Codes based on Graphs

<u>Problem</u>: For sufficiently large n, construct "explicit" codes

with poly-time encoding & decoding $(O(n^2))$,

Rate $\approx C - \varepsilon$, $\Pr(\text{error}) \leq 10^{-5}$ $\quad 2^{-\Theta(n)}$

$\Theta(1/n^2)$

Convolutional Codes

So far: Block codes (linear block codes)

Encoding: linear (matrix multiplication)

Encoding: Filter

Message: Signal $m(t) \to$ discrete time

Codeword: $c(t) \to$ discrete time

$$m(t) \rightarrow \boxed{\begin{matrix}\text{ENC} \\ g(t)\end{matrix}} \rightarrow c(t)$$

Convolutional code: LTI system!

- Nonsystematic feed forward CC (convolutional code)

  FIR

$$g_1(t) = \delta(t) + \delta(t-1) + \delta(t-2)$$

$$g_2(t) = \delta(t) + \delta(t-1)$$

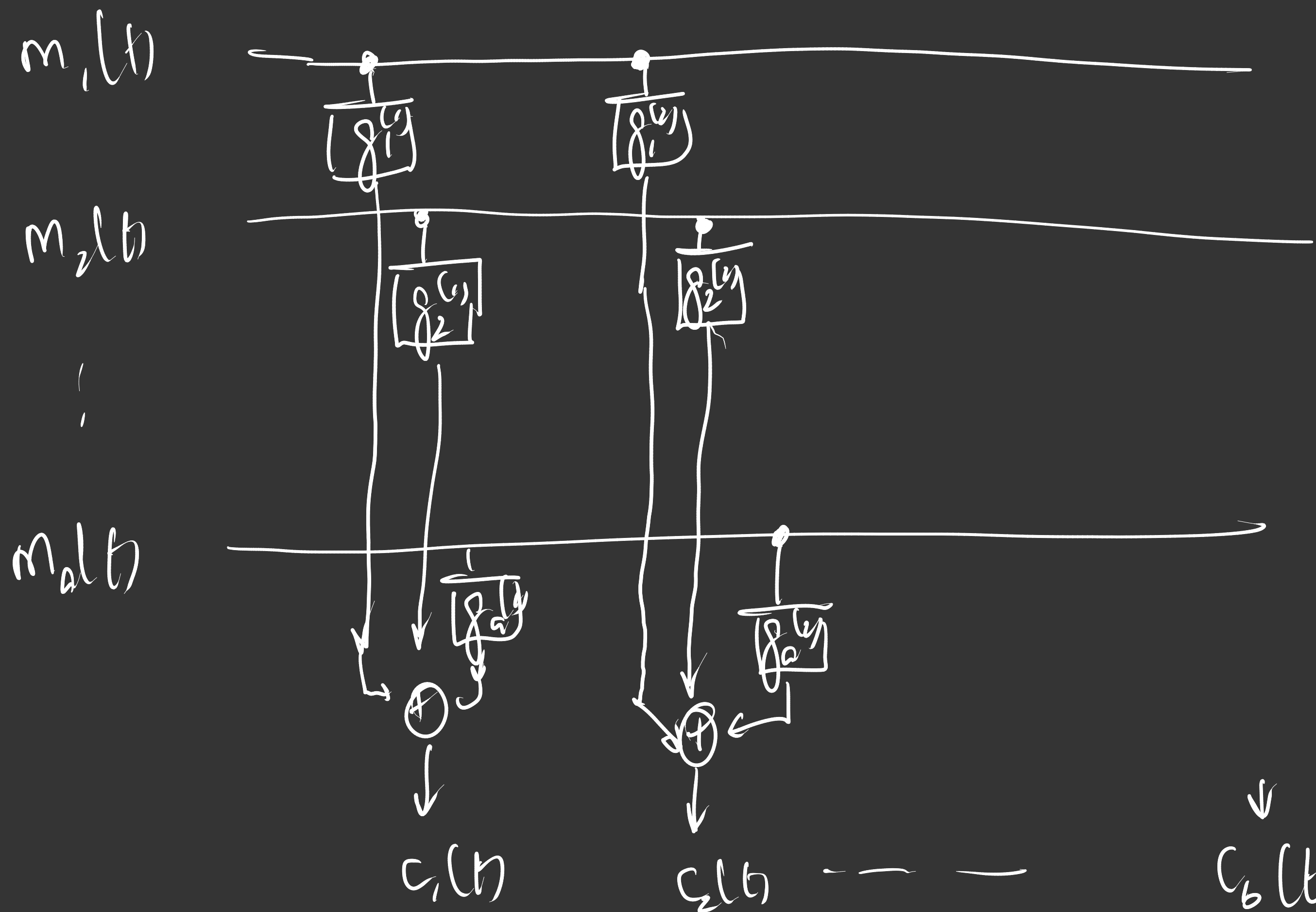
$$c(t) = \begin{bmatrix} c_1(t) = m(t) * g_1(t) \\ c_2(t) = m(t) * g_2(t) \end{bmatrix}$$

Rate = $1/2$

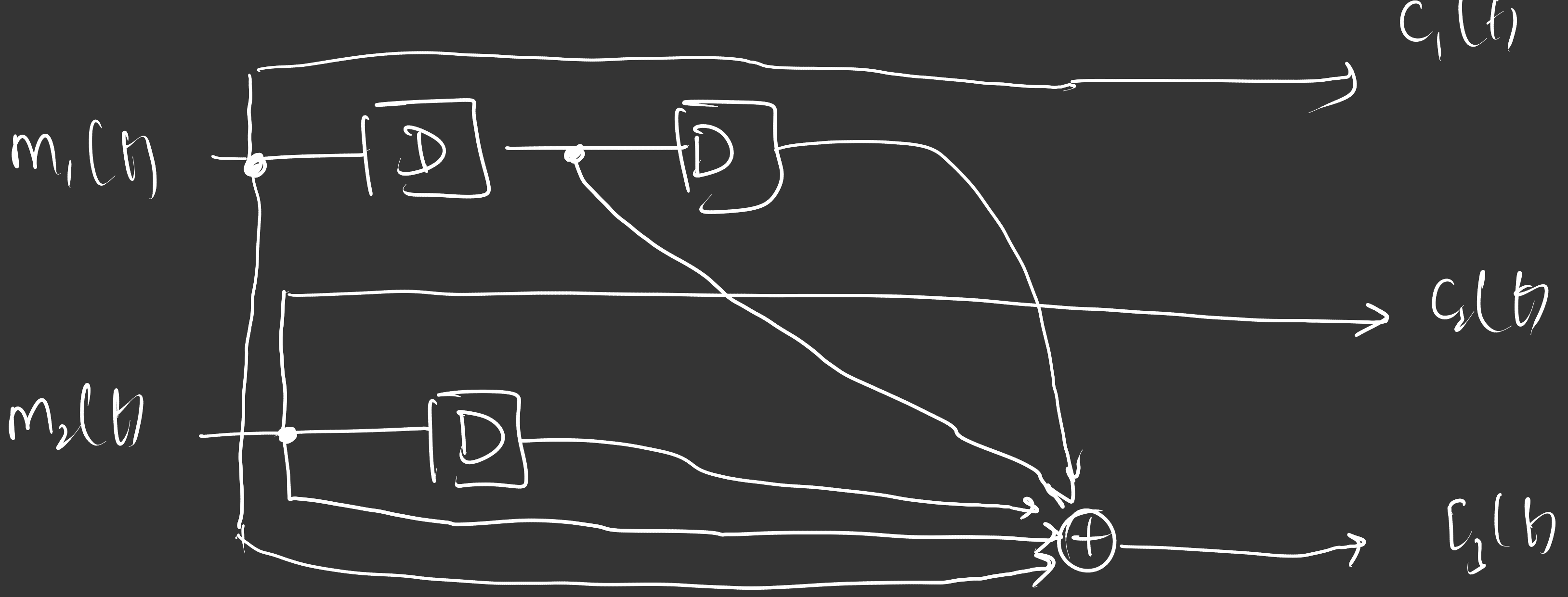
Dfn: A rate a/b convolutional code has a i/p streams $m_1(t), m_2(t), \dots, m_a(t)$ & b o/p streams $c_1(t), c_2(t), \dots, c_b(t)$ & is defined using $a \times b$ impulse responses $g_1^{(1)}, g_1^{(2)}, \dots, g_1^{(b)}, \dots, g_a^{(b)}$

$g_i^{(j)}$: impulse response / generator sequence for i/p i, & o/p j

$$c_j(t) = g_1^{(j)}(t) * m_1(t) + g_2^{(j)}(t) * m_2(t) + \cdots + g_a^{(j)}(t) * m_a(t)$$



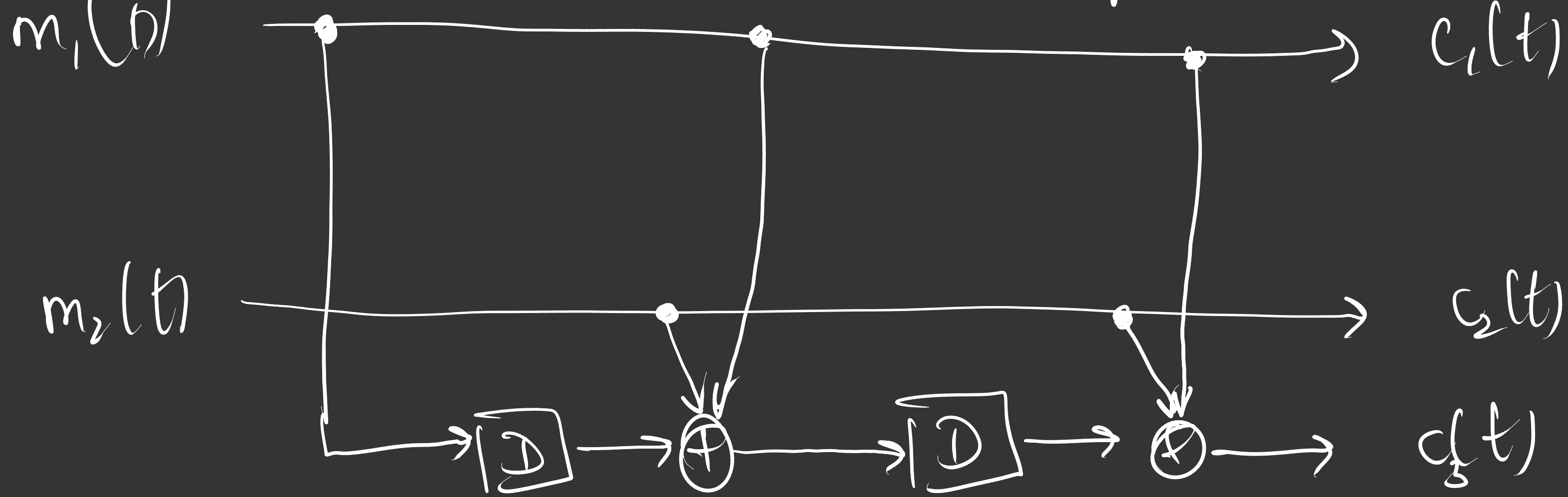
☆ All operations performed over $\mathbb{F}_q$.



$$c_3(t) = m_1(t) + m_1(t-1) + m_1(t-2) + m_2(t) + m_2(t-1)$$

↑ Controller canonical form

& Observer canonical form



Generalized distributive law

For any $a, b, c \in \mathbb{F}$: $a(b+c) = ab + ac$ $\longrightarrow$ Distributive law.

$f(x,y)$ $\quad$ $g(y,z)$ $\quad$ $x \in X$ $\quad$ $y \in Y$, $z \in Z$

$$A = \sum_{x,y,z} f(x,y)\, g(y,z) \longrightarrow \text{Sum-of-products}$$

Complexity : $|X||Y||Z| + |X||Y||Z| - 1$

$= \Theta(|X||Y||Z|)$

If $|X| = |Y| = |Z| = 10^3$, complexity $= \Theta(10^9)$

$$A = \sum_{y}\sum_{z}\sum_{x} f(x,y)\, g(y,z) = \sum_{y}\Big(\underbrace{\sum_{x} f(x,y)}_{|X|-1}\Big)\Big(\underbrace{\sum_{z} g(y,z)}_{|Z|-1}\Big)$$

complexity : $\Theta(|Y|(|X|+|Z|))$

$|Y|(|X|+|Z|-1-1+1)$

$\Theta(10^6)$

Commutative semiring: $(K, +, \cdot)$ is a commutative semiring if:

① $(K, +)$ is a commutative monoid (commutative associative unique identity exists)

② $(K, \cdot)$ is a commutative monoid

③ $\forall\, a, b, c \in K,$ $a \cdot (b + c) = ab + ac.$

Examples: ① $K = [0, \infty]$ $+, \cdot$

YES

② $K = [0, \infty]$ $\min, \cdot$ → multiplication

(Min-prod semiring)

$\min(1, x) = x \quad \forall x \in K.$

Identity 1: ∞

Identity 2: 1

$a \cdot (\min(b, c)) = \min(ab, ac)$

② $K = [-\infty, \infty]$ $\quad$ min, $\cdot$ → multiplication

Not distributive

④ $K = [0, \infty]$ $\quad$ max, $\cdot$

Max-prod semiring

Identity 1 : 0

Identity 2 : 1

⑤ $K = [-\infty, \infty]$ $\quad$ min, + $\quad$ Min-sum semiring

$$a + \min(b, c) = \min(a+b, a+c)$$

$$\min(a, b+c) = \min(a,c) + \min(a,b)$$

YES

(c) $K = [-\infty, \infty]$ max, $\cdot$

Id1 : $-\infty$

Id2 : 1

$a \cdot \max(b, c) = \max(ab, ac)$

$-2 \max(1, 2) = -4$

$\max(-2, -4) = -2$

(d) $K = \{0, 1\}$ OR AND

Id1: 0

Id2: 1

Marginalize a product function

R Sum-prod $\rightarrow$ $\sum_{x,y,z} f(x,y)\; g(y,z)$ — (1) $\qquad O(|x||y||z|)$

$\sum_{y,z} f(x,y)\; g(y,z)$

Max-prod $\rightarrow$ $\max_{x,y,z} f(x,y)\; g(y,z)$ — (2)

Min-sum $\rightarrow$ $\min_{x,y,z} \left[f(x,y) + g(y,z) \right]$ — (3)

Definitions :

$x_1, x_2, \ldots, x_n$ local variables

$x_i \in A_i \rightarrow$ domains

$\alpha_1, \alpha_2 \ldots \alpha_m$ local functions

For each i, $S_i \subseteq \{1, 2, \ldots n\}$

which variables participate in α_i

$\alpha_1 - S_1 = \{1, 2\}$

$\alpha_2 - S_2 = \{3, 4\}$

$\alpha_1 : A_1 \times A_2 \longrightarrow K$ commutative semiring

$$\mathcal{A}_{S_i} = \{ \mathcal{A}_{i_1} \times \mathcal{A}_{i_2} \times \cdots \mathcal{A}_{i_{|S_i|}} : i_j \in S_i \}$$

eg: $\mathcal{A}_{S_1} = \mathcal{A}_1 \times \mathcal{A}_2$

$\mathcal{A}_{S_2} = \mathcal{A}_3 \times \mathcal{A}_4$

$x_{S_1} = (x_1, x_2) \qquad x_{S_2} = (x_3, x_4)$

x_{S_i}

$\alpha_i(x_{S_i})$

$\alpha_i : \mathcal{A}_{S_i} \to K$

local function or

local kernel

Global function / global kernel

$$\beta(x_1 - x_n) = \alpha_1(x_{S_1})\,\alpha_2(x_{S_2}) \cdots \alpha_m(x_{S_m})$$

Goal: Marginalize β

$S \subseteq \{1, 2, - n\}$

In fact, choose S to be one of the S_i's

$$\beta(x_{S_i}) = \sum_{x_{S_i^c}} \beta(x_1 - x_n)$$

Brute force : Complexity : $\Theta\left(|A_1| |A_2| \cdots |A_n| \right)$

Eg : $\beta(x_1) = \sum_{x_2 x_3 x_4} f(x_1, x_2)\, g(x_1, x_3, x_4) \cdot 1$

$\alpha_1 = f \qquad , \qquad \alpha_2 = g$

$S_1 = \{1, 2\} \qquad S_2 = \{1, 3, 4\}$

$A_{S_1} = A_1 \times A_2 \qquad A_{S_2} = A_1 \times A_3 \times A_4$

$S_3 = \{1\} \qquad \alpha_3 = 1$

Hadamard transform

$$F(y_1 \cdots y_n) = \sum_{x_1 \cdots x_n} f(x_1 \cdots x_n) \, (-1)^{x_1 y_1 + x_2 y_2 + \cdots + x_n y_n}$$

$$= \sum_{x_1 \cdots x_n} f(x_1 \cdots x_n) \, (-1)^{x_1 y_1} (-1)^{x_2 y_2} \cdots (-1)^{x_n y_n}$$

$n+2$

$\alpha_1 = f$, $\quad S_1 = [n] = \{1, 2, \ldots n\}$

$\alpha_2 = (-1)^{x_1 y_1}$ $\quad S_2 = \{1_x \; 1_y\}$

$\alpha_3 = (-1)^{x_2 y_2}$ -

$\alpha_{n+2} = 1$ $\quad S_{n+2} = \{1_y \; 2_y \; \ldots \; n_y\}$

$$F(y) = \sum_{x} f(x)\, e^{j \phi x y}$$

$$F(y_1 \cdots y_n) = \sum_{x_1 \cdots x_n} f(x_1 \cdots x_n)\, e^{j 2\pi \sum_{i=1}^{n} x_i y_i}$$

Eg: Decoding linear codes

$$x_1 \cdots x_n \longrightarrow \boxed{\text{Ch}} \longrightarrow y_1 \cdots y_n$$

$$\arg \max_{(x_1 \cdots x_n) \in C} p(y_1 \cdots y_n \mid x_1 \cdots x_n)$$

$$\arg\max_{(x_1 \ldots x_n) \in C} \prod_{i=1}^{n} p(y_i \mid x_i)$$

$$= \arg\max_{(x_1 \ldots x_n) \in \{0,1\}^n} \left(\prod_{i=1}^{n} p(y_i \mid x_i) \right) \mathbb{1}\{x_1 \ldots x_n \in C\}$$

$$= \arg\max_{(x_1 \ldots x_n) \in \{0,1\}^n} \left(\prod_{i=1}^{n} p(y_i \mid x_i) \right) \mathbb{1}\{H\underline{x}^T = \underline{0}\}$$

$$= \arg\max_{(x_1 \ldots x_n) \in \{0,1\}^n} \left(\prod_{i=1}^{n} p(y_i \mid x_i) \right) \prod_{j=1}^{n-k} \mathbb{1}\{\underline{h}_j \underline{x}^T = 0\}$$

If H is sparse, then $\{\underline{h}_j \underline{x}^T = 0\}$ involves small # of x_i's

Compute $\quad x_i^{(ML)} = \arg\max_{x_i} \Big[\underbrace{\max_{x_{\{i\}^c}} p(y_1 \dots y_n \mid x_1 \dots x_n) \, \mathbb{1}_{\{x_1 \dots x_n \in C\}}}_{\text{MPF}} \Big]$

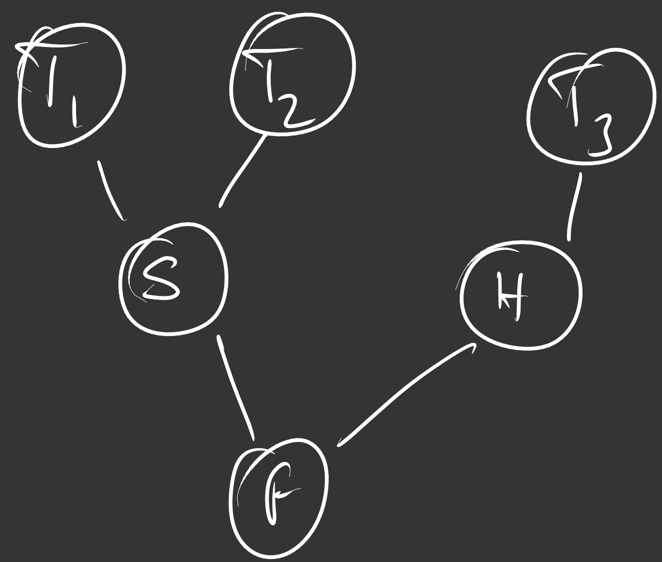
$$\max_{x_{i^c}} \left(\prod_{i=1}^{n} p(y_i \mid x_i) \right) \prod_{j=1}^{n-k} \mathbb{1}_{\{\underline{h}_j \underline{x}^T = 0\}}$$

$$\Leftrightarrow \quad \min_{x_{i^c}} \sum_{i=1}^{n} -\log \left(p(y_i \mid x_i) \right) + \sum_{j=1}^{n-k} -\log \mathbb{1}_{\{\underline{h}_j \underline{x}^T = 0\}}$$

(3) Probabilistic Graphical Models

$T_1 \quad T_2 \quad T_3 \rightarrow$ Temperature

$F \rightarrow$ fire



$$p(F, S, H, T_1, T_2, T_3) = p(T_1)\, p(T_2)\, p(T_3)\, p(S \mid T_1, T_2)\, p(H \mid T_3)\, p(F \mid S, H)$$

$$\underset{f \in \{0,1\}}{\arg\max} \; p(f \mid T_1 T_2 T_3)$$

$$=$$

$$\sum_{S, H} p(S \mid T_1 T_2)\, p(H \mid T_3)\, p(f \mid S, H)$$

<u>Convolutional</u> <u>Codes</u> (HMMs)

$$x_t = g_0 m_t + g_1 m_{t-1} + \cdots + g_{k-1} m_{t-k+1}$$

find $\underset{m_t}{\arg\max} \; p(m_t \mid y_1 \ldots y_n)$

$$\sum_{\underline{x}\, m_{i,c}} p(y, \underline{x}, \underline{m}) = \sum_{\underline{x}\, m_{i,c}} p(y \mid \underline{x})\, p(\underline{x} \mid \underline{m})$$

Recall :

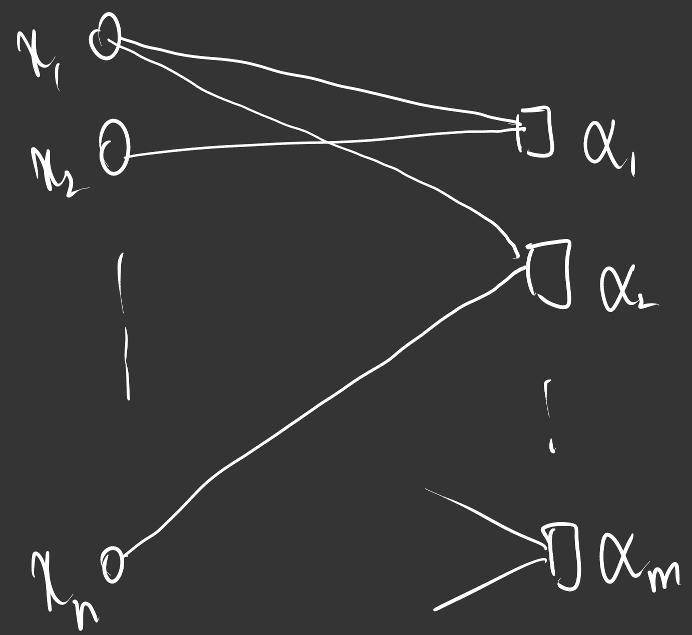
$$\beta = \sum_{x_1 x_2 \cdots x_n} \alpha_1(x_{S_1})\, \alpha_2(x_{S_2}) \cdots \alpha_m(x_{S_m})$$

$$x_i \in \mathcal{A}_i \qquad \Theta\left(m\, |\mathcal{A}_1||\mathcal{A}_2| \cdots |\mathcal{A}_n|\right)$$

$$\beta(x_{S_j}) = \sum_{x_{S_j^c}} \alpha_1(x_{S_1}) \cdots \alpha_m(x_{S_m})$$

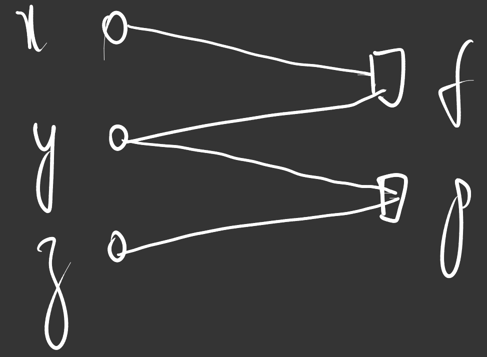
Graphical representations of the MPF problem

① Factor graph

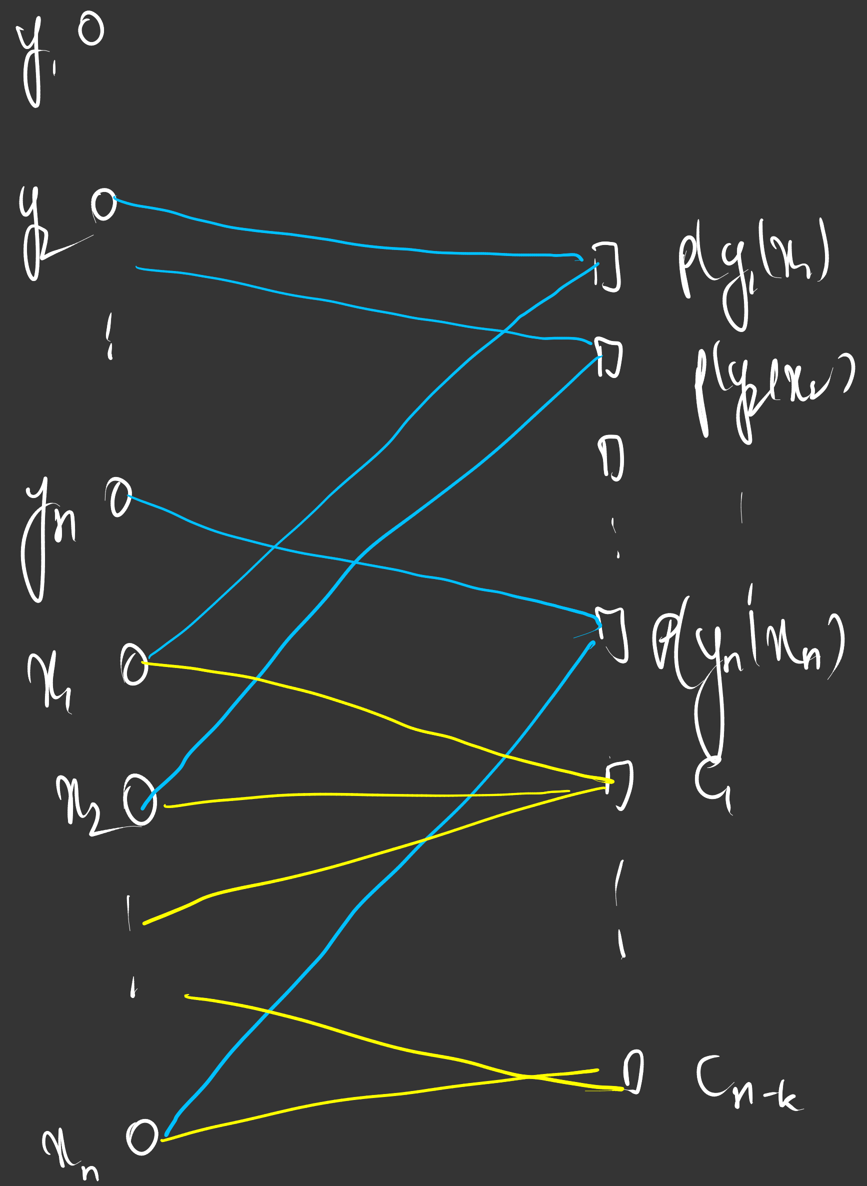


Eg:

① $f(x,y)\, g(y,z)$

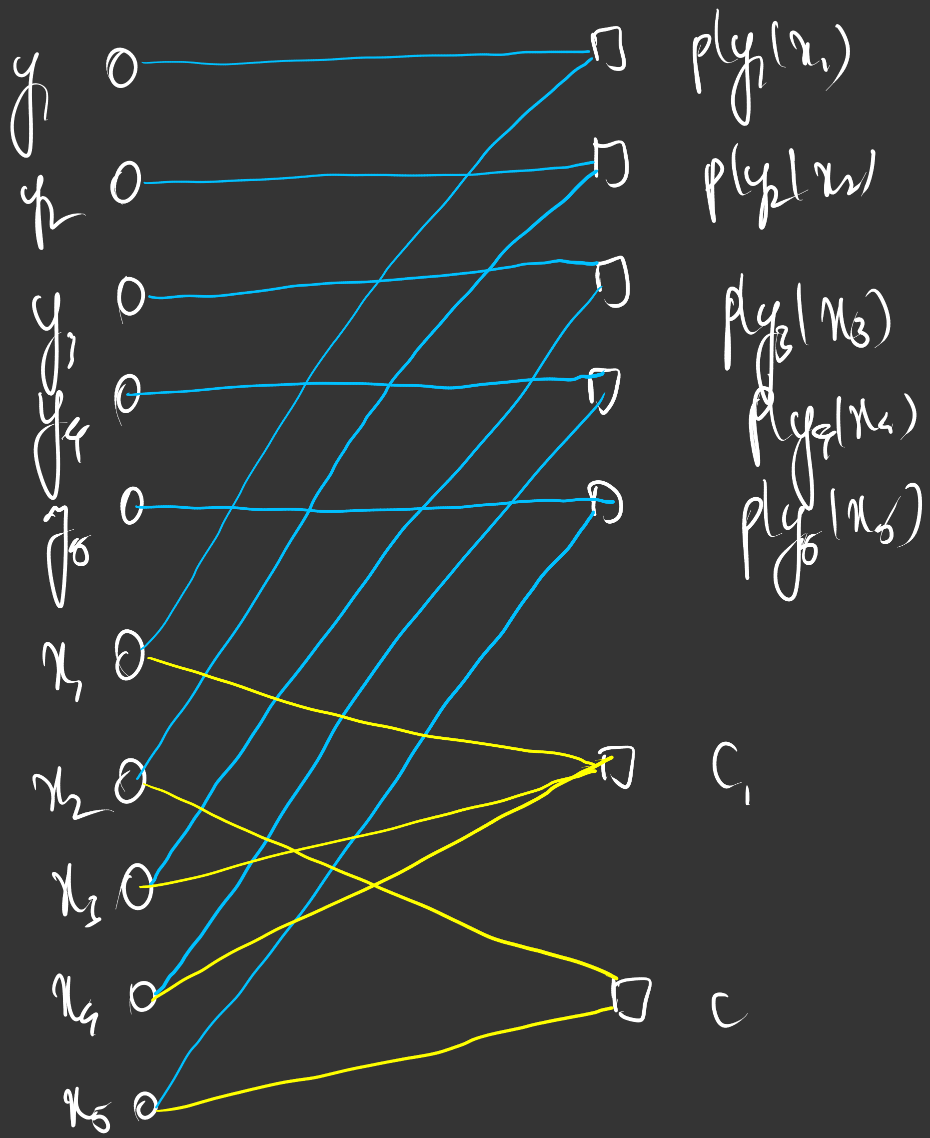


② $p(\underline{x},\underline{y}) = \left(\prod_{i=1}^{n} p(y_i \mid x_i) \right) \prod_{j=1}^{n-k} \mathbb{1}\{\underline{h}_j \underline{x}^T = 0\}$



y_1
y_2
y_n
x_1
x_2
x_n
$p(y_1|x_1)$
$p(y_2|x_2)$
$p(y_n|x_n)$
c_1
c_{n-k}

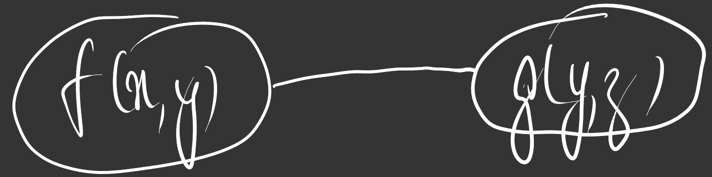
$$H = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$



Graphical representation 2: Junction tree

- Tree
- Each vertex corresponds to a local function
- The subgraph of all local domains/ local fns that involve any x_i is connected.

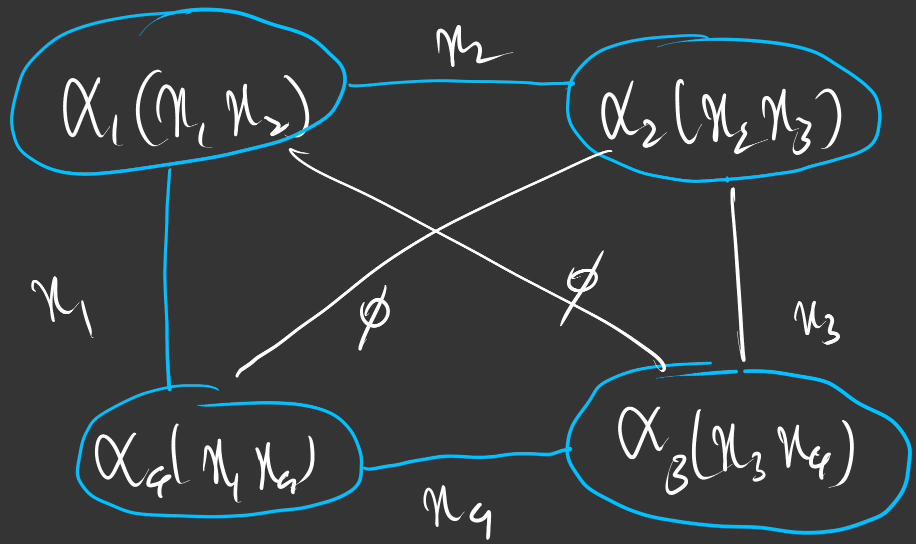
Eg: ① $f(x,y)$ $g(y,z)$

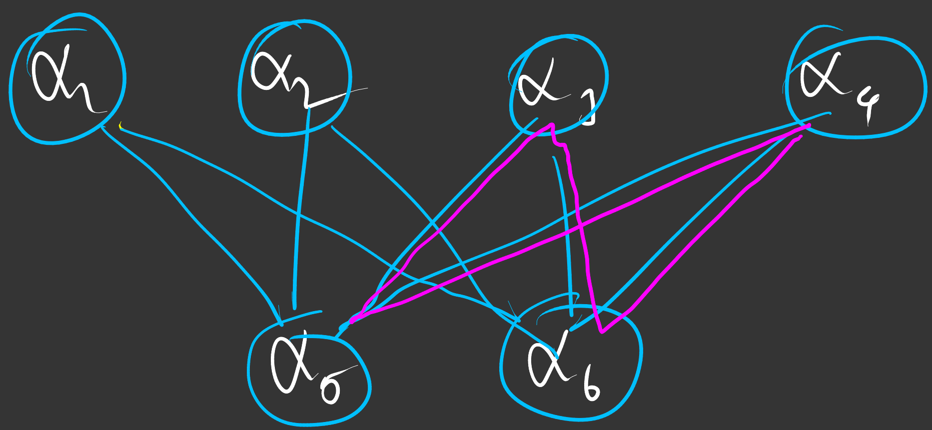


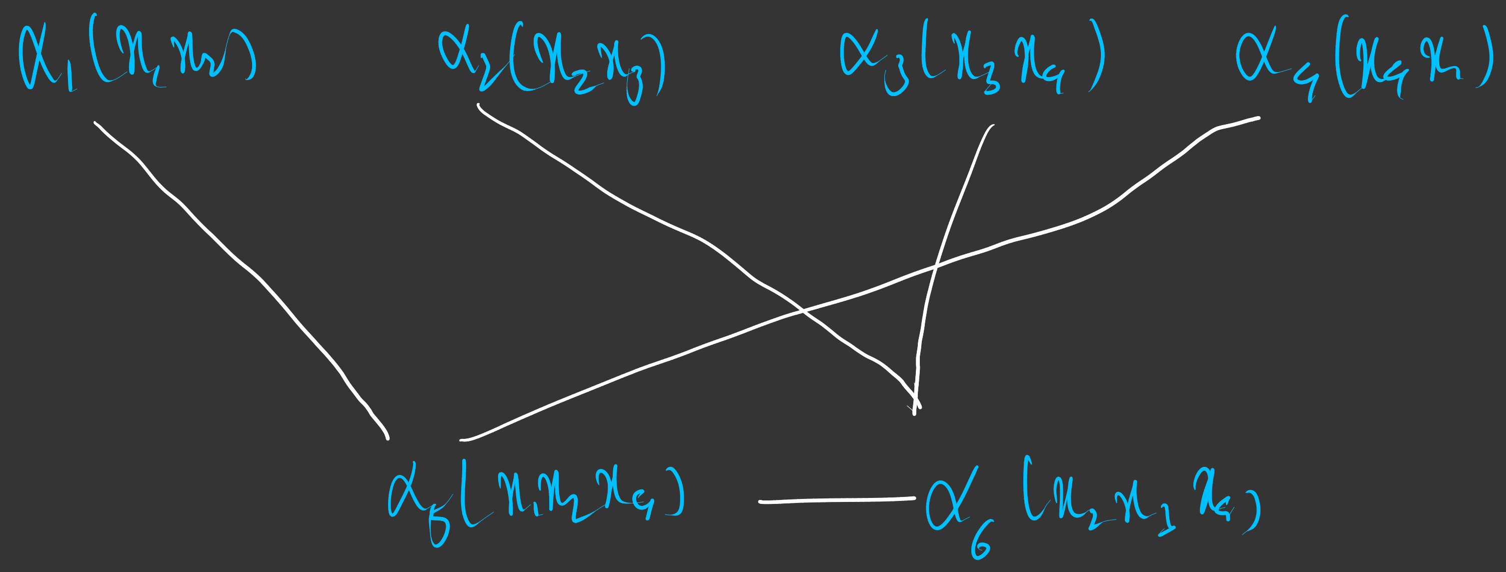
② $\beta(x_1, x_2, x_3, x_4) = \alpha_1(x_1, x_2)\ \alpha_2(x_2 x_3)\ \alpha_3(x_3 x_4)\ \alpha_4(x_1 x_4)$

$\alpha_5(x_1\ x_2 x_4)$

$\alpha_6(x_2\ x_3\ x_4)$






$\alpha_1(x_1 x_2)$
$\alpha_2(x_2 x_3)$
$\alpha_3(x_3 x_4)$
$\alpha_4(x_4 x_1)$
$\alpha_5(x_1 x_2 x_4)$
$\alpha_6(x_2 x_3 x_4)$

Constructing Junction Tree

$$\alpha_1(x_{S_1})\,\alpha_2(x_{S_2})\,\cdots\,\alpha_m(x_{S_m})$$

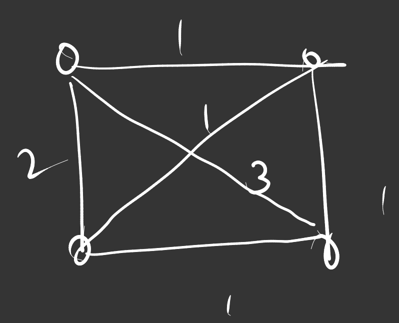
① Construct local domain (LD) graph G_{LD}:

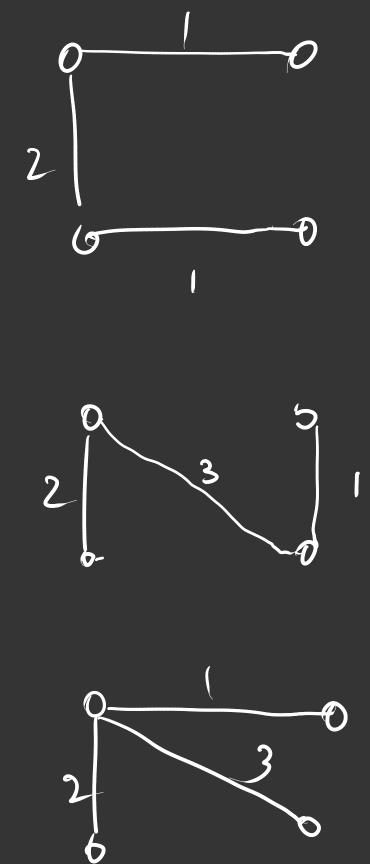
- Complete graph on m vertices $\alpha_1 \cdots \alpha_m$
- Each edge (i,j) has weight $w_{ij} = |S_i \cap S_j|$

② Find max wt spanning tree of G_{LD} : T

Max wt: spanning tree: ↳ tree that connects all vertices of G_{LD}

Wt of spanning tree = sum of wts of edges





(3) If $wt(T) = w^* = \sum_{i=1}^{m} |S_i| - n$

then T is a junction tree.

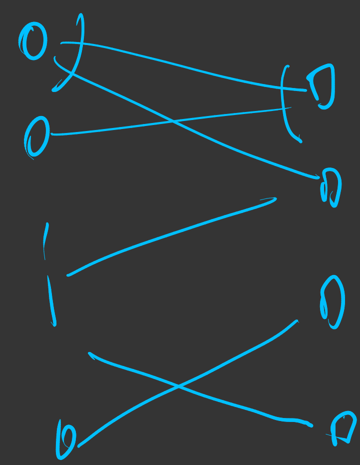
Theorem $wt(T) \leq W^*$

with equality iff T is a junction tree.

Proof: m_k : the number of local fns that x_k participates in

$f(x,y)$
$g(y,z)$

$$\sum_{k=1}^{n} m_k = \sum_{i=1}^{n} |S_i|$$



Consider T: Let w_k: no of edges in T that contain x_k

$$w_k \le m_k - 1$$ with equality iff subgraph of T corresp to x_k is connected

$$W(T) = \sum_{k=1}^{n} w_k \le \sum_{k=1}^{n} (m_k - 1)$$ → equality iff T is a junction tree

$$= \left(\sum_{k=1}^{n} m_k \right) - n$$

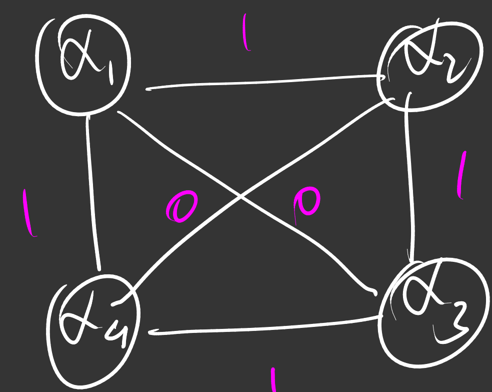
$$= \sum_{i=1}^{m} |S_i| - n = w^*.$$

Eg1 :

$f(x,y)\ g(y,z)$

Eg2 :

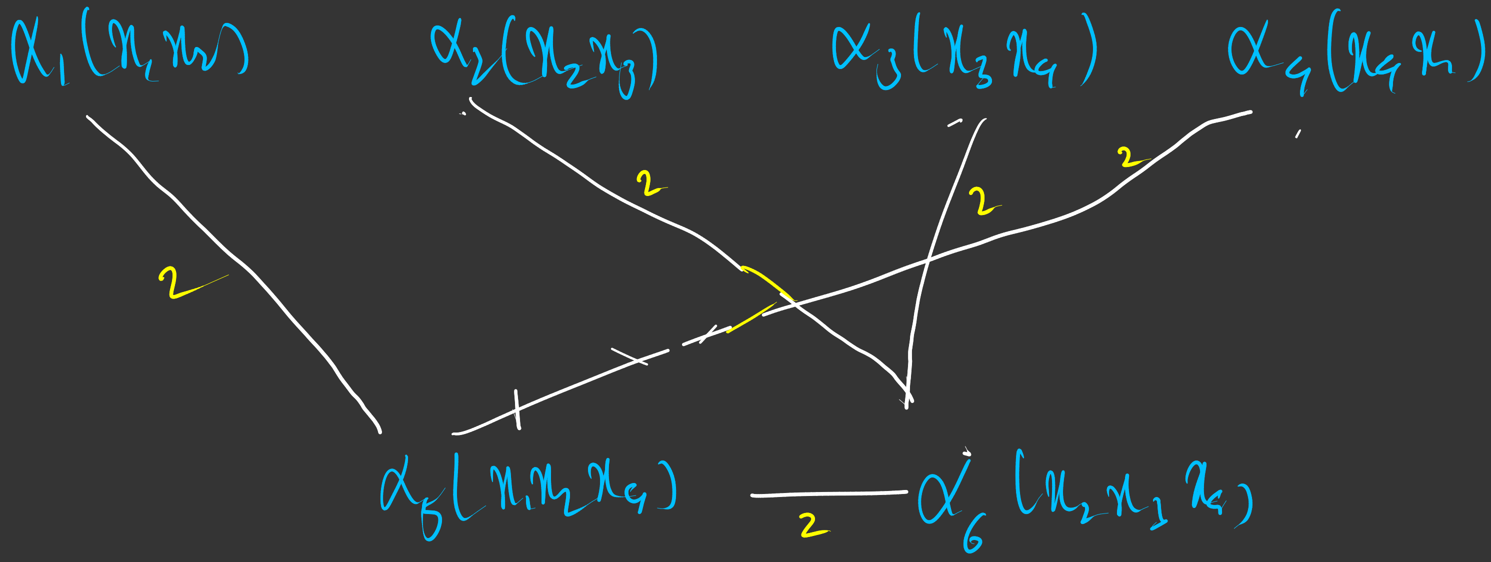
$\alpha_1(x_1 x_2)\ \alpha_2(x_2 x_3)\ \alpha_3(x_3 x_4)\ \alpha_4(x_1 x_4)$



$$w^* = \sum_{i=1}^{m} |S_i| - n$$

$$= 4$$

$w(T) < 4$ $\therefore$ No junction tree.



$$w^* = \sum_{i=1}^{m} |S_i| - n = 14 - 4 = 10$$

Algorithm to solve MPF problem:

$$\begin{tikzcd} \overset{1}{f(x,y)} & \overset{2}{g(y,z)} \arrow[l, "\mu_{2\to 1}(y)"'] \end{tikzcd}$$

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

$$O(|y||z|)$$

$$\mu_{2\to 1}(y) = \sum_{z} g(y,z)$$

$$\hat{\beta} = \sum_{x,y} f(x,y)\, \mu_{2\to 1}(y) = \sum_{x,y} f(x,y) \Big(\sum_{z} g(y,z) \Big)$$

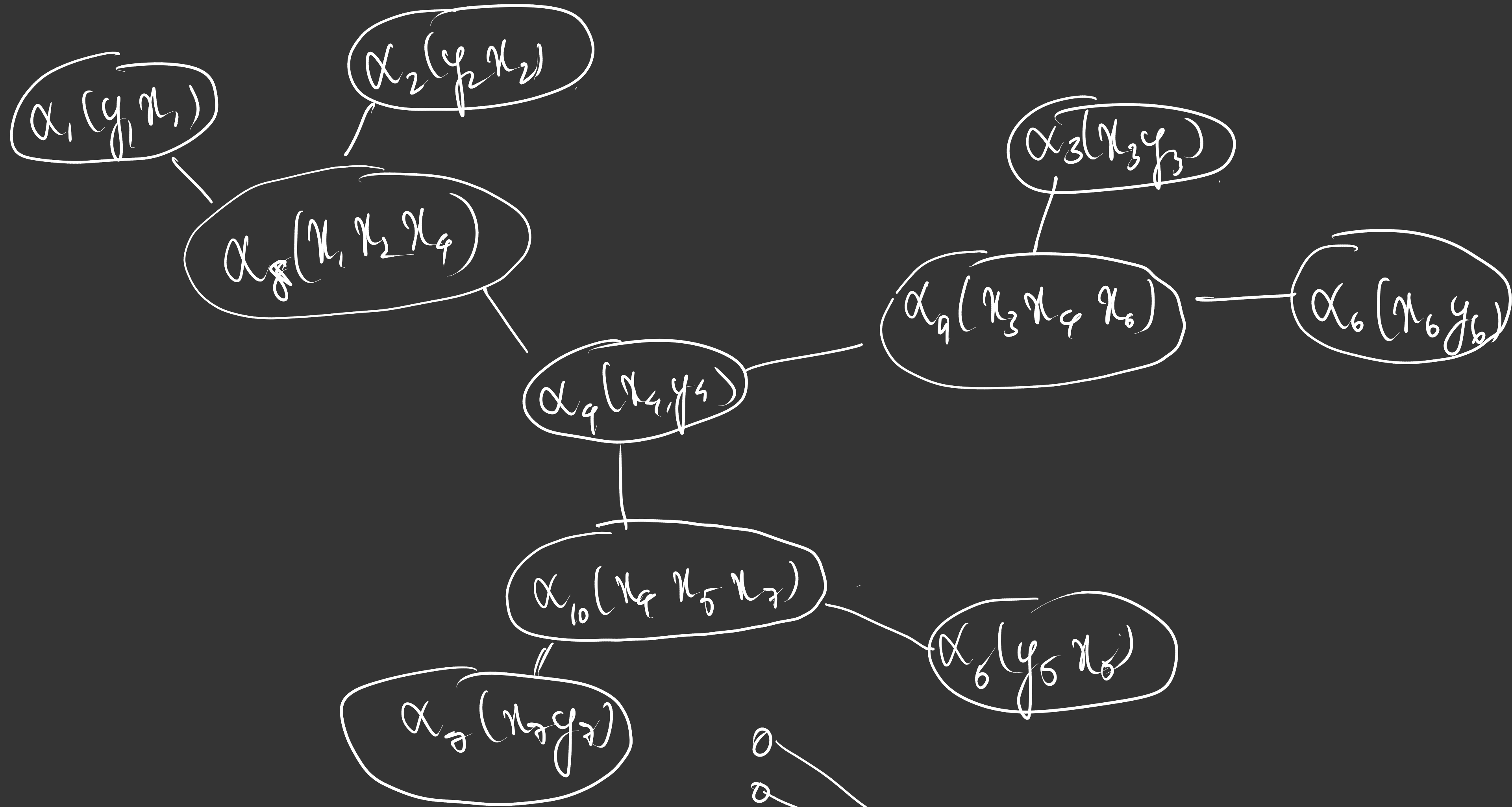
$$O(|y||z|) + O(|x||y|)$$

Eg Decoding linear codes

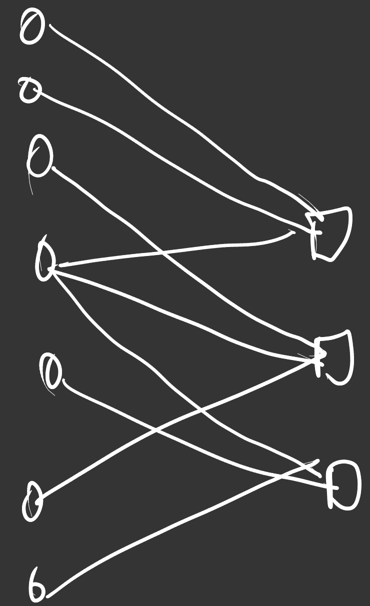
$$H = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

$$p(x_1 \ldots x_n, y_1 \ldots y_n) = \prod_{i=1}^{n} p(y_i | x_i) \prod_{j=1}^{n-k} \mathbb{1}_{\{\sum_{l \in C_j} x_l = 0\}}$$

$$= \prod_{i=1}^{7} p(y_i | x_i) \; \mathbb{1}_{\{x_1 + x_2 + x_4 = 0\}} \; \mathbb{1}_{\{x_3 + x_4 + x_6 = 0\}} \; \mathbb{1}_{\{x_4 + x_5 + x_7 = 0\}}$$

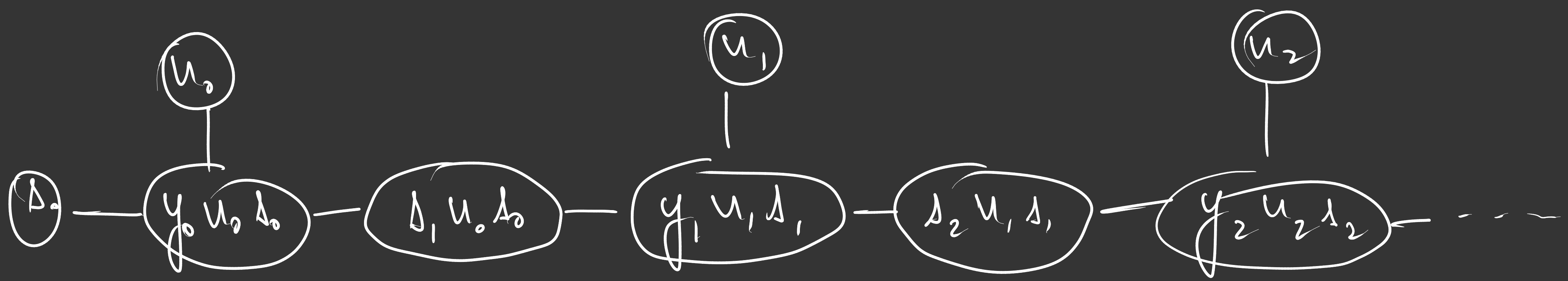


Tanner graph $\rightarrow$



Eg: Convolutional Codes/ Probabilistic State machine

$$P(\underbrace{u_0 \cdots u_n}_{\substack{\text{message/}\\ \text{var of interest/}\\ \text{i/p}}}, \underbrace{s_0 \cdots s_n}_{\text{states}}, \underbrace{y_0 \cdots y_n}_{\substack{\text{o/p or}\\ \text{observables}}}) = p(u_0)\, p(s_0)\, p(y_0 | u_0 s_0) \prod_{i=1}^{n} \Big(p(s_i | u_{i-1} s_{i-1})\, p(u_i) \times p(y_i | s_i u_i) \Big)$$

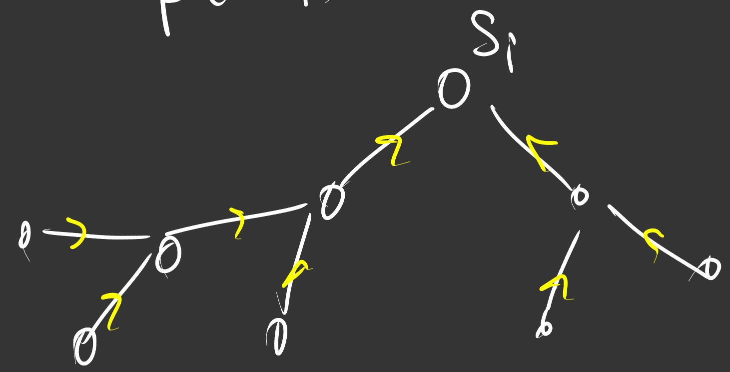


MPF for junction trees :

$\beta(x_{S_i})$

$-$ Root : $\alpha_i(x_{S_i})$

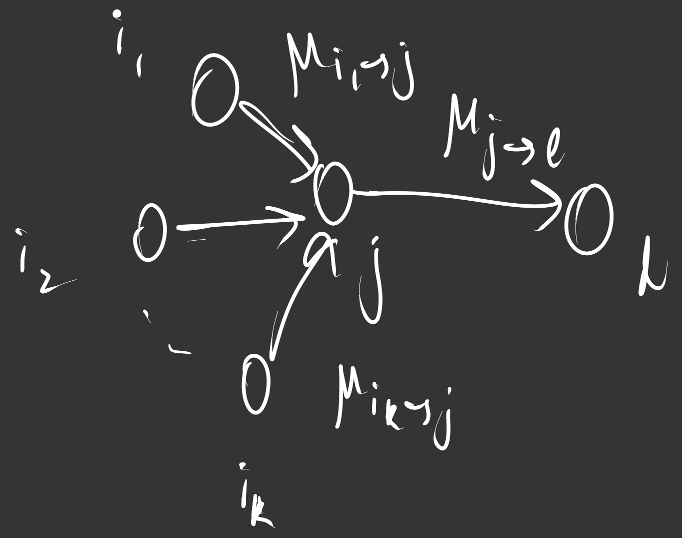
$\beta(x_{S_i})$ $\qquad$ x_{S_i}



Directed graph with flow towards the root

- Message from node i to the parent node
- Message computed only after node gets message from all children

$M_{i \to j}(x_{S_i \cap S_j})$: Msg from node i to node j



$$\mu_{j \to l}(x_{S_j \cap S_l}) = \sum_{x_{S_i \setminus S_j}} \alpha_j(x_{S_j}) \prod_{m=1}^{k} \mu_{i_m \to j}(x_{S_{i_m} \cap S_j})$$

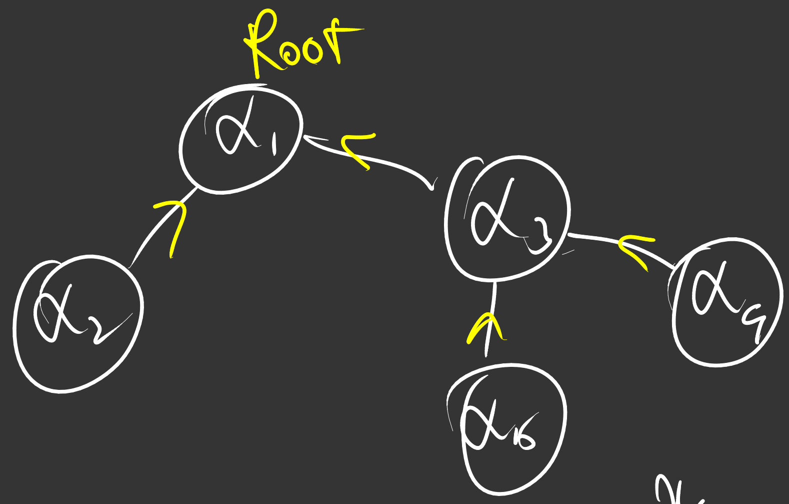
At root:

$$\beta(x_{S_i}) = \alpha_i(x_{S_i}) \prod_{j \in N_i} \mu_{j \to i}(x_{S_j \cap S_i})$$

$$\beta = \sum_{x_{S_i}} \alpha_i(x_{S_i}) \prod_{j \in N_i} \mu_{j \to i}(x_{S_j \cap S_i})$$

$\alpha_1(x_1, x_2)$ $\alpha_2(x_1, x_3)$ $\alpha_3(x_1, x_4\, x_5)$ $\alpha_4(x_4\, x_5$

$\alpha_5(x_4)$



x_j

$$M_{5\to 3}(x_4) = \alpha_5(x_4)$$

$$M_{4\to 3}(x_4, x_5) = \alpha_4(x_4, x_5)$$

$$\mu_{3\to 1}(x_1) = \sum_{x_4 x_5} \alpha_3(x_1 x_4 x_5)\ \alpha_4(x_4 x_5)\ \alpha_5(x_4)$$

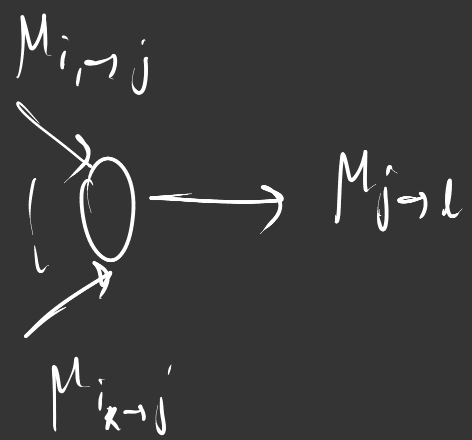
$$\mu_{2\to 1}(x_1) = \sum_{x_3} \alpha_2(x_1 x_3)$$

$$\sum_{x_1, x_2} \alpha_1(x_1 x_2) \left(\sum_{x_3} \alpha_2(x_1 x_3) \right) \left(\sum_{x_4 x_5} \alpha_3(x_1 x_4 x_5)\ \alpha_4(x_4 x_5)\ \alpha_5(x_4) \right)$$

Complexity

Brute force: $O(|A_1||A_2| \cdots |A_n|)$

New algo (GDL):



$$\mu_{j \to l}(x_{S_j \cap S_l}) = \sum_{x_{S_i \setminus S_j}} \alpha_j(x_{S_j}) \prod_{m=1}^{k} \mu_{i_m \to j}(x_{S_{i_m} \cap S_j})$$

At each node : $\underbrace{k\,|\mathcal{A}_{S_j}|}_{\text{Multiplication}} + \underbrace{|\mathcal{A}_{S_j}| - 1}_{\text{Additions}}$

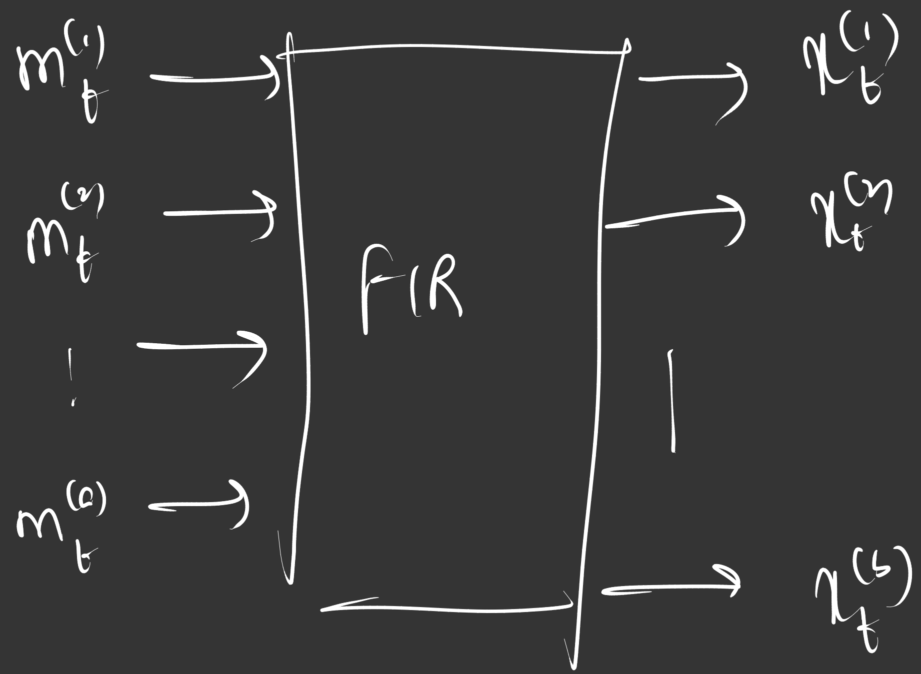
$$\leq (k+1)\,|\mathcal{A}_{S_j}|$$

$$= \deg(\alpha_j)\,|\mathcal{A}_{S_j}|$$

Total complexity : $\sum_{j=1}^{m} \deg(\alpha_j)\,|\mathcal{A}_{S_j}|$

CONVOLUTIONAL CODES

Feed forward Convolutional code: Encoder is an FIR filter



Rate $:= \frac{a}{b}$

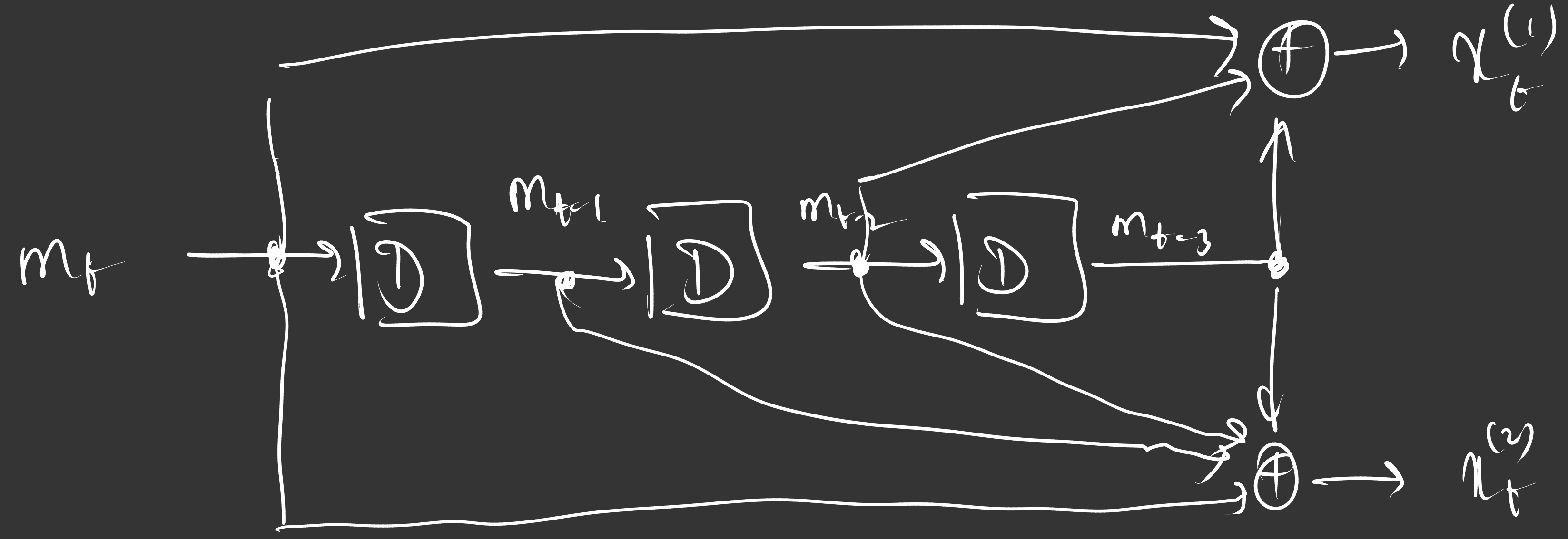
a/b

$$x_t^{(1)} = g_{10}^{(1)} m_t^{(1)} + g_{11}^{(1)} m_{t-1}^{(1)} \cdots + g_{1\nu_1}^{(1)} m_{t-\nu_1}$$

$$+ g_{20}^{(1)} m_t^{(2)} + g_{21}^{(1)} m_{t-1}^{(2)} + \cdots$$

$$x_t^{(i)} = \sum_{j=1}^{a} \sum_{l=1}^{\nu_j} g_{jl}^{(i)} \, m_{t-l}^{(j)}$$

$$\nu = \nu_1 + \nu_2 + \cdots + \nu_a$$



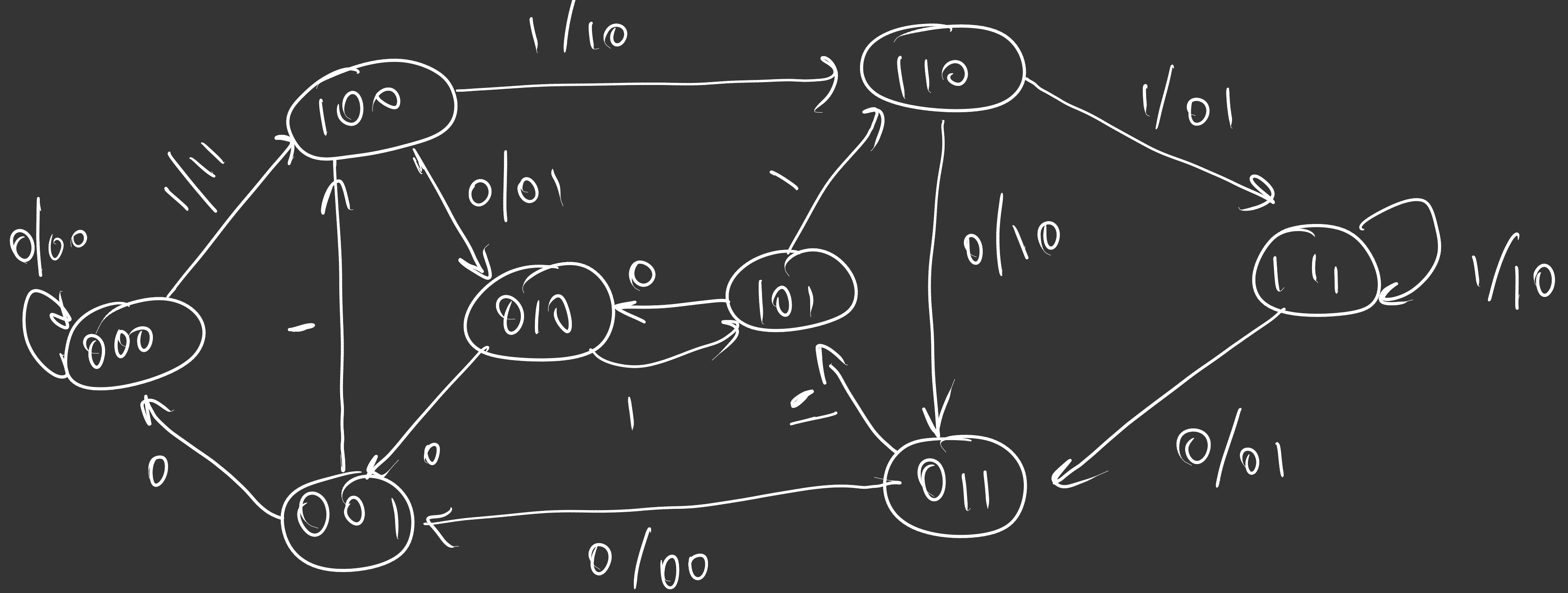
Rate 1/2

Overall constraint length : $\nu = 3$

* $[b, a, \nu]$ convolutional encoder

State diagram

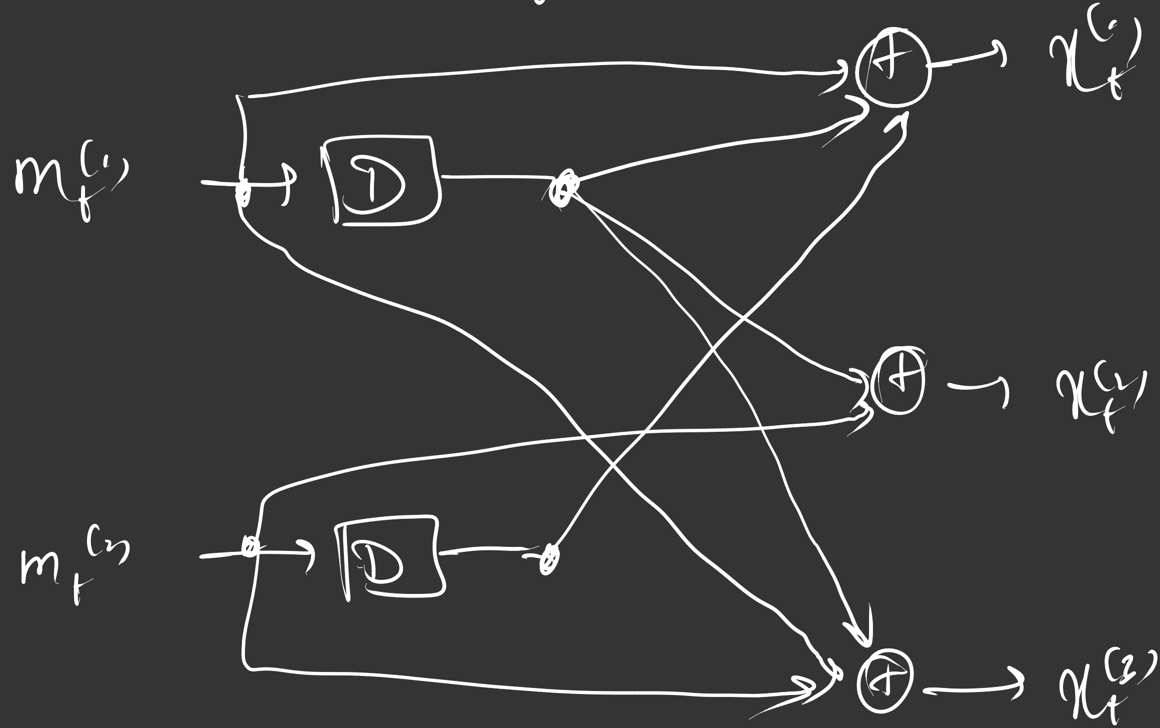
States : O/ps of delay elements



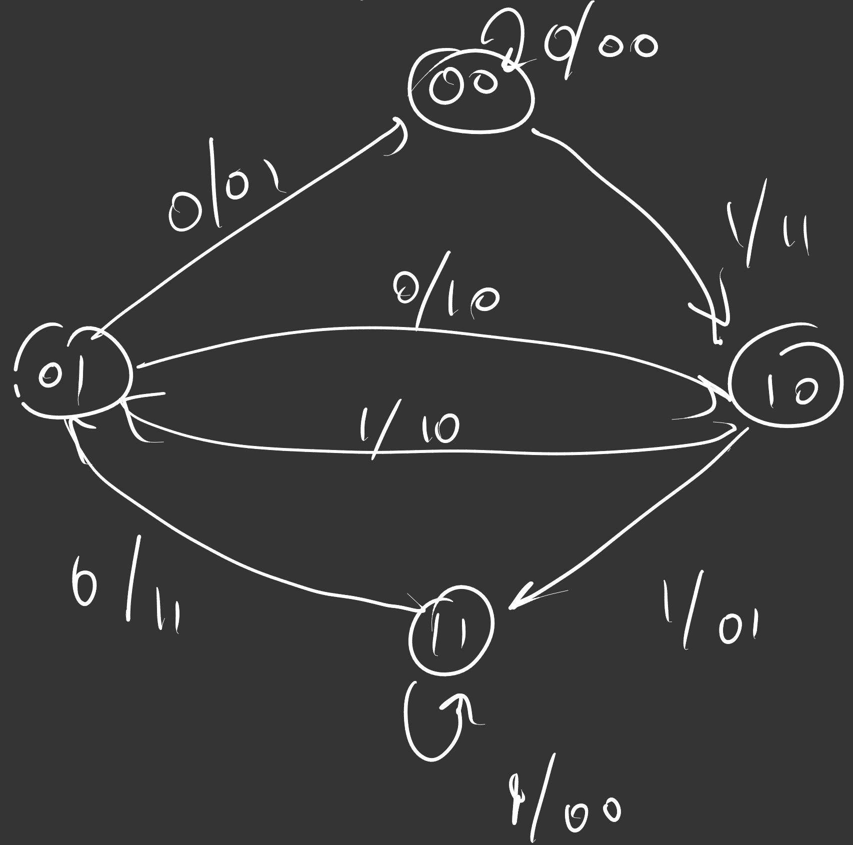
Directed graph : - Each node is a state
- Edges labelled by i/p / o/p

- Edge from state i to state j with edge labeled k if i/p k, current state i, then next state is j

HW1: Construct state diagram.



HW 2: Find ENC & generating seq



This is a "BAD" convolutional code

Viterbi algorithm: example