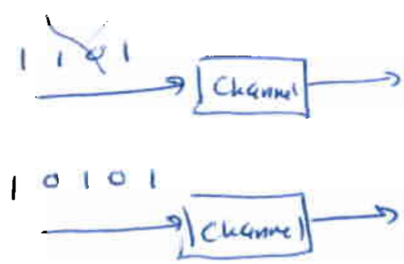
Constrained Channels:

(*) Some channels have constraints: not all sequences are permissible into the channel. (In fact, virtually all channels have some type of constraint).

- $X \in \mathcal{C}$ where $\mathcal{C}$ is the constraint set.

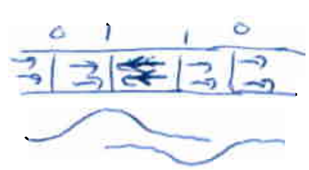
Examples:

1) BSC $\quad X \in \{0, 1\}$
2) Recording channel (saturation constraint) $X \in \{+1, -1\}$
3) Sequence constrained channels (runlength limited channel).



Recording channels are great examples of constrained channels.

$\Rightarrow$ one cannot write two 1's adjacent to one another. Why? 1's denote change in magnetic flux



adjacent 1's result in severe intersymbol interference.

⇒ Same is true in optical recording

∴ → One requires that at least d 0's appear between consecutive 1's.

→ One requires that at most k 0's appear between consecutive 1's.

Example:

Note: - Normally we think of noise as being the limiting factor in information transmission.

- Channel constraints severely restrict the potential capacity of system. Even if channel is noiseless, constraints impose a capacity on the channel.

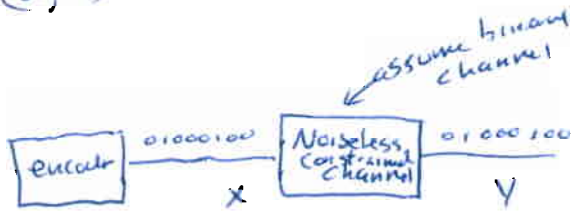
Fundamental questions:

1) What effect on channel capacity is there by imposing a constraint

2) How does one design efficient encoders to satisfy the constraints?

Consider a noiseless, run length limited, i.e. (d,k) constrained channel:

In practice IBM: (1,7), (2,7) constraints are common.



<u>Note</u>: even though channel is noiseless

$C < 1$ for any (d,k) constraint.

<u>Compute capacity</u>:

$$C = \max_{p(x)} I(X;Y)$$

$$= \max_{p(x)} H(X) - \cancel{H(X|Y)}^{\,0}$$

$\to 0$ because channel is noiseless

$$= \max_{p(x)} H(X)$$

$\Rightarrow$ Therefore we need only find the entropy rate of a sequence satisfying the (d,k) constraint.

<u>Combinatorial Computation of Entropy for a (d,k) sequence</u>

<u>Defn</u>: Let $A(n)$ be set of all sequences of length n satisfying (d,k) constraint

Let $N(n) = |A(n)|$

<u>Fact</u>: If $N(n)$ is a "suitable set" then if we used them equally likely, we would have maximum entropy. ("suitable" mean concatenatable)

$A(n)$: suitable sequences of length n

$|A(n)| = N(n)$

- using this set we could encode $\log_2(N(n))$ bits.

$$\Rightarrow R \text{ (in bits/channel symbol)} = \frac{\log_2 N(n)}{n}$$

$$R < H = \lim_{n\to\infty} \frac{\log_2 N(n)}{n}$$

(eliminates suitability condition and is an entropy rate).

<u>Compute $N(n)$ for (d,k) sequence:</u>

- Count # of (d,k) sequences of length n and then take limit.
- we will use the following state diagram

<u>G</u>:

- Fact: - any walk on this state diagram gives a valid (d,k) sequence
  - all length n walks can be found from this.

Ex: (1,3) constraint

Defn: Let A^T be a matrix whose (i,j)'th entry is # paths from state i to state j in G

Ex: (1,3)

$$A^T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Fact: $N(n) = \sum_{i=0}^{k} X_i(n)$ where $X_i(n)$ is # of (d,k) sequences in G ending in state i.

Ex: (1,3)

$$N(n) = X_0(n) + X_1(n) + X_2(n) + X_3(n)$$

Fact: we can compute $X_i(n)$'s recursively.

$$\left.\begin{aligned} X_0(n) &= X_1(n-1) + X_2(n-1) + X_3(n-1) \\ X_1(n) &= X_0(n-1) \\ X_2(n) &= X_1(n-1) \\ X_3(n) &= X_2(n-1) \end{aligned}\right\} \text{for } (1,3) \text{ constraint}$$

In vector form:

$$\underline{X}(n) = \begin{bmatrix} X_0(n) \\ X_1(n) \\ X_2(n) \\ X_3(n) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} X_0(n-1) \\ X_1(n-1) \\ X_2(n-1) \\ X_3(n-1) \end{bmatrix}}_{\underline{X}(n-1)}$$

Further:

$$\underline{X}(n) = A\,x(n-1) = AA\,\underline{x}(n-2) = A^3 x(n-3) = \cdots = A^{n-1}\underline{x}(1)$$

recall A^T is $(k \times k)$ matrix $a_{ij} = \#$ paths from i to j.

(A) $\Rightarrow$ $\underline{X}(n) = A^{n-1}\underline{X}(1)$

Fact: Can diagonalize A using a similarity transformation consisting of the orthonormal eigenvectors of A

$$A = U^{-1}\Lambda U \qquad \Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_k \end{bmatrix}$$

$$U = \begin{bmatrix} - u_1 - \\ - u_2 - \\ \vdots \\ - u_k - \end{bmatrix}$$ u_i are orthonormal eigenvector for λ_i

Using (A)

$$\Rightarrow \underline{X}(n) = A^{n-1}\underline{X}(1)$$

$$= (U^{-1}\Lambda U)^{n-1}\underline{X}(1)$$

$$= U^{-1}\Lambda^{n-1}\underbrace{U\,\underline{X}(1)}_{\underline{V}}$$

$$= U^{-1}\begin{bmatrix} \lambda_1^{n-1} & & & \\ & \lambda_2^{n-1} & & \\ & & \ddots & \\ & & & \lambda_k^{n-1} \end{bmatrix}\begin{bmatrix} \underline{V} \end{bmatrix} \qquad \underline{V} = \begin{bmatrix} V_1 \\ \vdots \\ V_k \end{bmatrix}$$

Perron Frobenius Theorem, largest mag. eigenvalue is real, $\lambda_{max} > 1$

$$\underline{X}(n) = U^{-1}\begin{bmatrix} \lambda_1^{n-1} v_1 \\ \lambda_2^{n-1} v_2 \\ \vdots \\ \lambda_k^{n-1} v_k \end{bmatrix} = \begin{bmatrix} - & \omega_1 & - \\ - & \omega_2 & - \\ & \vdots & \\ - & \omega_k & - \end{bmatrix}\begin{bmatrix} \lambda_1^{n-1} v_1 \\ \lambda_2^{n-1} v_2 \\ \vdots \\ \lambda_k^{n-1} v_k \end{bmatrix}$$

$$= \begin{bmatrix} \lambda_1^{n-1}\omega_{1,1} v_1 + \lambda_2^{n-1}\omega_{1,2} v_2 + \cdots + \lambda_k^{n-1}\omega_{1,k} v_k \\ \vdots \\ \lambda_1^{n-1}\omega_{k,1} v_1 + \lambda_2^{n-1}\omega_{k,2} v_2 + \cdots + \lambda_k^{n}\omega_{k,k} v_k \end{bmatrix}$$

$$\Rightarrow \quad \underline{X}(n) = \lambda_1^{n-1}\underline{Y}_1 + \lambda_2^{n-1}\underline{Y}_2 + \cdots + \lambda_k^{n-1}\underline{Y}_k$$

where: $\underline{Y}_j = \begin{bmatrix} \omega_{1,j} v_j \\ \vdots \\ \omega_{k,j} v_j \end{bmatrix}$

and $\quad X_i(n) = \lambda_1^{n-1} Y_{1,i} + \lambda_2^{n-1} Y_{2,i} + \cdots + \lambda_k^{n-1} Y_{k,i}$

of (d,k) sequences of ~~[illegible]~~ length n ending in state i

eigenvalues of A

<u>Defn</u>: Let $H_i(X)$ be entropy given we end in i

$$\therefore H(X) = E_i[\,H_i(X)\,]$$

$$H_i(X) = \lim_{n\to\infty} \frac{\log_2 X_i(n)}{n}$$

$$= \lim_{n\to\infty} \frac{\log_2\left(\lambda_1^{n-1} Y_{1,i} + \lambda_2^{n-1} Y_{2,i} + \cdots + \lambda_k^{n-1} Y_{k,i}\right)}{n}$$

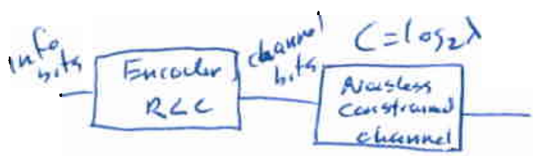
$$= \lim_{n\to\infty} \frac{\log_2(\lambda^{n-1} y)}{n}$$

where λ is largest (real) eigenvalue of A and y is corresponding y-term.

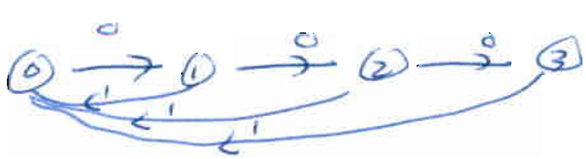
$$= \lim_{n\to\infty}\left[\frac{n-1}{n}\log_2\lambda + \overset{\to 0}{\frac{\log_2 y}{n}}\right]$$

$$H(X) = \log_2 \lambda$$

Recall $C = H(X)$, so $\boxed{C = \log_2 \lambda}$ where λ is largest (real) eigenvalue of A.



Example: (1,3) constraint

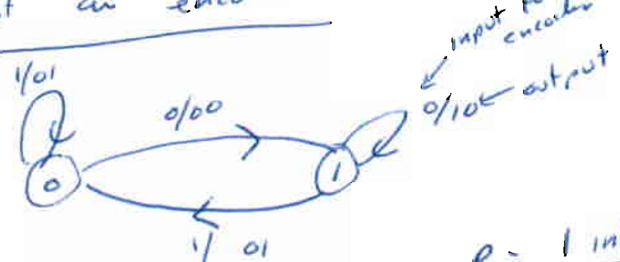


$$A^T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$\lambda = 1.456$

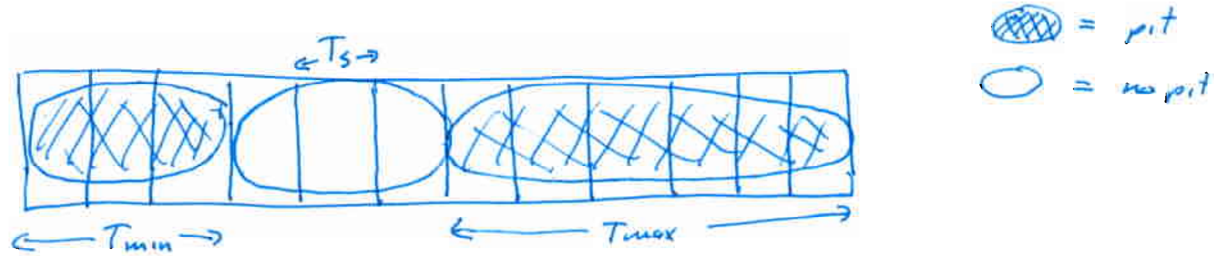
$R = \log_2 \lambda = .542$ info bits/channel bit

Example of an encoder:



$$R = \frac{1 \text{ info bit}}{2 \text{ channel bits}} = .5 < .542$$

Optical Recording: conventional saturation recording



⇒ Data is stored by putting pits/ no bits on disk.

⇒ There is some minimum possible mark T_{min} (optical resolution of system does not allow smaller marks).

⇒ This is some maximum possible mark T_{max} (system derives timing from edges in written marks).

⇒ 2 early options for converting information bits to marks.

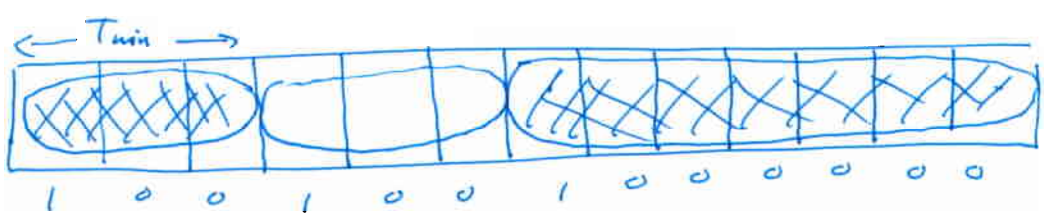
1) Each mark/ no mark represents ≈ 1 bit. (no!!)
2) Slice time smaller than T_{min}.

Let $T_{min} = (d+1)\,T_s$

$T_{max} = (k+1)\,T_s$

Assume the following writing ~~s~~ scheme: (allowable marks on disk).

a) let 1's change state of written marks.
b) let 0's modulate width of mark.

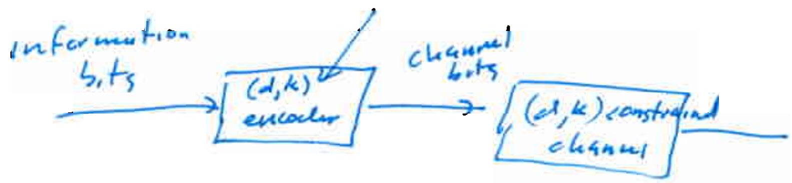


This is a constrained channel: at least d 0's
at most k 0's after
each '1'.

What is density of this recording scheme?

measure density in information bits/T_{min}

has rate R $\frac{\text{info bits}}{\text{channel bits}}$



$$D = \frac{\#\ \text{information bits}}{T_{min}} = \frac{(d+1)\ \text{channel bits}}{T_{min}} \cdot \frac{R\ \text{info bits}}{\text{channel bits}}$$

$$= \frac{(d+1)R}{T_{min}}$$

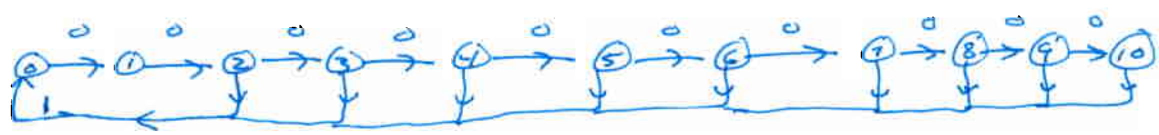
Best density: $D_c = \frac{(d+1)C}{T_{min}} \geq \frac{(d+1)R}{T_{min}}$

where C is capacity of the (d,k) constrained channel.

Example: CD-ROM / Audio CD

$d=2,\ k=10$

G:



What is largest density achievable by a code?

$$D_c = \frac{(1+d)C}{T_{min}} = \frac{3 \cdot C}{T_{min}}$$

Find C:

$C = \log_2 \lambda$ where λ is largest real eigen value of A^T for G.

$$A^T = \begin{array}{c|ccccccccccc} & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 1 & & & & 1 & & & & & & \\ 4 & 1 & & & & & 1 & & & 0 & & \\ 5 & 1 & & & & & & 1 & & & & \\ 6 & 1 & & & & & & & 1 & & & \\ 7 & 1 & & 0 & & & & & & 1 & & \\ 8 & 1 & & & & & & & & & 1 & \\ 9 & 1 & & & & & & & & & & 1 \\ 10 & 1 & & & & & & & & & & 0 \end{array}$$

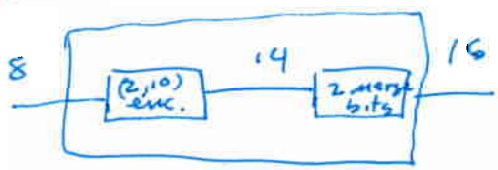
$\lambda = 1.4558$

$C = \log_2 \lambda = .5418$

$\Rightarrow D = \dfrac{(3)(.5418)}{T_{min}} = 1.625 \dfrac{\text{bits}}{T_{min}}$

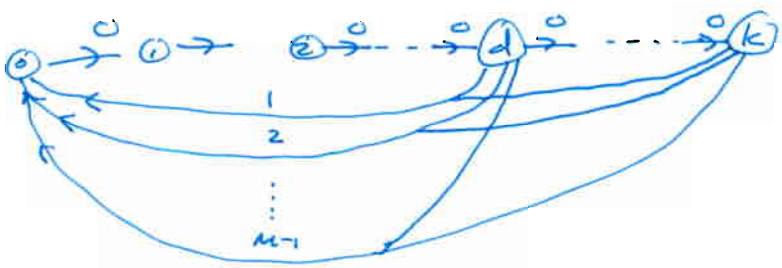
In practice: Code developed by Kees Immink @ Philips

Eight-to-Fourteen (EFM).



$R = \dfrac{8}{16} = .5 < .5418$ (92% efficient).

Sum: Non binary recording (m-ary (d,k) codes)



Lots of fun!!