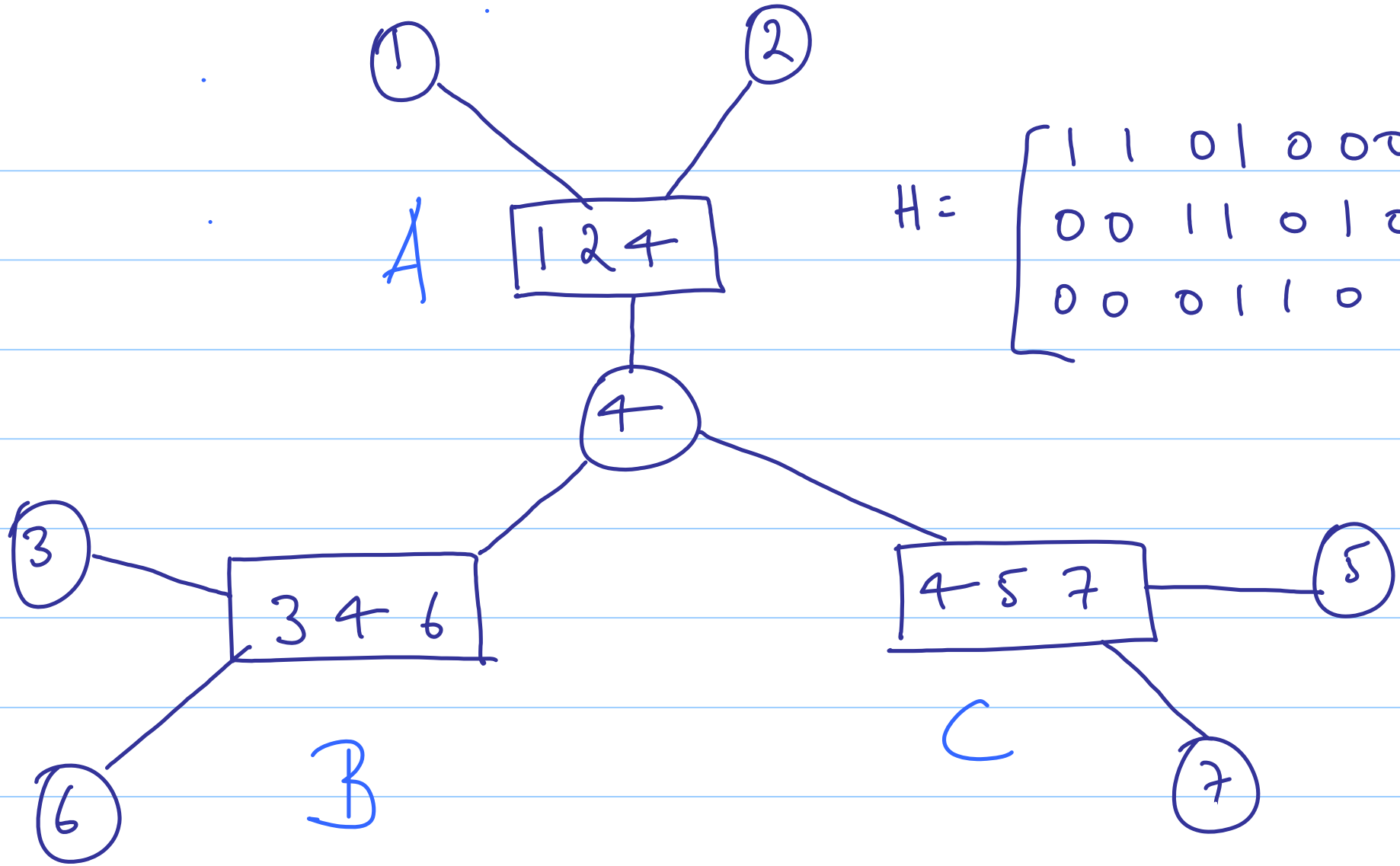
1 2 4

3 4 6

4 5 7

note: this code is not (d_v, d_c) regular

{Tanner graph} the [7, 4, 2] code

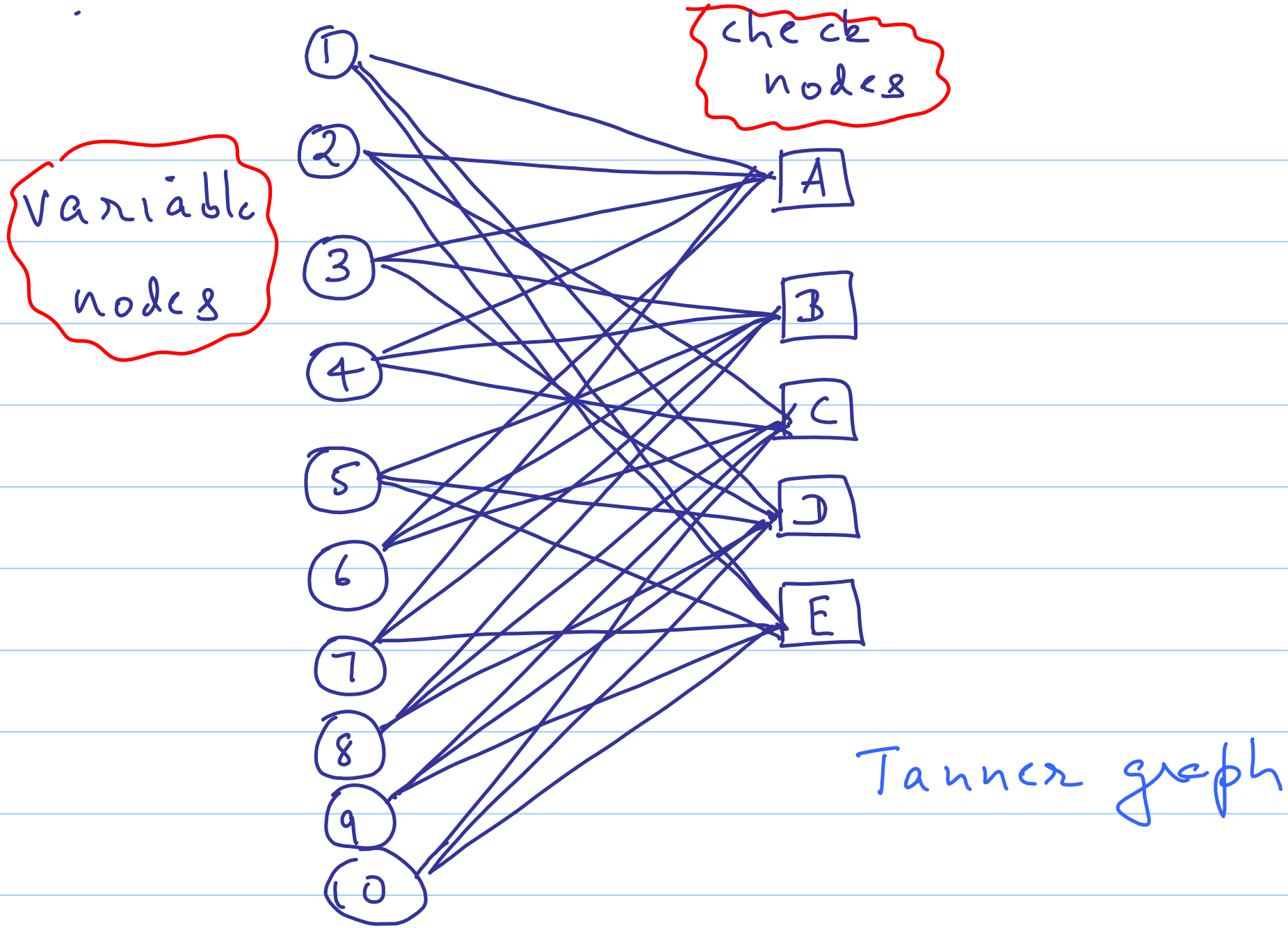


$$H = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

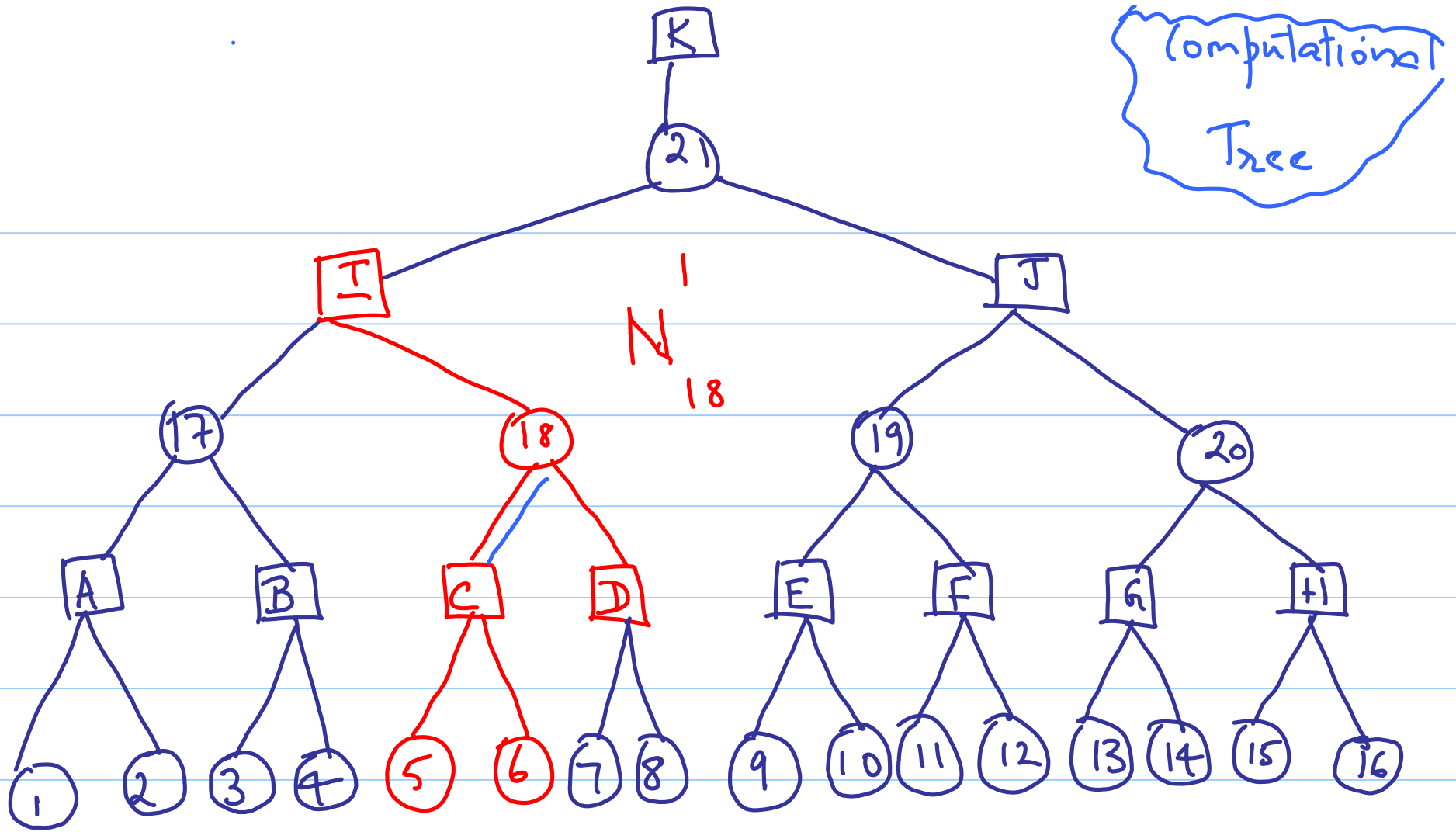
Lec 30 LDPC Code Terminology

Recap

* introduced LDPC codes
  - (d_v, d_c) regular code
  - rate
  - Tanner graph
  - decoding complexity



Tanner graph


K
21
I
J
17
18
19
20
A
B
C
D
E
F
G
H
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
1
N
18
Computational
Tree

Edges, Paths & Neighborhoods

$\vec{e} = (v, c)$

A path in the graph (Tanner graph) is a directed sequence of directed edges

$\vec{e}_1 \quad \vec{e}_2 \quad \cdots \quad \vec{e}_k$ 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

$$u_1 \in N_{u_2}^{d} \iff u_2 \in N_{u_1}^{d}$$

If $\vec{e} = (v, c)$ then the undirected

nbhd to depth d of $\vec{e} = N_v^d \cup N_c^d$

$$= N_c^d$$

$$e \in N_{e'}^{d} \iff e' \in N_e^{d}$$

The directed nbhd to depth d of edge $\vec{e} = (v, c)$ denoted by $H_{\vec{e}}^{d}$

= the induced subgraph containing all edges and nodes on paths $\vec{e_1}\ \vec{e_2}\ldots$ $\vec{e_d}$ starting from v, but with $\vec{e_1} \neq \vec{e}$.

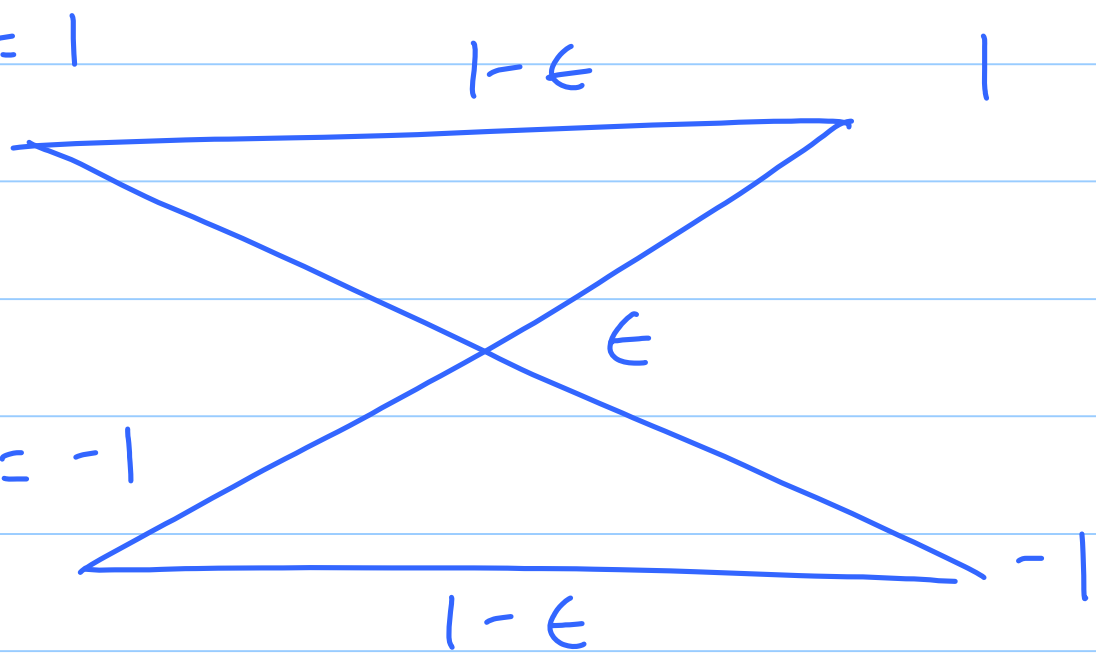
Channel Models

BSC

u_t

$x_t = \pm 1$

$u_t \in \{0, 1\}$



$$y_t = x_t z_t$$

$z_t \in \{\pm 1\}$

$p_{z_t}(-1) = \epsilon$

$p_{z_t}(+1) = 1-\epsilon$

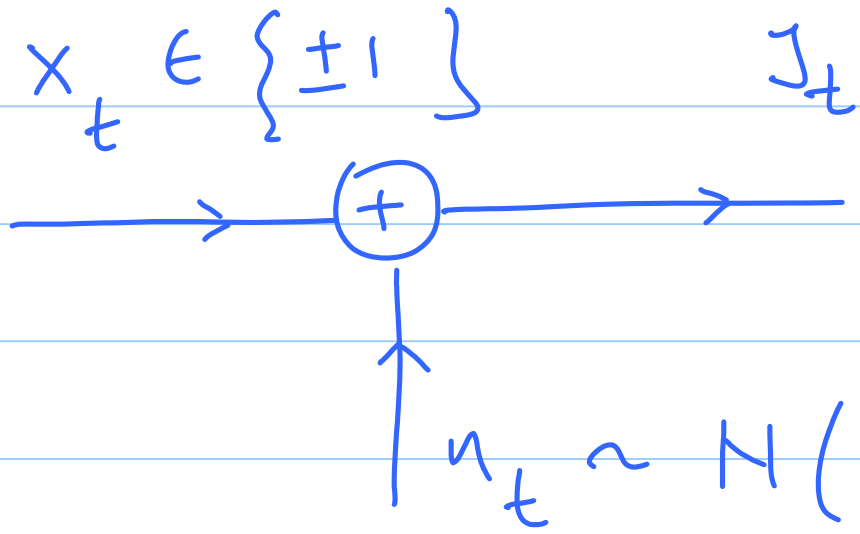
$$p_{Y_t|X_t}(y|x) = p_{Z_t}\left(\frac{y}{x}\right)$$

$$= p_{Y_t|X_t}(-y|-x)$$

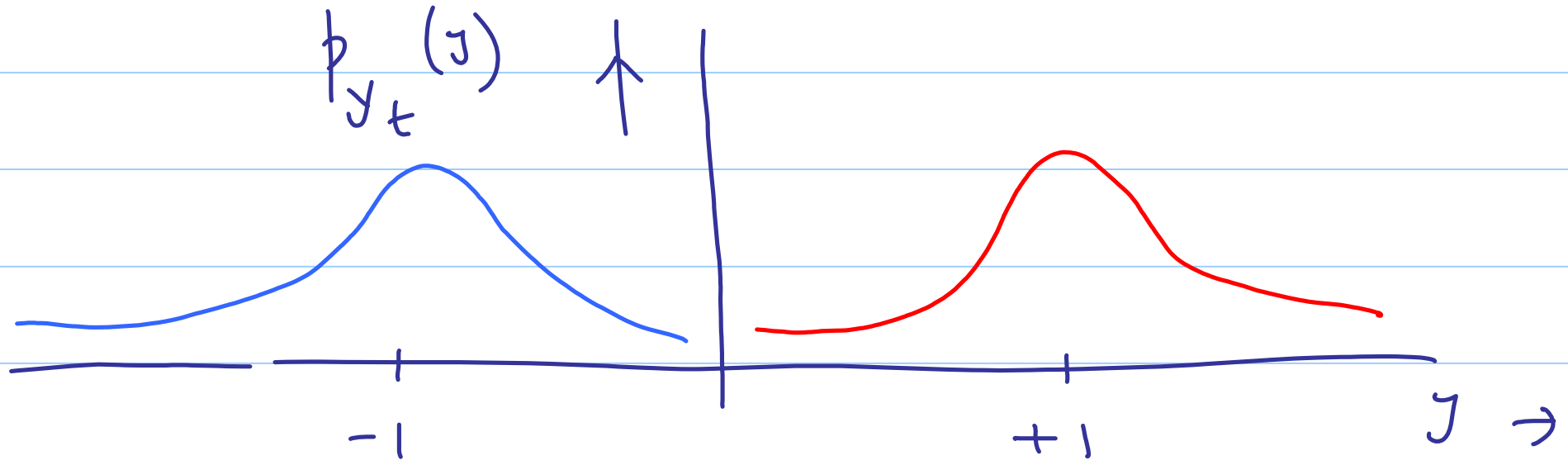
known as the channel symmetry condition

(as applied to the BSC)

Binary-Input AWGN Channel



$$y_t = x_t + n_t$$



Can write:

$$y_t = x_t z_t$$

$$z_t \in N(1, \sigma^2)$$

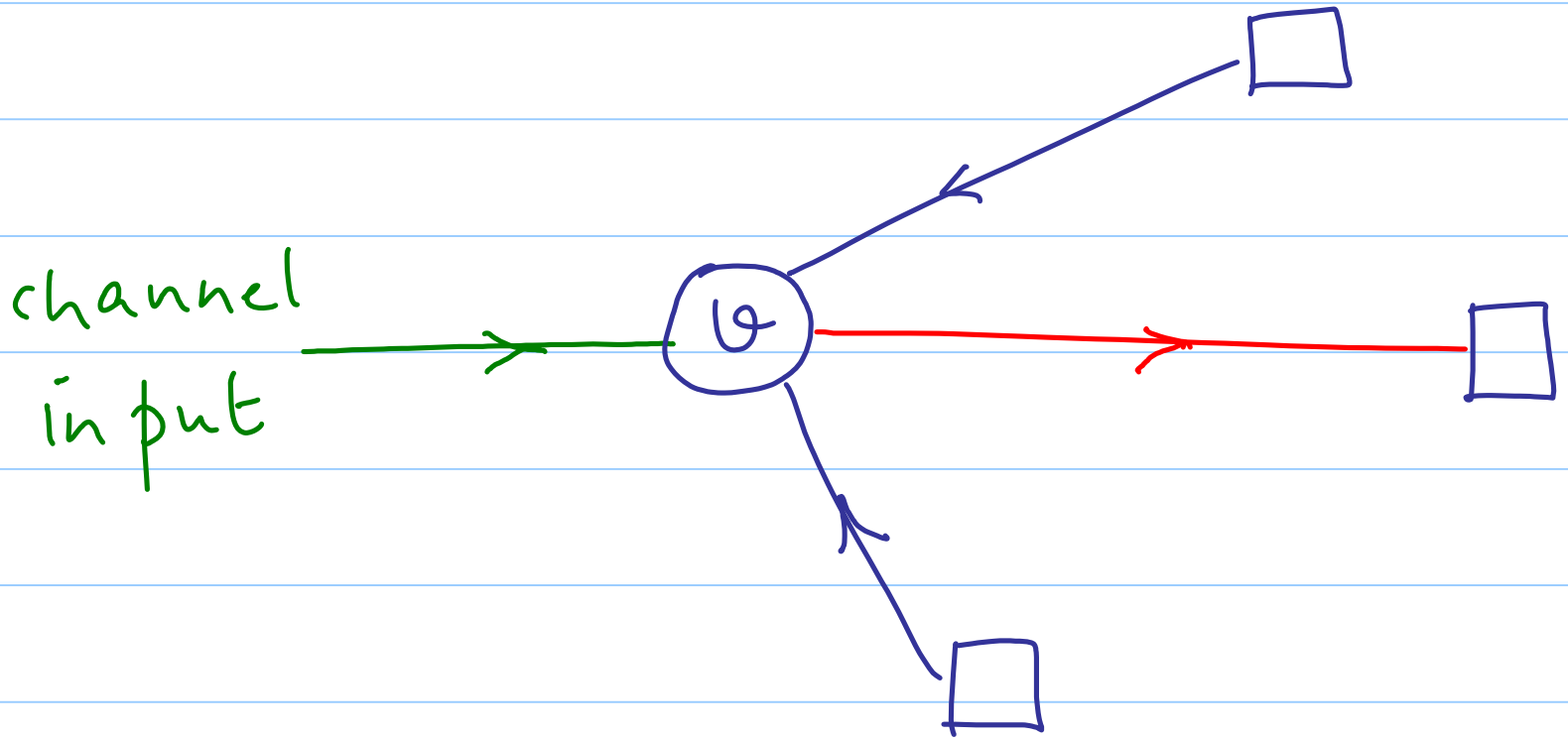
z_t independent of x_t

$$p_{Y_t|X_t}(y|x) = p_{z_t}\left(\frac{y}{x}\right)$$

$$= p_{Y_t|X_t}(-y|-x) = p_{z_t}\left(\frac{-y}{-x}\right) = p_{z_t}\left(\frac{y}{x}\right)$$

(called the channel symmetry condition)

Message-Passing Terminology



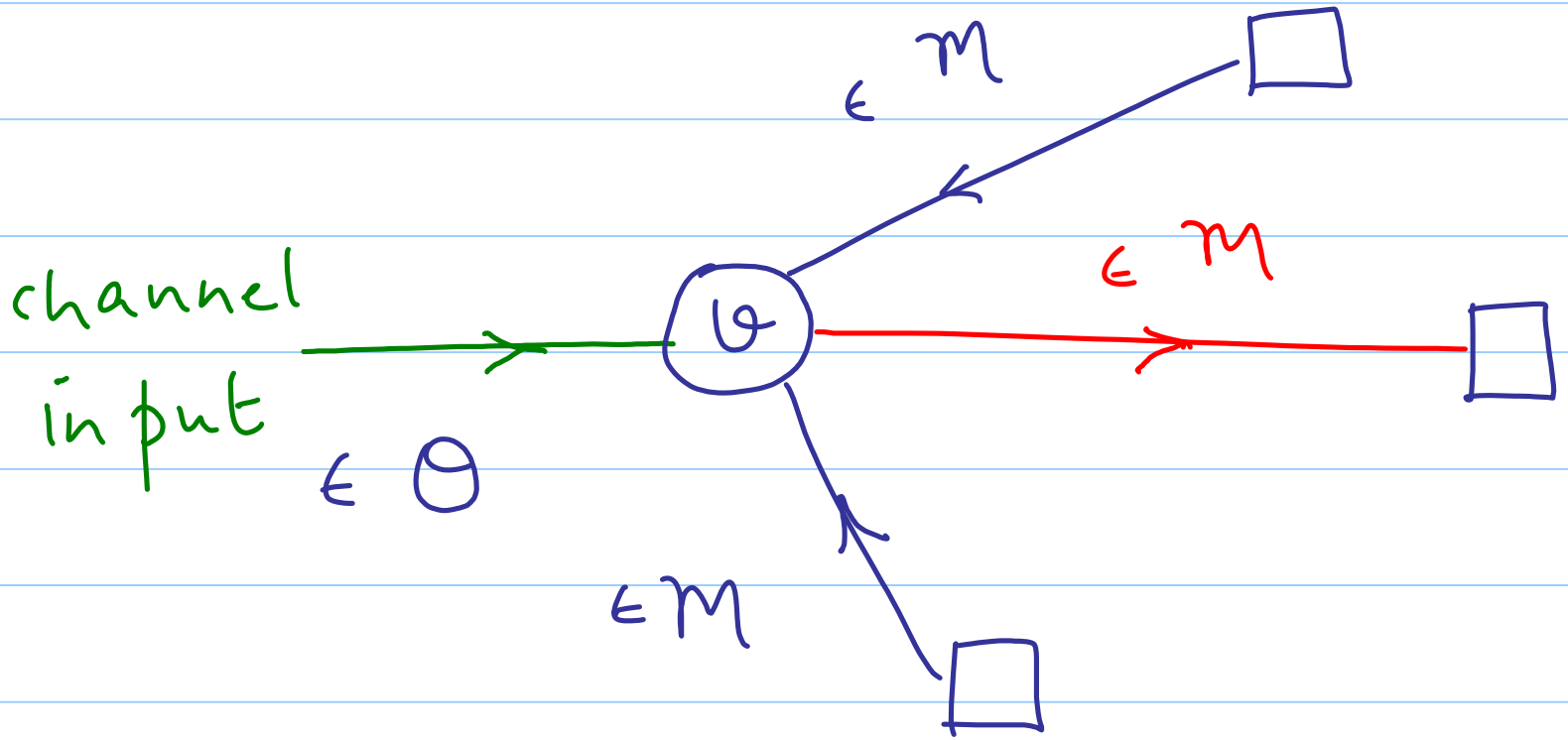
Lec 31 Gallager Decoding Algorithm A

Recap

* Terminology relating to LDPC codes:
  - edges, paths, nbhds
  - channel models

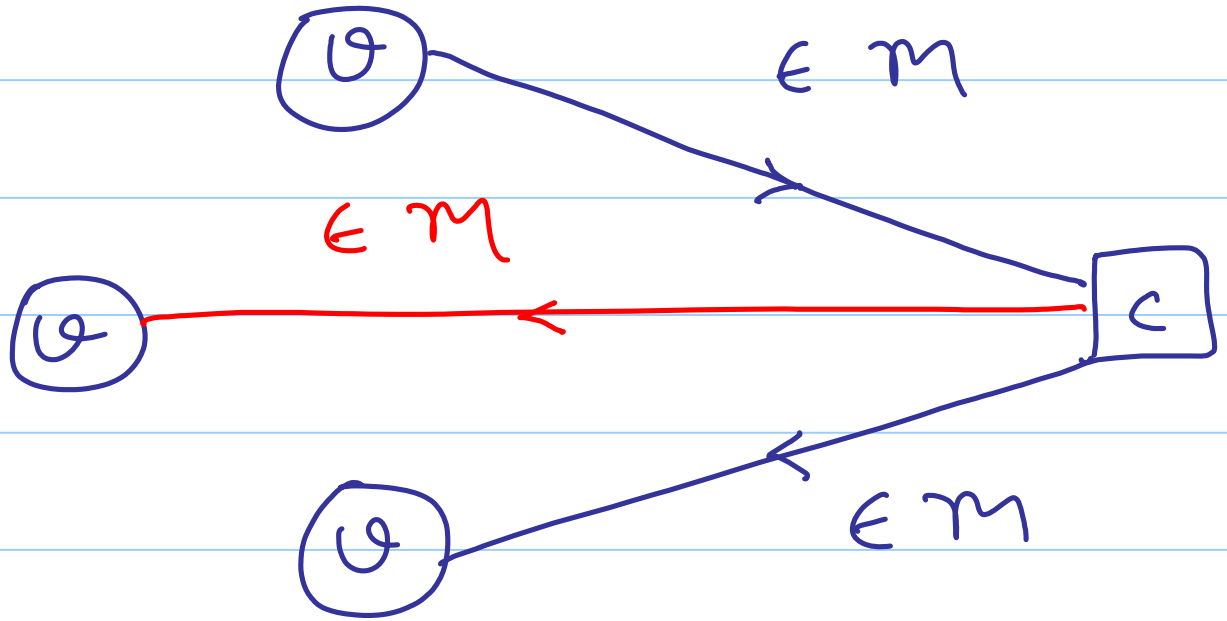
(called the channel symmetry condition)

Message-Passing Terminology



Θ = output alphabet of the channel

$\mathcal{M}$ = the common alphabet employed to pass messages from variable node to check nodes or vice-versa.



$\psi_v^{(0)} : \Theta \to m$ { the initial message map

$\psi_c^{(\ell)} : m^{d_c - 1} \to m$ $\ell = \#$ the iteration

$\psi_v^{(\ell)} : \Theta \times m^{d_v - 1} \to m$

Assumptions concerning message passing at a variable / check node:

$$\psi_v^{(0)}(bm) = \left[\psi_v^{(0)}(m)\right] b, \quad b \in \{\pm 1\} \tag{1}$$

$m \in \mathcal{M}$

$$\psi_v^{(\ell)}(bm_0, bm_1, \ldots, bm_{d_v-1}) = b\,\psi_v^{(\ell)}(m_0 m_1 \cdots m_{d_v-1}) \tag{2}$$

$b \in \{\pm 1\}$

$$\psi_c^{(\ell)}\left(b_1 m_1 , b_2 m_2 , \cdots , b_{d_{c-1}} m_{d_{c-1}} \right)$$

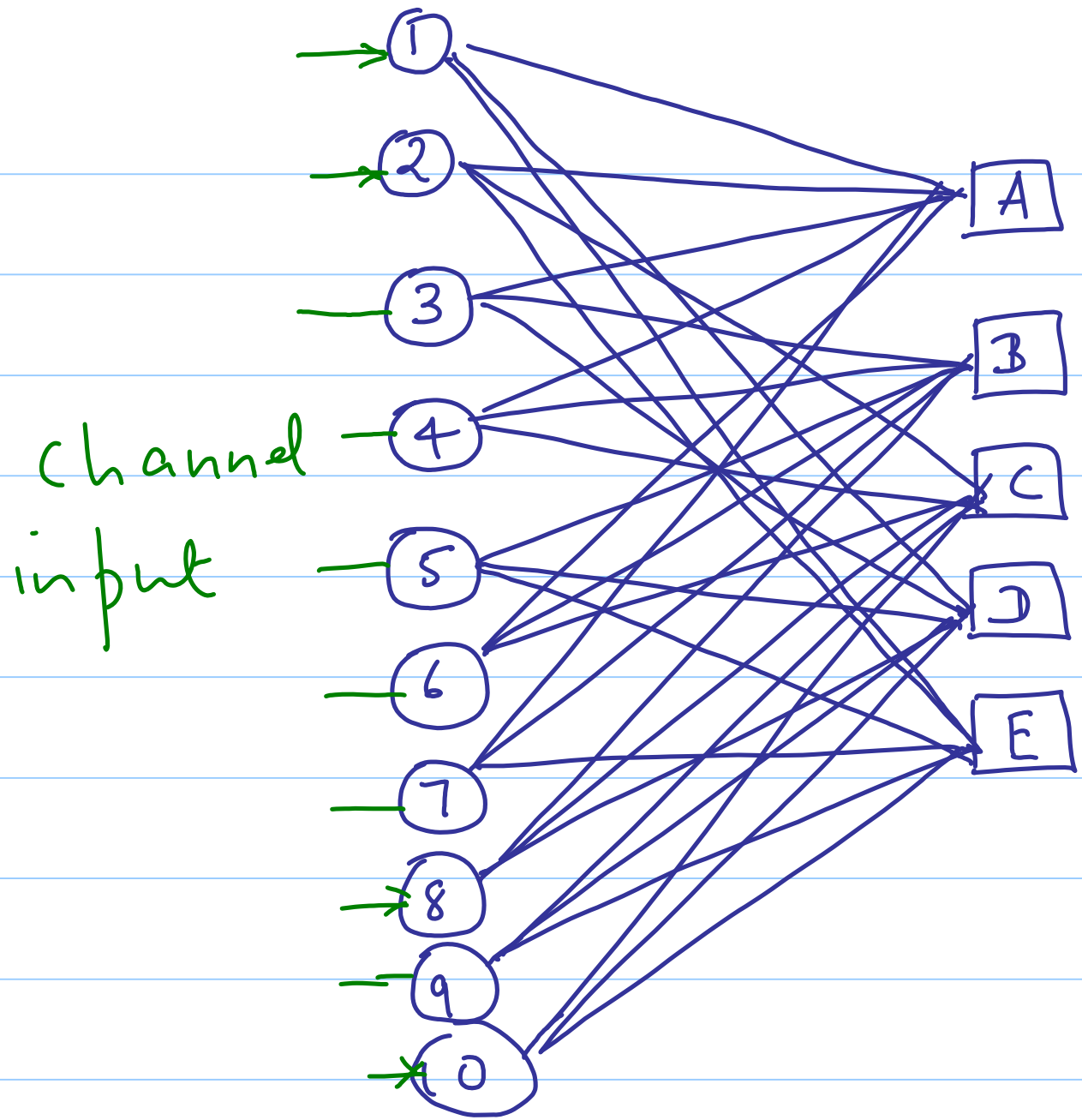
$$= \left[\prod_{j=1}^{d_{c-1}} b_j \right] \psi_c^{(\ell)} \left(m_1 m_2 \cdots m_{d_{c-1}} \right) \quad (3)$$

$$b_j \in \{ \pm 1 \}$$

$$m_j \in \mathcal{M} .$$

Equations (1) – (3) are known as the

variable and check-node symmetry conditions.



we will use the sign of the message

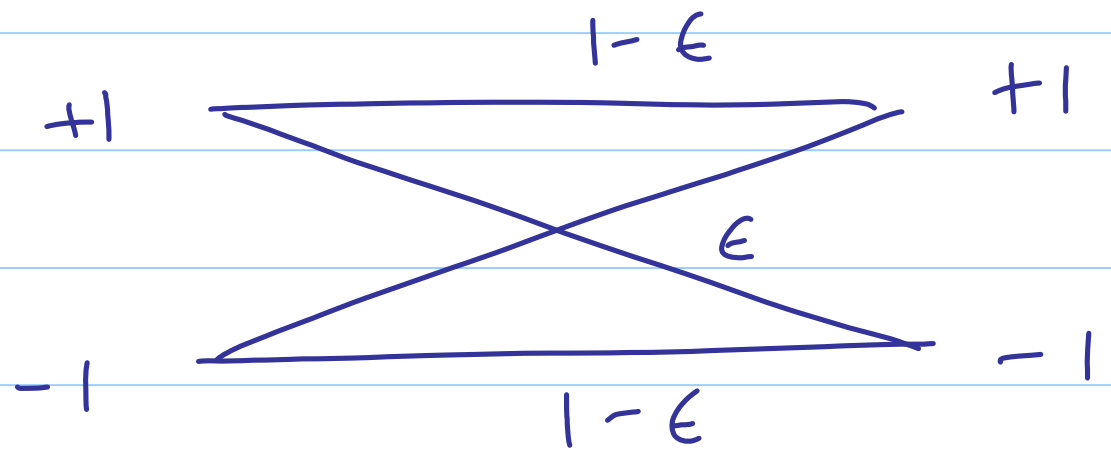
as an indication as to the value of (v).

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



$\Theta = \{\pm 1\}$

$m = \{\pm 1\}$

(0)

$\psi_v(m) = m$

$$\psi_v^{(l)}(m_0\, m_1 \cdots m_{d_v-1})$$

$$= \begin{cases} -m_0 & \text{if } m_j = -m_0 \text{ all } 1 \le j \le d_v - 1 \\ m_0 & \text{else} \end{cases}$$

$$\psi_c^{(l)}(m_1 m_2 \cdots m_{d_c-1}) = \prod_{j=1}^{d_c-1} m_j$$

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.

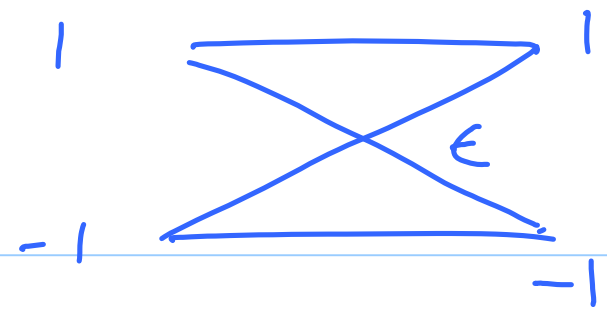
$p_1^{(0)}$ = {prob that the message passed by a variable node during the 0^{th} iteration = +1

$p_{-1}^{(0)}$ = similarly for −1

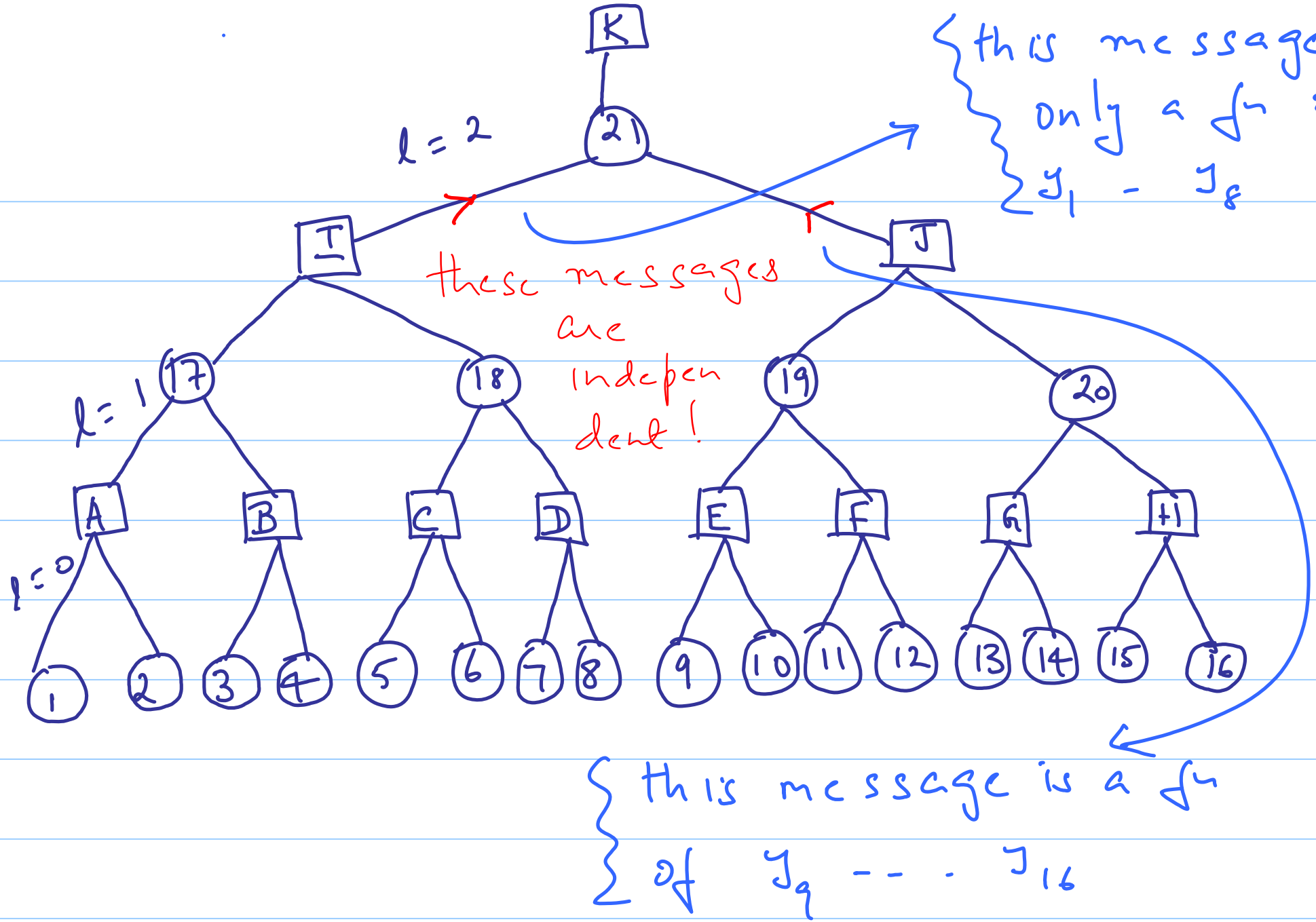
$q_1^{(\ell)}$ = {prob that on the ℓ^{th} iteration the check node message = 1

$q_{-1}^{(\ell)}$ = similarly for −1

$p_1^{(0)} = 1-\epsilon$ $\quad$ $p_{-1}^{(0)} = \epsilon$



note that by symmetry the probability mass f_k of messages along an edge only depends upon the number of the iteration (i.e. on k) as well as the direction of the message (variable node to check node or vice-versa).


K
21
l = 2
T
J
17
18
19
20
l = 1
A
B
C
D
E
F
G
H
l = 0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
this message is only a fn of $y_1 - y_8$
these messages are independent!
this message is a fn of $y_9 - - - y_{16}$

$$p_1^{(0)} = 1 - \epsilon \qquad p_{-1}^{(0)} = \epsilon$$

$$p_{-1}^{(\ell)} = p_{-1}^{(0)} \left\{ 1 - \left(1 - q_{-1}^{(\ell)} \right)^{d_v - 1} \right\} + p_1^{(0)} \left[q_{-1}^{(\ell)} \right]^{d_v - 1}$$

$q_{-1}^{(\ell)}$ represents the probability that not all incoming message are $= +1$

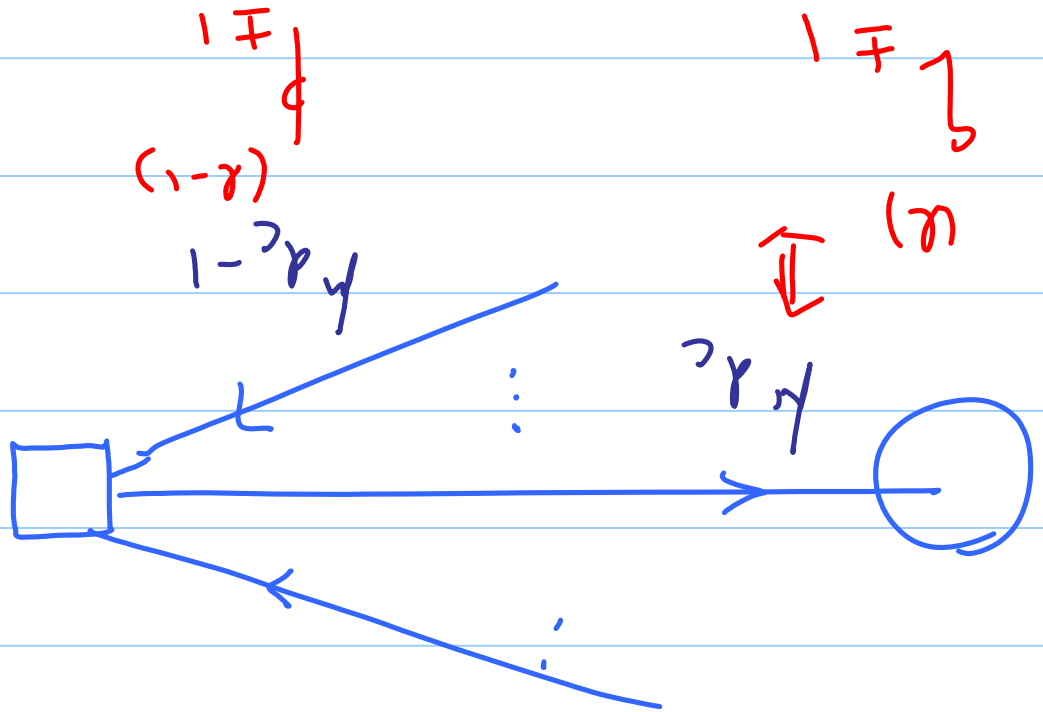
(uses independence)

Lec 32 {BP Decoding of LDPC Codes

(BP ≡ Belief Propagation)

Recap

* {Alphabets arising in message passing
* independence assumptions
* {Gallager decoding algorithm A



$$\therefore \boxed{\mu_{d_c} = \prod_{j=1}^{d_c-1} \mu_j}$$

$$\therefore \mathbb{E}\{\mu_{d_c}\} = \prod_{j=1}^{d_c-1} \mathbb{E}\{\mu_j\} \quad \cdots (1)$$

$$\begin{cases} P_r(\mu_j = 1) = p_1^{(l-1)} \\ P_r(\mu_j = -1) = p_{-1}^{(l-1)} \end{cases}$$

$\therefore$ (1) gives us that

$$\left[q_1^{(\ell)} - q_{-1}^{(\ell)} \right] = \left[p_1^{(\ell-1)} - p_{-1}^{(\ell-1)} \right]^{d_c - 1}$$

$$\therefore \quad 1 - 2 q_{-1}^{(\ell)} = \left[1 - 2 p_{-1}^{(\ell-1)} \right]^{d_c - 1}$$

$$\therefore \boxed{q_{-1}^{(\ell)} = \frac{1}{2} \left\{ 1 - \left[1 - 2 p_{-1}^{(\ell-1)} \right]^{d_c - 1} \right\}}$$

.... (2)

$$p_{-1}^{(\ell)} = p_{-1}^{(0)} \left\{ 1 - \left(1 - q_{-1}^{(\ell)}\right)^{d_c - 1} \right\} \quad \text{- (3)}$$

$$+ \; p_{1}^{(0)} \left[q_{-1}^{(\ell)} \right]^{d_c - 1}$$

From (2) and (3) we get:

$$p_{-1}^{(\ell)} = p_{-1}^{(0)} \left\{ 1 - \left[\frac{1}{2} \left\{ 1 + \left[1 - 2 p_{-1}^{(\ell-1)} \right]^{d_c - 1} \right\} \right]^{d_v - 1} \right\}$$

$$+ \left[1 - p_{-1}^{(0)} \right] \left\{ \left[\frac{1}{2} \left[1 - \left[1 - 2 p_{-1}^{(\ell-1)} \right]^{d_c - 1} \right] \right]^{d_v - 1} \right\}$$

{ This is the desired expression for density evolution

Can be shown that

a) $\phi_{-1}^{(e)}$ increases monotonically with increase in $\phi_{-1}^{(o)}$

b) when $\phi_{-1}^{(o)}$ is below a certain threshold, $\phi_{-1}^{(e)} \longrightarrow 0$ as the # of iterations increases.

Eg when $(d_v, d_c) = (3, 6)$

then designed rate satisfies:

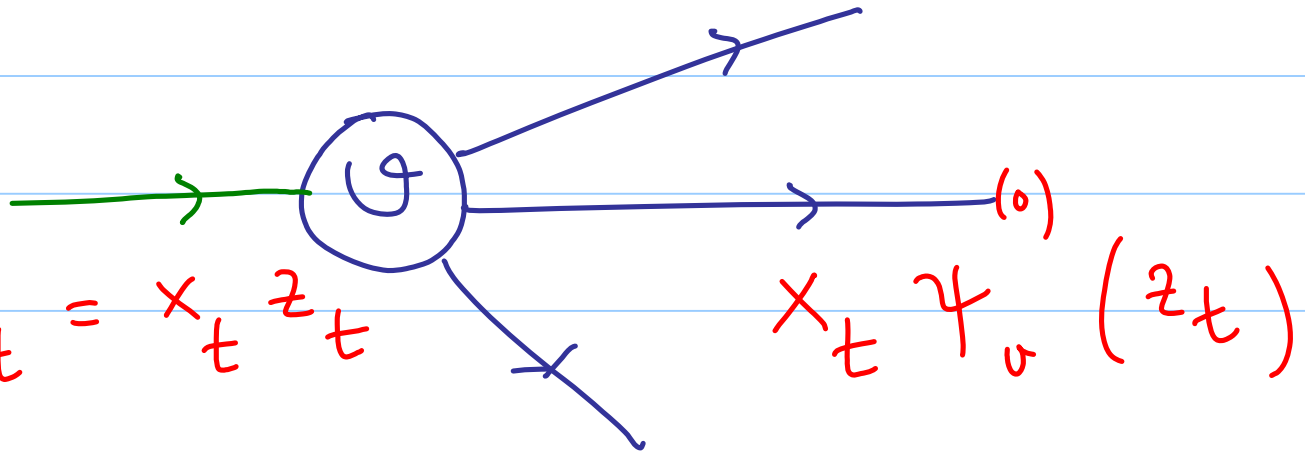
$$\frac{k}{n} \geq 1 - \frac{3}{6} = \frac{1}{2}$$

and the threshold exhibited by Gallager decoding algorithm A is $\epsilon = 0.04$.

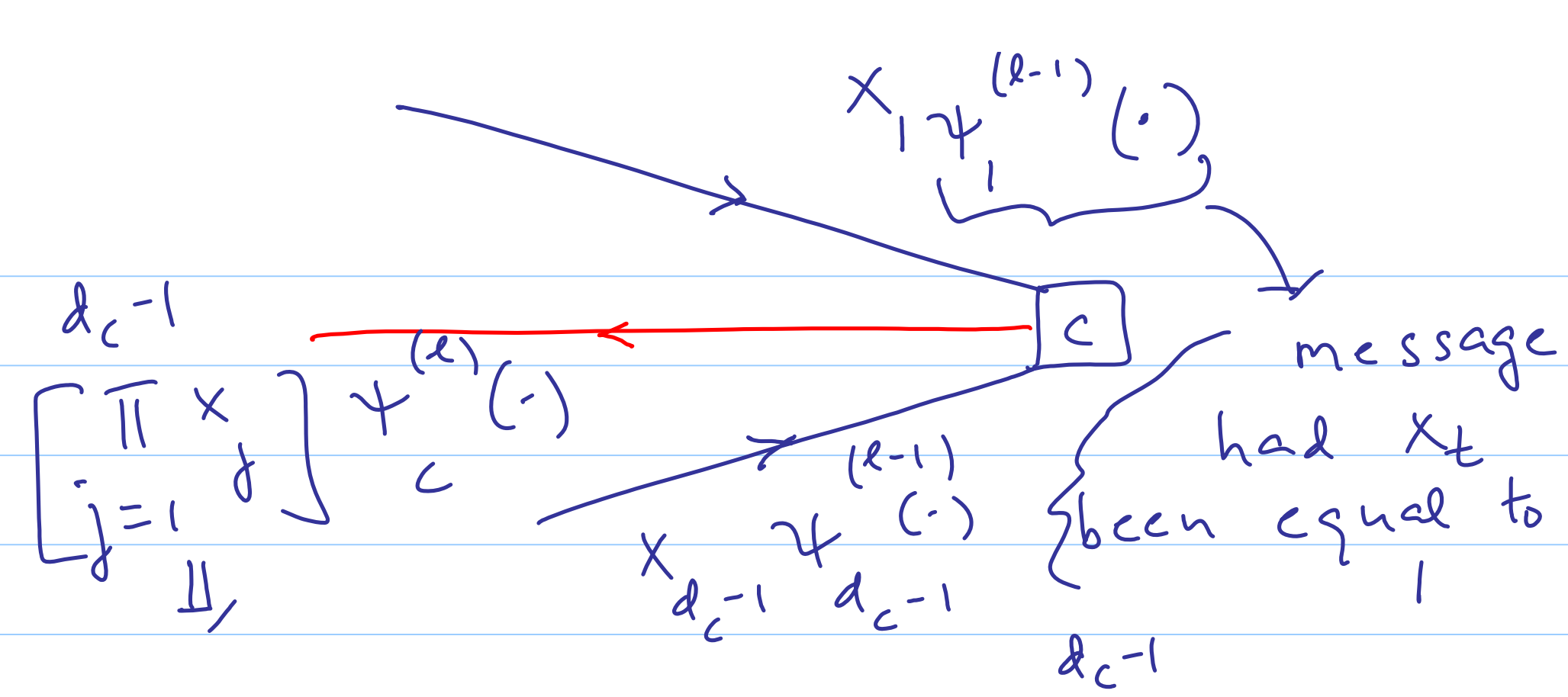
(in comparison channel capacity

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.



χ_{α_v}
$\psi_v(\cdot)$
φ
χ_{α_u}
$\psi_u(\cdot)$
χ_{α_s}
$\psi_s(\cdot)$
χ_{α_c}
$\psi_c(\cdot)$



$= x_{d_c}$

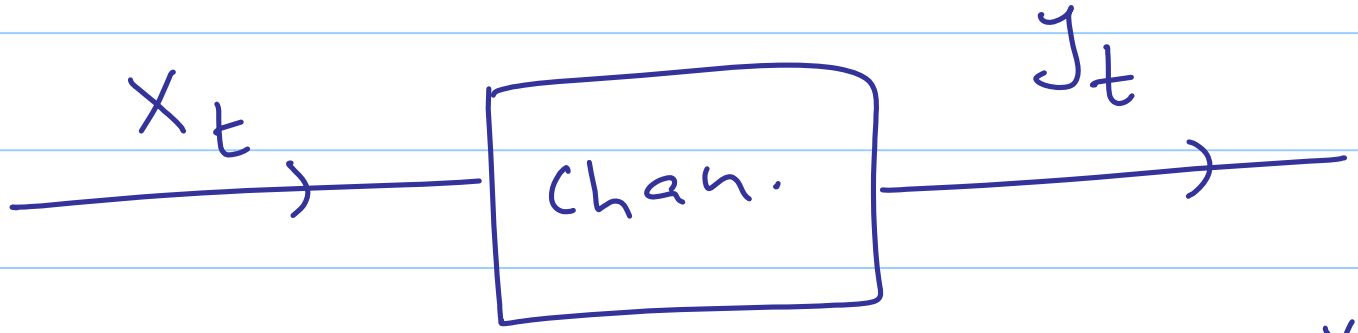
(because of the parity check)

$$u_{d_c} = \sum_{j=1}^{d_c-1} u_j \pmod 2$$

$$x_j = (-1)^{u_j}$$

Channel Symmetry Assumption

$$p(y_t | x_t) = p(-y_t | -x_t) \quad (4)$$



$x_t \in \{\pm 1\}$

Define $\boxed{y_t = x_t z_t}$

We will now show that x_t is independent of z_t.

$$p_{z|x}(z_t \mid x_t = 1) = p_{r|x}(z_t \mid x_t = 1)$$

$$p_{z|x}(z_t \mid x_t = -1) = p_{r|x}(-z_t \mid x_t = -1)$$

$$= p_{r|x}(z_t \mid x_t = 1)$$

by (4)

$$\therefore p_{z|x}(z_t \mid x_t) = p_z(z_t) \quad ///$$

Lec 33 BP Decoding (continued)

Recap

* Gallager's decoding algorithm A
  - density evolution
* completed proof showing that
  assuming { check node, variable node, channel output } symmetry

one can assume for the purpose of estimating error probability that the all-1 codeword was transmitted.

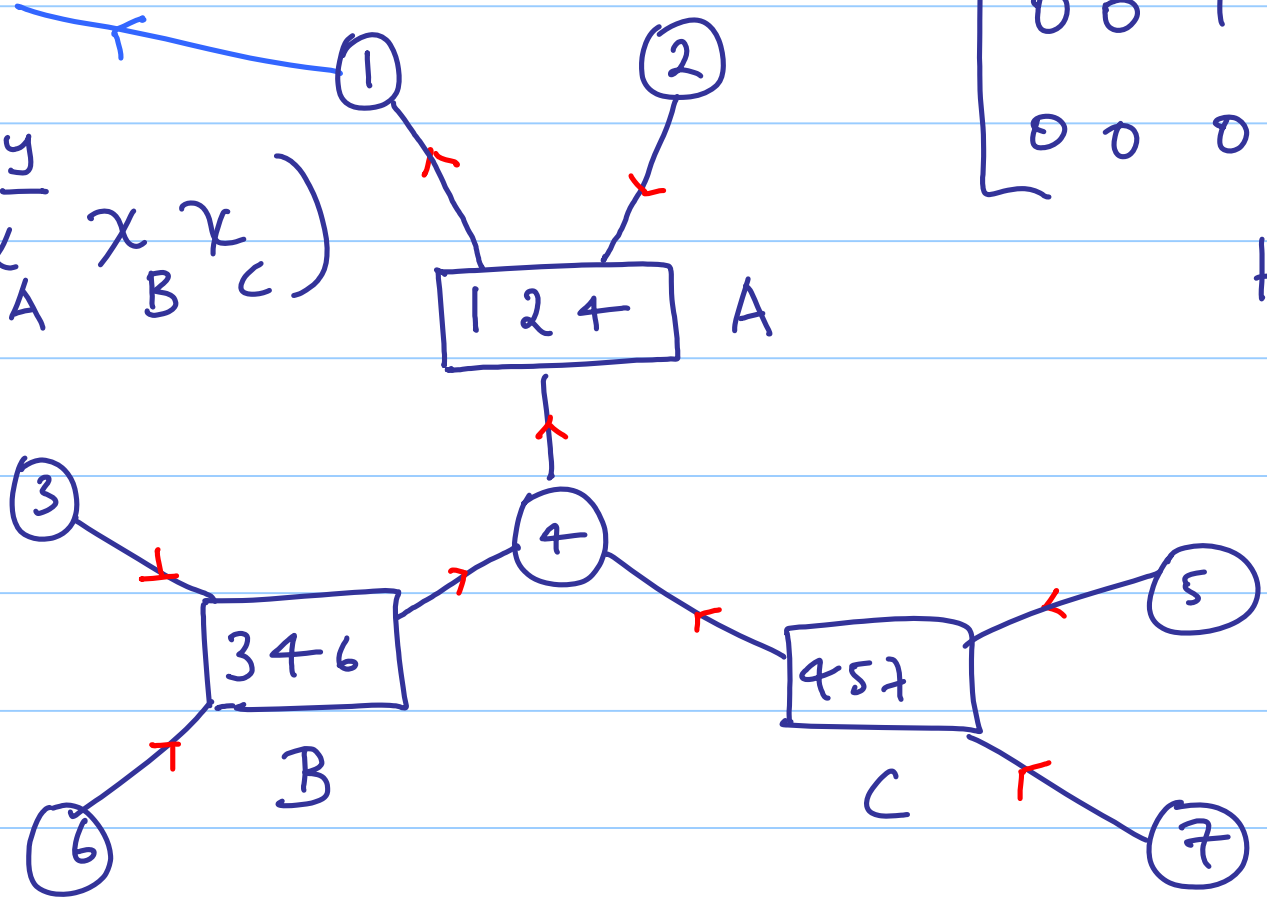
objective fn.

$\beta_1(x_1)$

$$\alpha\, p\left(x_1 \middle| \underline{y}_A\, x_B\, x_C\right)$$

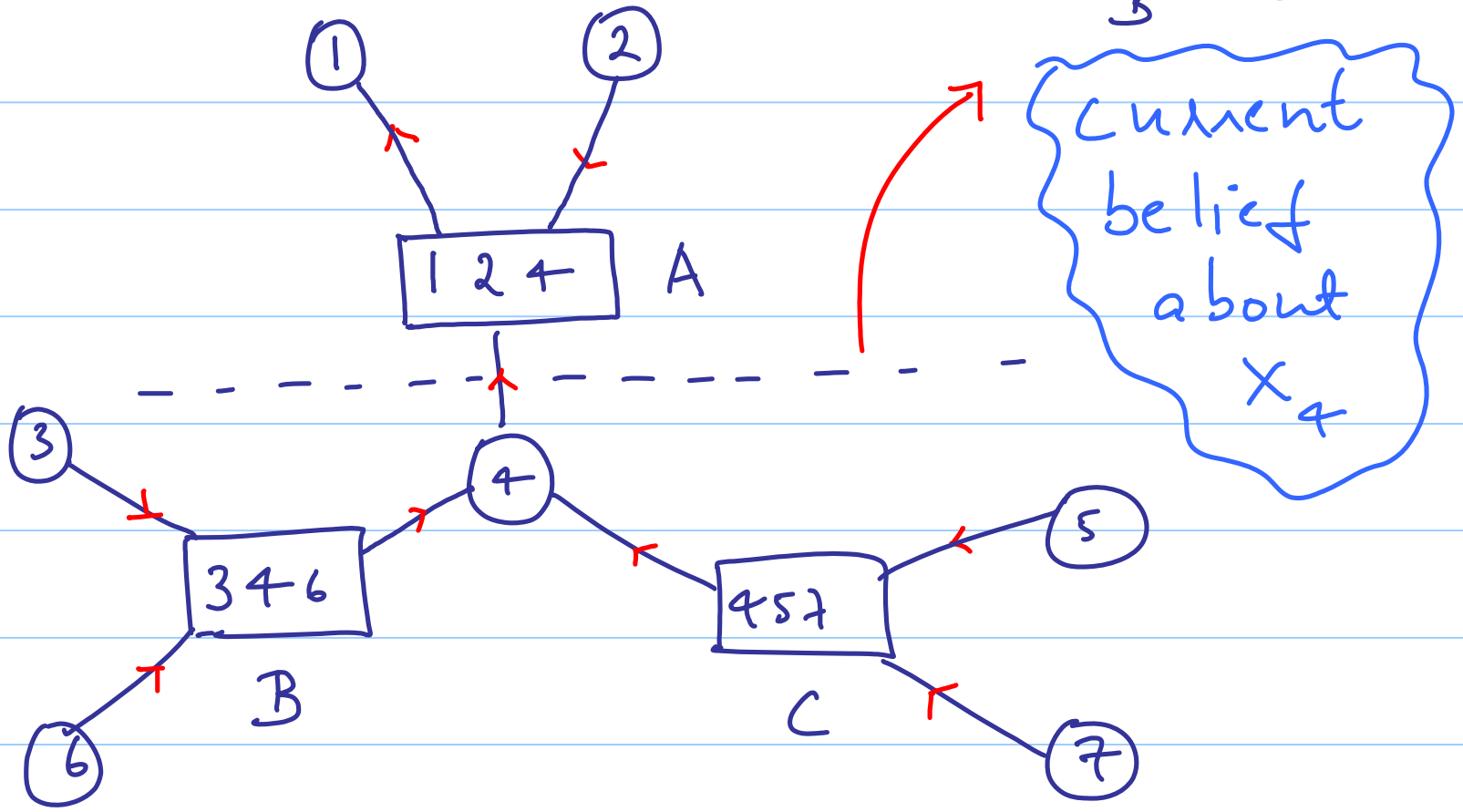
$$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

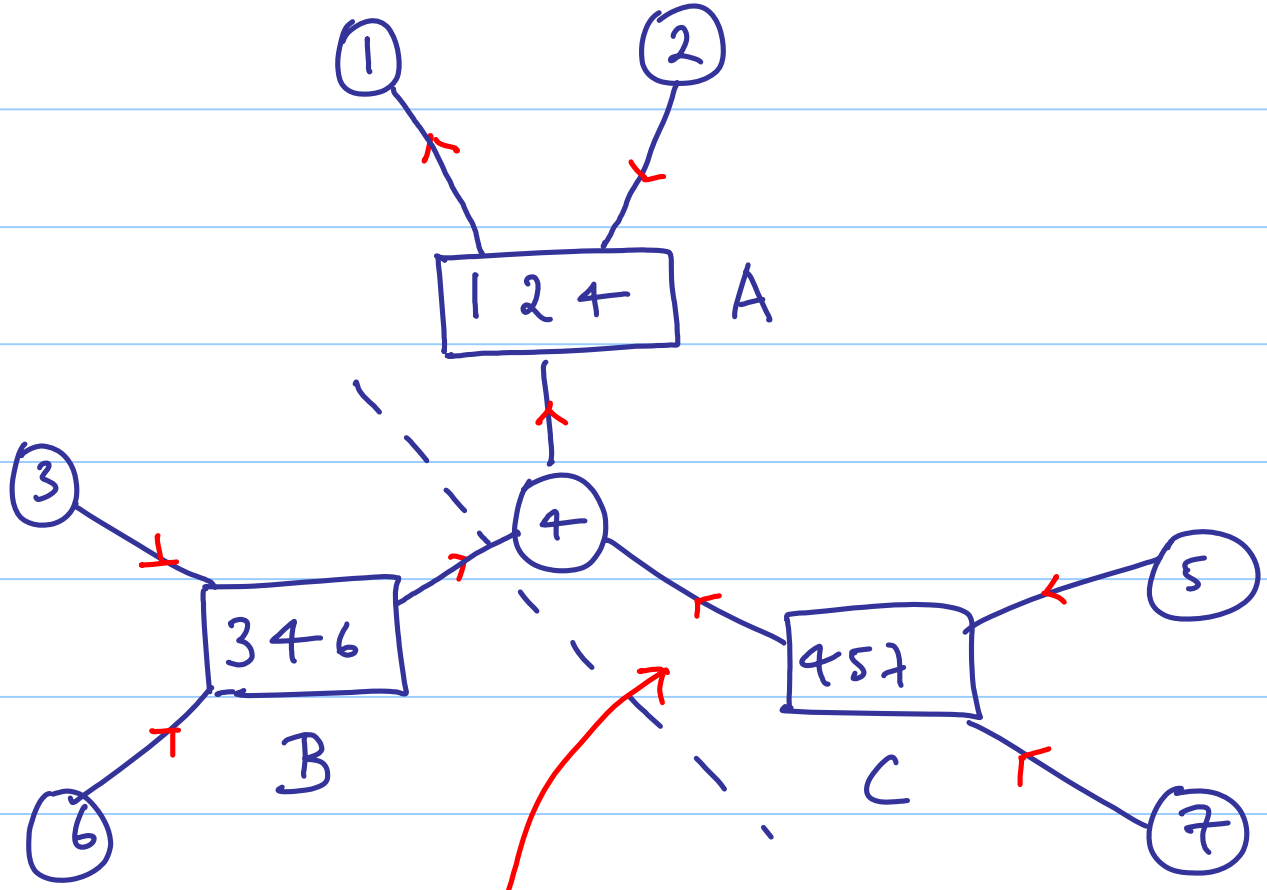
H



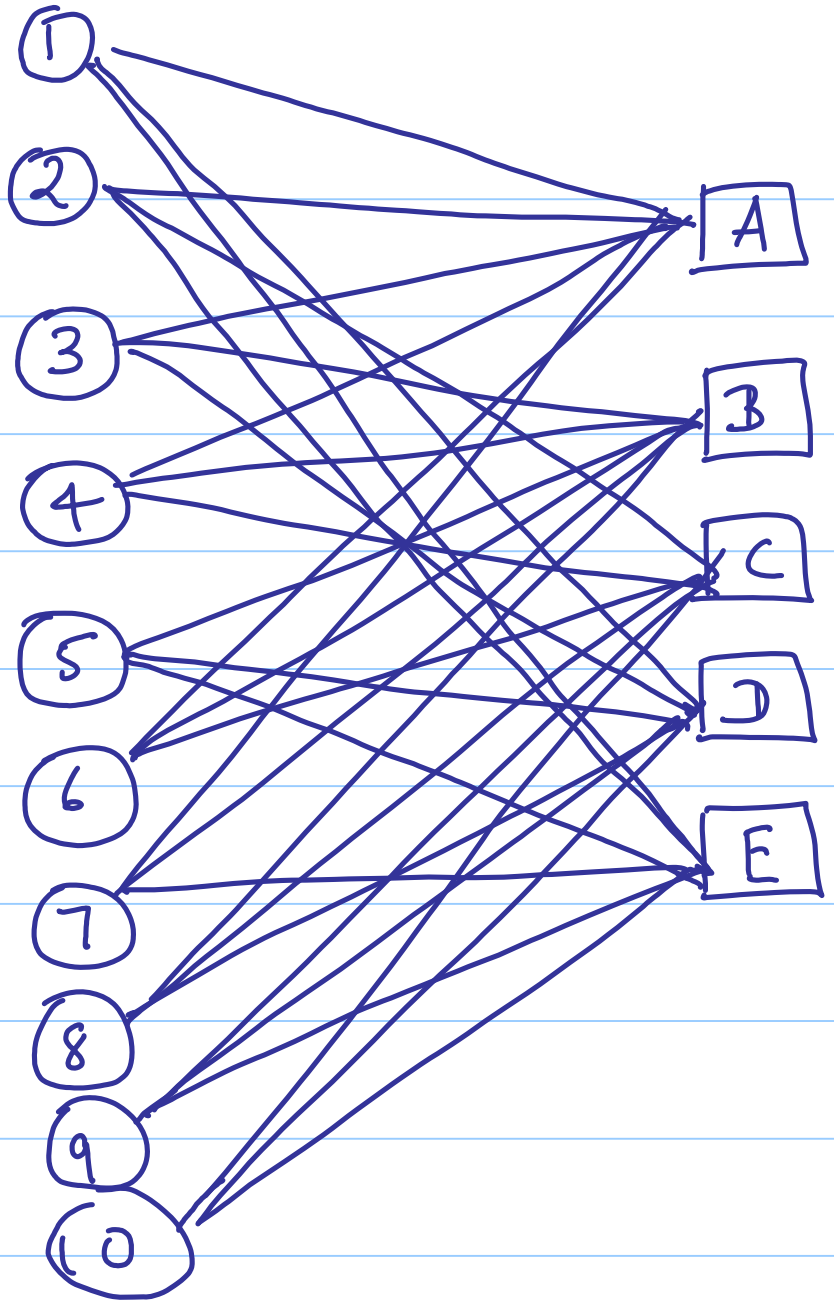
$$\propto p(x_4 \mid y_3\, y_4\, y_5\, y_6\, y_7)$$

$x_B \; x_C$





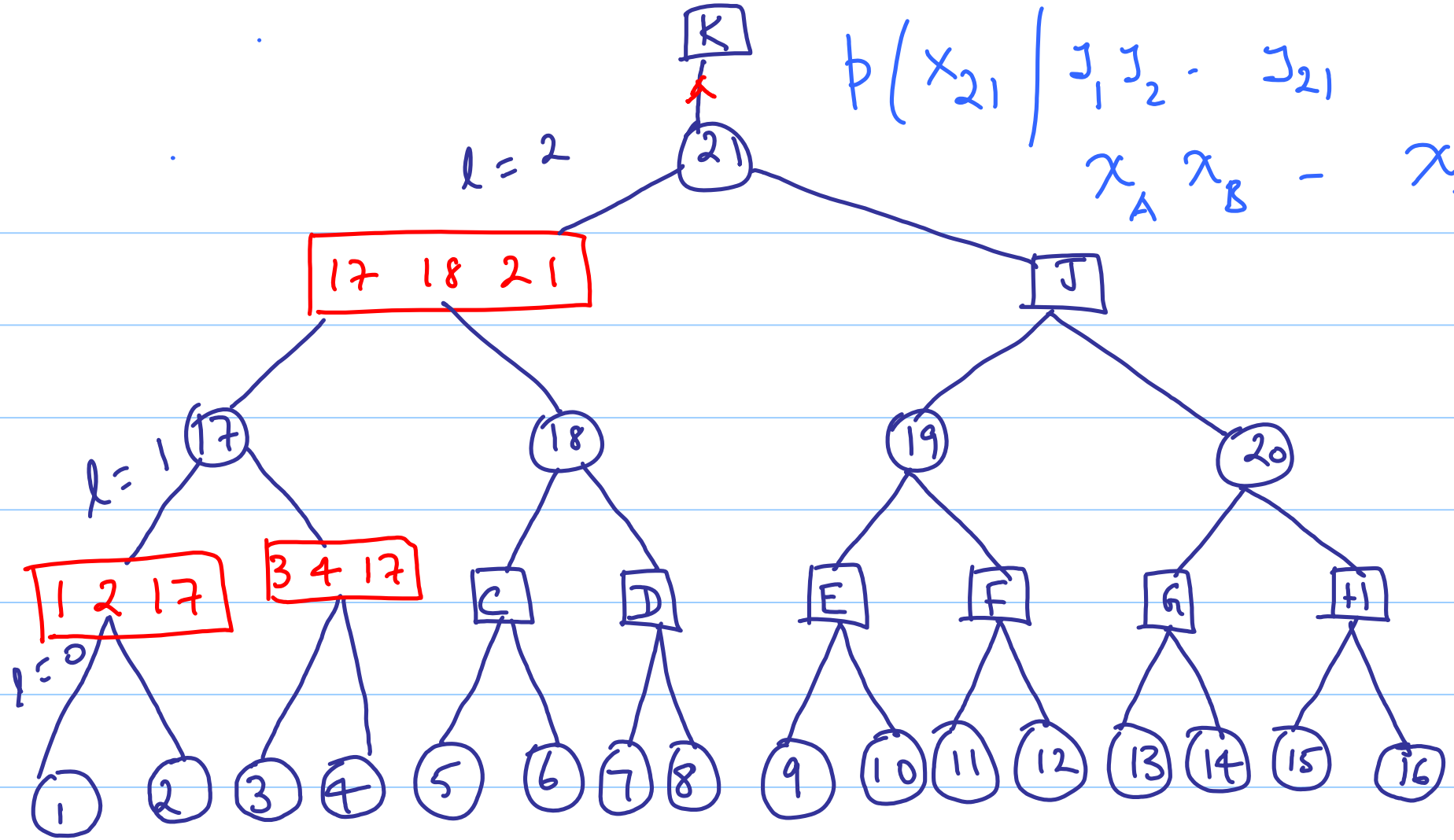
$p(x_4 \mid y_3\, y_6\, x_B)$ { current belief about the prob of x_4



Tanner graph

we assume that the nbhd of every edge is tree like to depth 2ℓ

$N_{\vec{e}}^{2\ell}$



$$p\left(x_{21} \mid J_1 J_2 \cdots J_{21}\right)$$

$$x_A \; x_B \; - \; x_I$$

this is the jn tree that one would obtain if one posed the problem

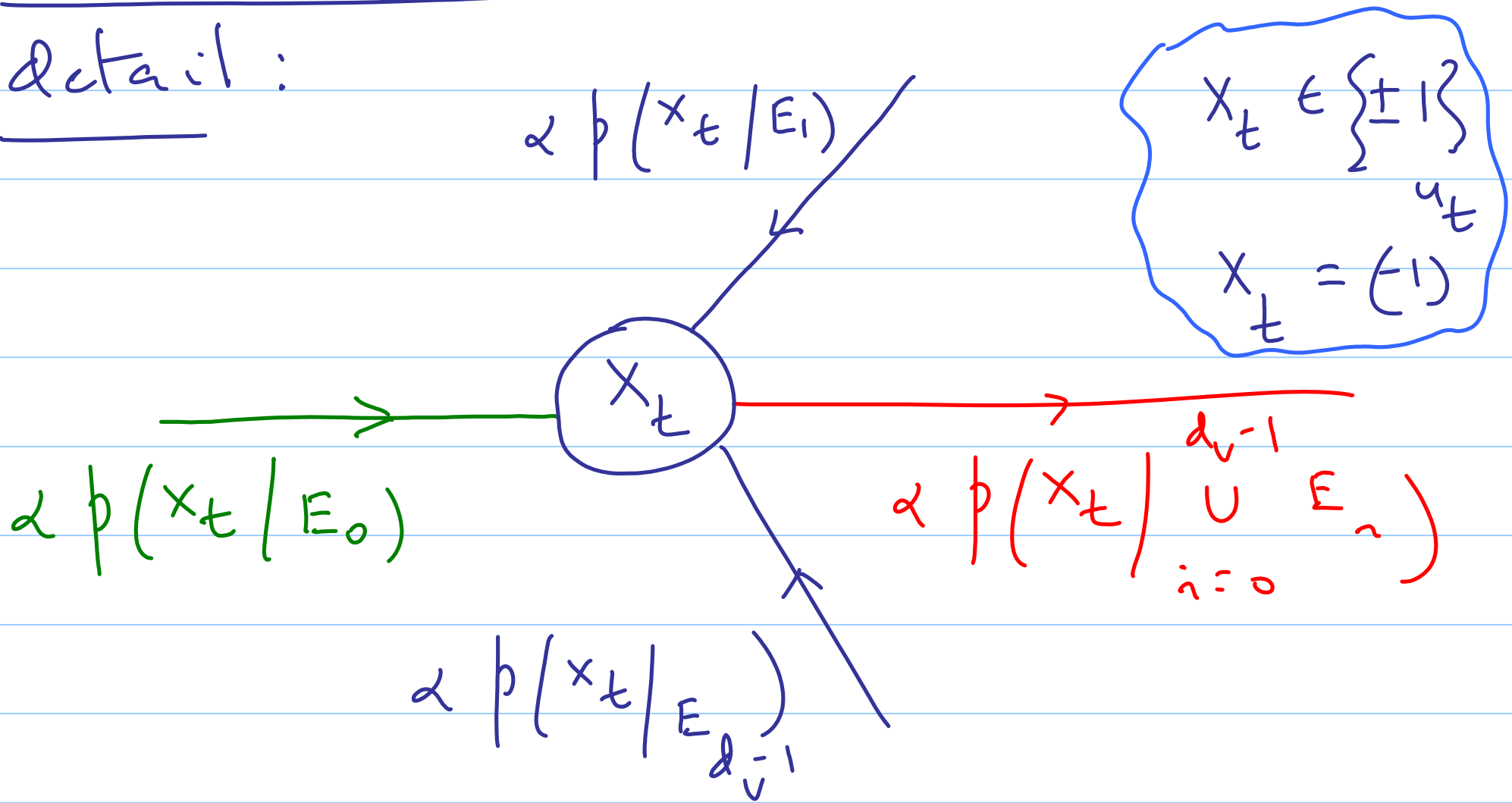
of computing $p\left(x_{21} \,\middle|\, \begin{matrix} y_1 y_2 \cdots y_{21} \\ x_A x_B - x_I \end{matrix}\right)$

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 graph 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 more detail:



Under the GDL we have:

$$p\left(x_t \mid \bigcup_{i=0}^{d_v-1} E_i\right) \propto \prod_{j=0}^{d_v-1} p(x_t \mid E_j)$$

$$\Rightarrow \frac{p\left(x_t=1 \mid \bigcup E_i\right)}{p\left(x_t=-1 \mid \bigcup E_i\right)} = \prod_{j=0}^{d_v-1} \frac{p(x_t=1 \mid E_j)}{p(x_t=-1 \mid E_j)}$$

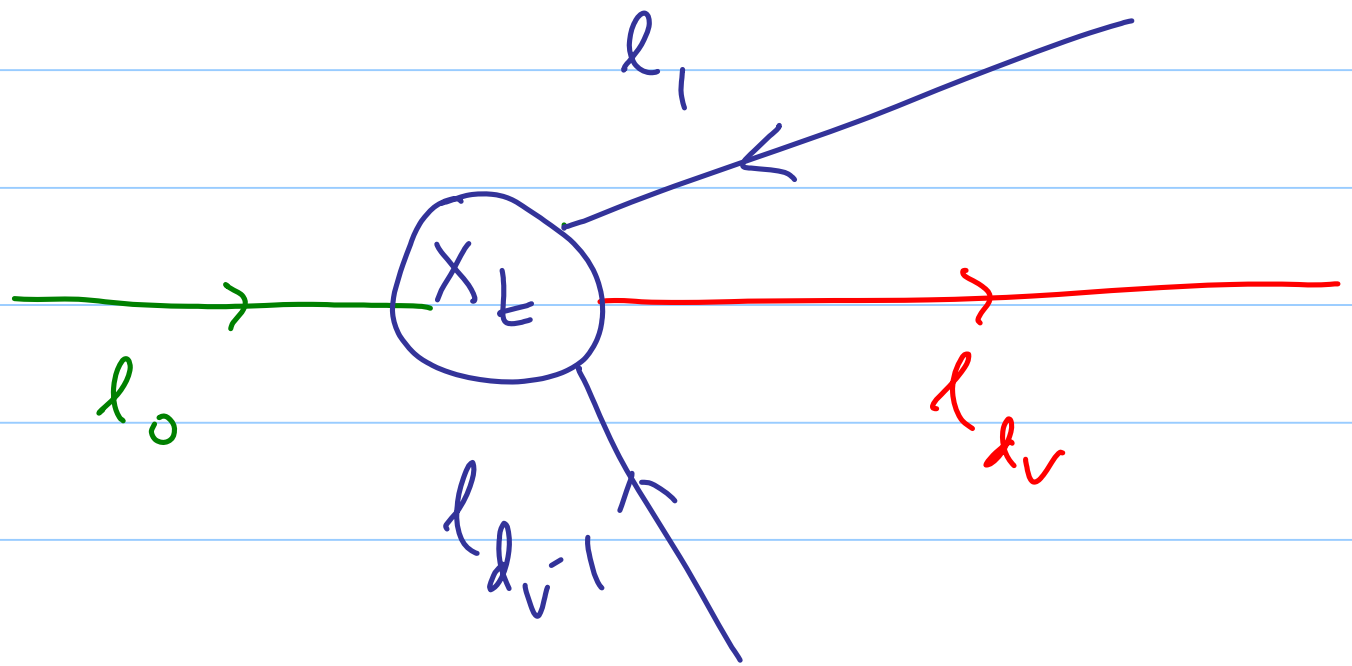
$$\Rightarrow \boxed{\frac{p\left(u_t=0 \mid \bigcup E_i\right)}{p\left(u_t=1 \mid \bigcup E_i\right)} = \prod_{j=0}^{d_v-1} \frac{p\left(u_t=0 \mid E_j\right)}{p\left(u_t=1 \mid E_j\right)}}$$

$$l_j = \ln\left\{\frac{p\left(u_t=0 \mid E_j\right)}{p\left(u_t=1 \mid E_j\right)}\right\} \quad \cdots\cdots (1)$$

$$E_{d_v} = \bigcup_{i=0}^{d_v-1} E_i$$

Taking logs on both sides of (1) gives us:

$$\ell_{d_V} = \sum_{i=0}^{d_V - 1} \ell_i$$



X_{d_c}
$\propto p(X_{d_c} \mid \bigcup_i E_i)$
$X_{d_{c-1}}$
$\propto p(X_{d_{c-1}} \mid E_{d_{c-1}})$
X_1
$\propto p(X_1 \mid E_1)$
C
$X_1 X_2 \cdots X_{d_c}$

$$p(x_{d_c} \mid \cup E_j) \propto \sum_{\sim x_{d_c}} \prod_{j=1}^{d_c-1} p(x_j \mid E_j)$$

$$\sum_{u_{d_c}=0}^{1} p(u_{d_c} \mid \cup E_j)(-1)^{\times u_{d_c}}$$

$$x_j = (-1)^{u_j}$$

$$\propto \sum_{u_{d_c}=0}^{1} (-1)^{\times u_{d_c}} \sum_{u_1 \cdots u_{d_c-1}} \prod_{j=1}^{d_c-1} p(u_j \mid E_j)$$

$$\sum_{j=1}^{d_c-1} u_j = u_{d_c}$$

$$= \alpha \sum_{u_1 \cdots u_{d_c}} \prod_{j=1}^{d_c-1} (-1)^{\times u_j} \, p(u_j \mid E_j)$$

$$\therefore \frac{p(u_{d_c}=0 \mid E_{d_c}) - p(u_{d_c}=1 \mid E_{d_c})}{p(u_{d_c}=0 \mid E_{d_c}) + p(u_{d_c}=1 \mid E_{d_c})}$$

$$= \prod_{j=1}^{d_c-1} \frac{p(u_j=0 \mid E_j) - p(u_j=1 \mid E_j)}{p(u_j=0 \mid E_j) + p(u_j=1 \mid E_j)}$$

In terms of log-likelihood ratios, reduces to:

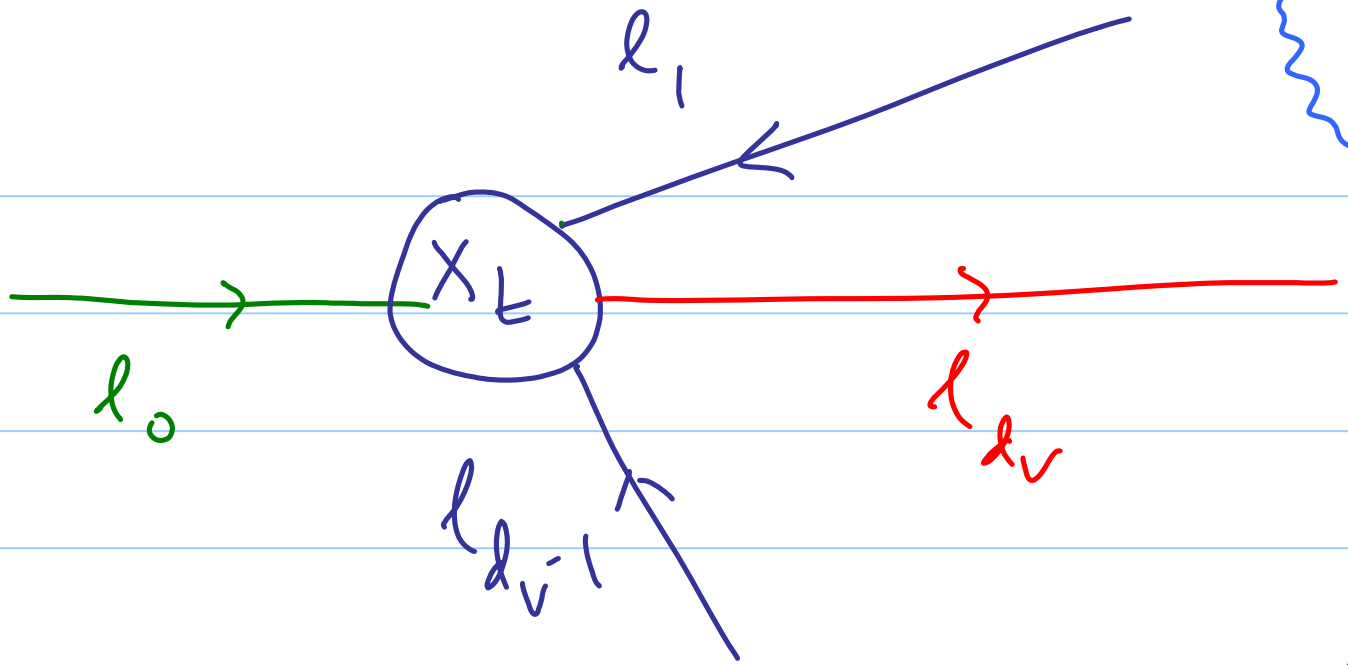
$$\frac{e^{\ell_{d_c}} - 1}{e^{\ell_{d_c}} + 1} = \prod_{j=1}^{d_c - 1} \frac{e^{\ell_j} - 1}{e^{\ell_j} + 1}$$

$$\therefore \quad \tanh\left(\frac{\ell_{d_c}}{2}\right) = \prod_{j=1}^{d_c - 1} \tanh\left(\frac{\ell_j}{2}\right)$$

Lec 34 Density Evolution under BP decoding

Recap

* Discussion of BP decoding of LDPC codes and relation to message passing along a junction tree
* messages (beliefs) expressed in terms of LLRs

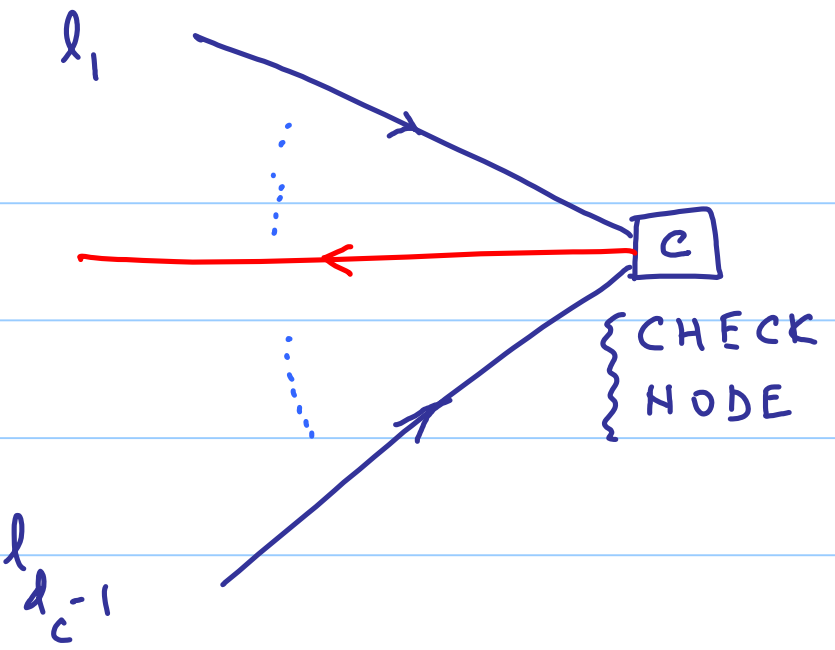


Log-Likelihood Ratios (LLR) at input and output of a variable node

$$\ell_{d_V} = \sum_{i=0}^{d_V-1} \ell_i$$

Note: variable-node symmetry condition is met.

$$\tanh\left(\frac{l_{d_c}}{2}\right) = \prod_{j=1}^{d_c-1} \tanh\left(\frac{l_j}{2}\right) \quad (2)$$



Or,

$$l_{d_c} = 2\tanh^{-1}\left\{\prod_{j=1}^{d_c-1} \tanh\left(\frac{l_j}{2}\right)\right\} \quad (3)$$

Can verify that if $l_j \Rightarrow l_j b_j$

then $b_j \in \{\pm 1\}$

$$l_{d_c} = 2\tanh^{-1}\left\{\prod_{j=1}^{d_c-1} \tanh\left(\frac{b_j l_j}{2}\right)\right\}$$

$$= \left(\prod_{j=1}^{d_c-1} b_j\right) 2\tanh^{-1}\left\{\prod_{j=1}^{d_c-1} \tanh\left(\frac{l_j}{2}\right)\right\}$$

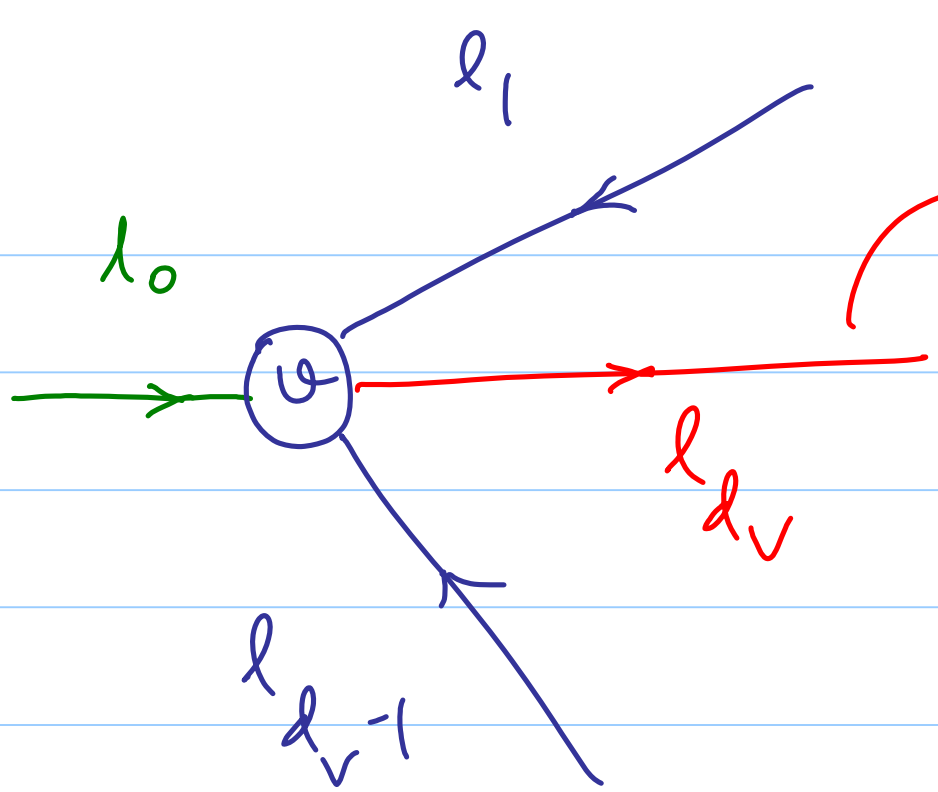
and hence the check-node symmetry condition is also met.

With this we are done with the description of BP-based message-passing-decoding of LDPC codes.

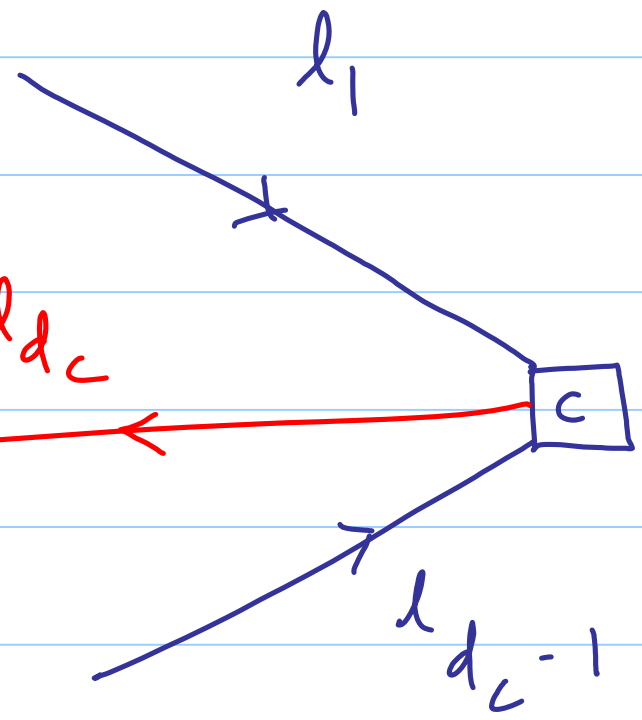
We now turn our attention to performance analysis.

$\Downarrow$

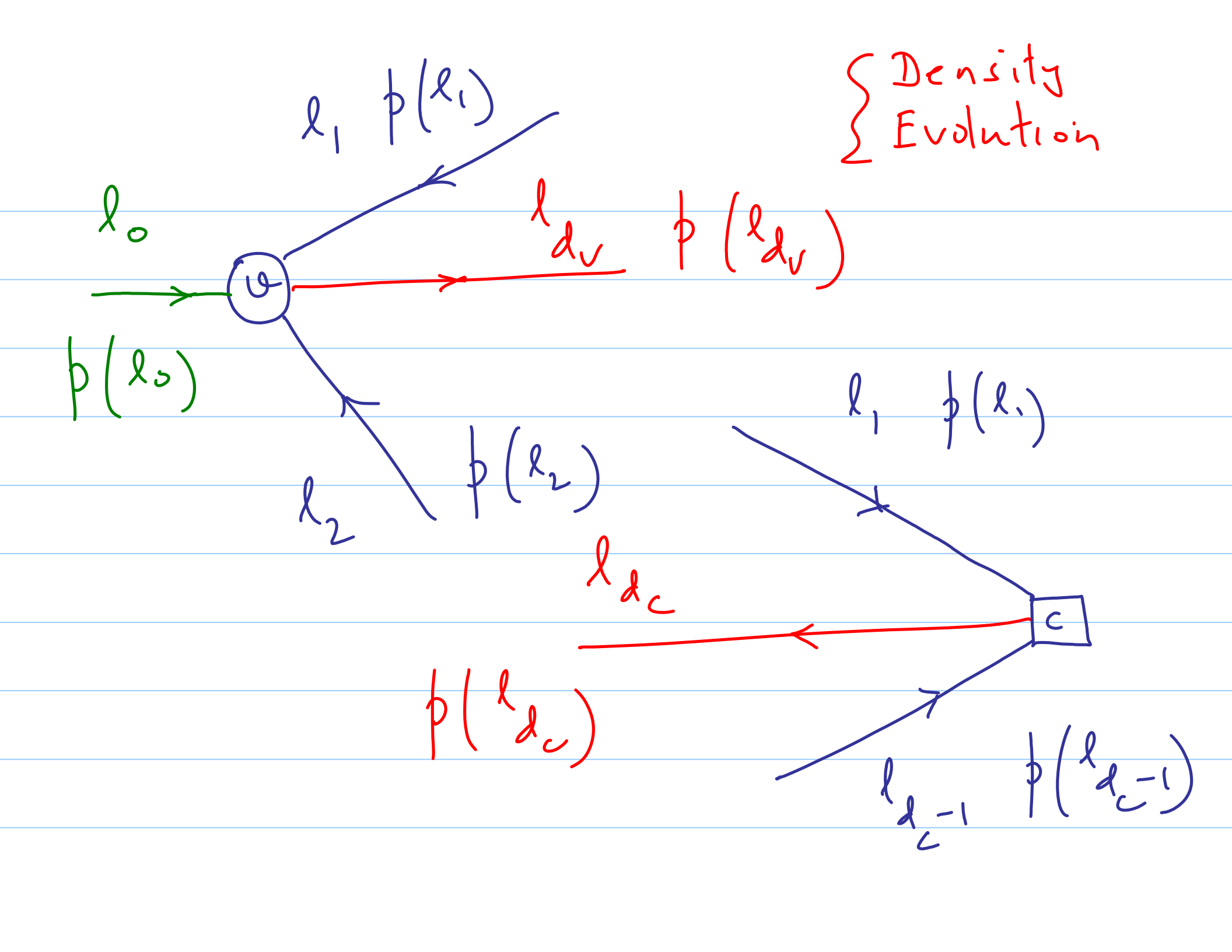
to be carried out using density evolution.



$$l_{d_v} = \ln\left\{\frac{p(x_t=1 \mid E)}{p(x_t=-1 \mid E)}\right\}$$



In carrying out performance analysis we assume that 1 was transmitted


Density Evolution
l_1 $p(l_1)$
l_0
l_{d_v} $p(l_{d_v})$
v
$p(l_0)$
l_2 $p(l_2)$
l_1 $p(l_1)$
l_{d_c}
c
$p(l_{d_c})$
l_{d_c-1} $p(l_{d_c-1})$

Replace $\tanh\left(\frac{\ell}{2}\right)$ by (x, y)

where:

$$x = \operatorname{sgn}(\ell)$$

$$y = -\ln\left|\tanh\left(\frac{\ell}{2}\right)\right|$$

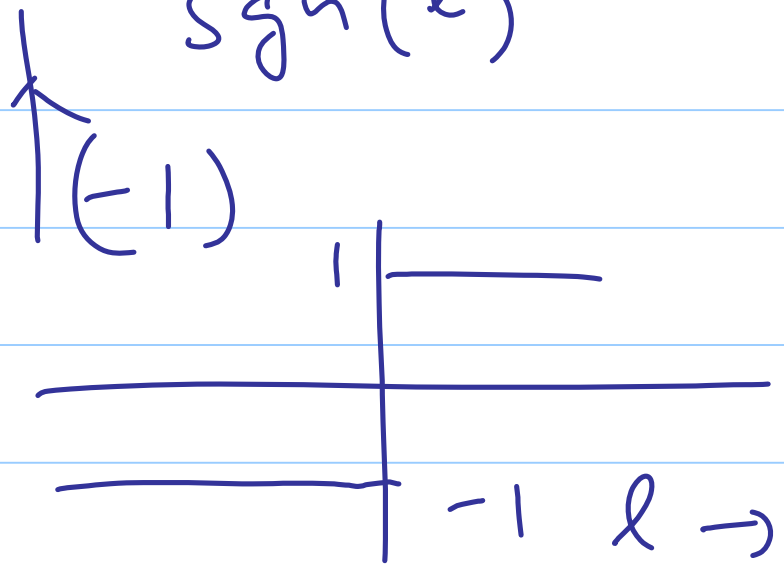
keeps track of the sign of $\tanh\left(\frac{\ell}{2}\right)$

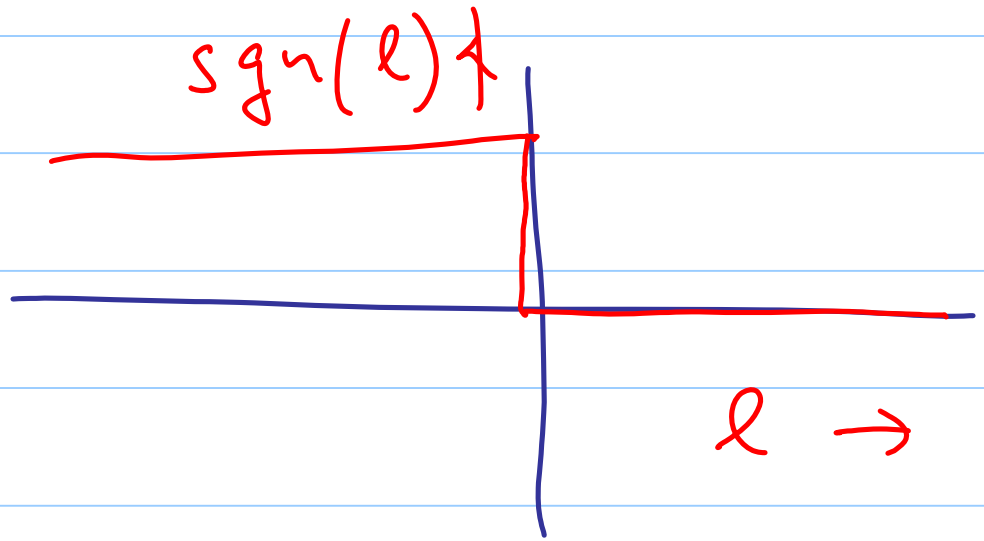
keeps track of the magnitude of $\tanh\left(\frac{\ell}{2}\right)$

Where

$$\operatorname{sgn}(\ell) = \begin{cases} 0 & \ell \geq 0 \\ 1 & \ell < 0 \end{cases}$$

$\operatorname{sgn}(\ell)$





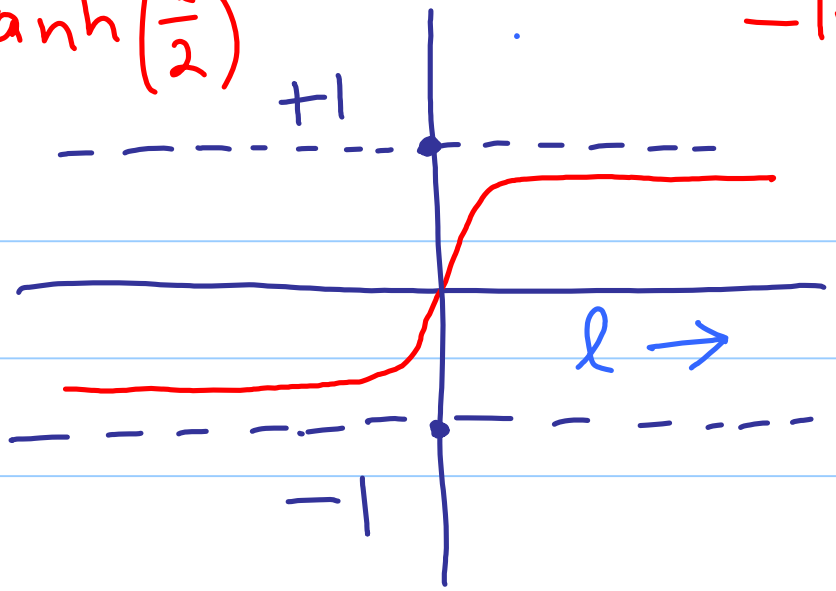
Thus

$$\tanh\left(\frac{l_{d_c}}{2}\right) = \prod_{j=1}^{d_c-1} \tanh\left(\frac{l_j}{2}\right)$$

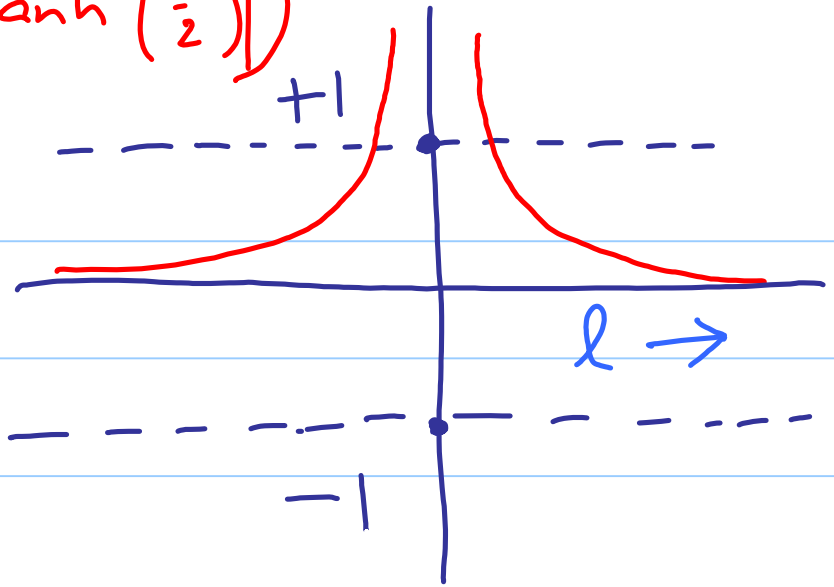
is equivalent to:

$$\left(x_{d_c}, y_{d_c}\right) = \left(\sum_{n=1}^{d_c-1} x_i \pmod 2 \;,\; \sum_{i=1}^{d_c-1} y_i\right)$$

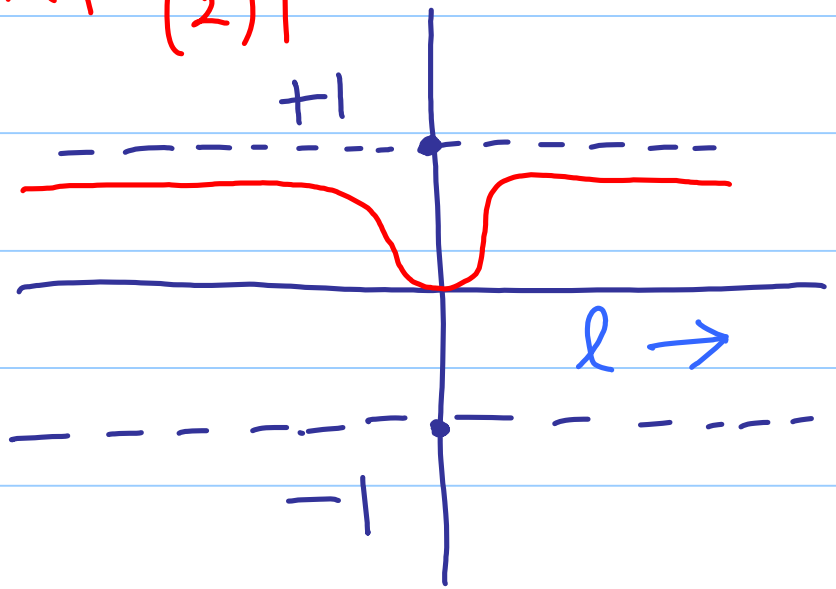
$\tanh\left(\frac{\ell}{2}\right)$



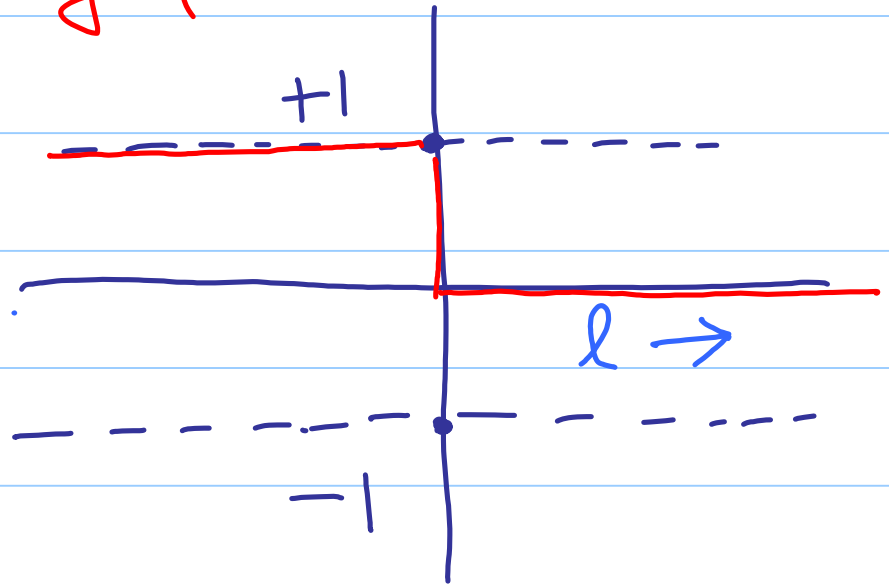
$-\ln\left(\left|\tanh\left(\frac{\ell}{2}\right)\right|\right)$



$\left|\tanh\left(\frac{\ell}{2}\right)\right|$



$\operatorname{sgn}(\ell)$



γ_2
T^{-1}
(x_1, x_2, x_3)
γ_1
T^{-1}
(x_1, x_2, x_3)
γ_2
T^{-1}
(x_1, x_2, x_3)
C

$$\omega \longmapsto \theta \longmapsto x \longmapsto \psi^{-1}(x)$$

$$\omega \longmapsto \theta \longmapsto \psi^{-1}(x)$$

$$\omega \longmapsto \omega \times \theta \longmapsto \psi^{-1}(x)$$

$$\omega \longmapsto \theta \longmapsto \psi^{*}(x)$$

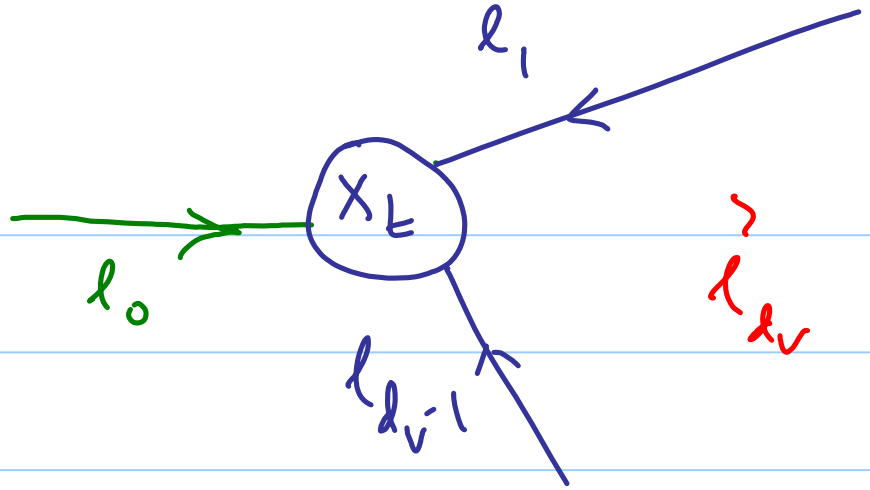
$$\omega \longmapsto \theta \longmapsto \psi^{*}(x)$$

$$\omega \longmapsto \theta \longmapsto \psi^{*}(x)$$

corresponding map in terms of density functions

t^{th} iteration

(open node at a check)



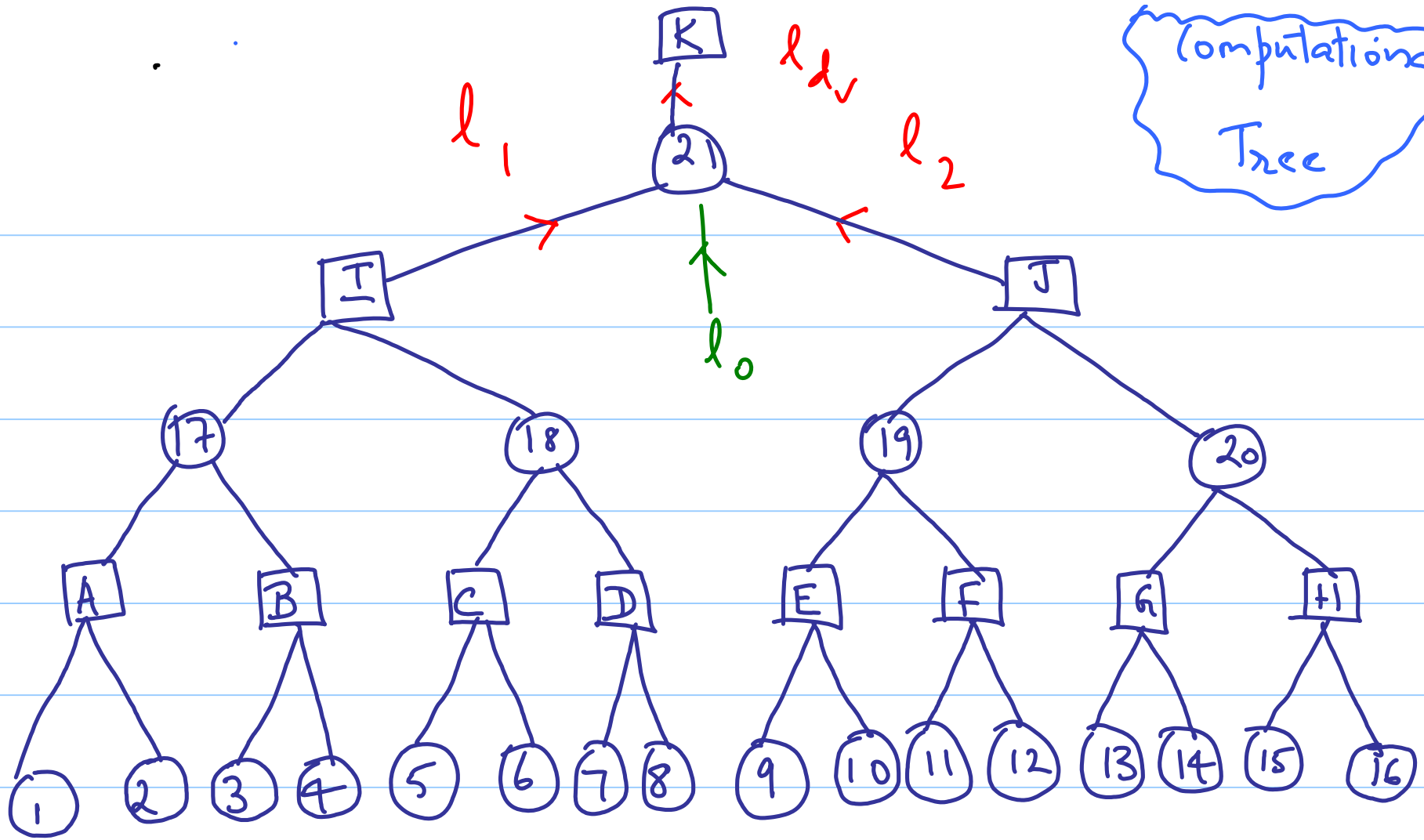
$$\ell_{d_v} = \sum_{i=0}^{d_v - 1} \ell_i$$

Assume that $\underline{1}$ was transmitted.

The $\{\ell_i\}$ are random since they are a function of the particular channel realization. Consider ℓ rounds

$\int$ 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 different l_i, $1 \le i \le d_v - 1$ are linearly

independent as they are functions of disjoint subsets of received variables $\{y_j\}_{j=1}^{n}$.



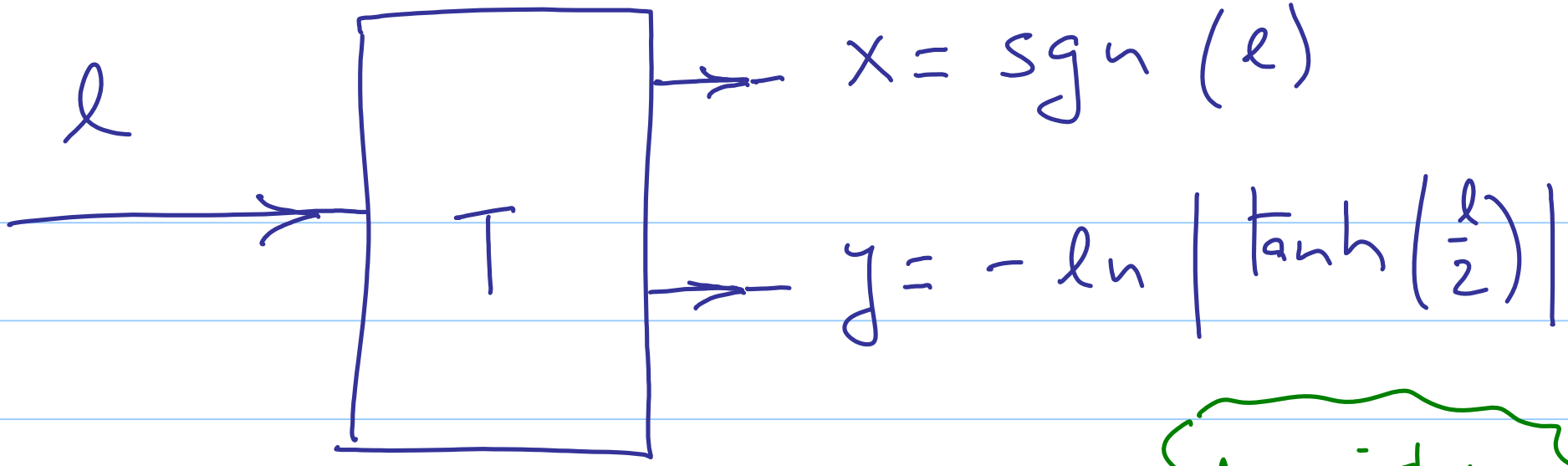
Computational Tree

Scenario where the nbhd is tree-like.

and hence by symmetry

$$p_{l_{t_v}} = \Gamma^{-1}\left\{\left[\Gamma\left\{p_{l_i}\right\}\right]^{d_v - 1}\right\}$$

this is a description of density evolution at a variable node.



densities after the transfmn

$$p_{XY}(0,y) = \frac{p_L\left(-\ln \tanh\left(\frac{y}{2}\right)\right)}{\sinh(J)}$$

$$p_{XY}(1,y) = \frac{p_L\left(\ln \tanh\left(\frac{y}{2}\right)\right)}{\sinh(J)}$$

Define

$$\phi_j(\lambda, s) = E\left\{ (-1)^{\lambda x_j} e^{-s y_j} \right\}$$ joint char. fn.

$$= \sum_{u=0}^{1} (-1)^{\lambda u} \int e^{-sy} p_{x_j r_j}(u, y)\, dy$$

$$= \mathcal{L}\left(p_{x_j r_j}(0, y) \right) + (-1)^{\lambda} \mathcal{L}\left(p_{x_j r_j}(1, y) \right)$$

Laplace transform

Since the $\{l_j\}$ and hence the $\{(x_j, y_j)\}$ are statistically independent it follows that:

$$x_{d_c} = \sum_j x_j \pmod 2$$

$$y_{d_c} = \sum_j y_j$$

$$\Phi_{d_c}(\lambda, s) = E\left\{(-1)^{\lambda x_{d_c}} e^{-s y_{d_c}}\right\}$$

$$= \prod_{j=1}^{d_c - 1} \Phi_j(\lambda, s)$$

Hence

$$\left\{ \begin{array}{l} x \cdot p_1 \\ \vdots \\ x \cdot p_r \end{array} \right\} \Vdash (x, p) \left\{ \begin{array}{l} p_1^{-1} \\ \vdots \\ p_r^{-1} \end{array} \right\} c^{-1}$$

$$\Vdash \left\{ \begin{array}{l} p_1 \cdot \theta \\ \vdots \\ p_r \cdot \theta \end{array} \right\} (s, x) \left\{ \begin{array}{l} \theta^{-1} \cdot p_1^{-1} \\ \vdots \\ \theta^{-1} \cdot p_r^{-1} \end{array} \right\}$$

$$\Vdash \theta \, (s, x) \, \theta^{-1}$$

$$\Vdash (s, x)$$

The last computation use that:

$$\phi_{d_c}(\lambda s)$$

$$= \int \left(\phi_{x_{d_c} r_{d_c}}(0,y) \right) + \hat{\lambda}(-1) \int \left(\phi_{x_{d_c} r_{d_c}}(1,y) \right)$$

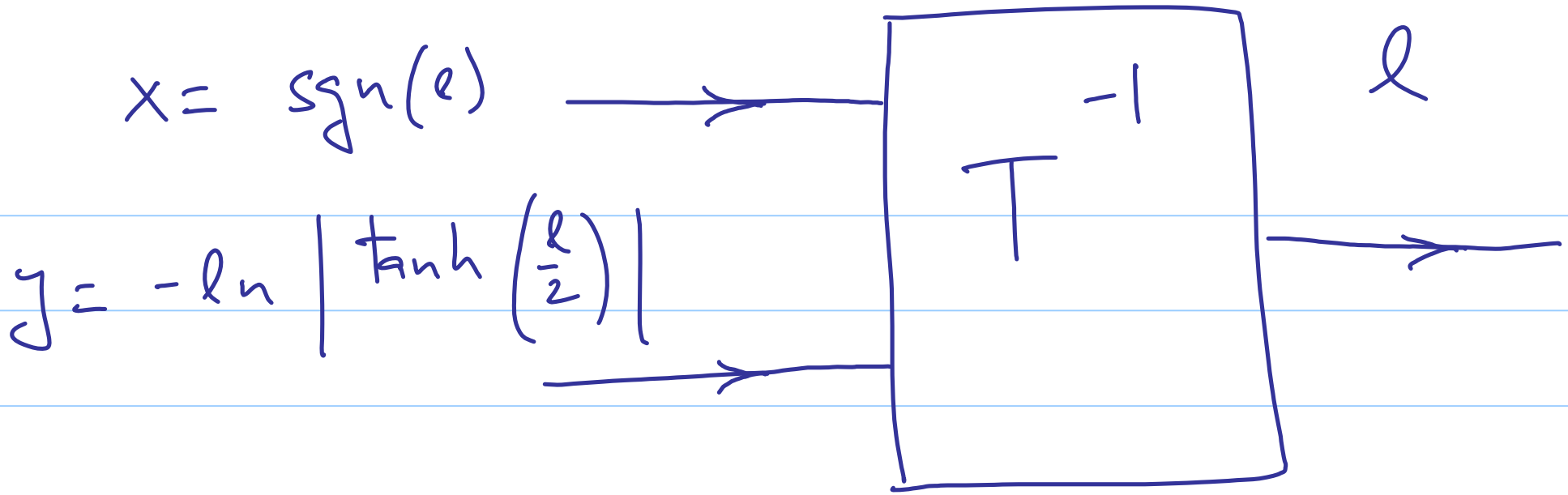
$$\therefore \phi_{x_{d_c} r_{d_c}}(0,y) =$$

$$\mathcal{I}^{-1}\left\{ \frac{\phi_{d_c}(0,s) + \phi_{d_c}(1,s)}{2} \right\}$$

$$\phi_{x_d r_{d_c}}(1,J) =$$

$$\mathcal{I}^{-1}\left\{ \frac{\phi_{d_c}(0,s) - \phi_{d_c}(1,s)}{2} \right\}$$

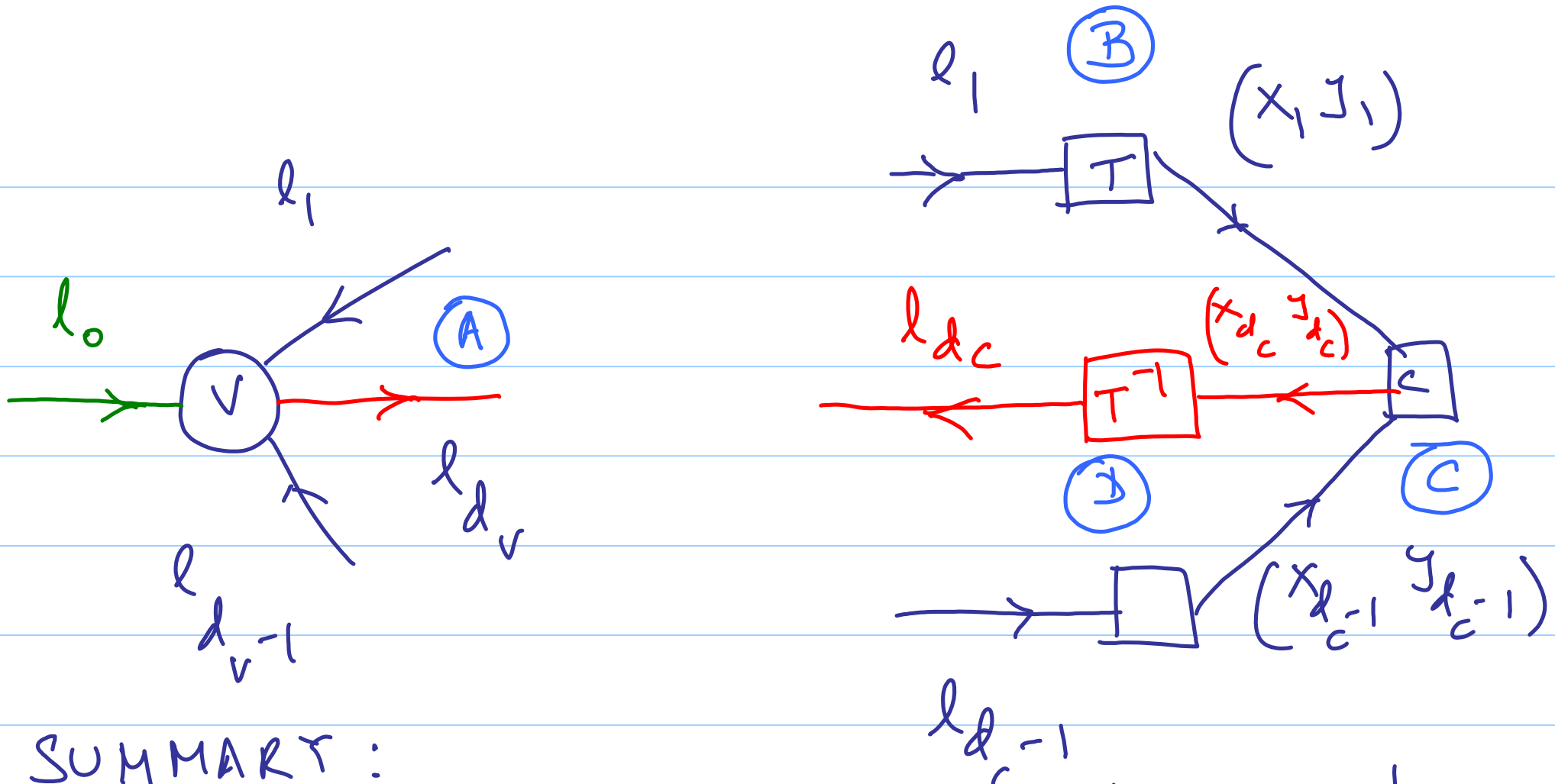
Letting $x = x_{d_c}$ $r = r_{f_c}$ $l = l_{d_c}$:



Finally

$$p_L(\ell) = \frac{p(0, -\ln \tanh(\frac{\ell}{2}))}{\sinh(\ell)} \qquad \ell > 0$$

$$= \frac{p(1, -\ln \tanh(\frac{-\ell}{2}))}{-\sinh(\ell)} \qquad \ell \leq 0$$



SUMMARY:

density evolution was accomplished by tracking densities across location's:

(A) (B) (C) (D).

Lec 35 { Convergance { 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"

T. J. Richardson & Ruediger L. Urbanke

{IEEE Trans. on. Inform. Theory

Feb. 2001

Theorem For any $\epsilon > 0$,

a) $$P_x\left\{ \left| Z - \mathbb{E}(Z) \right| > \frac{n d_v \epsilon}{2} \right\} \leq 2e^{-\beta \epsilon^2 n} \quad (1)$$

(concentration around the mean) (1)

b) For any $\epsilon > 0$ and $n > \frac{2\gamma}{\epsilon}$,

$$\left| \mathbb{E}(Z) - n d_v p \right| < \frac{n d_v \epsilon}{2} \quad (2)$$

(convergance to the cycle-free case)

c) For any $\epsilon > 0$ and $n > \frac{2r}{\epsilon}$,

we have:

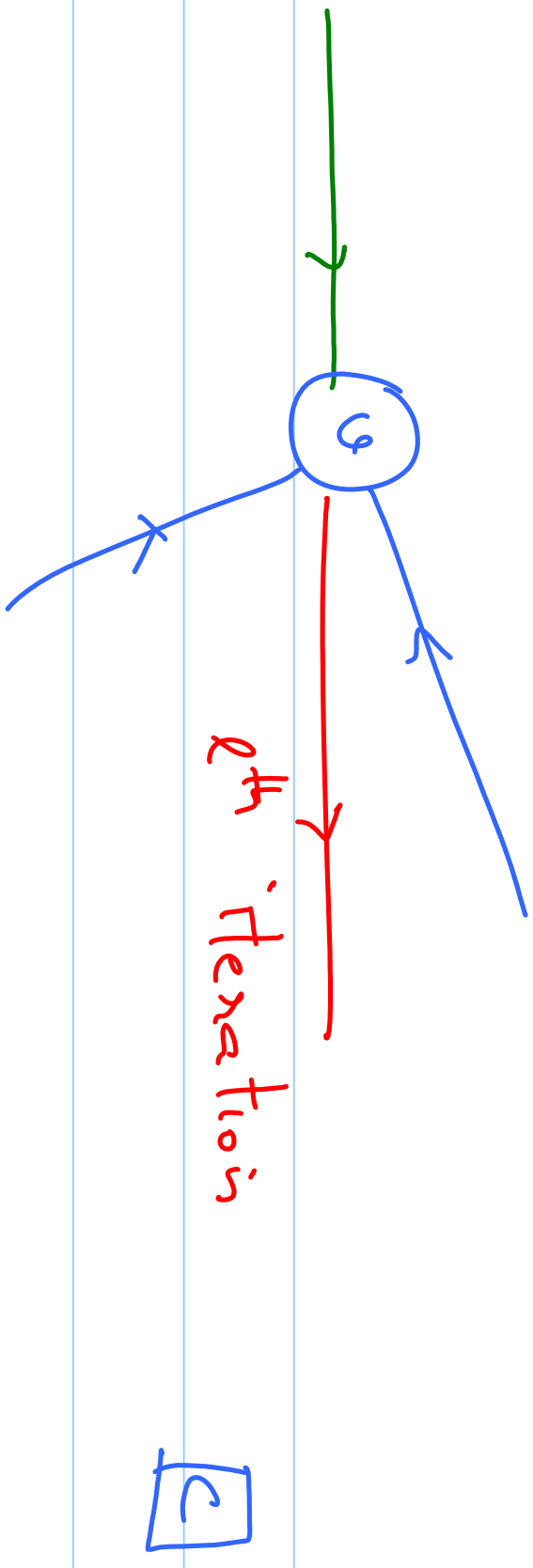
$$P_{\mathcal{H}}\left\{ |z - nd_V p| > nd_V \epsilon \right\} \leq 2e^{-\beta \epsilon^2 n} \quad (3)$$

(Concentration around the cycle-free case)

where:

(i) the probabilities are computed over all choices of (d_v, d_c)-regular codes and over all channel realizations

(ii) z = # of incorrect messages passed from the nd_v variable nodes to the rd_v check nodes in the ℓ^{th} iteration. (r denotes the # of check nodes and $nd_v = rd_c$).


6
4th Iteration

p below is the probability of an incorrect message being passed during the ℓ^{th} iteration in the tree-like case.

b) For any $\epsilon > 0$ and $n > \frac{2\gamma}{\epsilon}$,

$$\left| \mathbb{E}(z) - n d_v p \right| < \frac{n d_v \epsilon}{2} \qquad (2)$$

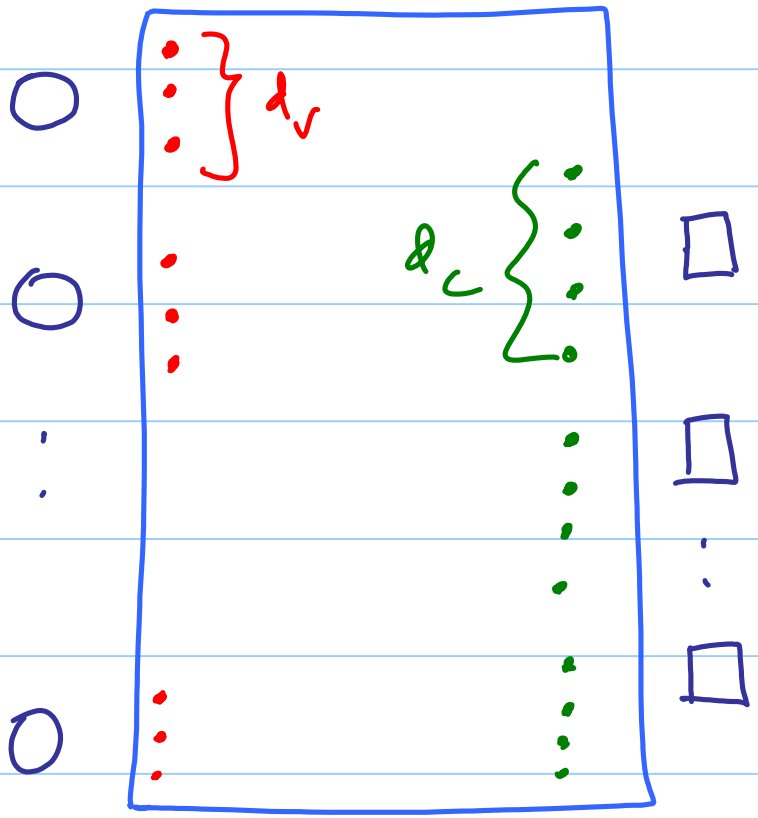
p derived from density evolution

Also, in the theorem,

$\beta = \beta(d_v, d_c, \ell)$ and $\gamma = \gamma(d_v, d_c, \ell)$

are constants.

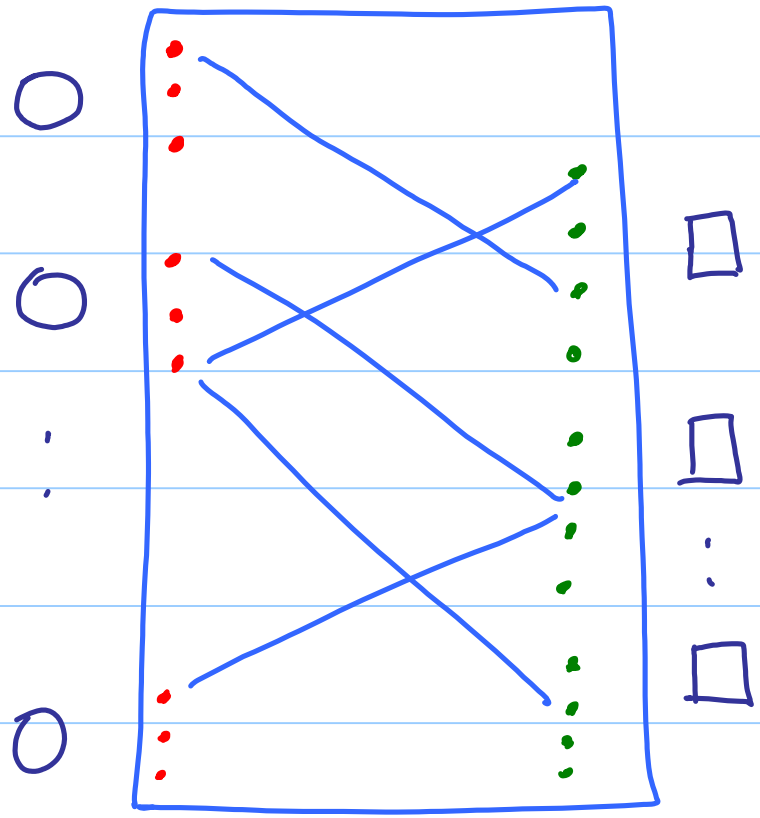
The ensemble of (d_v, d_c) - regular LDPC codes.



n variable nodes, each of degree $= d_v$

r check nodes, each of degree $= d_c$

each variable is associated with d_v "sockets"



each check node is linked to d_c "sockets"

$n d_v = r d_c$

$(n d_v)!$ 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^{th} edge e_i during the ℓ^{th} iteration. $z_i \in \{0,1\}$.

$$z = \sum_{i=1}^{n d_v} z_i$$

$$\therefore \mathbb{E}\{z\} = \sum_{i=1}^{n d_v} \underbrace{\mathbb{E}\{z_i\}}_{\Downarrow} = n d_v \mathbb{E}\{z_1\} \quad \text{(by symmetry)}$$

focus on this

$$\mathbb{E}\{z_1\} = \mathbb{E}\{z_1 \mid N_{c_1}^{2\ell} \text{ is T-L}\} \to \phi$$

T-L

$$P_r\{N_{c_1}^{2\ell} \text{ is T-L}\}$$

true

like

$$1 - \frac{\gamma}{n} \le \cdot \le 1 + \mathbb{E}\{z_1 \mid N_{c_1}^{2\ell} \text{ is not T-L}\} \to \le 1$$

$$\le \frac{\gamma}{n} \leftarrow P_r\{N_{c_1}^{2\ell} \text{ is not T-L}\}.$$

When n is large, turns out that

$$\{P_r\{N_c^{2\ell} \text{ is T-L}\} \ge 1 - \frac{\gamma}{n}\}$$

γ is a constant.

$$p - \frac{\gamma}{n} \leq p\left(1 - \frac{\gamma}{n}\right) \leq \mathbb{E}\{z_1\} \leq p + \frac{\gamma}{n}$$

$$\therefore \quad \left| \mathbb{E}\{z_1\} - p \right| \leq \frac{\gamma}{n}$$

$$\therefore \quad \left| \mathbb{E}\{z\} - n d_v p \right| \leq n d_v \left(\frac{\gamma}{n}\right)$$

We assume n large enough so that

$$\frac{\gamma}{n} < \frac{\epsilon}{2}$$

$$\therefore \quad \boxed{\left| \mathbb{E}\{z\} - n d_v p \right| \leq \frac{n d_v \epsilon}{2}}$$

This proves ②. We will skip the lengthy proof of ①.

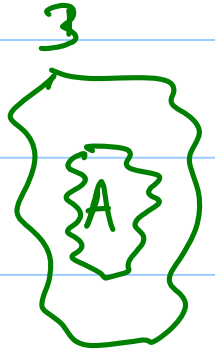
To prove ③ given ① and ②, we argue as follows:

$$P_r\left\{ \left| z - n d_v \phi \right| > n d_v \epsilon \right\}$$

$$= P_r\left\{ \left| \left[z - \mathbb{E}\{z\} \right] + \left[\mathbb{E}\{z\} - n d_v \phi \right] \right| > n d_v \epsilon \right\}$$

but for large enough n when

$$\left| \mathbb{E}\{z\} - n d_v \phi \right| < \frac{n d_v \epsilon}{2}$$



$$\leq P_R \left\{ \left| z - \mathbb{E}\{z\} \right| > \frac{n d_V \epsilon}{2} \right\}$$

$$- \beta \epsilon^2 n$$

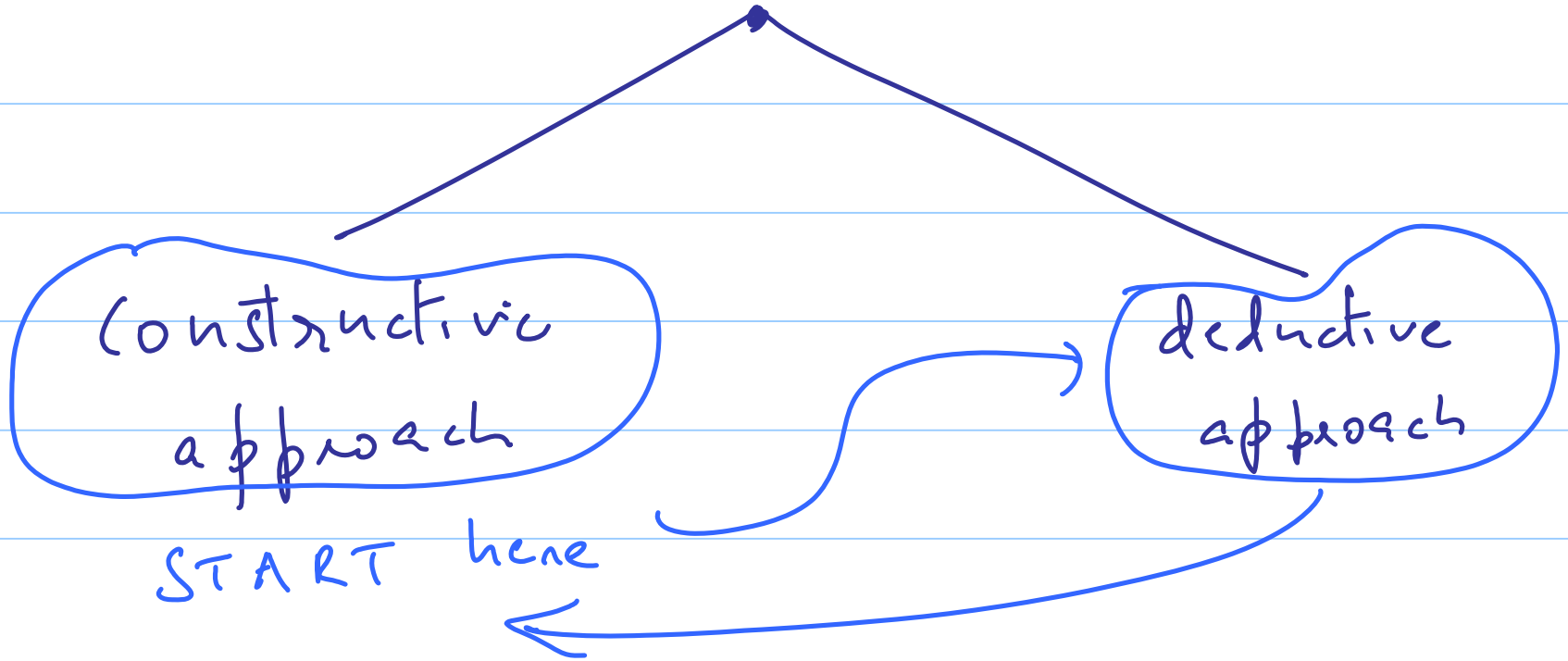
$$\leq 2c \qquad \text{from part a)}$$

Lec 36 — A construction for Finite Fields

Recap

* reviewed discussion of density evolution w.r.t the BP decoding of LDPC codes
* concentration & convergence theorem

* finite fields are the basis upon which the widely-used classes of BCH and Reed-Solomon codes are built



Eg of a finite field

$\mathbb{C} = \mathbb{R}[i]$, $i^2 = -1$ → imaginary $\sqrt{-1}$

{ collection of all polynomials in i having real coefficients

$\mathbb{R}[x] / (x^2+1)$

$\mathbb{R}[x]$ (modulo x^2+1)

$\mathbb{R}[x]$ the ring of all polynomials in x over the real numbers

$\mathbb{Z}$ (mod n)

$\mathbb{Z}$ (mod p) ⇒ (p prime)

⇐ forms a field $\mathbb{Z}_p$

In $\mathbb{R}[x]/(x^2+1)$,

$$x^2 = \underbrace{(x^2+1)}_{0} - 1 = -1$$

$\mathbb{R}[i]$ $\qquad$ $i^2+1=0 \Rightarrow i^2 = -1$

$$= \{ a + ib \mid a, b \in \mathbb{R} \}$$

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 $m = 1$, $\mathbb{Z}_p$

$\mathbb{Z}_p = \{$ set of integers modulo $p \}$

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 $= 1$

Ex. $f(x) = \sum_{i=0}^{m} f_i x^i$

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 $p = 2$ $\quad \mathbb{F}_2 = \{0, 1\}$

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

$g_1 \sim g_2 \iff f(x)$	$g_1(x) - g_2(x)$

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

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

$+$: i.e. $\triangleq 0$

$\cdot$: i.e. $= 1$

$F_{\{\}}$ contains 1, $\therefore$ it contains

$1,\ 1+1,\ \underbrace{1+1+1}_{3},\ 1+1+1+1,\ \cdots$

This list must repeat entries.

$\therefore$ $\boxed{m = n}$ some $n > m$.

$\underbrace{1+1+1+\cdots+1}_{m \text{ times}}$

$\Rightarrow$ $\boxed{(n-m)1 = 0}$

Let p be the smallest integer s.t.

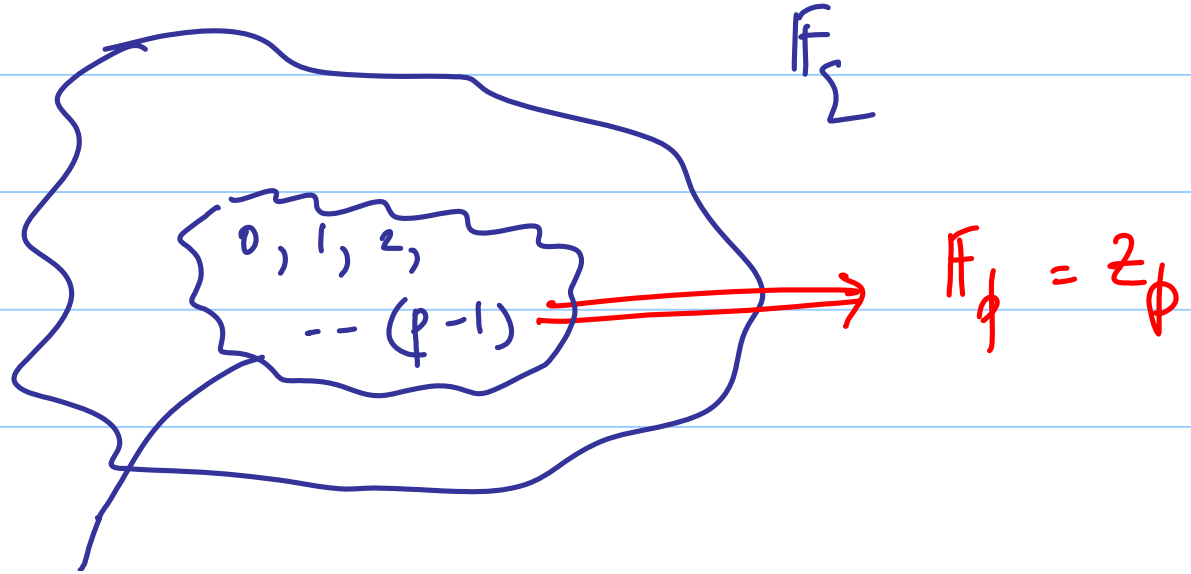
$p \cdot 1 = 0$. Then, p must be prime,

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 modulo a prime p.

Lec 37

Finite Fields

A deductive approach

Recap

* Motivating a construction for finite fields

* $\mathbb{F}_p[x] \,/\, (f(x))$

  $\deg(f(x)) = m.$

$\mathbb{F}_p[x] / (f(x))$ { is a collection of equivalence classes

where we define

$$g_1(x) \sim g_2(x) \quad \text{if} \quad f(x) \mid g_1(x) - g_2(x)$$

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

$$R \triangleq \mathbb{F}_p[x] / (f(x))$$

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 $(R, +)$ is an Abelian group

- CLOSURE ✓
- i.e. ✓
- ASSOCIATIVE ✓
- INVERSE ✓
- COMMUTATIVE ✓

$$[a(x)] + [b(x)] = [a(x) + b(x)].$$

$$\text{i.e.} = [0] = \{\text{set of all multiple of } f(x)\}$$

$$\text{inverse of } [a(x)] = [-a(x)]$$

Step 2 Under $\cdot$ $(R, \cdot)$ satisfies:

- CLOSURE ✓
- i.e. ✓
- commutative ✓
- ASSOCIATIVE ✓
- inverse

$$[a(x)] \cdot [b(x)] = [a(x) \cdot b(x)]$$

$$\text{i.e.} \quad = [1].$$

We will demonstrate through example, the presence of a multiplicative inverse.

Eg $\quad p = 2 \qquad f(x) = x^4 + x + 1$

$$[a(x)] = [(x+1)]$$

$$[a(x)]^{-1} = ?$$

$$[a(x)][b(x)] = [a(x)\ b(x)]$$
$$= [1].$$

We make use of the extended Euclidean division algorithm (EDA).

	$f(x) = x^4 + x + 1$	$(X+1)$	quotient
	1	0	
$x^4 + x + 1$			
$X+1$	0	1	$x^3 + x^2 + x$
(Rem) 1	1	$x^3 + x^2 + x$	
0			

$$
\begin{array}{r}
x^3 + x^2 + x \\
\hline
x+1 \;\big)\; x^4 + x + 1 \\
x^4 + x^3 \\
\hline
x^3 + x + 1 \\
x^3 + x^2 \\
\hline
x^2 + x + 1 \\
x^2 + x \\
\hline
1
\end{array}
$$

$$\therefore 1 = 1(x^4 + x + 1) + (x+1)(x^3 + x^2 + x)$$

$$\therefore [1] = [(x+1)(x^3 + x^2 + x)]$$

$$= [(x+1)][(x^3 + x^2 + x)]$$

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.

Eg (of the construction of a finite field of size = 16 as well as of its representation in terms of an imaginary element α).

$p = 2 \quad m = 4 \quad f(x) = x^4 + x + 1$

$$\mathbb{F}_p[x] / (f(x)) = \mathbb{F}_2[x] / (x^4 + x + 1)$$

We introduce the imaginary element α which is a zero of x^4+x+1:

$$\boxed{\alpha^4 + \alpha + 1 = 0}$$

$$\mathbb{F}_2[x] \Big/ (x^4+x+1) = \mathbb{F}_2[\alpha]$$

$$\mathbb{F}_2[\alpha] = \left\{ \sum_{i=0}^{3} a_i \alpha^i \;\middle|\; a_i \in \{0,1\} \right\}$$

0	$\alpha^4 = \alpha + 1$	$\alpha^{11} = \alpha^3 + \alpha^2 + \alpha$
$\alpha^0 = 1$	$\alpha^5 = \alpha^2 + \alpha$	$\alpha^{12} = \alpha^4 + \alpha^3 + \alpha^2 = \alpha^3 + \alpha^2 + \alpha + 1$
$\alpha^1 = \alpha$	$\alpha^6 = \alpha^3 + \alpha^2$	$\alpha^{13} = \alpha^4 + \alpha^3 + \alpha^2 + \alpha = \alpha^3 + \alpha^2 + 1$
$\alpha^2 = \alpha^2$	$\alpha^7 = \alpha^4 + \alpha^3$	$\alpha^{14} = \alpha^4 + \alpha^3 + \alpha = \alpha^3 + 1$
$\alpha^3 = \alpha^3$	$= \alpha^3 + \alpha + 1$	
	$\alpha^8 = \alpha^4 + \alpha^2 + \alpha$	$\alpha^{15} = \alpha^4 + \alpha = \alpha + \alpha + 1$
	$= \alpha^2 + 1$	$= 1$!!
	$\alpha^9 = \alpha^3 + \alpha$	
	$\alpha^{10} = \alpha^4 + \alpha^2$	
	$= \alpha^2 + \alpha + 1$	

From the table, we observe that the finite field also has representation in terms of the element zero and the various powers of α:

$$\mathbb{F}_{2^4} = \{0\} \cup \{\alpha^i \mid 0 \leq i \leq 14\}$$

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
  - multiplication distributes over addition
  - closure
  - Assoc.
  - i.e. $= 1$
  - inverse
  - Comm.

$1 \in \mathbb{F}_q \Rightarrow$ all $1, \; 1+1, \; 1+1+1, \; \ldots \; 1+1+1, \ldots \in \mathbb{F}_q$

$\Rightarrow m = n$ some $n > m$

$\triangleq \underbrace{1+1+\ldots+1}_{m \text{ times}} \overset{!!}{=} 0 \Rightarrow \boxed{n-m=0}$

Defn. The characteristic p of a finite field $\mathbb{F}_q$ is the smallest integer p s.t.

$$p = \underbrace{1+1+\cdots+1}_{p \text{ times}} = 0 \quad \text{in } \mathbb{F}_q$$

Thm 1 The char. p is a prime.

pf. Suppose $p = a \cdot b$, $1 < a, b < p$

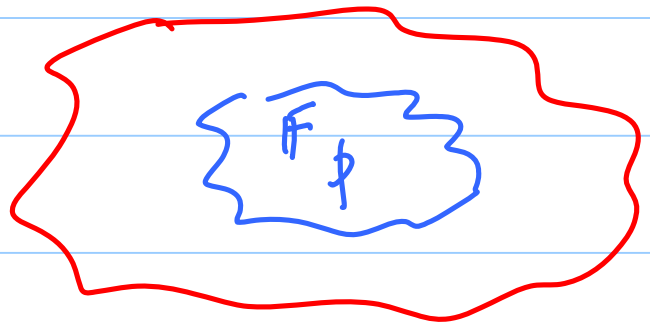
Then $a \cdot b = 0 \Rightarrow a = 0$ or $b = 0$

which contradicts the minimality of p.

$\Leftarrow$ //

$\mathbb{F}_q$ contains the set $\{0, 1, 2, \ldots, p-1\}$

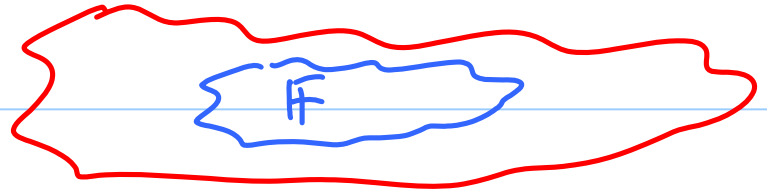
the arithmetic used to operate on these elements is $(\mathrm{mod}\ p)$ arithmetic since $p \equiv 0$ in $\mathbb{F}_q$. Hence $\mathbb{F}_q$ contains a copy of $\mathbb{F}_p$.



$|$

$\mathbb{F}_p = \mathbb{Z}_p$

It can be shown that $\mathbb{E}$, $\mathbb{F}$ are fields and $\mathbb{E} \supseteq \mathbb{F}$, then $\mathbb{E}$ is a vector space over $\mathbb{F}$.



It follows that $\mathbb{F}_\Sigma$ is a vector space over $\mathbb{F}_p$. Let m be the dimension of this vector space. Then since

$$\mathbb{F}_\Sigma = \left\{ \sum_{i=1}^{m} a_i \underbrace{r_i}_{\in \mathbb{F}_\Sigma} \;\middle|\; a_i \in \mathbb{F}_p \right\},$$

where $\{r_1 \dots r_m\}$ is a basis of $\mathbb{F}_q | \mathbb{F}_p$,

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, $m \geq 1$ (moreover p is the characteristic of $\mathbb{F}_q$)

$q \in \{2, 2^2, 3, 3^2, 2^3, 5, \dots\}$

Multiplicative Structure of $\mathbb{F}_q$

Let $\beta \in \mathbb{F}_q^* \triangleq \{ x \in \mathbb{F}_q \mid x \neq 0 \}$.

Then $\mathbb{F}_q^*$ contains

$$1 = \beta^0, \quad \beta, \quad \beta^2, \quad \ldots \quad \beta^a \quad \ldots \quad \beta^b \quad \ldots$$

by finiteness of $\mathbb{F}_q$, $\beta^a = \beta^b$ for some integers $b > a \Rightarrow \beta^{b-a} = 1$.

This motivates:

Defn. The (multiplicative) order of $\beta \in \mathbb{F}_q^*$ is the smallest nonzero exponent e s.t. $\beta^e = 1$

Lemma 1 Let $\beta \in \mathbb{F}_q^*$ have order e. Then $\beta^\ell = 1$ iff $e \mid \ell$.

Pf Let $\ell = ue + v$, $0 \le v \le e-1$

(u: quotient, v: remainder)

Then, $\beta^{\ell} = \beta^{ue} \cdot \beta^{v} = (\beta^{e})^{u} \cdot \beta^{v}$

$= (1)^{u} \cdot \beta^{v} = \beta^{v} = 1$

but this contradicts the minimality of e unless $v = 0$. Hence $e \mid \ell$ ///

Lemma 2 Let $\beta \in \mathbb{F}_q^*$ have order e.

Then β^ℓ has order $= \dfrac{e}{(\ell, e)}$.

Eg $e = 15 \quad \ell = 10$

order $(\beta^{10}) = \dfrac{15}{(15,10)} = 3$

Pf. (Exercise, straightforward).

Lemma 3 Let $\beta, \gamma \in \mathbb{F}_q^*$ have orders a, b s.t. $(a, b) = 1$. Then

$(\beta\gamma)$ has order $= ab$.

Pf. Let $(\beta\gamma)$ have order $= \ell$.

Then $\beta^{\ell} = \gamma^{-\ell}$

$\Rightarrow \dfrac{a}{(a,\ell)} = \dfrac{b}{(b,\ell)}$ { since γ, γ^{-1} have the same order

$\Rightarrow a(b,\ell) = b(a,\ell)$

$\therefore a \mid (a,\ell) \Rightarrow a \mid \ell$

Similarly $b \mid (b,\ell) \Rightarrow b \mid \ell$

$\therefore ab \mid \ell$.

On the other hand,

$$(\beta\gamma)^{ab} = (\beta^a)^b \cdot (\gamma^b)^a = 1$$

hence $\ell \mid ab$. $\therefore \ell = ab$ ///

Amongst all the elements in $\mathbb{F}_\Sigma^*$, let β have maximal order $= r$.

Claim: Every element $\theta \in \mathbb{F}_\Sigma^*$ has

order dividing n.

If Let θ have order $= \ell$, $\ell \nmid n$.

($\Leftarrow$ via Lemma 3)

Then we can write $n = p_1^{e_1} \cdots p_s^{e_s}$

$$\ell = p_1^{a_1} \cdots p_s^{a_s}$$

with $a_i > e_i$ some i. WOLOG assume $i \geq 1$.

Then, $\left.\begin{aligned} n &= p_1^{e_1}(n_1) \\ \ell &= p_1^{a_1}(\ell_1) \end{aligned}\right\}$ (say)

Now consider $\beta^{p_1^{e_1}} \cdot \theta^{\ell_1}$ has order $= \frac{\ell}{(\ell_1, \ell_1)} = p_1^{q_1}$

has order

$$= \frac{r}{(r_1, p_1^{e_1})} = \frac{r}{p_1^{e_1}} = r_1$$

Since $(r_1, p_1^{q_1}) = 1$, it follows

that $\beta^{p_1^{e_1}} \cdot \theta^{\ell_1}$ has order

$= \phi_1^{a_1} \cdot r_1 > \phi_1^{b_1} r_1 = r$

which contradicts the maximality of r.

Hence, $\ell \mid r$. ///

Lemma The maximal order of an element $\beta \in \mathbb{F}_\Sigma^*$ is $q-1 = \phi^m - 1$.

Pf. Let β have maximal order $= r$.

To show that $r = p^m - 1$. Since $\theta \in \mathbb{F}_q^*$

$\Rightarrow \theta$ has order dividing r,

$$\theta^r = 1 \Rightarrow x^r - 1 \Big|_{x=\theta} = 0.$$

It follows that every nonzero element in $\mathbb{F}_q$ is a zero of $(x^r - 1)$.

$\therefore \quad r \geq p^m - 1.$

On the other hand, consider

$$1, \beta, \beta^2, \ldots, \beta^i, \ldots, \beta^j$$

clearly for some $a, b,$ $b > a$ $\Big\}$ $\beta^a = \beta^b \Leftrightarrow \beta^{b-a} = 1$

with $b - a \leq p^m - 1.$

Hence $r \leq p^m - 1.$ $\therefore \boxed{r = p^m - 1}$ ///

Corollary Every ff $\mathbb{F}_q$ contains an element α of order $= q-1$. In terms of α, $\mathbb{F}_q$ has the representation:

$$\mathbb{F}_q = \{0\} \cup \{\alpha^i \mid 0 \le i \le q-2\}$$

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 α for the equivalence class

$\alpha = [x]$ in $\mathbb{F}_2[x]/(x^4+x+1)$.

(alternately we may regard α as an imaginary element satisfying $\alpha^4 + \alpha + 1 = 0$

⇑

more practical !)

0

1

α

α^2

α^3

$\alpha^4 = \alpha + 1$

$\alpha^5 = \alpha^2 + \alpha$

$\alpha^6 = \alpha^3 + \alpha^2$

$\alpha^7 = \alpha^4 + \alpha^3$

$= \alpha^3 + \alpha + 1$

$\alpha^8 = \alpha^4 + \alpha^2 + \alpha$

$= \alpha^2 + 1$

$\alpha^9 = \alpha^3 + \alpha$

$\alpha^{10} = \alpha^4 + \alpha^2$

$= \alpha^2 + \alpha + 1$

$\alpha^{11} = \alpha^3 + \alpha^2 + \alpha$

$\alpha^{12} = \alpha^4 + \alpha^3 + \alpha^2$

$= \alpha^3 + \alpha^2 + \alpha + 1$

$\alpha^{13} = \alpha^4 + \alpha^3 + \alpha^2$

$+ \alpha =$

$\alpha^3 + \alpha^2 + 1$

$\alpha^{14} = \alpha^4 + \alpha + \alpha^3$

$= \alpha^3 + 1$

$\alpha^{15} = \alpha^4 + \alpha = 1 !$

Thus α as defined above, is a

p.e. (primitive element) of $\mathbb{F}_{16}$.

{ Preview of Minimal Polynomials (by example)

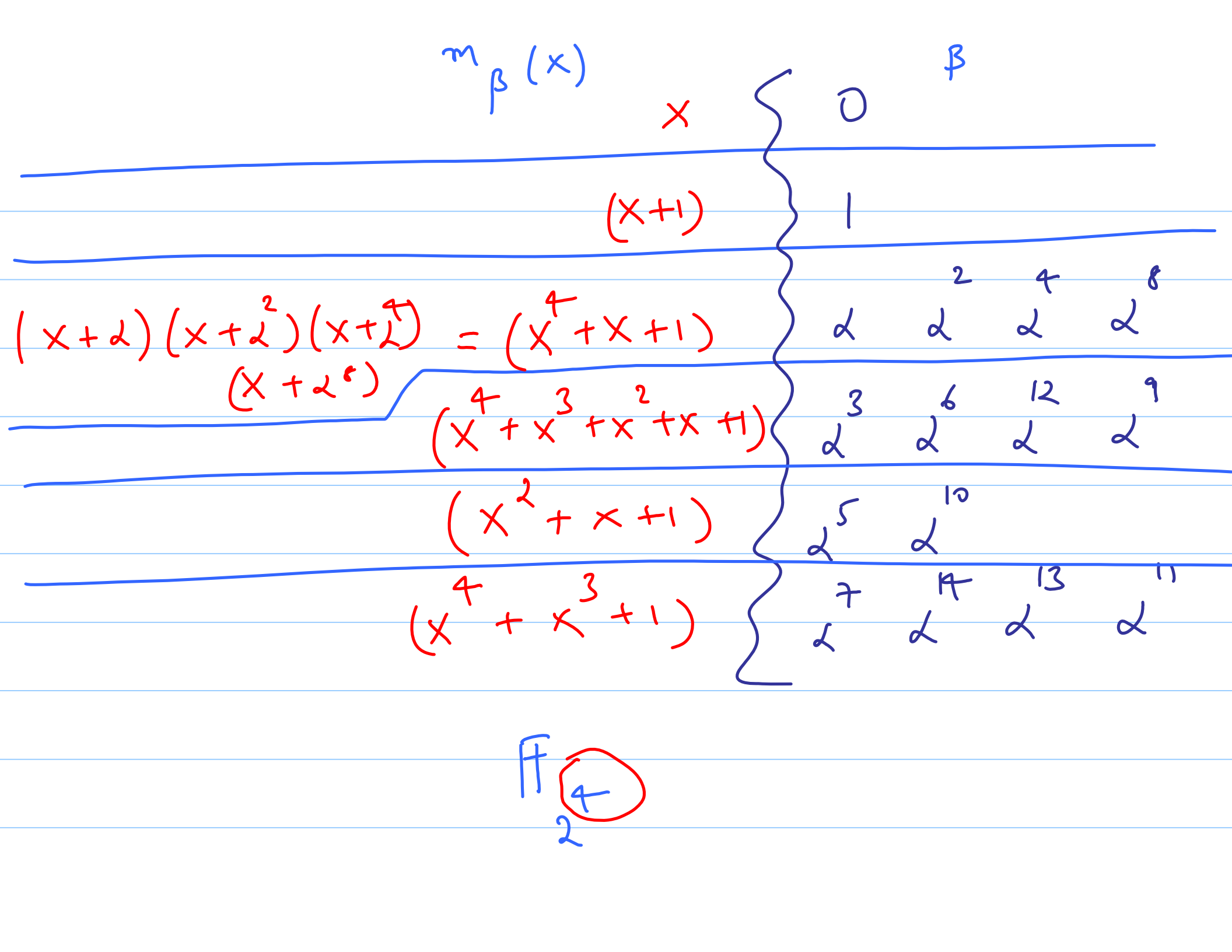
Eg $p = 2$ $\quad f(x) = x^4 + x + 1$

$$\mathbb{F}_2 = \mathbb{F}_2[x] \Big/ (x^4 + x + 1)$$

$$\mathbb{F}_q = \left\{ \sum_{i=0}^{3} a_i \alpha^i \;\middle|\; a_i \in \{0,1\} \right\}$$

where $\alpha \triangleq [x]$ in $\mathbb{F}_2[x]/(x^4+x+1)$

and hence satisfies $\alpha^4 + \alpha + 1 = 0$


$m_\beta(x)$
x
β
0
$(x+1)$
1
$(x+\alpha)(x+\alpha^2)(x+\alpha^4)(x+\alpha^8)(x+\alpha^{16}) = (x^4+x+1)$
$\alpha \quad \alpha^2 \quad \alpha^4 \quad \alpha^8$
$(x^4+x^3+x^2+x+1)$
$\alpha^3 \quad \alpha^6 \quad \alpha^{12} \quad \alpha^9$
(x^2+x+1)
$\alpha^5 \quad \alpha^{10}$
(x^4+x^3+1)
$\alpha^7 \quad \alpha^{14} \quad \alpha^{13} \quad \alpha^{11}$
$\mathbb{F}_{2^4}$

$$\therefore \left(x^{2^4} - x\right) = (x)(x+1)\cdot$$

$$(x^2 + x + 1)\cdot$$

$$(x^4 + x + 1)(x^4 + x^3 + 1)(x^4 + x^3 + x^2 + x + 1)$$

Minimal Polynomials

Note: Every element $\beta \in \mathbb{F}_q^*$ is a zero of $x^{q-1} - 1$. Hence every element of $\mathbb{F}_q$ is a zero of

$$x^q - x.$$

This motivates:

Defn The minimal polynomial of $\beta \in F_q^*$ is the smallest degree monic polynomial $m_\beta(x)$ of which β is a zero.

Lemma $m_\beta(x)$ is irreducible

pf Suppose not. Then

$$m_\beta(x) = f(x)\, g(x) \quad \text{with}$$

$$0 < \deg(f), \deg(g) < \deg(m_\beta).$$

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:

$$f(x) = a(x)\, m_\beta(x) + b(x)$$

$a(x)$: quotient

$b(x)$: remainder of deg $< m_\beta(x)$

$\therefore f(\beta) = 0 \Rightarrow a(\beta)\, m_\beta(\beta) + b(\beta) = 0$

$\Rightarrow b(\beta) = 0$

but this once again,

contradicts the minimality of $m_\beta(x)$

unless $b(x) = 0$. Hence $m_\beta(x) \mid f(x)$ ///

Corollary $m_\beta(x) \mid x^q - x$

Pf. Every element in $\mathbb{F}_q$ is a zero of $x^q - x$. Hence $\beta^q - \beta = 0$

$\Rightarrow m_\beta(x) \mid x^q - x$. //

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 nonzero elements)

Lemma 1 Let $\beta \in \mathbb{F}_q^*$, $q = p^m$. Then

$$\deg\left(m_\beta(x)\right) \leq m.$$

Pf Consider

$$\underbrace{1, \beta, \beta^2, \beta^3, \ldots, \beta^m}_{(m+1) \text{ elements}}$$

$\mathbb{F}_{p^m}$

1

$\mathbb{F}_p$

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 $|\mathbb{F}_q| = p^m$. Thus there exists a linear dependence expression of the form:

$$\sum_{i=0}^{m} c_i \beta^i = 0 \quad \left\{ \text{with at least one } c_i \neq 0 \right.$$

$\Leftrightarrow$ β is a zero of $\sum_{i=0}^{m} c_i x^i$.

Hence $m_\beta(x) \mid \sum_{i=0}^{m} c_i x^i$

$\Rightarrow$ $\boxed{\deg\left(m_\beta(x)\right) \leq m}$

///

Lemma 2 If $\beta \in \mathbb{F}_q^*$, $q = p^m$, is a primitive element, then

$$\deg\left(m_\beta(x)\right) = m.$$

Pf. Let $\deg\left(m_\beta(x)\right) = s$. From Lemma 1, $s \le m$. Since β is a p.e. of $\mathbb{F}_q$, every element θ in $\mathbb{F}_q$ can be expressed as a polynomial in β and

hence has an expression of the form

$$\theta = \sum_{i=0}^{s-1} a_i \beta^i \quad , \quad a_i \in \mathbb{F}_p$$

Hence $p^s \geq p^m$ by counting

$\therefore s \geq m$

$\therefore \boxed{s = m}$ //

Defn. The minimal polynomial $m_\beta(x)$ of a primitive element β in $\mathbb{F}_q$, $q = p^m$, is called a **primitive polynomial** (note that $m_\beta(x)$ has degree $= m$)

Eg $q = 2^4$ $\quad \mathbb{F}_q = \mathbb{F}_2[x] / (x^4 + x + 1)$

$= \mathbb{F}_2[\alpha]$ where $\alpha = [x]$

and thus satisfies $\boxed{\alpha^4 + \alpha + 1 = 0.}$

As seen before α is a p.e.

α^k has order $= \dfrac{(p^m - 1)}{(p^m - 1, k)} = \dfrac{15}{(15, k)}$

and hence is primitive iff $(15, k) = 1$.

Given an integer $n = \prod_{i=1}^{r} p_i^{e_i}$ the number of integers l, $0 \le l \le n-1$ s.t. $(l, n) = 1$ equals $\phi(n) = \prod_{i=1}^{r} p_i^{(e_i - 1)} (p_i - 1)$

(Eulers totient fn.)

where $\{p_i\}$ are odd primes.

$$\therefore \quad \phi(15) = \phi(5 \cdot 3) = (5-1)(3-1) = 8.$$

Hence $\mathbb{F}_{16}$ contains 8 p.e :

PRIMITIVE ELEMENT	COMMON MIN. POLY.
$\{\alpha \quad \alpha^2 \quad \alpha^4 \quad \alpha^8\}$	$\Longleftrightarrow X^4 + X + 1$
$\{\alpha^7 \quad \alpha^{11} \quad \alpha^{13} \quad \alpha^{14}\}$	$\Longleftrightarrow X^4 + X^3 + 1$

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 $\mathbb{F}_{p^m}$.

Eg $p = 2 \quad m = 12$

$\mathbb{F}_{2^{12}}$ has the following subfield structure:

$\mathbb{F}_{2^3}$
$\mathbb{F}_{2^6}$
$\mathbb{F}_{2^{12}}$
$\mathbb{F}_2$
$\mathbb{F}_{2^2}$
$\mathbb{F}_{2^4}$
indicates
subfield
relationship

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 $\mathbb{F}$ of characteristic p,

$$(x+y)^p = x^p + y^p.$$

Pf.

$$(x+y)^p = \sum_{i=0}^{p} \binom{p}{i} x^i y^{p-i}$$

$$= (x^p + y^p)$$ because:

$$\binom{p}{i} = \begin{cases} 1 & i = 0 \text{ or } i = p \\ 0 \pmod p & \text{else} \end{cases}$$

Since:

$$\binom{p}{i} = \frac{p!}{(p-i)!\, i!} = p \frac{(p-1)\cdots(p-i+1)}{1 \cdot 2 \cdots i}$$

//

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

$$\left(\alpha^k\right)^{2^2} = \alpha \Rightarrow \alpha^{3k} = 1 \Rightarrow$$

$$k \in \{0, 5, 10\}$$

$$\therefore \mathbb{F}_4 = \{0, 1, \alpha^5, \alpha^{10}\}$$

$\mathbb{F}_{2^4}$

|

$\mathbb{F}_{2^2}$

|

$\mathbb{F}_2$

Similarly,

$F_2 \Rightarrow$ **Test: $x^2 = x$**

$\Rightarrow x = 0$ or else $x = \alpha^k$ with

$\alpha^{2k} = \alpha^k \Rightarrow \alpha^k = 1 \Rightarrow k = 0$

$\therefore F_2 = \{0, 1\}$.

$$F_{2^4} = \{0, 1, x, x^2, \ldots, x^{14}\} \cup \{x^k \mid 0 \le k \le \infty\}$$

$$F_{2^2} = \{0, 1, x, x^2\}$$

$$F_2 = \{0, 1\}.$$

Thm Finite fields of size p^m exist for every prime p, $m \geq 1$

Pf When $m = 1$, the set of integers modulo 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 $x^{p^n} - x$. Then this collection of p^n zeros can be shown to form the desired finite field.

(further details may be found in the Appendix)

Thm The polynomial $x^{p^m} - x$ over $\mathbb{F}_p$ has the factorization:

$$x^{p^m} - x = \prod_{1 \le d \le m,\ d \mid m} \prod f(x)$$

$f(x)$ irred. $\deg(f) = d$

We illustrate by example. The proof may be found in the Appendix.

Eg. Let $q = 2^4$ so $p = 2$. Then

$$x^{2^4} - x = \underbrace{(x)(x+1)}_{\deg = 1}\ \underbrace{(x^2+x+1)}_{\deg = 2}$$

$$\Rightarrow \underbrace{(x^4+x+1)(x^4+x^3+1)(x^4+x^3+x^2+x+1)}_{\deg = 4}$$

irreducible factors

Thm. Any two finite fields $\mathbb{F}_q$, $\mathbb{F}_{q'}$ of the same size $q = p^m$ are isomorphic.

Pf. (sketch) Let β be a p.e. of $\mathbb{F}_q$ and $m_\beta(x)$ its min. poly.

Then $m_\beta(x) \mid x^{p^m} - x$.

Over in $\mathbb{F}_\Sigma'$, every element is a zero of $x^{p^m} - x$. Hence some element θ must be a zero of $m_\beta(x)$. Then the map $\phi: \beta \to \theta$

$$\sum_{i=0}^{m-1} a_i \beta^i \longrightarrow \sum_{i=0}^{m-1} a_i \theta^i$$

$(a_i \in \mathbb{F}_p)$

$\mathbb{F}_q$ $\mathbb{F}_\Sigma'$

can be shown to be an isomorphism.

We illustrate below with an example.

Let $q = 2^4$.

Eg — We note from the factorization:

$$X^{2^4} - X = (X)(X+1)(X^2+X+1)$$
$$(X^4+X+1)(X^4+X^3+1)(X^4+X^3+X^2+X+1)$$

that there are 3 different irreducible polynomials of degree 4 over $\mathbb{F}_2$.

Let

$$\mathbb{F}_\Sigma = \mathbb{F}_2[x] \Big/ (x^4 + x^3 + x^2 + x + 1)$$

and

$$\mathbb{F}'_\Sigma = \mathbb{F}_2[x] \Big/ (x^4 + x + 1) .$$

Let $\beta = [x]$ in $\mathbb{F}_5$ so that

$$m_\beta(x) = x^4 + x^3 + x^2 + x + 1$$

and

$\alpha = [x]$ in $\mathbb{F}_5^1$ so that

$$m_\alpha(x) = x^4 + x + 1.$$

Then in $\mathbb{F}_{2^4}$ the element

$\theta = \alpha^3$ has min poly

$$m_\theta(x) = x^4 + x^3 + x^2 + x + 1.$$

Hence the map:

$$\phi : \begin{cases} \mathbb{F}_2 \longrightarrow \mathbb{F}_{2^4} \\ \beta \longrightarrow \theta \end{cases}$$

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 $p = 2 \quad q = 2^4$

$$\mathbb{F}_q = \mathbb{F}_2[x] \,/\, (x^4 + x + 1)$$

with $\alpha = [x]$ so that $\alpha^4 + \alpha + 1 = 0$.

Then

$$\mathbb{F}_q = \{0\} \cup \{\alpha^i \mid 0 \le i \le 14\}.$$

POLYNOMIAL REPRESENTATION

0	$\alpha^4 = \alpha + 1$	$\alpha^{11} = \alpha^3 + \alpha^2 + \alpha$
$\alpha^0 = 1$	$\alpha^5 = \alpha^2 + \alpha$	$\alpha^{12} = \alpha^4 + \alpha^3 + \alpha^2 = \alpha^3 + \alpha^2 + \alpha + 1$
$\alpha^1 = \alpha$	$\alpha^6 = \alpha^3 + \alpha^2$	$\alpha^{13} = \alpha^4 + \alpha^3 + \alpha^2 + \alpha = \alpha^3 + \alpha^2 + 1$
$\alpha^2 = \alpha^2$	$\alpha^7 = \alpha^4 + \alpha^3$	$\alpha^{14} = \alpha^4 + \alpha^3 + \alpha = \alpha^3 + 1$
$\alpha^3 = \alpha^3$	$= \alpha^3 + \alpha + 1$	
	$\alpha^8 = \alpha^4 + \alpha^2 + \alpha$	$\alpha^{15} = \alpha^4 + \alpha = \alpha + \alpha + 1$
	$= \alpha^2 + 1$	$= 1$!!
	$\alpha^9 = \alpha^3 + \alpha$	
	$\alpha^{10} = \alpha^4 + \alpha^2$	
	$= \alpha^2 + \alpha + 1$	

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	α^5	α^{10}	α^{11}	α^{12}
1	0	α^6	α^{13}	α^{13}	α^6
α	α^4	α^7	α^9	α^{14}	α^3
α^2	α^8	α^8	α^2		
α^3	α^{14}	α^9	α^7		
α^4	α	α^{10}	α^5		

The add-1 table is frequently used in conjunction with Hamer's method to add a string of powers of α: Eg:

$$\alpha^3 + \alpha + \alpha^7 + \alpha^8 = \alpha + \alpha^3 + \alpha^7 + \alpha^8$$

$$= \alpha\left(1 + \alpha^2\left(1 + \alpha^4\left(1+\alpha\right)\right)\right) = \alpha^2 .$$

α^4

α^2

α

CYCLOTOMIC COSETS

These cosets will be used to explain the structure of minimal polynomials as well as to construct cyclic codes.

Let p be prime, $m \geq 1$.

If α is a p.e. of $\mathbb{F}_{p^m}$, then

all arithmetic in the exponent of α

is conducted (mod $p^m - 1$) since

α 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-cyclotomic cosets $\pmod{p^m-1}$.

Proof of equivalence:

(i) $a = p^0 a$ $\therefore a \sim a$ REFLEXIVE

(ii) $a = p^k b \Rightarrow b = p^{m-k} a \pmod{p^m - 1}$

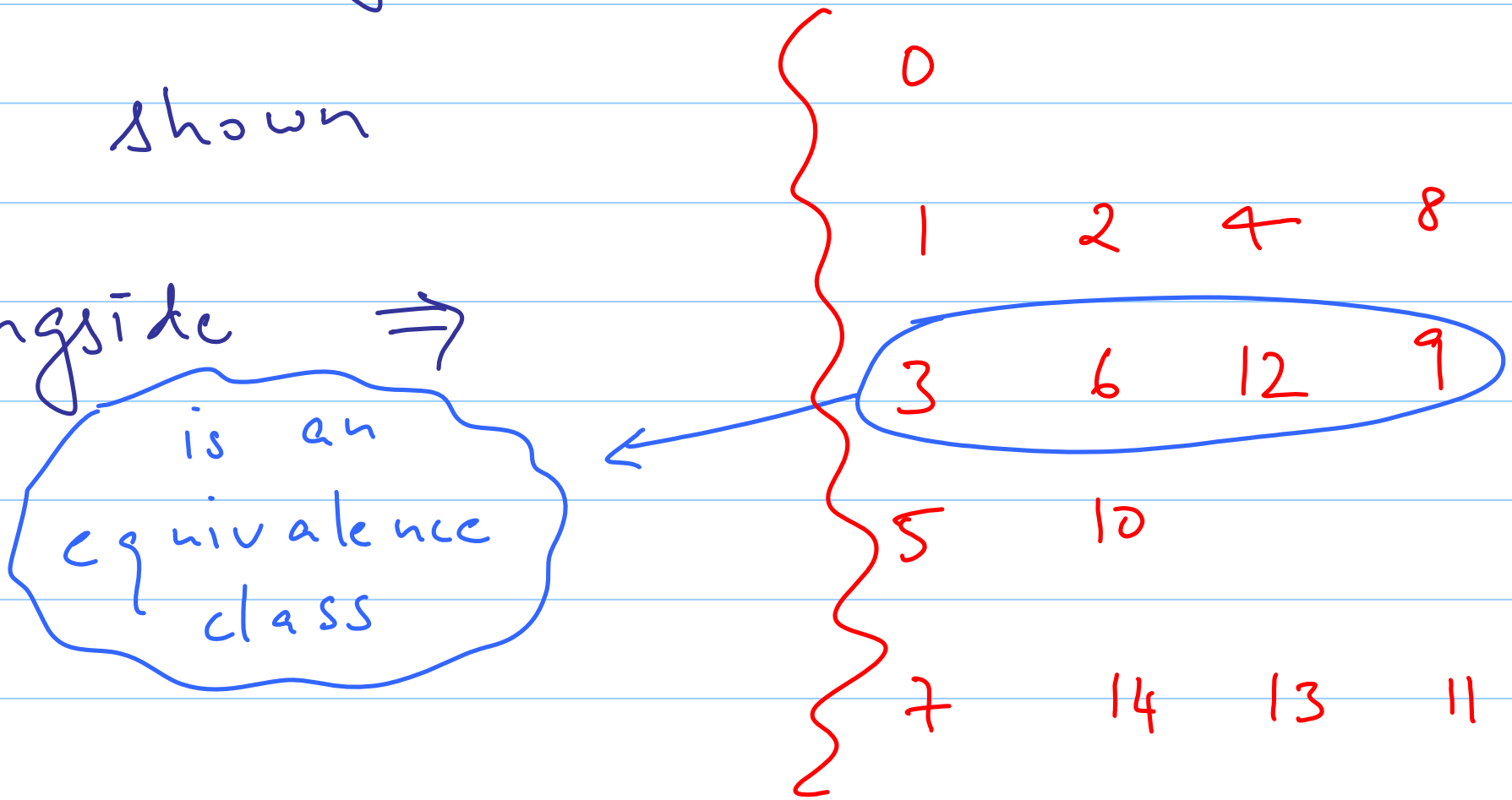
SYMMETRY

(iii) $a = p^k b$, $b = p^l c \Rightarrow a = p^{k+l} c$

TRANSITIVE

Eg $p = 2 \quad q = 2^4$

The 2-cyclotomic cosets (mod $2^m - 1$) are shown alongside $\Rightarrow$



The smallest element within each cyclotomic coset is called a coset leader.

Thus here, 0, 1, 3, 5, 7 are the coset leaders.

Thm. Let $\beta \in \mathbb{F}_{p^m}^{*}$. Then

$$m_\beta(x) = \prod_{\ell=0}^{d-1} \left(x - \beta^{p^\ell}\right)$$

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 β is an element.

Pf (see the Appendix)

Defn. The elements β^{p^l}, $1 \le l \le d-1$ are called the conjugates of β.

Thus the theorem also asserts that all the conjugates of β share the same minimal polynomial.

APPENDIX

— proofs to complete the lecture notes

Lemma 1 Let $n \geq 2$, $r, s \geq 1$, then

$$n^r - 1 \mid n^s - 1 \iff r \mid s$$

Pf Write: $s = ar + b$ $\quad b < r$

$$(n^s - 1) \pmod{n^r - 1}$$

$$= (n^{ar+b} - 1) \pmod{n^r - 1}$$

$$= \left((n^r)^b \cdot n^b - 1 \right) \pmod{n^r - 1}$$

$$= (n^b - 1) = 0 \text{ iff } b = 0$$

i.e., iff $r \mid s$ ///

Lemma 2 Let $s > r \geq 1$. 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)

$$= (x^b - 1) \pmod{x^n - 1}$$

$$= 0 \text{ iff } b = 0$$

i.e. iff $n \mid s$. //

Corollary $x^d - 1 \mid x^m - 1$ iff $d \mid m$.

Thm $\mathbb{F}_{p^d} \subseteq \mathbb{F}_{p^m}$ iff $d \mid m$.

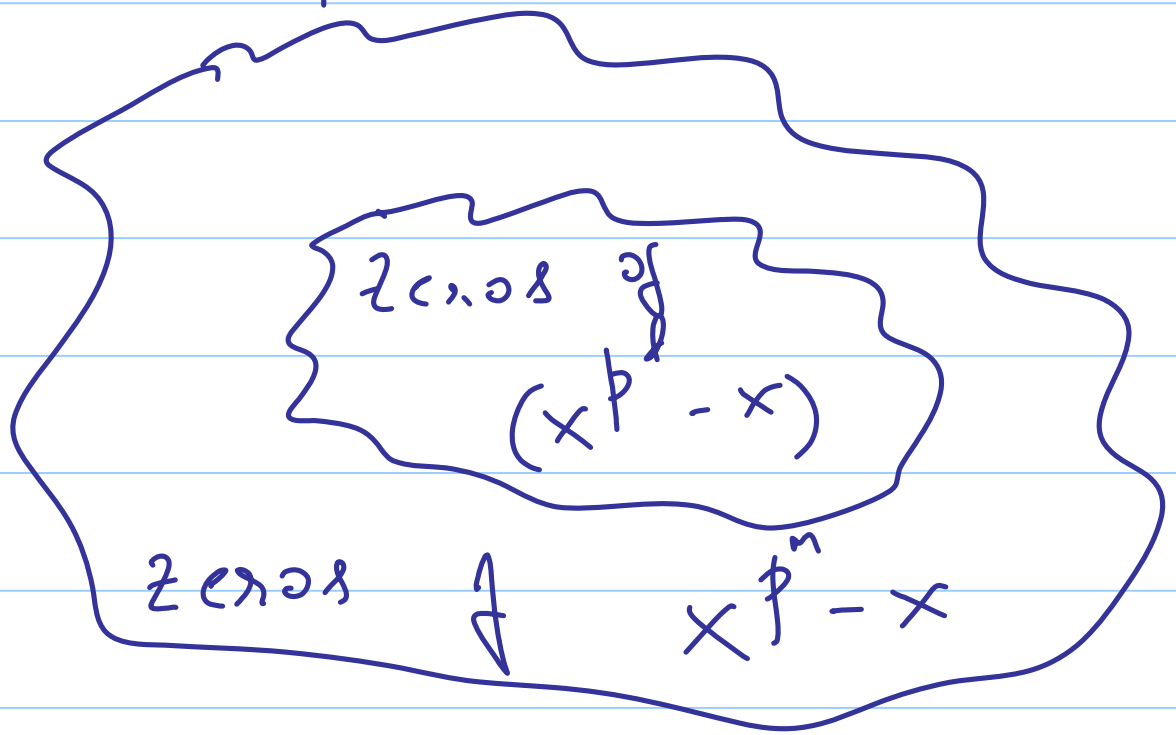
Pf a) Suppose $d \mid m$

$\Rightarrow (x^{p^d-1} - 1) \mid (x^{p^m-1} - 1)$

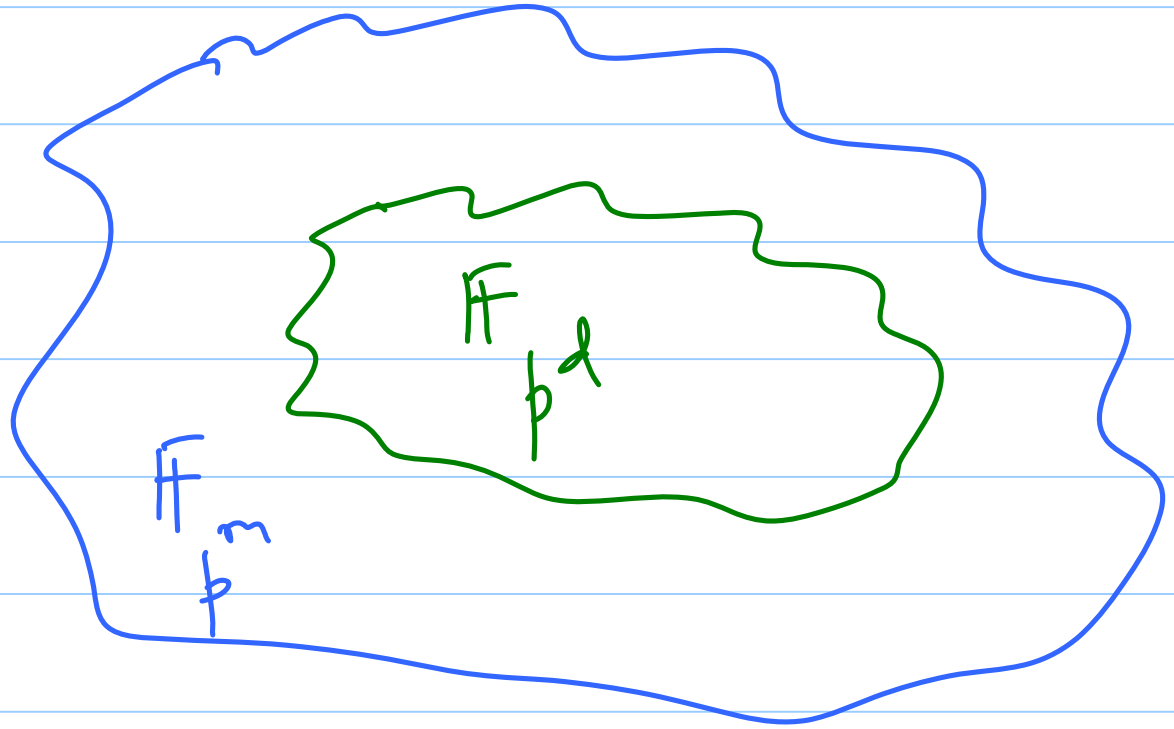
$= (x^{p^d} - x) \mid (x^{p^m} - x)$

But the elements of $\mathbb{F}_{q^m}$ are precisely the q zeros of $(x^q - x)$.

Thus we have the picture:



and Lemma 3 will show that this picture has a subfield interpretation:



Lemma 3: The collection of p^d zeros of $x^{p^d} - x$ in $\mathbb{F}_{p^m}$ form a sf of size p^d.

Pf Let

$$S = \left\{ \theta \in \mathbb{F}_{p^m} \;\middle|\; \theta^{p^d} - \theta = 0 \right\}.$$

To show that $(S, +)$ is an Abelian group, it suffices to show that:

$a, b \in S \Rightarrow a - b \in S.$

but

$$(a-b)^{p^q} = (a + (-b))^{p^q}$$

$$= a^{p^q} + (-1)^{p^q} b^{p^q}$$

$$= a^{p^q} - b^{p^q}$$

$$= a - b$$

$\therefore (a-b)$ is also a zero of f

$(x^{p^\ell} - x) \quad \therefore \quad (a-b) \in S$

$(S, \cdot)$ Hence it suffices to show that

$a, b \in S \Rightarrow ab \in S$ and $a \in S \Rightarrow$

$a^{-1} \in S$. But

$$a^{p^\ell} = a, \quad b^{p^\ell} = b = (ab)^{p^\ell} = ab$$

$a^{p^\ell} = a \Rightarrow (a^{-1})^{p^\ell} = a^{-1}$

$\therefore a^{-1} \in S$

$\therefore$ S is a subfield of size $\mathbb{F}_{p^\ell}$.

b) next, suppose $\mathbb{F}_{p^\ell} \subseteq \mathbb{F}_{p^m}$.

Let β be a p.e. of $\mathbb{F}_{p^\ell}$.

Then β has order $= p^\ell - 1$.

But since $\beta \in \mathbb{F}_{p^m}$, $\beta^{p^m - 1} = 1$

$\therefore (p^d - 1) \mid (p^m - 1) \Rightarrow d \mid m$ ///

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 $\theta^{p^d} = \theta$

$\Rightarrow$ θ is a zero of $x^{p^d} - x$.

But the collection of zeros of $x^{p^d} - x$ in $\mathbb{F}_{p^m}$ forms a subfield $\mathbb{F}_{p^d}$ of size p^d. $\therefore \theta \in \mathbb{F}_{p^d}$.

(conversely, suppose $\theta \in \mathbb{F}_{p^d}$

$\Rightarrow \theta^{p^d} = \theta$.

Thm Finite fields of size p^m exist for every prime p, $m \geq 1$

Pf When $m = 1$, the set of integers modulo p, $\mathbb{Z}_p$, is an example of a 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:

$$(x^{p^m} - x) = \prod_{i=1}^{r} l_i(x) \prod_{j=1}^{s} f_j(x)$$

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 poly 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 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 $x^{p^m} - x$ over $\mathbb{F}_p$ has the factorization:

$$x^{p^m} - x = \prod_{1 \le d \le m,\ d \mid m} \prod f(x)$$

$f(x)$ irred. $\deg(f) = d$

Pf Let $f(x)$ be irred of deg $= d$ with $d \mid n$. Then

$$\mathbb{F}_p[x] \Big/ (f(x)) \cong \mathbb{F}_{p^d}.$$

Let $\beta = [x]$ in $\mathbb{F}_p[x] \Big/ (f(x))$. 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 $\beta = [x]$ in $\mathbb{F}_p[x] / (f(x))$ $\cong \mathbb{F}_{p^d}$. Then $m_\beta(x) = f(x)$ in $\mathbb{F}_{p^d}$.

$f(x) \mid x^{p^m} - x \Rightarrow \beta^{p^m} = \beta$.

Let α be a g.e. of $\mathbb{F}_{p^d}$. Then

$$\alpha = \sum_{i=0}^{d} a_i \beta^i, \quad a_i \in \mathbb{F}_p$$

$$\Rightarrow \alpha^{p^m} = \left[\sum_{i=0}^{d} a_i \beta^i \right]^{p^m} = \sum_{i=0}^{d} a_i^{p^m} \beta^{p^m i}$$

$$= \sum_{i=0}^{d} a_i \beta^i = \alpha.$$

Thus $(p^d - 1) \mid p^m - 1 \Rightarrow d \mid m$

Thm. Let $\beta \in \mathbb{F}_{p^m}^{*}$. Then

$$m_\beta(x) = \prod_{\ell=0}^{d-1} \left(x - \beta^{p^\ell}\right)$$

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 β is an element.

Pf Let

$$m_\beta(x) = \sum_{i=0}^{d} g_i x^i \quad (g_i \in \mathbb{F}_p)$$

Then

$$m_\beta(\beta) = \sum_{i=0}^{d} g_i \beta^i = 0$$

$$\Rightarrow m_\beta(\beta^{p^\ell}) = \sum_{i=0}^{d} g_i \beta^{p^\ell \cdot i} = 0$$

$$= \sum_{i=0}^{d} g_i \left(\beta^{p^k}\right)^i = 0$$

$\Rightarrow \beta^{p^k}$ is also a zero of $m_\beta(x)$

On the other hand, define

$$h(x) = \sum_{j=0}^{d} h_j x^j = \prod_{i=0}^{d-1} (x - \beta^{p^i})$$

The coefficients of $h(x)$ are elem.

Symm. fns $f\left\{\beta^{p^\ell}\right\}_{\ell=0}^{d-1}$.

Eg $h_{d-1} = -\sum_{\ell=0}^{d-1} \beta^{p^\ell}$

$\therefore h_{d-1}^{p} = -\sum_{\ell=0}^{d-1} \beta^{p^{\ell+1}} = -\sum_{i=0}^{d-1} \beta^{p^\ell}$

since $\beta^{p^d} = \beta$.

Hence $h(x) \in \mathbb{F}_p[x]$

$$\Rightarrow \quad m_\beta(x) = \prod_{\ell=0}^{d-1} \left(x - \beta^{p^\ell}\right) .$$

Lec 40-41 { Transform Approach to Cyclic Codes

Recap

* { Completed discussion on finite fields

* completed discussion on min. poly.

* subfield structure

* $(x+y)^p = x^p + y^p$

* $\theta \in \mathbb{F}_{p^d} \iff \theta^{p^d} = \theta$

* finite fields of every size p^m exist

* $$x^{p^m} - x = \prod_{d \mid m} f(x)$$
$\deg f = d$, f irreducible

* any two ff of the same size are isomorphic

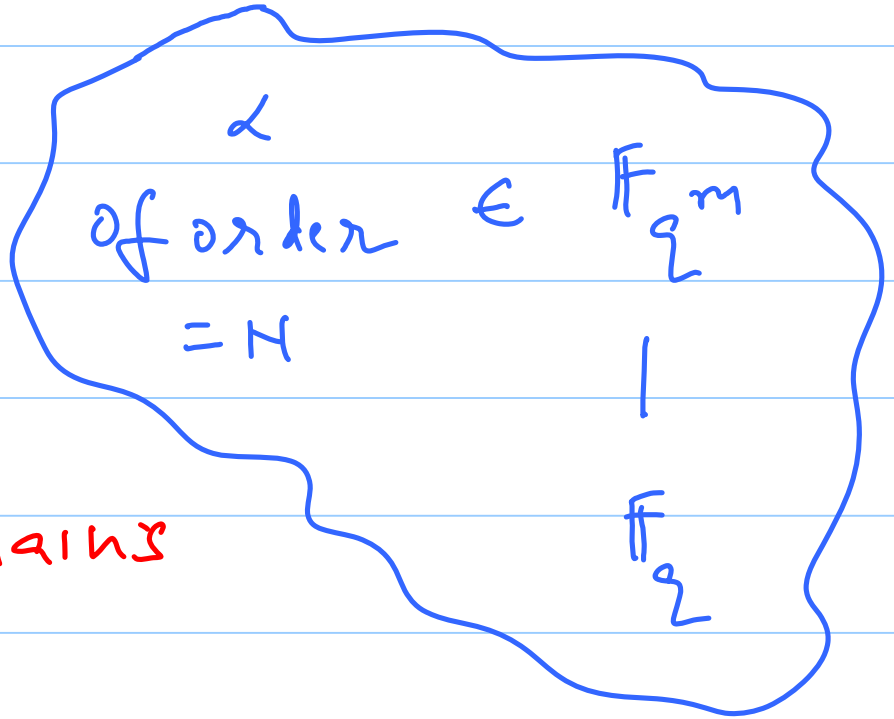
* add-1 table

* p-cyclotomic cosets (mod $p^m - 1$)

* $m_\beta(x) = \prod_{l=0}^{d-1} (x - \beta^{p^l})$

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

$$1, \Sigma, \Sigma^2, \ldots, \Sigma^k \ldots \Sigma^\ell \ldots \quad (\text{modulo } N)$$

Since $\mathbb{Z}_H$ is finite,

Some $\Sigma^k \equiv \Sigma^\ell$, $\ell > k$,

$(\text{mod } N)$

i.e.,

$$q^{\ell-k} = 1 \pmod{N}.$$

Let m be the smallest power of q s.t $\boxed{q^{m} = 1 \pmod{N}}$, m is then called the multiplicative order of q $\pmod{N}$.

Note: this implies that

$$q^m \equiv 1 \pmod{N}$$

$$\Rightarrow \quad N \mid (q^m - 1)$$

$\Rightarrow$ $\mathbb{F}_{q^m}$ contains an element of order N, for example, $\alpha = \beta^{\left(\frac{q-1}{N}\right)}$ where β is a p.e. of $\mathbb{F}_{q^m}$.

Eg $q = 2 \quad N = 15 \Rightarrow n = 4$

$\alpha = \beta^1$, β primitive in $\mathbb{F}_{2^4}$.

$q = \phi^e \quad N = (q-1) \Rightarrow n = 1$

$\alpha = \beta$, β p.e. of $\mathbb{F}_{\phi^e}$


FINITE FIELD
TRANSFORM

Defn Let $(q, N) = 1$, m be the multiplicative order of $q \pmod N$.

Let α 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

$$\hat{a}_\lambda \triangleq \sum_{t=0}^{N-1} a_t \alpha^{\lambda t}, \quad 0 \le \lambda \le N-1,$$

is called the finite field transform (fft) of (a_t).

Properties

1). **Linearity:**

$$(a_t) \Leftrightarrow (\hat{a}_\lambda) \qquad (b_t) \Leftrightarrow (\hat{b}_\lambda)$$

$$\Rightarrow (a_t + \theta b_t) \Leftrightarrow (\hat{a}_\lambda + \theta \hat{b}_\lambda)$$

(Pf. Exercise!) all $\theta \in \mathbb{F}_q^m$.

CYCLIC SHIFTS

2). Let

$$b_t = a_{(t-\tau) \pmod N} \qquad 0 \le t \le N-1$$

for some $0 \le \tau \le N-1$. Then (b_t) is called a cyclic shift of (a_t).

In the sequel, we will always assume that subscripts of a or $\hat{}$

are always computed $(\text{mod } N)$

and will therefore simply write

$(a_{t-\tau})$ in place of $(a_{t-\tau} \ (\text{mod } N))$.

Now

$$\sum_{t=0}^{N-1} a_{t-\tau} \alpha^{\lambda t}$$

$$= \sum_{t=0}^{N-1} a_{t-\tau} \alpha^{\lambda(t-\tau)} \alpha^{\lambda\tau}$$

$$= \omega^{\lambda\tau} \left[\sum_{s=0}^{N-1} a_s \omega^{\lambda s} \right], \quad \text{Setting } t-\tau = s \pmod N$$

$$= \omega^{\lambda\tau} \hat{a}_\lambda .$$

Thus $(a_t) \Leftrightarrow (\hat{a}_\lambda)$

$$\Rightarrow (a_{t-\tau}) \Leftrightarrow \omega^{\lambda\tau} (\hat{a}_\lambda)$$

3) INVERSION FORMULA:

$$a_t = (N)^{-1} \sum_{\lambda=0}^{N-1} \hat{a}_\lambda \, \alpha^{-\lambda t}$$

Pf. $$RHS = (N)^{-1} \sum_{\lambda=0}^{N-1} \sum_{s=0}^{N-1} a_s \, \alpha^{\lambda(s-t)}$$

$$= (N)^{-1} \sum_{s=0}^{N-1} a_s \left[\sum_{\lambda=0}^{N-1} \alpha^{\lambda(s-t)} \right]$$

$= N$ (when $s=t$)

$$= \left[\frac{\alpha^{N(s-t)} - 1}{\alpha^{(s-t)} - 1} \right] \quad s \neq t$$

$= 0$ since $\alpha^N = 1$

$= a_t$

---

4). **CYCLIC CONVOLUTION**

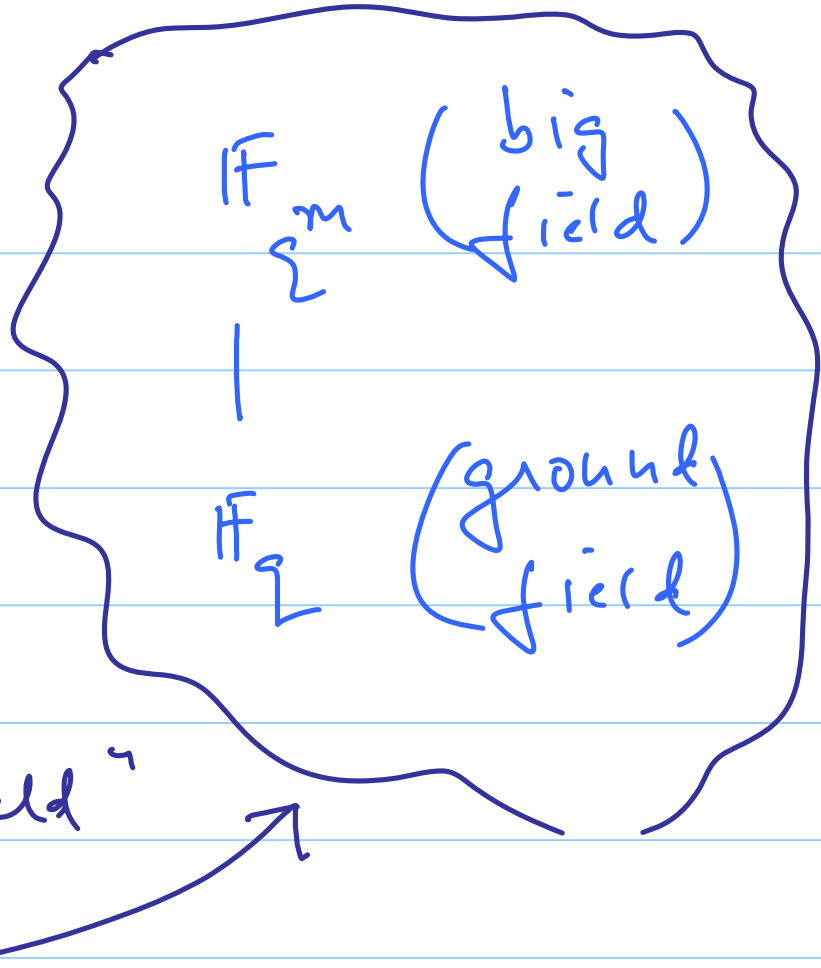
$(a_t) \Leftrightarrow (\hat{a}_\lambda) \qquad (b_t) \Leftrightarrow (\hat{b}_\lambda)$

$\Rightarrow$ $(\hat{u}_\lambda) = (\hat{a}_\lambda \, \hat{b}_\lambda)$ where $u_t = \sum_{\tau=0}^{N-1} a_{t-\tau} \, b_\tau$

Pf. (Exercise !)

5). CONJUGACY

Suppose $a_t \in \mathbb{F}_\Sigma$, all t, i.e. (a_t) lies in the "ground field"



Then (conjugacy relation)

$$\hat{a}_{\lambda\xi} = \left[\hat{a}_{\lambda}\right]^{\xi} \quad \forall\, \lambda$$

(where $\lambda\xi$ in the subscript is computed $(\mathrm{mod}\ r)$).

(conversely $\hat{a}_{\lambda\xi} = \left[\hat{a}_{\lambda}\right]^{\xi}$ $\forall\, \lambda$

$\Rightarrow$ $a_t \in \mathbb{F}_{\xi}$, $\forall\, t$.

PERIODIC SEQUENCE INTERPRETATION

since arithmetic in the subscript of a_t or else $\hat{a}_x$ is always computed (mod N), it is often convenient to think of both (a_t) as well as $(\hat{a}_x)$ as being periodic sequences over $\mathbb{F}_q^m$ of

period N, i.e.,

$$\ldots a_{N-1} \underbrace{a_0 \; a_1 \; a_2 \; \ldots \; a_{N-1}}_{\text{one period}} a_0 \; a_1 \; \ldots \; a_{N-1} \; a_0 \ldots$$

$$\ldots \hat{a}_{N-1} \underbrace{\hat{a}_0 \; \hat{a}_1 \; \hat{a}_2 \; \ldots \; \hat{a}_{N-1}}_{\text{one period}} \hat{a}_0 \; \hat{a}_1 \; \ldots \; \hat{a}_{N-1} \; \hat{a}_0 \ldots$$

This is consistent with the defn:

$$\hat{a}_\lambda = \sum_{t=0}^{N-1} a_t \, \alpha^{\lambda t}$$

$$\hat{a}_{\lambda+N} = \hat{a}_\lambda$$

since α has order N.

Pf (of $\hat{a}_{\lambda q} = \left[\hat{a}_{\lambda}\right]^q$ when $a_t \in \mathbb{F}_q$ all t)

$$\hat{a}_{\lambda q} = \sum_{t=0}^{N-1} a_t \alpha^{\lambda q t}$$

$$= \sum_{t=0}^{N-1} \left(a_t \alpha^{\lambda t}\right)^q \qquad \text{since } a_t^q = a_t$$

$$= \left[\sum_{t=0}^{N-1} a_t \alpha^{\lambda t}\right]^q \qquad \text{since } q = p^e$$

$$= \left[\hat{a}_\lambda\right]^q$$

Converse: Next suppose

$$\hat{a}_{\lambda\xi} = \left[\hat{a}_\lambda\right]^\xi \quad \text{all } \lambda, \text{ then}$$

$$\left[a_t\right]^\xi = \left[N^{-1} \sum_\lambda \hat{a}_\lambda \alpha^{-\lambda t}\right]^\xi$$

$$= (N^{-1})^\xi \sum_\lambda \left[\hat{a}_\lambda\right]^\xi \alpha^{-\lambda\xi t}$$

$$= N^{-1} \sum_{\lambda=0}^{N-1} \hat{a}_{\lambda q} \, \alpha^{-\lambda q t}$$

(Set $\mu = \lambda q \pmod N$. Then as λ varies over $\{0, 1, \ldots, N-1\}$, so does λq.)

$$= N^{-1} \sum_{\mu=0}^{N-1} \hat{a}_\mu \, \alpha^{-\mu t} = a_t$$

Thus $[a_t]^q = a_t$ all t

$\Rightarrow$ $a_t \in \mathbb{F}_q$ all t.

CYCLIC CODES

Defn A linear cyclic code $\mathcal{C}$ of block length N over $\mathbb{F}_q$ is a collection of N-tuples $(c_t)_{t=0}^{N-1}$ such that:

(i) **LINEARITY**

$$(c_t^{(1)}), \quad (c_t^{(2)}) \in \mathcal{C}$$

$$\Rightarrow (c_t^{(1)} + \theta\, c_t^{(2)}) \in \mathcal{C}, \quad \forall \theta \in \mathbb{F}_q$$

(thus $\mathcal{C}$ is a vector space over $\mathbb{F}_q$)

(ii) CYCLIC SHIFTS

$$(c_t) = (c_0\ c_1\ \cdots\ c_{N-1}) \in \mathcal{C}$$

$$\Rightarrow (c_{t-\tau}) = (c_{N-\tau}\ c_{N-\tau+1}\ \cdots\ c_{N-\tau-1}) \in \mathcal{C}$$

(i.e., $\mathcal{C}$ is closed under cyclic shifts).

Given a codeword $(c_t) \in \mathbb{F}_q^N$ in $\mathcal{C}$,

we define the transform $(\hat{c}_\lambda)$ of

(c_t) via

$$\hat{c}_\lambda = \sum_{t=0}^{N-1} c_t \alpha^{\lambda t}$$

where α has order N in $\mathbb{F}_{q^m}$,

in which m is the multiplicative

order of $q \pmod N$.

Note: Thus while the code symbols

belong to the ground field $\mathbb{F}_q$,

their transforms lie in $\mathbb{F}_{\Sigma^m}^H$ in general.

In this context, we will regard $\hat{c}_\lambda$ as the transform coefficient of (c_t) at frequency λ.

EQUIVALENCE RELATION ON FREQUENCIES

Define

$$\lambda_2 \sim \lambda_1 \qquad 0 \leq \lambda_1, \lambda_2 \leq N-1$$

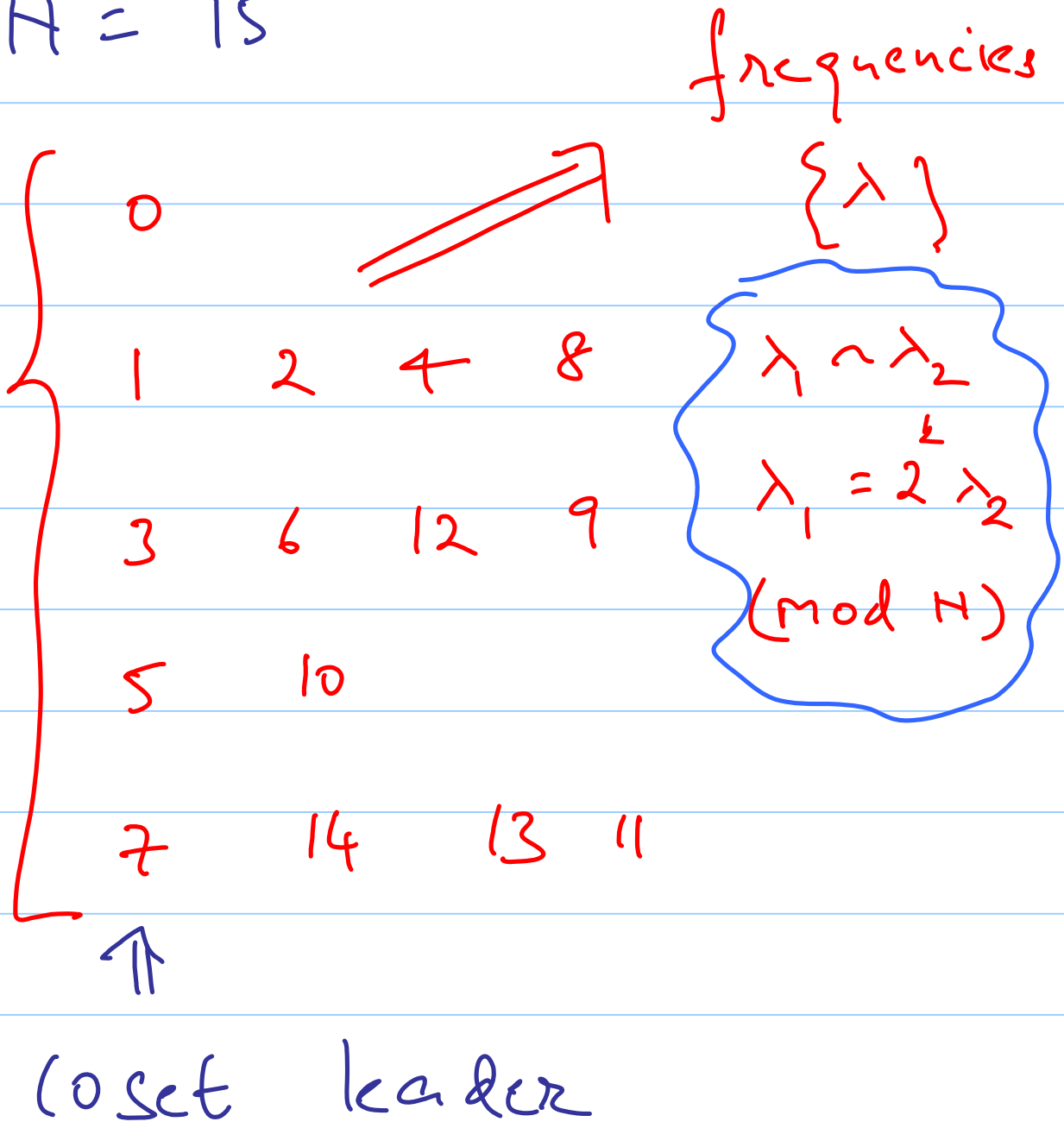
if $\lambda_2 \equiv q^i \lambda_1 \pmod{N}$ some $i \geq 0$.

This can be verified to be an equivalence relation.

The corresponding equivalence classes are called the

q-cyclotomic cosets (mod N).

Eg $q = 2$ $N = 15$



there are

5 equivalence

classes ⇒

The smallest element within each cyclotomic cosets is called the coset leader.

In general, if there k cosets with n_i elements in the i^{th} coset, then $\sum_{i=1}^{k} n_i = n$.

Defn. A subset

$$S \subseteq \{0, 1, 2, \ldots, N-1\}$$

is said to be a closed set of frequencies if

$$\lambda \in S \Rightarrow q\lambda \pmod{N} \in S.$$

It is straightforward to show that every closed set is the union of cyclotomic cosets.

NULL SPECTRUM

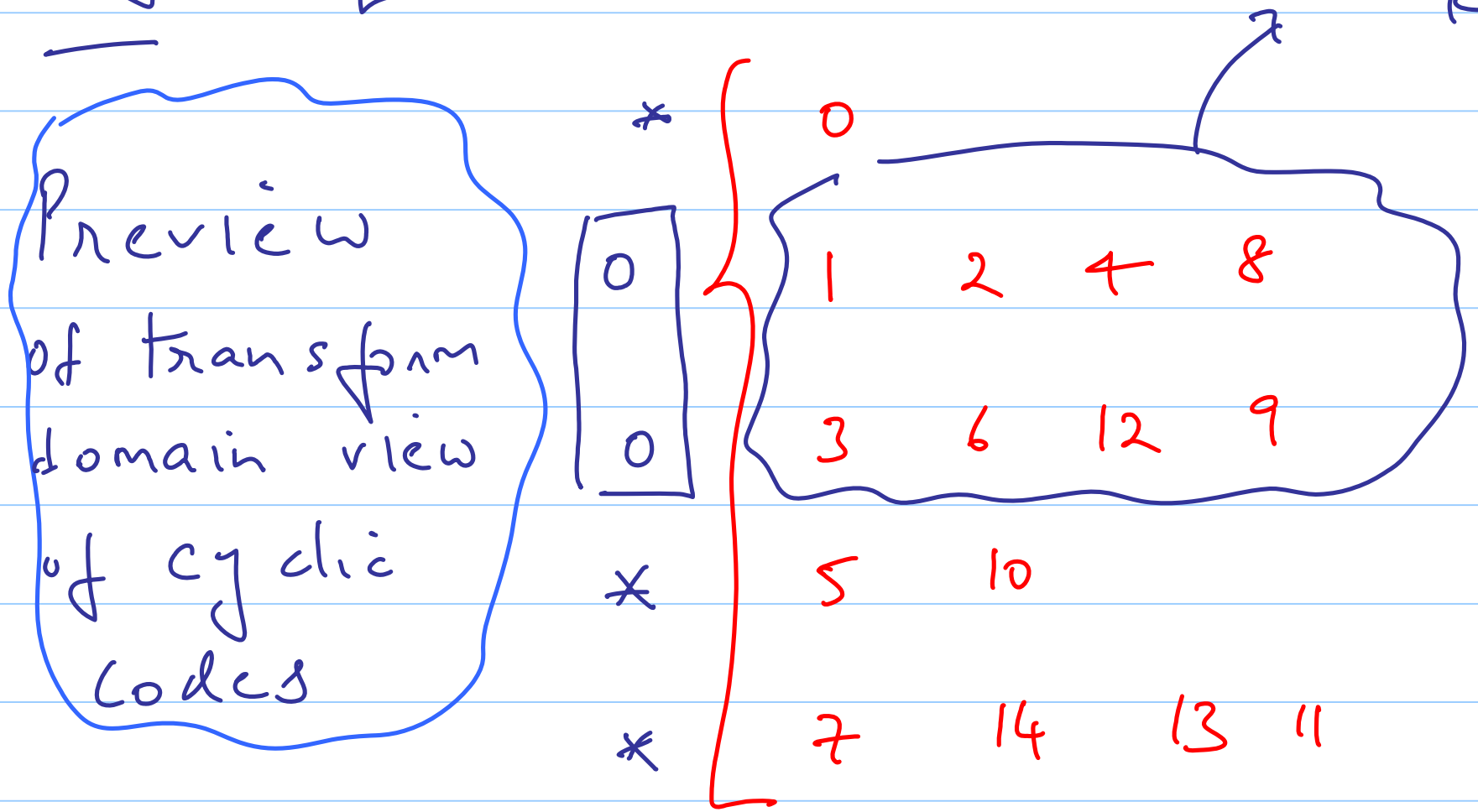
Defn Let $\mathcal{C}$ be a linear, cyclic code of block length N over $\mathbb{F}_q$.

Then the collection of frequencies

$$NS_{\mathcal{C}} = \left\{ \lambda \;\middle|\; \begin{array}{l} 0 \le \lambda \le N-1 \\ \hat{c}_\lambda = 0 \text{ all } (c_t) \in \mathcal{C} \end{array} \right\}$$

is called the **null spectrum** of $\mathcal{C}$.

Eg $q = 2$ $N = 15$


Preview
of transform
domain view
of cyclic
codes
*
0
0
*
*
0
1 2 4 8
3 6 12 9
5 10
7 14 13 11
NSR

Lec 41

Estimating the parameters of a cyclic code.

Recap

* finite field transform
  - expanding $\mathbb{F}_{\Sigma} \to \mathbb{F}_{\Sigma^m}$
  - defn
  - properties
* cyclic codes
  - closed set
  - cyclotomic cosets
  - null spectrum.

Lemma 1 The null spectrum $NS_{\mathcal{C}}$ of a cyclic code $\mathcal{C}$ is a closed set

Pf $\lambda \in NS_{\mathcal{C}} \Leftrightarrow \hat{c}_\lambda = 0 \quad \forall (c_t) \in \mathcal{C}$

$\hat{c}_\lambda = 0 \Rightarrow \left[\hat{c}_\lambda\right]^2 = 0 \Rightarrow \hat{c}_{\lambda^2} = 0$

since $(c_t) \in \mathbb{F}_q^n$ (conjugacy)

$\Rightarrow \lambda q \in (NS)_R$ Hence $(NS)_R$ is a closed set.

Lemma 2 Let $S \subseteq \{0, 1, \ldots, N-1\}$ be a closed set of frequencies.

Then

$$C \triangleq \left\{ (c_t) \in \mathbb{F}_q^N \;\middle|\; \hat{c}_\lambda = 0, \text{ all } \lambda \in S \right\}$$

is a linear cyclic code.

Pf. Let $(a_t), (b_t) \in R$

$\Rightarrow \hat{a}_\lambda = 0 \quad \hat{b}_\lambda = 0$ for all $\lambda \in S$

$\Rightarrow \hat{a}_\lambda + \theta \hat{b}_\lambda = 0$ for all $\lambda \in S$, $\theta \in \mathbb{F}_q$

$\Rightarrow (a_t + \theta b_t)_{t=0}^{N-1} \in C.$

This proves that C is linear.

Next, if $(c_t) \in R$

and $a_t = c_t - \tau$ all t

$$\Rightarrow \hat{a}_\lambda = \lambda^{\tau} \hat{c}_\lambda$$

$$\Rightarrow \hat{a}_\lambda = 0 \quad \text{all } \lambda \in S$$

$$\Rightarrow (a_t)_{t=0}^{N-1} \in C.$$

$\therefore$ C is cyclic

REMINDER: COMMON SETTING

$q = p^e$ $\quad (q, N) = 1$.

m is the order of $q \pmod N$

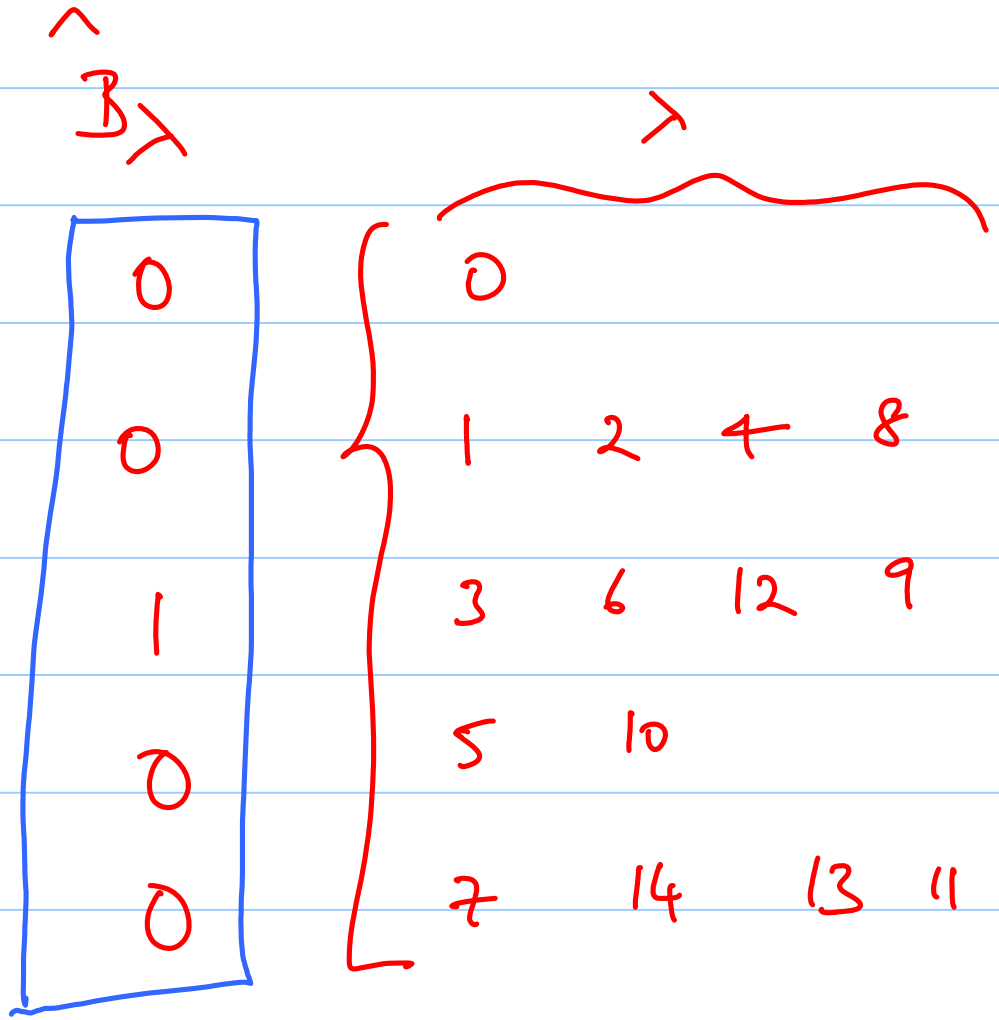
$\mathcal{C}$ is a cyclic code of block length N over $\mathbb{F}_q$.

Defn A basic sequence (B_t) of frequency λ_0, $0 \le \lambda_0 \le N-1$ is a sequence satisfying:

$$\hat{B}_\lambda = \begin{cases} 1 & \lambda = q\,\lambda_0 \quad q \in \mathbb{Z} \\ 0 & \text{else} \end{cases}$$

Eg $q = 2$ $N = 15$ $m = 4$

$\lambda_0 = 3 \Rightarrow$



Note

$\hat{B}_3 = 1 \Rightarrow$

$\hat{B}_6 = \hat{B}_{12} = \hat{B}_9$

$= 1$

as well !

Lemma 3 Let $\mathcal{C}$ be a cyclic code having null spectrum $NS_{\mathcal{C}}$.

Then

$$\mathcal{C} = \left\{ (a_t) \in \mathbb{F}_q^N \;\middle|\; \hat{a}_\lambda = 0 \;\; \forall \lambda \in NS_{\mathcal{C}} \right\}$$

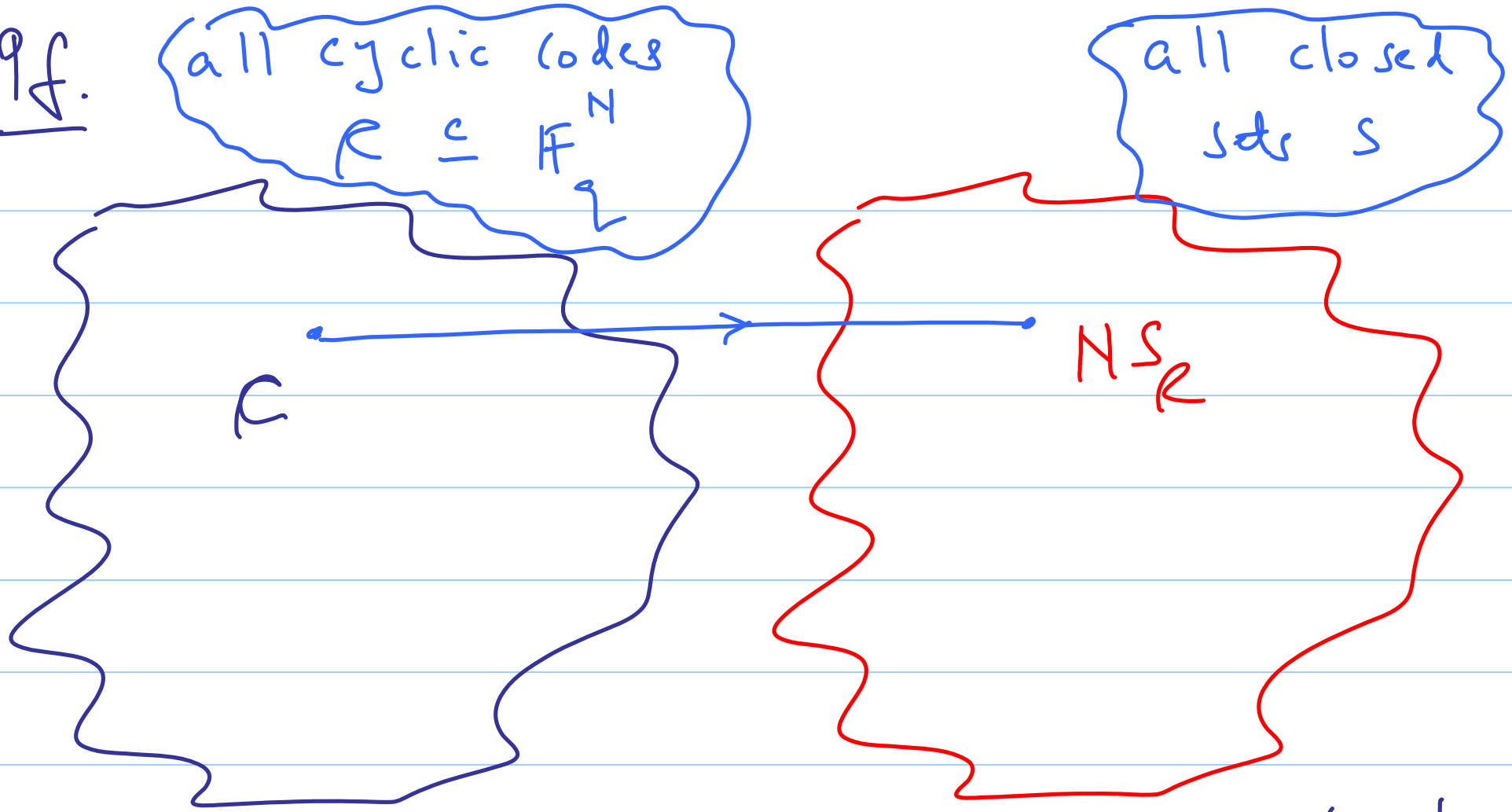
In particular, the basic sequence (β_t) of every frequency $\lambda \notin NS_{\mathcal{C}}$ belongs to $\mathcal{C}$.

(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 $\subseteq \{0, 1, \ldots, N-1\}$

Pf.



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 $\mathbb{F}_q$ equals the size of its non-null spectrum, i.e.,

$$k = |NS_R^c| = N - |NS_R|$$

Pf (presented in the Appendix).

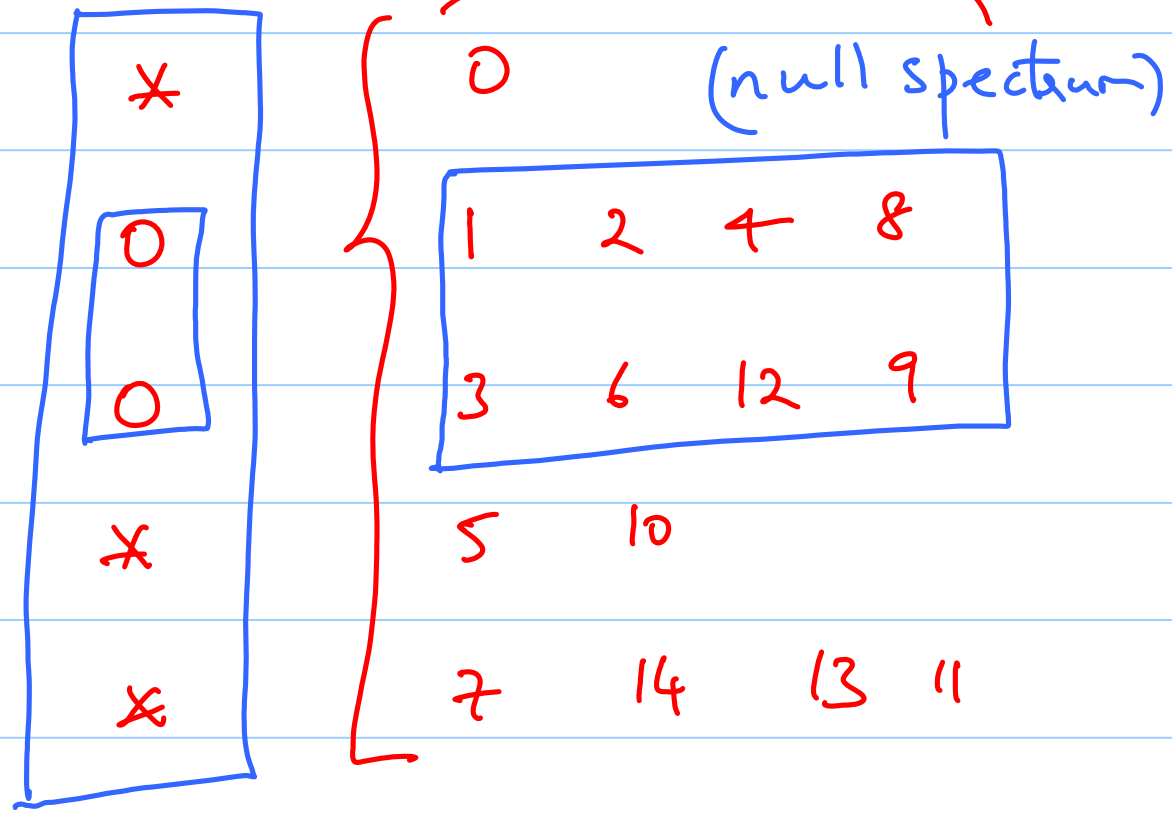
Eg $q = 2 \quad N = 15 \quad m = 4$

$$NS_{\mathcal{C}} = \{1, 2, 4, 8, 3, 6, 12, 9\}$$

$\Rightarrow \dim(\mathcal{C})$

$= 15 - 8$

$= 7.$



Thm (weight theorem) Let $(a_t) \in \mathbb{F}_q^N$ 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 $u_i \in \mathbb{F}_q$.

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

$$\{m_0, m_0+1, m_0+2, \ldots, m_0+d-2\} \subseteq \{0, 1, 2, \ldots, N-1\}$$

for some integers m_0, d with $d \geq 2$.

Then,

$$d_{min}(R) \geq d$$

d is called the designed distance of the code.

and hence $\mathbb{C}$ is a

$$[N, \; N - |NS_R|, \; \geq d]$$ code

over $\mathbb{F}_q$.

Pf. Since R is a linear code, it suffices to prove that the min Hamming weight $\geq d$ the code $\geq d$.

Suppose $(c_t) \in R$ has Hamming weight $w > 0$.

Then $(\hat{c}_\lambda)$ satisfies a linear recursion of the form:

$$\hat{c}_\lambda = \sum_{i=1}^{w} \hat{c}_{\lambda - i}\, u_i \quad \forall \lambda$$

So in particular

$$\hat{c}_{m_0+d-1} = \sum_{i=1}^{w} \hat{c}_{m_0+d-1-i}\, u_i$$

$$\hat{c} \rightarrow$$

x x . 0 0 0 * * *

0 1 . m_0 m_0+1 . . m_0+d-2 m_0+d-1 - - (N-1)

$$\leq (NS)_R$$

If $w < d$, then

$$m_0 + d - 1 - i \geq m_0 + d - 1 - (d-1)$$

$$= m_0 .$$

i.e. $\hat{c}_{m_0 + d - 1 - i} = 0$

one can then proceed to show inductively that

$$\hat{c}_\lambda = 0 \quad \forall \lambda \Rightarrow (c_t) = \underline{0}$$

which contradicts our initial assumption that $w \geq 0$.

Hence $w \geq d$ and

i. $d_{min}(C) \geq d$.

Eg Binary, (primitive) BCH codes

Here $q = 2$, $N = 2^m - 1$,

($\Rightarrow$ the multiplicative order of

$q \pmod N = m$ CHECK!)

$NS_R \geq \{ m_0 \; m_0+1 \; - \; m_0+d-2 \}$

$\therefore$ binary, primitive BCH codes

have parameters

$$\left[N = 2^m - 1, \quad N - |NS_c| > d_{min} \geq d \right]$$

A popular choice is $m_0 = 1$.

$$\therefore NS_R \geq \{1, 2, \ldots, m_0 + d - 2 = d - 1\}$$

$$= \{1, 2, \ldots, 2t\} \text{ if } d = 2t + 1$$

Note that in the 2- cyclotomic cosets mod N (i.e., the

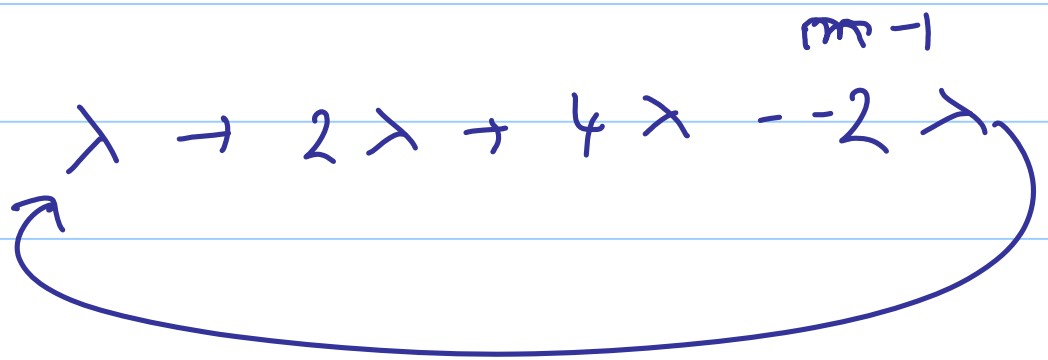
2- cyclotomic cosets $(\bmod 2^m - 1)$

(i) the size of a cyclotomic coset is at most m since

$$\lambda 2^m \pmod{n}$$

$$= \lambda 2^m \pmod{2^m - 1}$$

$$= \lambda 1$$



(ii) if $k \in \{1, 2, \ldots, 2t\}$ is even,

$k = 2\ell$, then $\ell \in \{1, 2, \ldots, 2t\}$.

and $\{k, \ell\}$ belong to the same

cyclotomic coset.

i. it suffices to ensure that

$\{1, 3, 5, \ldots, 2t-1\} \subseteq NS_C$

$\Downarrow$

set is of size t

∴ the null spectrum need be no larger than mt.

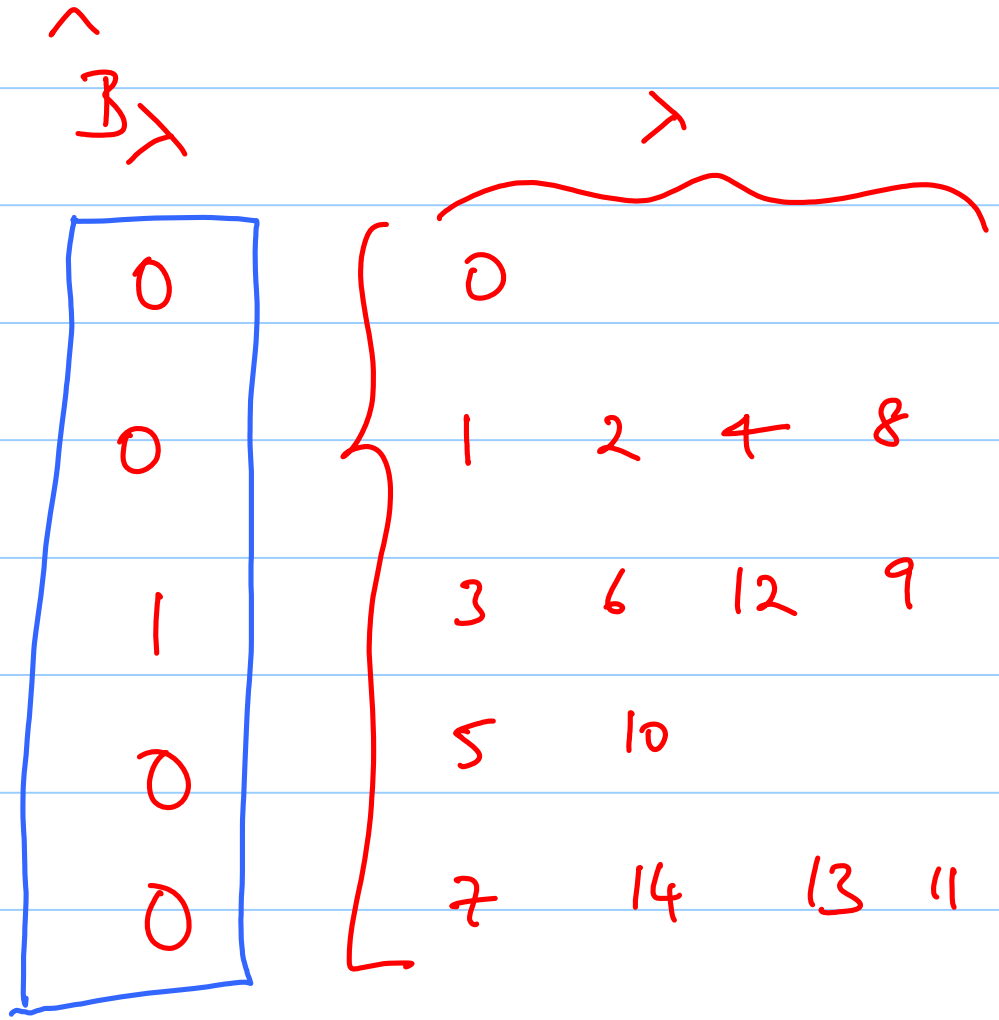
$$\therefore \dim(\mathcal{R}) = |\mathcal{N}_{\mathcal{C}}^{\mathcal{C}}| = N - |\mathcal{N}_{\mathcal{C}}| \geq 2^m - 1 - mt.$$

BCH code parameters:

$$\left[2^m - 1,\ 2^m - 1 - mt,\ d_{min} \geq 2t+1 \right]$$

Eg $q = 2 \quad N = 15 \quad m = 4$

$\lambda_0 = 3 \Rightarrow$



Note

$\hat{B}_3 = 1 \Rightarrow$

$\hat{B}_6 = \hat{B}_{12} = \hat{B}_9$

$= 1$

as well !

[

APPENDIX

Lemma 3 Let $\mathcal{C}$ be a cyclic code having null spectrum NS_{ℓ}. Then $\mathcal{C}$ contains the basic sequence of every frequency $\lambda \notin NS_{\ell}$.

Lec 42 Decoding Cyclic Codes

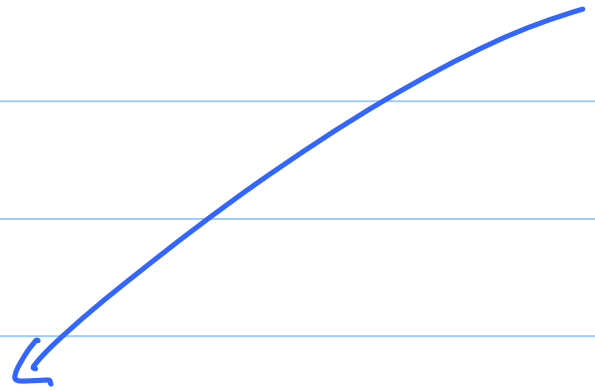
Recap

* The null spectrum theorem
* $\dim(\mathcal{C}) = N - |N^S_{\mathcal{C}}|$
* BCH codes
  - $d_{min} \geq d$
  - {proof via the

{ weight theorem

— primitive binary BCH codes

— dimension estimate



$$\left[2^m - 1, \geq 2^m - 1 - mt, d_{min} \geq 2t+1 \right]$$

(parameters of a t-error-correcting BCH code).

Thm (BCH codes) 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

$$\{m_0, m_0+1, m_0+2, \ldots, m_0+d-2\}$$

$$\subseteq \{0, 1, 2, \ldots, N-1\}$$

for some integers m_0, d with $d \geq 2$.

Then,

$$d_{min}(R) \geq d$$

d is called the designed distance of the BCH code.

and hence C is a

$$[N, \quad N - |N_{s,1}|, \quad \geq d]$$ 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 $q \pmod{N}$

$= 1$.

Eg $q = 2^4$ $\quad N = q - 1 = 15$

$m_0 = 2 \quad d = 5$

$\therefore NS_{\mathcal{C}} \supseteq \{ m_0, m_0 + 1, \ldots\ldots, m_0 + d - 2 \}$

$= \{ 2, 3, \ldots, 5 \}$

$\therefore \boxed{d_{min} \geq 5}$

q cyclotomic cosets (mod n)

$a \sim b$ iff $a = q^i b \pmod n$

$= b \pmod n$

Since $q = 1 \pmod n$!

$\therefore a \sim b \iff a = b.$

Thus each equivalence class contains

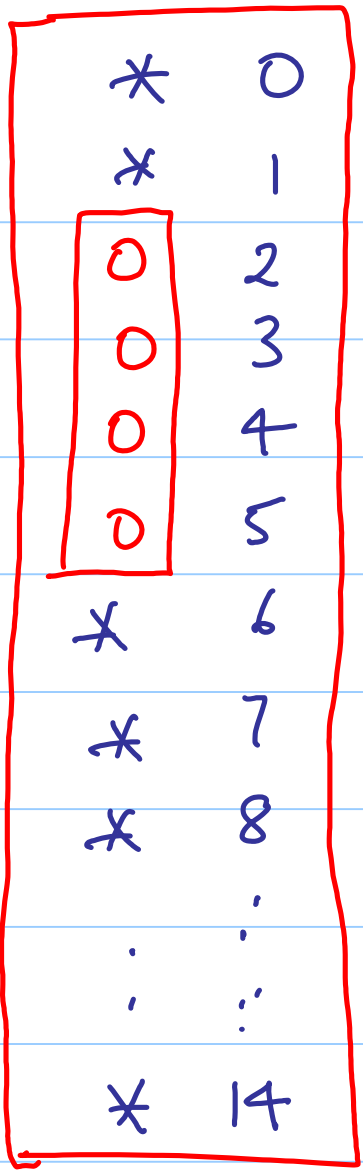
only a single element:

$\therefore \dim(\mathcal{C}) = N - (d-1)$

$\therefore k = N - d + 1 \geq N - d_{min} + 1$

hence the code is MDS

and $d_{min} = d$.



General parameters of a R-S code:

$$[N, \quad N-d+1, \quad d]$$

DECODING BCH CODES

BCH Code Setting

$q = p^e \qquad (q, N) = 1$

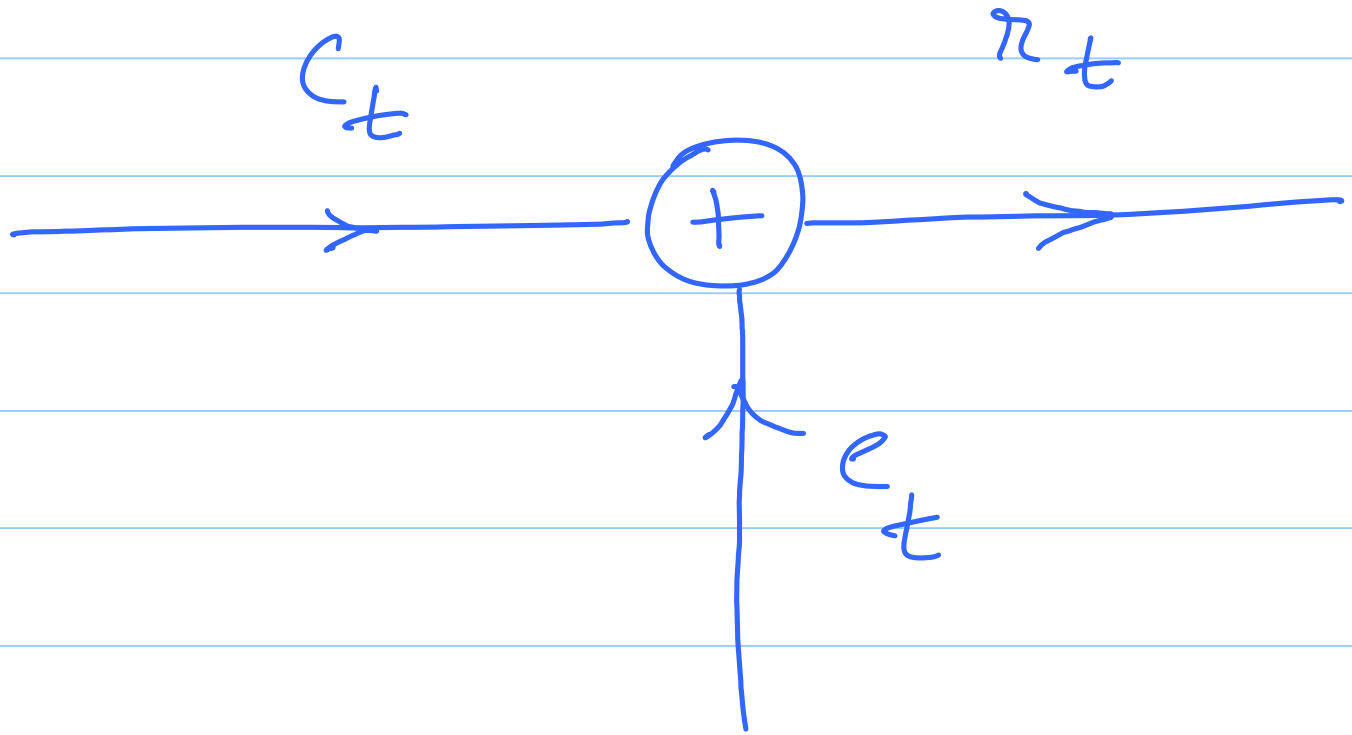
m is the order of q $\pmod N$

C – cyclic code $\subseteq \mathbb{F}_q^N$

$\alpha \in \mathbb{F}_{q^m}$ has order N

$NS_C \supseteq \{m_0, m_0+1, \ldots\ldots, m_0+d-2\}$

Channel Model:



$c_t, e_t, r_t \in \mathbb{F}_q$, all $0 \le t \le n-1$

$$\hat{r}_\lambda \triangleq \sum_{t=0}^{N-1} r_t \, \alpha^{\lambda t}$$

$$= \sum_t c_t \alpha^{\lambda t} + \sum_t e_t \alpha^{\lambda t}$$

$$= \hat{c}_\lambda + \hat{e}_\lambda$$

$$\therefore \boxed{\hat{r}_\lambda = \hat{e}_\lambda \qquad m_0 \le \lambda \le m_0 + d - 2}$$

"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 $n \triangleq d-1$ and define:

$$S_{\infty}(z) = \sum_{\lambda=0}^{\infty} \hat{e}_{\lambda+m_0} z^{\lambda}$$

$$= \sum_{\lambda=0}^{\infty} \sum_{t=0}^{N-1} e_t \, \alpha^{(\lambda+m_0)t} \, z^{\lambda}$$

$$= \sum_{i=1}^{w} e_{t_i} \, \alpha^{m_0 t_i} \sum_{\lambda=0}^{\infty} \left(z \alpha^{t_i}\right)^{\lambda}$$

$$= \sum_{i=1}^{w} \frac{e_{t_i} \, \alpha^{m_0 t_i}}{\left(1 - z \alpha^{t_i}\right)}$$

where
w = # of errors in locations $\{t_i\}$

$$= \frac{w(z)}{\sigma(z)} \quad (\text{say})$$

where:

$$\sigma(z) \triangleq \prod_{i=1}^{w} (1 - \alpha^{t_i} z)$$

ERROR LOCATOR POLYNOMIAL

$$\omega(z) \triangleq \sum_{i=1}^{w} e_{t_i} \alpha^{m_0 t_i} \prod_{j=1, j \neq i}^{w} (1 - \alpha^{t_j} z)$$

ERROR EVALUATOR POLYNOMIAL

Note:

$$\deg(\tau(z)) = w \leq \left\lfloor \frac{d-1}{2} \right\rfloor = \left\lfloor \frac{r}{2} \right\rfloor$$

$$\deg\left(w(z)\right) = w-1 \leq \left\lfloor \frac{n}{2} \right\rfloor - 1$$

Replacing $S_0(z)$ by $S(z)$

$$S(z) = \sum_{\lambda=0}^{n-1} \hat{e}_{\lambda+m_0} z^{\lambda}$$

$$= S_0(z) \pmod{z^r}$$

$$\therefore \quad S(z) = \frac{w(z)}{\sigma(z)} \pmod{z^{2t}}$$

$$\therefore \quad S(z) = \frac{w(z)}{\sigma(z)} + A(z)\, z^{2t}$$

$\Downarrow$

power series in z with no neg. exponents

$$\therefore \quad \sigma(z)\, S(z) - w(z) = \underbrace{A(z)\, \sigma(z)}_{\text{polynomial } B(z)}\, z^{r}$$

$$\therefore \quad \boxed{w(z) = \sigma(z)\, s(z) + B(z)\, z^{r}}$$

$\deg \leq \left\lfloor \frac{r}{2} \right\rfloor - 1$ $\qquad$ $\deg \leq \left\lfloor \frac{r}{2} \right\rfloor$

this is reminiscent of gcd computation via the Euclidean division algorithm.

DECODING EXAMPLE

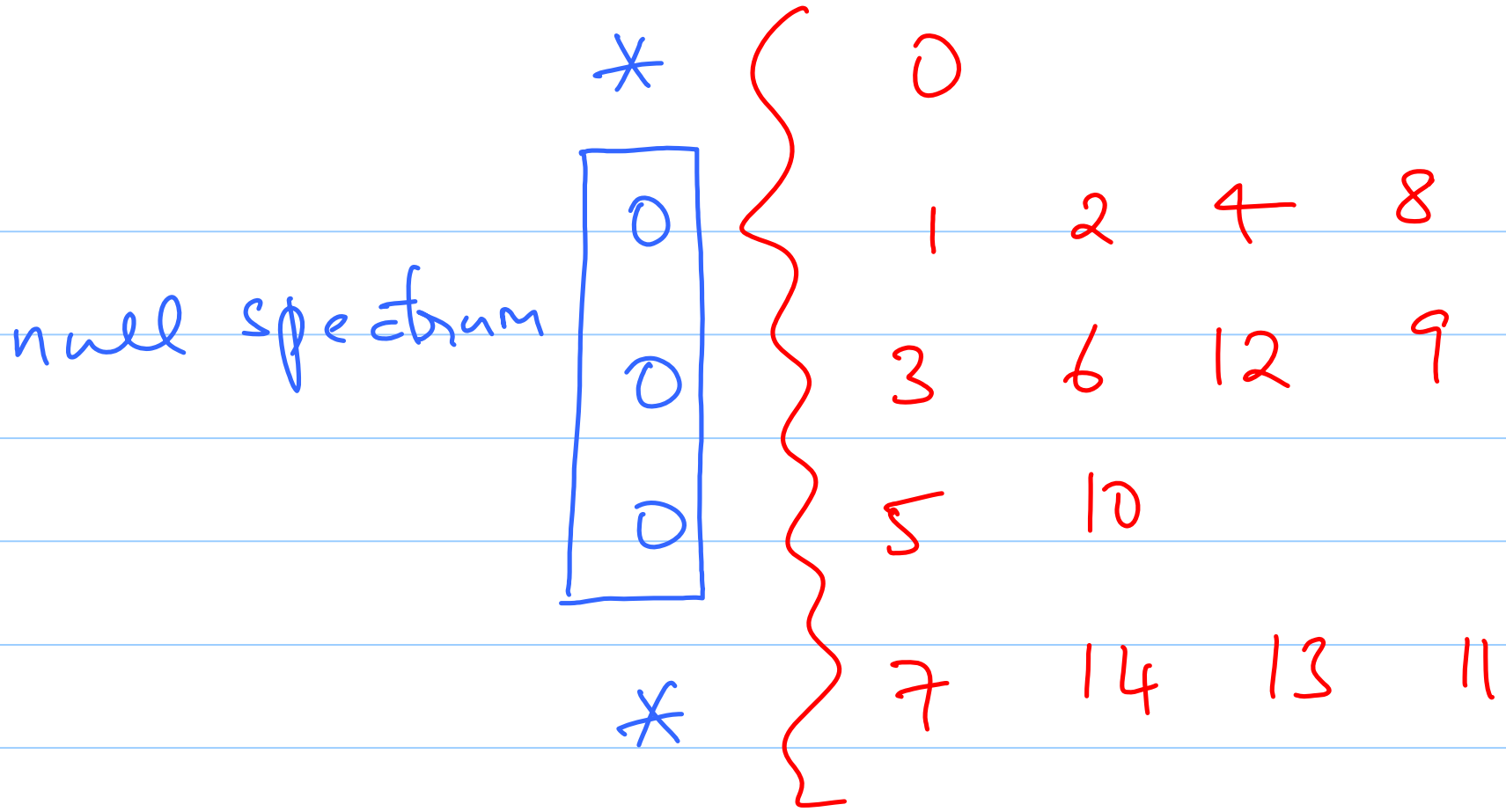
$q = 2 \quad N = 15 \quad m = 4 \quad d = 5$

$$\{m_0, m_0+1, \ldots, m_0+d-2\}$$

$$= \{1, 2, 3, 4, 5, 6\}$$

(Triple-error correcting BCH code)

consequent choice of null spectrum:



$$x(t) = \begin{cases} 1 & t \in \{0, 3, 4, 5, 6, 7, 8, 12, 13\} \\ 0 & \text{else} \end{cases}$$

STEP 1 Compute $\hat{r}_\lambda$

$$m_0 \le \lambda \le m_0 + d - 2$$

$$\therefore \hat{r}_\lambda = \sum_{t=0}^{14} r_t \alpha^{\lambda t}$$

$$= \alpha^0 + \alpha^3 + \alpha^4 + \alpha^5 + \alpha^6 + \alpha^7 + \alpha^8 + \alpha^{12} + \alpha^{13}$$

$$= \alpha^0 \left(1 + \alpha^3 \left(1 + \alpha \left(1 + \alpha \left(1 + \alpha \left(1 + \alpha \left(1 + \alpha^4 \underbrace{(1+\alpha}_{\alpha^4}\right)\right)\right)\right)\right)\right)\right)$$

$$\underbrace{\qquad\qquad}_{\alpha^2}$$

$$\underbrace{\qquad\qquad\qquad\qquad}_{\alpha^{14}} \quad \alpha_c$$

$$= \alpha^6$$

$$\hat{n}_2 = \left[\hat{n}_1\right]^2 = \alpha^{12}$$

$$\hat{\pi}_4 = [\hat{\pi}_1]^4 = \alpha^9$$

$$\hat{\pi}_3 = \alpha^{13} \quad \text{(direct computation)}$$

$$\hat{\pi}_5 = \alpha^5 \quad (\text{"} \quad \text{"})$$

$$\hat{\pi}_6 = [\hat{\pi}_3]^2 = \alpha^{11}$$

STEP 2 :

$$S(z) = \sum_{\lambda=0}^{d-2} \hat{\lambda}_{\lambda+m_0} z^{\lambda}$$

$$= \alpha^{11} z^5 + \alpha^5 z^4 + \alpha^9 z^3 + \alpha^{13} z^2 + \alpha^{12} z + \alpha^6$$

$$= \left(\alpha^{11} \; \alpha^5 \; \alpha^9 \; \alpha^{13} \; \alpha^{12} \; \alpha^6 \right)$$

SHORTHAND NOTATION

	z^2	$s(z)$	Quotient
$z^r = z^6$	1	0	
$\alpha^{11}\ \alpha^{5}\ \alpha^{9}\ \alpha^{13}\ \alpha^{12}\ \alpha^{6}$	0	1	$\alpha^{4}\ \alpha^{13}$
$\alpha^{8}\ \alpha^{12}\ \alpha^{6}\ 0\ \alpha^{4}$		$\alpha^{4}\ \alpha^{13}$	$\alpha^{3}\ \alpha^{2}$
deg $\alpha^{14}\ \alpha^{3}\ \alpha^{2}\ 0$		$\alpha^{7}\ \alpha^{11}\ 0$	$\alpha^{9}\ 0$
$> \lfloor \frac{r}{2} \rfloor - 1$			
$\alpha\ 0\ \alpha^{4}$		$\alpha\alpha^{5}\ \alpha^{4}\ \alpha^{13}$	
$= k\ w(z)$		$= k\ \sigma(z)$	

$\alpha^4\ \alpha^{13}$ ⇒ Quotient

$\alpha^{11}\ \alpha^{5}\ \alpha^{9}\ \alpha^{13}\ \alpha^{12}\ \alpha^{6}$) 1 0 0 0 0 0 0

$1\ \alpha^{9}\ \alpha^{13}\ \alpha^{2}\ \alpha\ \alpha^{10}$

$\alpha^{9}\ \alpha^{13}\ \alpha^{2}\ \alpha\ \alpha^{10}\ 0$

$\alpha^{9}\ \alpha^{3}\ \alpha^{7}\ \alpha^{11}\ \alpha^{10}\ \alpha^{4}$

Rem = $\alpha^{8}\ \alpha^{12}\ \alpha^{6}\ 0\ \alpha^{4}$

$\alpha^3 \quad \alpha^2$ = Quotient

$\alpha^8 \quad \alpha^{12} \quad \alpha^6 \quad \alpha^4$ | $\alpha^{11} \quad \alpha^5 \quad \alpha^9 \quad \alpha^{13} \quad \alpha^{12} \quad \alpha^6$

$\alpha \quad \alpha \quad \alpha \quad 0 \quad \alpha$ | $\alpha \quad \alpha \quad \alpha \quad \alpha \quad \alpha \quad \alpha$

$\alpha^{11} \quad 1 \quad \alpha^9 \quad 0 \quad \alpha^7$

$0 \quad \alpha^{10} \quad 0 \quad \alpha^{13} \quad \alpha^2 \quad \alpha^6$

$\alpha^{10} \quad \alpha^{14} \quad \alpha^8 \quad 0 \quad \alpha^6$

Rem = $\alpha^{17} \quad \alpha^3 \quad \alpha^2 \quad 0$

$2^9\ 0$ Quotient

$2^{14}\ 2^3\ 2^2\ 0 \;\big|\; 2^8\ 2^{12}\ 2^6\ 0\ 2^4$

$2^8\ 2^{12}\ 2^{11}\ 0$

Rem $= 2\ 0\ 2^4$

$$1 + (\alpha^4 z + \alpha^{13})(\alpha^3 z + \alpha^2)$$

$$= 1 + z^2 \alpha^7 + z(\alpha + \alpha^6) + 1$$

$$= z^2 \alpha^7 + z\alpha^{11}$$

$$(\alpha^4 z + \alpha^{13}) + (\alpha^9 z)(\alpha^7 z^2 + \alpha^4 z)$$

$$= \underbrace{z^3\alpha + z^2\alpha^5 + \alpha^4 z + \alpha^{13}}$$

$$\therefore \prod_{i=1}^{\omega} (1 - \alpha^{i} z) = k(\quad \rightarrow \quad)$$

$$\therefore \quad k = \alpha^2 .$$

$$\therefore \sigma(z) = z^3\alpha^3 + z^2\alpha^7 + \alpha^6 z + 1$$

Chien search

$z = 1 \Rightarrow \alpha^3 + \alpha^7 + \alpha^6 + 1$

$= 1 + \alpha^3 \left(1 + \alpha^3 \underbrace{(1+\alpha)}_{\alpha^4} \right)$

$\underbrace{\qquad\qquad}_{\alpha^7}$

$= \alpha^{11} \neq 0$

$z = \alpha^2 \Rightarrow$

$$\alpha^3\alpha^6 + \alpha^7\alpha^4 + \alpha^6\alpha^2 + 1$$

$$= 1 + \alpha^8\left(1 + \alpha\underbrace{\left(1 + \alpha^2\right)}_{\alpha^8}\right)$$

$$\underbrace{\qquad\qquad}_{\alpha^7}$$

$$= 0$$

$\therefore$ $\boxed{z = \alpha^2 \text{ is a zero}}$

Similarly it turns out that

$\prod_{i=1}^{w} (1 - \alpha^{t_i} z)$ has

$z = \alpha^{2}, \alpha^{11}, \alpha^{4}$ as the

3 zeros.

$\therefore$ error locations are the

reciprocals:

$\alpha^{-2} \quad \alpha^{-11} \quad \alpha^{-14} \Rightarrow \alpha^{13}, \alpha^{4}, \alpha$

$\therefore \quad t_1 = 1$

$t_2 = 4$

$t_3 = 13$