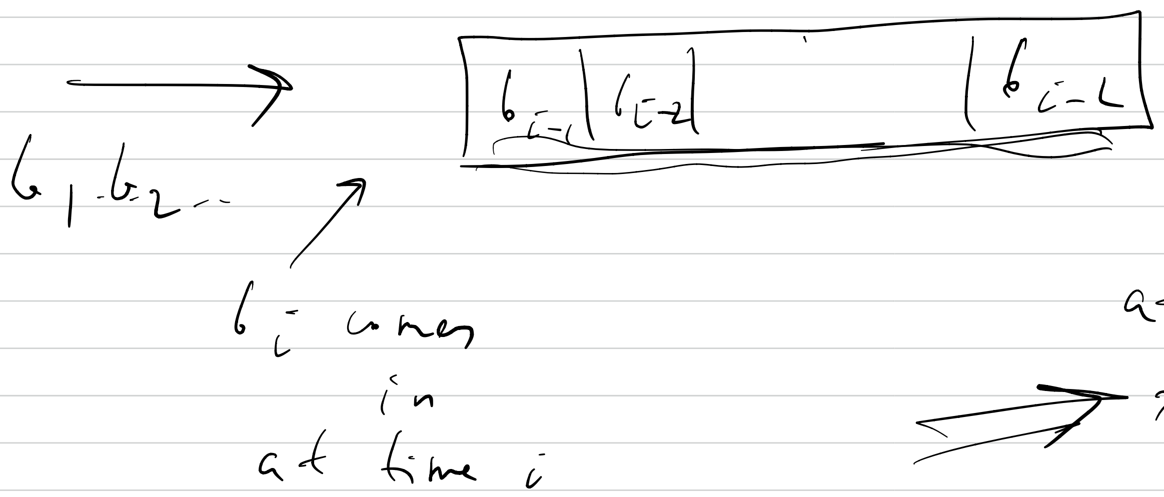
Principles of Digital Communication

May 7, 2021

Reading for next week: 6.4 (& 6.5)

↑ analysis of the Convolutional Codes.

A method of mapping k bits to n symbols ($Ex: n = 2k$) was proposed:



at time i

$\Rightarrow x'_i = \underline{funct(b_i, \ldots, b_{i-L})}$

$x''_i = \underline{funct''(\qquad)}$

$b_i \in \{\pm 1\}$

"Convolutional":

$\{\pm 1\}$ and "product"

$\equiv \{0,1\}$ and $\underline{\text{mod } 2}$ sum

$*$	+	−
+	+	−
−	−	+

+	0	1
0	0	1
1	1	0

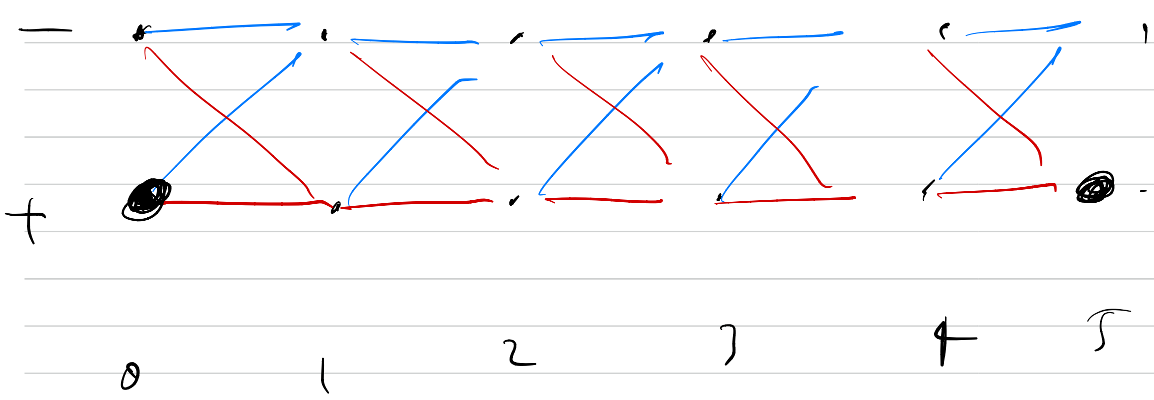
$\Rightarrow x'_i = \underline{funct(b_i \ldots b_{i-L})}$

$$\equiv \bar{x}'_i = \sum_{j \in S} \bar{b}_{i-j} \mod 2 = \sum_j h_j \bar{b}_{i-j} \mod 2$$

$$h_j = \begin{cases} 0 & \text{if } j \notin S \\ 1 & \text{if } j \in S \end{cases}$$

$= (h * b)_i$

Decoding / Encoding



—— : + is input

—— : − is "

when we send a block of k bits, we agree to place a "dummy" termination $b_{k+1} = +$.

with "L" $\ll k$ these dummy bits do not change the efficiency properties of the code

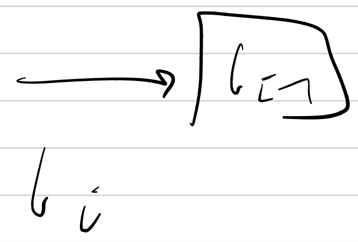
at the decoder we have received:

(with $k=4$, "$n=2k$") $(y_1 y_2)(y_3 y_4)\dots(y_9 y_{10})$

and we are searching for $(\hat{b}_1 \dots \hat{b}_4 +)$ that maximize

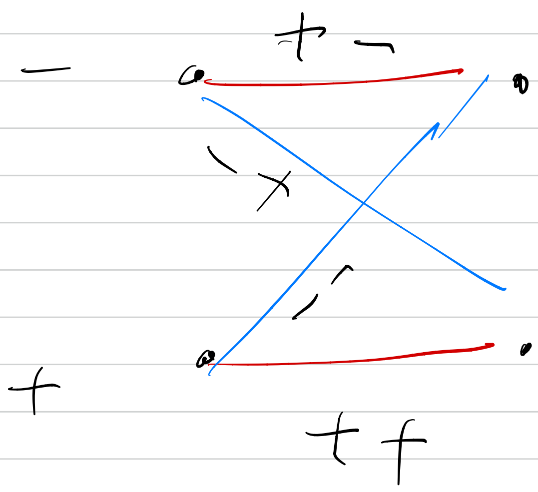
$\langle x(\hat{b}_1 \dots \hat{b}_4 +), y \rangle$

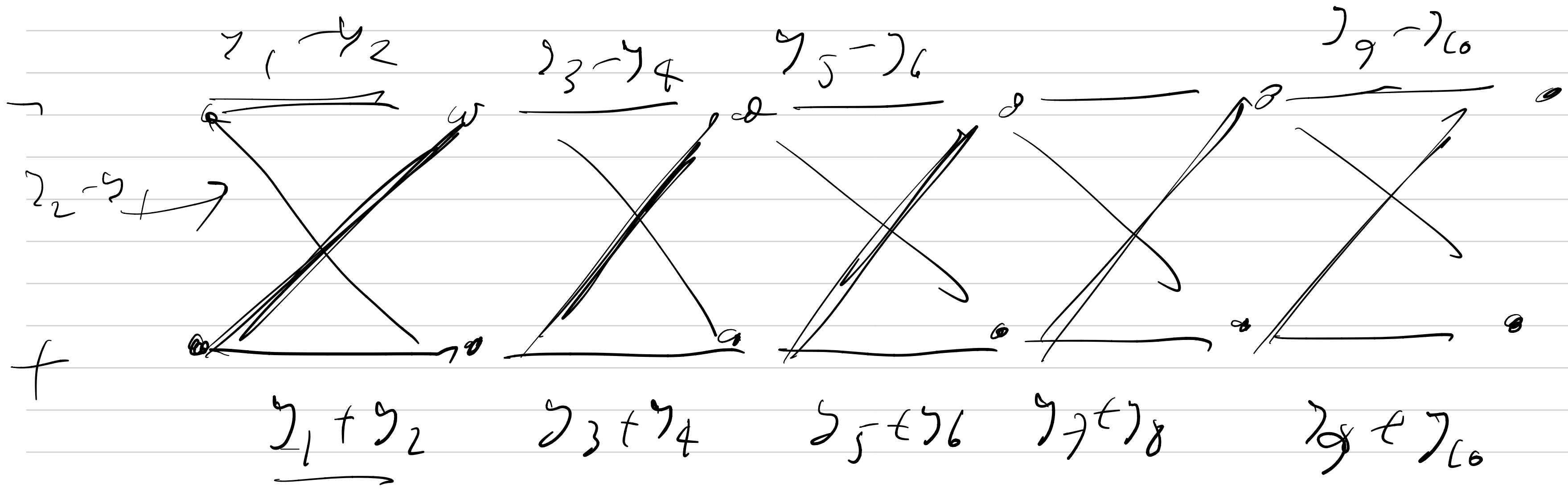
we find $\hat{b}_1 \dots \hat{b}_k$ by keeping track for each intermediate stage, the most rewardful (?) way to arrive to this stage starting from the starting node

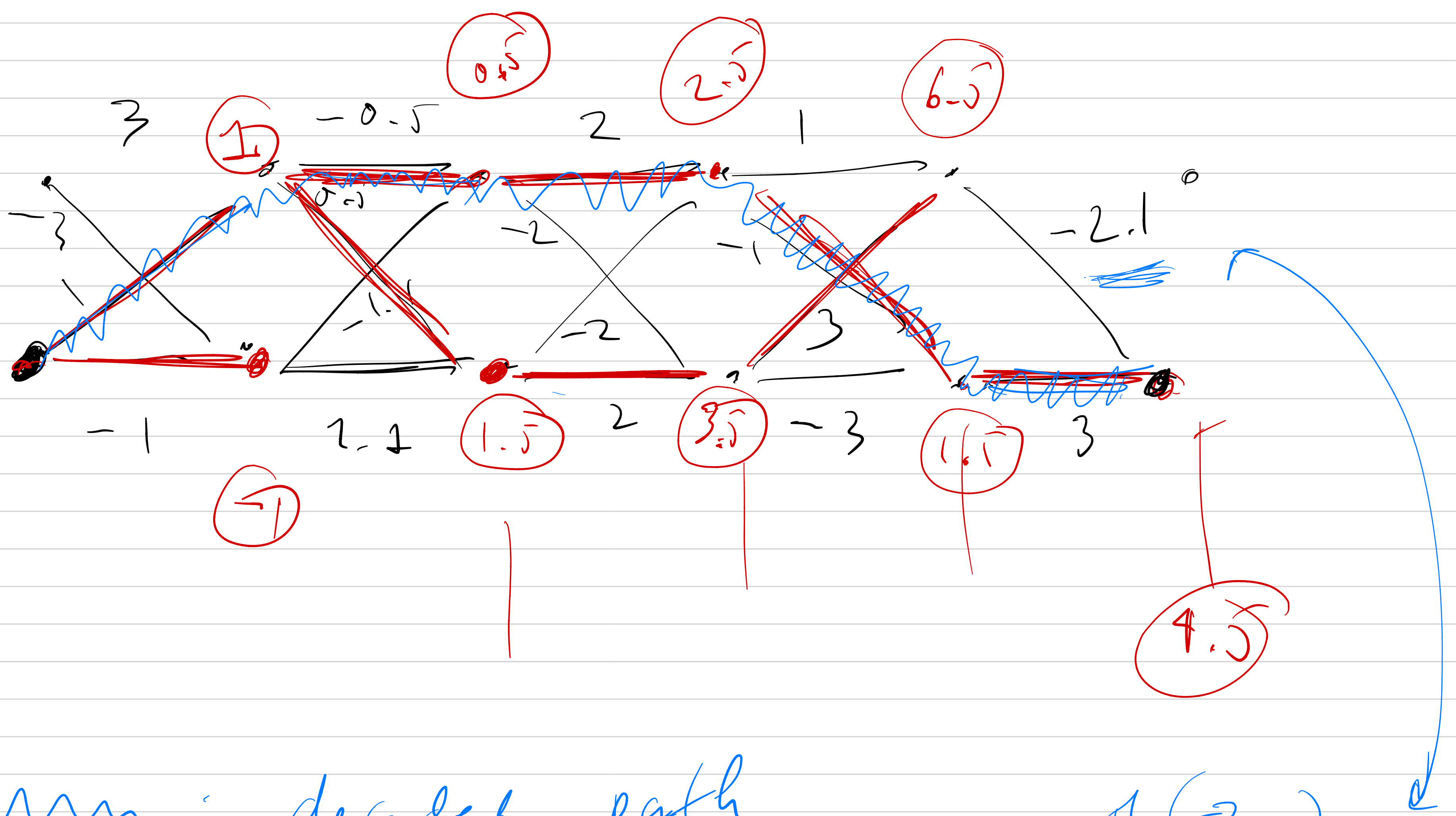


$x'_i = b_i$

$x''_i = b_i b_{i-1}$







mm : decoded path

with (−2.0)

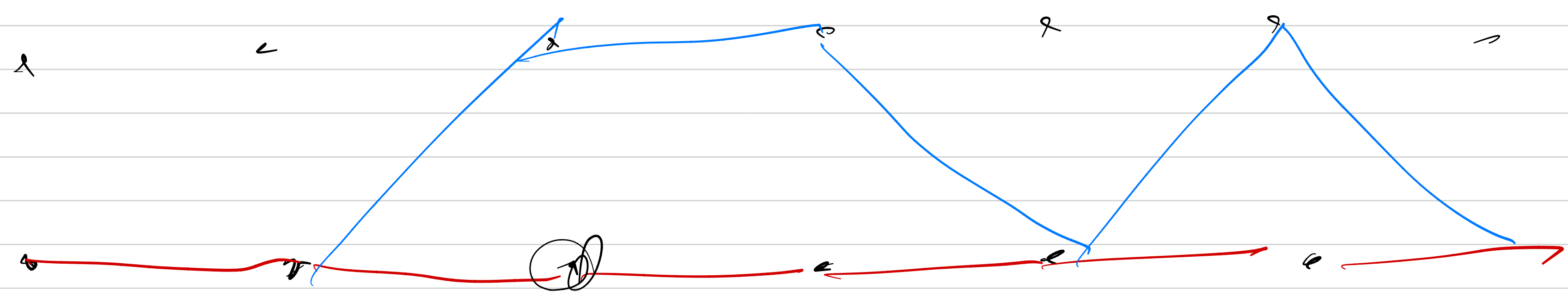
− + + − (+)

[+ − + − (+)]

Convolutional Codes: Peter Elias.

Decoding Method: Andrew Viterbi.
[Viterbi decoding] ↑ founder of Qualcomm
↑ David Forney realized that this is the MAP decoder.

Analysis of Convolutional codes:



Suppose the true bits are $t\,t\,t \ldots t$

what is the chance that the decoder follows a different path?

Ω_0 = # of incorrect decoded bits because of a departure at time 0 (0)

Ω_1 = ... time 1 (2)

So : # of bits decoded in error

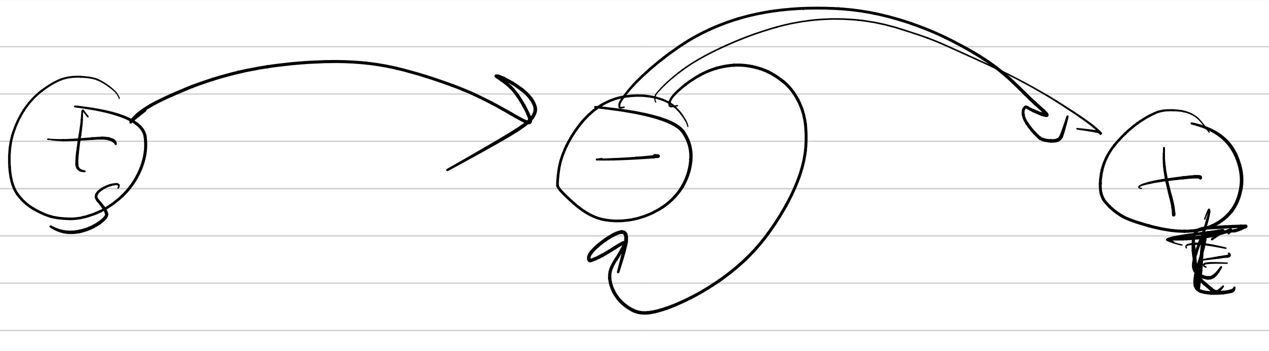
$$= \sum_{i=0}^{k} \Omega_i$$

$\Rightarrow$ the expected fraction of incorrect bits

$$= \frac{1}{k} \sum_{i=0}^{k} E[\Omega_i] \quad \text{(bit error rate)}$$

to study this we should understand the nature of these "detours" from the correct path.

In the simple example a detour is a path from $(+_S)$ to $(+_E)$



"$+_S - +_E$"

"$+_S - - - - +_E$"

"$+_S - - +_E$"

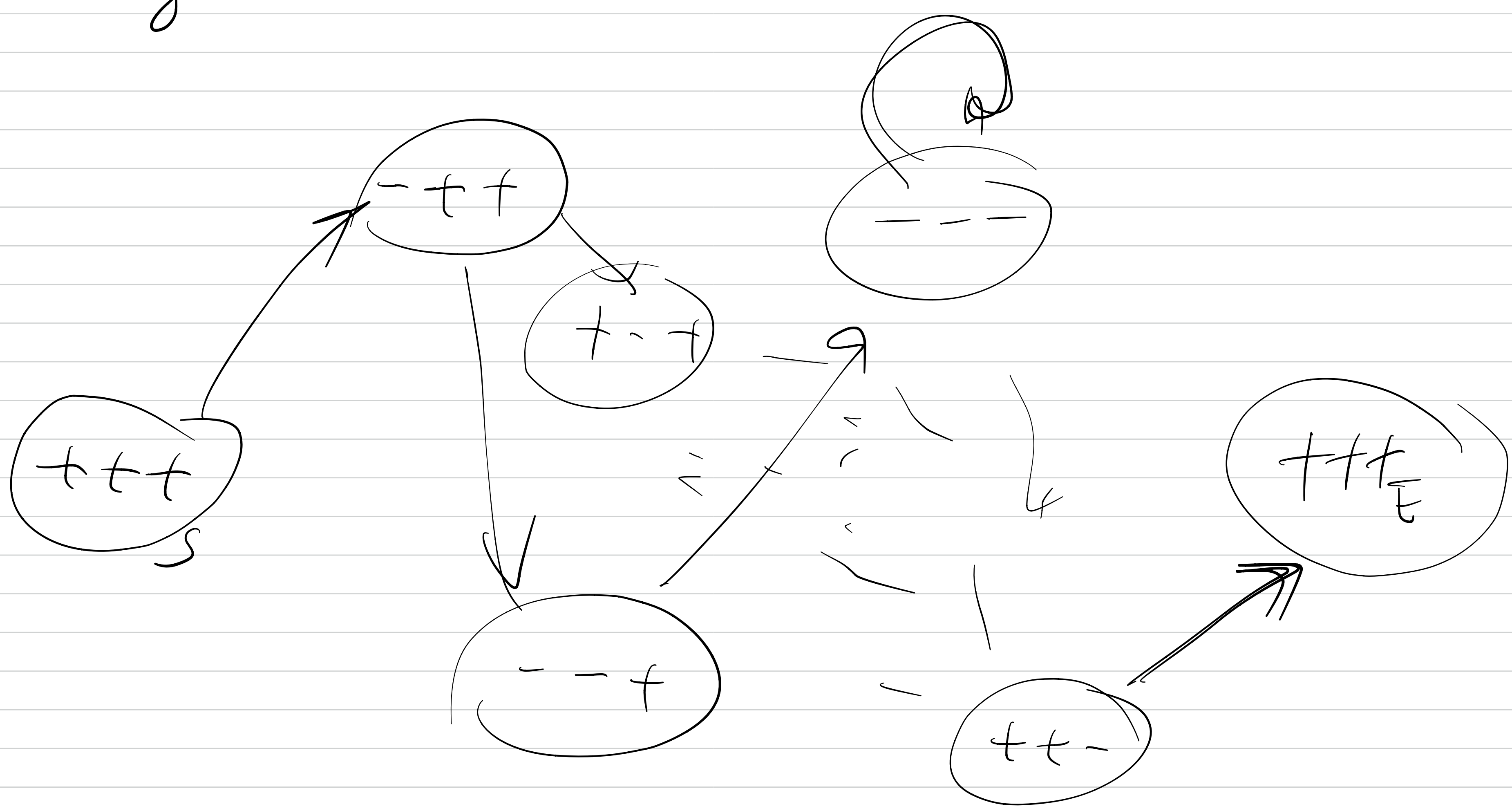
. . .

$\leftarrow$ possible detours.

We need to understand, for each detour "h":

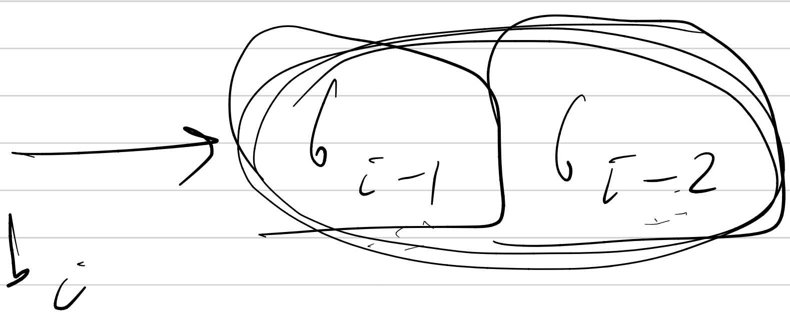
1. the prob $\pi(h)$ of taking the detour
2. the cost $i(h)$ of the detour in the # of incorrectly decoded bits caused by it.

In general:



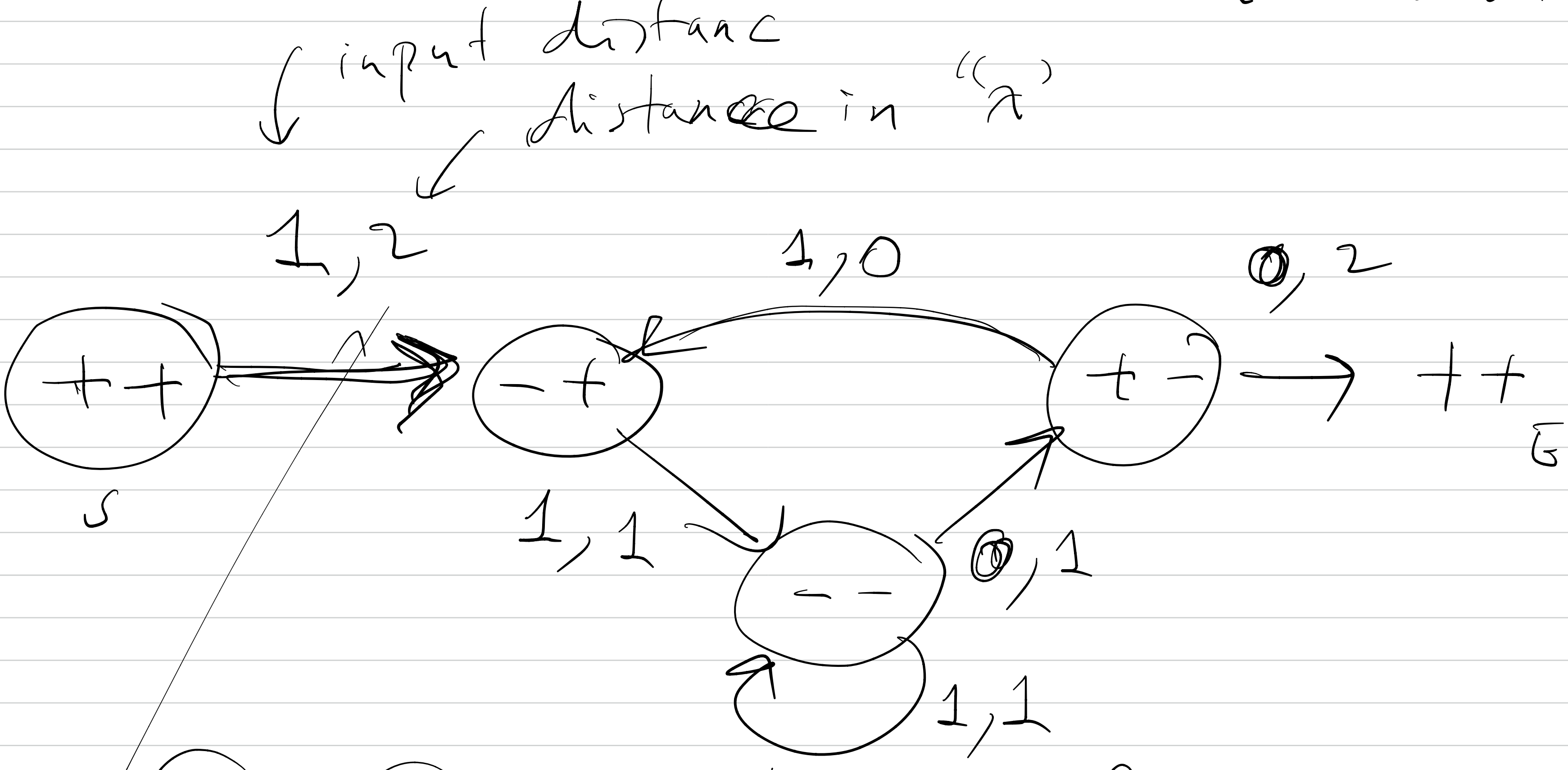
and we need to enumerate the paths from $+++_s$ to $+++_t$ in terms of (1): the distance (in the "x"'s) from the path to the "true path" and 2 : the number of hit errors.

Ex :



$x'_i = b_i b_{i-2}$

$x''_i = b_i b_{i-1} b_{i-2}$



(++) → (-+) generates -- for x', x''

2 : (++) → ++   ++ " "

(-, ++) → 1, 0

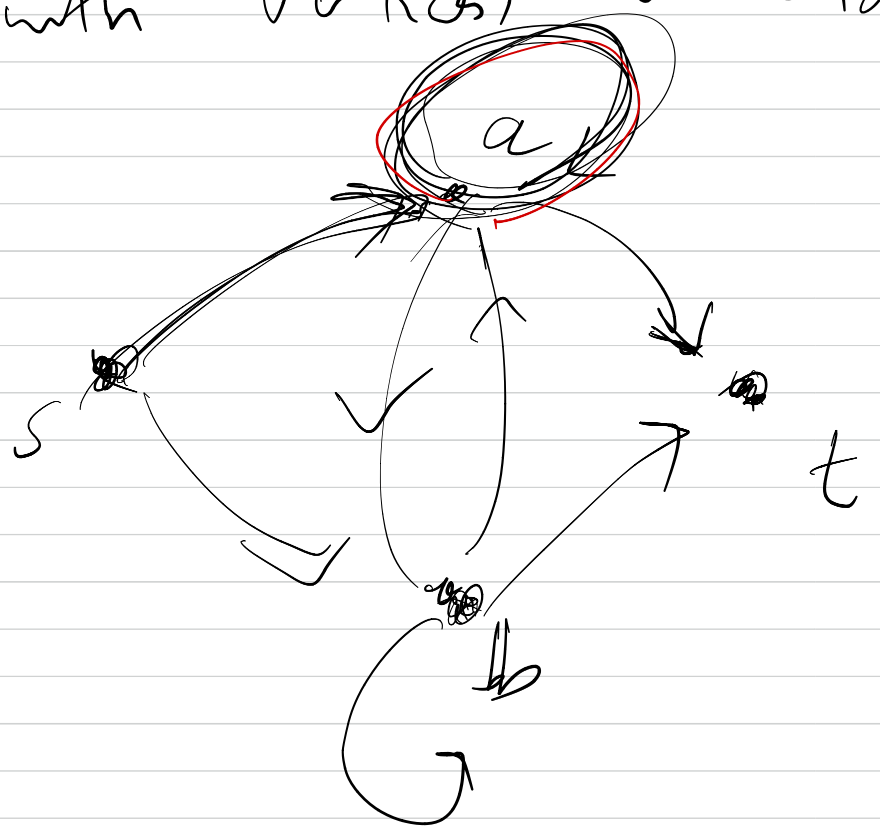
+- → -+

We need to count the paths h from

(++)$_S$ to (++)$_E$ according to "b" & "x" distances.

How to Count?

we have a graph with vertices V and edges E.



each edge is labled by (distance, time, gas stations, restaurants)

$$D^{d}\ T^{t}\ G^{g}\ R^{r}$$ (# of km, hrs, stations, rests.)

dummy variable.

Let us now form the following object

$$\sum_{\substack{h:\ \text{from } s \\ \text{to } t}} \underbrace{D^{(\text{km in } h)}\ T^{(\text{hrs in } h)}\ G^{(\text{stats in } h)}\ R^{(\text{rest in } h)}} = \boxed{A(D,T,G,R)}$$

if $A(D,T,G,R) = 5\, D^{17} T^{6} G^{1} R^{0} + 3 D^{18} T^{7} G^{2} R^{1}$
$+ \ldots$

if $A(\quad)$ was somehow computed, the coefficient $a(d,t,g,r)$ will give us the # of paths that is d km's long

take t hrs to traverse

with g gear steps

and r restaurants.

Let $A_a(D,T,G,R)$ be the sum $\sum_{h:s\to a} D^{\cdot}T^{\cdot}G^{\cdot}R^{\cdot}$

$A_b(\quad)$ " " " " " $\sum$

$A_t(\quad)$ what we want.

edge from $s\to a$ — edge($a\to a$)

$$\underline{\underline{A_a}} = DTGR + A_a\,DTGR$$

edge ($b\to a$)

$$+ A_b\,DTGR$$

edge($b\to b$) — edge $a\to b$ — edge($s\to b$)

$$\underline{\underline{A_b}} = A_b\,DTGR + A_a\,DTGR + DTGR$$

two eqns in two unknowns. Solve them

will give:

$$\begin{bmatrix} * & * \\ * & * \end{bmatrix} \begin{pmatrix} A_a \\ A_b \end{pmatrix} = \begin{bmatrix} * \\ * \end{bmatrix}$$

$$A_t = A_a \cdot \overset{\text{edge}(a \to t)}{\overset{\ell\,\ell\,\ell\,\ell}{DTGR}} + A_b \, \overset{\text{edge}(b \to t)}{\overset{\ell\,\ell\,\ell\,\ell}{DTGR}}$$

$A(D, T, G, R)$

We will use exactly this method to count the paths h with given "b" and "x" distances.