Convolutional Codes

Differences Between Block Codes And Convolutional Codes

1. LINEAR BLOCK CODES ARE DESCRIBED BY

   n, k AND g(X) (OR G)

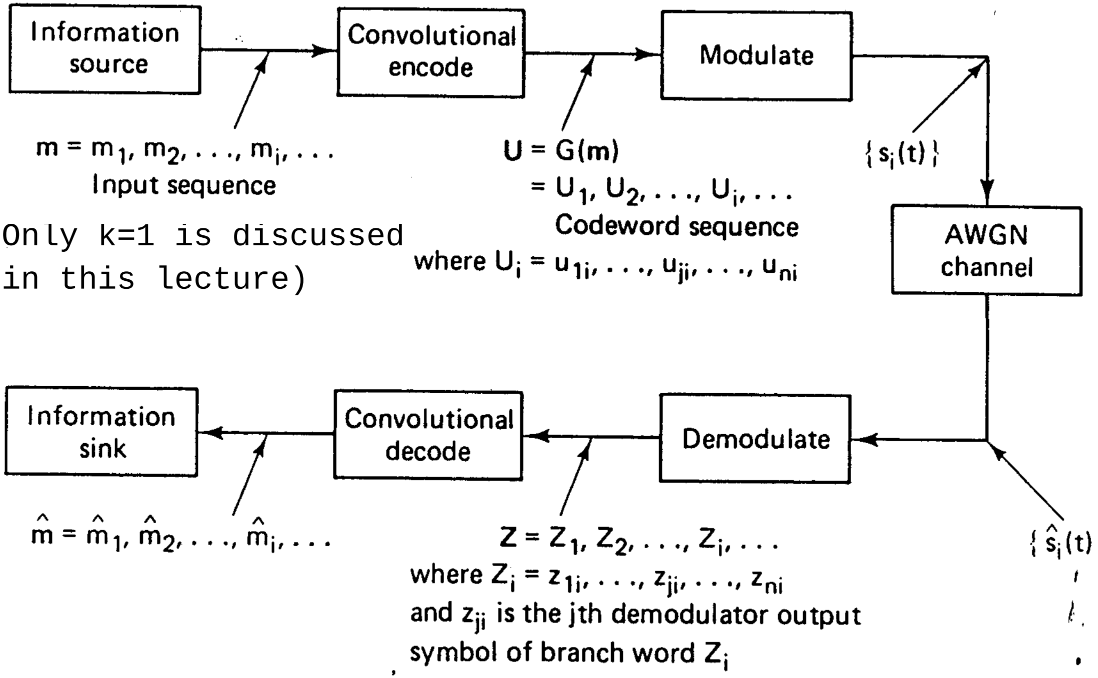
   CONVOLUTIONAL CODES ARE DESCRIBED BY

   n, k AND K

   WHERE ~~K~~ K-1 IS THE NUMBER OF SHIFT REGISTERS

2. FOR LINEAR BLOCK CODES, EACH CODEWORD n-TUPLE IS UNIQUELY DETERMINED BY THE INPUT MESSAGE k-TUPLE.
   FOR CONVOLUTIONAL CODES, n DOES NOT DEFINE A BLOCK OR CODEWORD LENGTH, HENCE NOT DETERMINED BY THE INPUT MESSAGE k-TUPLE ONLY, BUT ALSO A FUNCTION OF THE PREVIOUS (K-1) INPUT k-TUPLES.

3. DECODING OF LINEAR BLOCK CODES IS DONE n-BLOCK BY n-BLOCK AND EACH n-BLOCK WILL GIVE THE CORRESPONDING k-BLOCK MESSAGE.
   DECODING THE CONVOLUTIONAL CODES CANNOT BE DONE AS ABOVE, BUT THE LENGTH OF THE BLOCK DECODED AT ONE TIME DEPENDS ON THE ALGORITHMS OR METHODS ADOPTED.



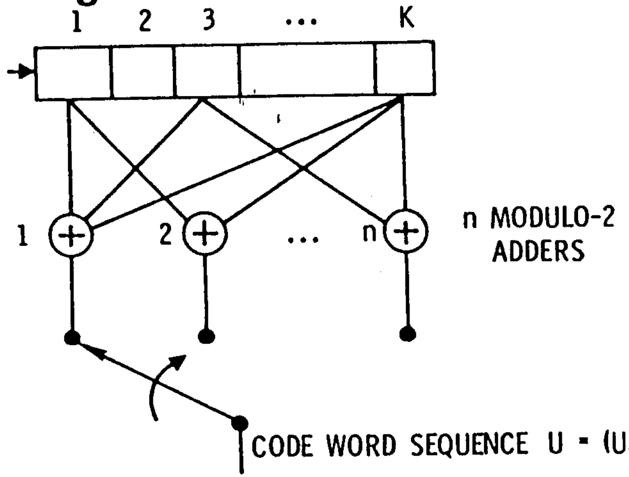
(Only k=1 is discussed in this lecture)

Figure 6.1 Encode/decode and modulate/demodulate portions of a communication link.

Convolutional Encoder with Constraint Length K and Rate 1/n

$m = (m_1, m_2, m_3, \ldots)$

INPUT DATA SEQUENCE



n MODULO-2 ADDERS

WHERE:

m_i = ith DATA BIT

CODE WORD SEQUENCE $U = (U_1, U_2, U_3 \ldots)$

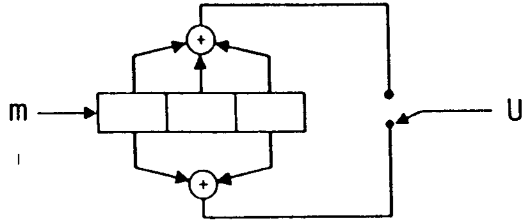
WHERE: $U_i = (u_{1i}, u_{2i} \cdots u_{ni})$

= ith CODE WORD, BRANCH

u_{ji} = jth CHANNEL BIT OF U_i

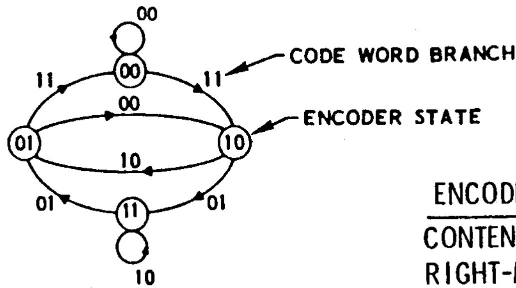
Encoder Characterization

1. CONNECTION PICTORIAL



ENCODER, K = 3, 1/n = 1/2

2. STATE DIAGRAM



ENCODER STATE DIAGRAM

ENCODER STATE
CONTENTS OF THE RIGHT-MOST K-1 STAGES

3. CONNECTION VECTORS OR POLYNOMIALS

$g_1 = 1 \quad 1 \quad 1$
$g_2 = 1 \quad 0 \quad 1$

$g_1(X) = 1 + X + X^2$
$g_2(X) = 1 + X^2$

Connection Pictorial

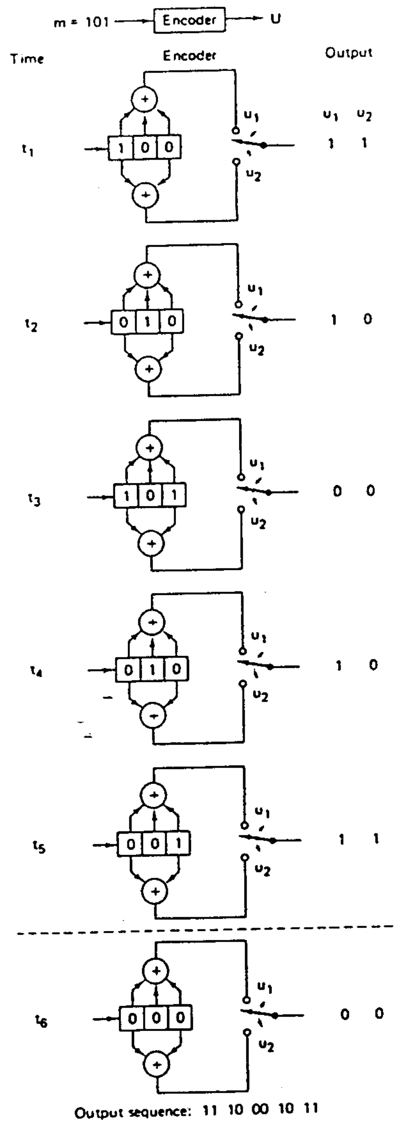


Figure 6.4 Convolutionally encoding a message sequence with a rate ½, K = 3 encoder.

Convolutional Encoding Example

INPUT DATA SEQUENCE = 1 1 0 1 1

OUTPUT CODE SEQUENCE = 11 01 01 00 01

INPUT	REGISTER	STATE t_i	STATE t_{i+1}	BRANCH WORD
1	1 0 0	00	10	11
1	1 1 0	10	11	01
0	0 1 1	11	01	01
1	1 0 1	01	10	00
1	1 $\underbrace{1\ 0}_{\text{state } t_i}$	10	11	01

$\underbrace{1\ 1}_{\text{state } t_{i+1}}$ 0

Connection Vectors Or Polynomials Representation

Connection Vectors

$$g_1 = 1\ 1\ 1$$
$$g_2 = 1\ 0\ 1$$

☞ IMPULSE RESPONSE OF THE ENCODER:

Register Contents	Branch Word u_1	u_2
100	1	1
010	1	0
001	1	1
	g_1	g_2

earliest transmission
↘
INPUT SEQUENCE : 1 0 0
OUTPUT SEQUENCE: 1 1 1 0 1 1

☞ TO FIND THE OUTPUT SEQUENCE FOR INPUT m = 1 0 1, USE SUPERPOSITION OR LINEAR ADDITION OF THE IMPULSE RESPONSES OF EACH INPUT BIT:

INPUT m	OUTPUT				
1	11	10	11		
0		00	00	00	
1			11	10	11
MODULO-2 SUM:	11	10	00	10	11

THIS IS POSSIBLE BECAUSE CONVOLUTION CODE IS LINEAR.

Polynomials

$g_1(X) = 1 + X + X^2$ AND $g_2(X) = 1 + X^2$

$m = 1\ 0\ 1 \longrightarrow m(X) = 1 + X^2$

OUTPUT SEQUENCE IS FOUND AS FOLLOWS:

$U(X) = m(X).g_1(X)$ INTERLACED WITH $m(X).g_2(X)$

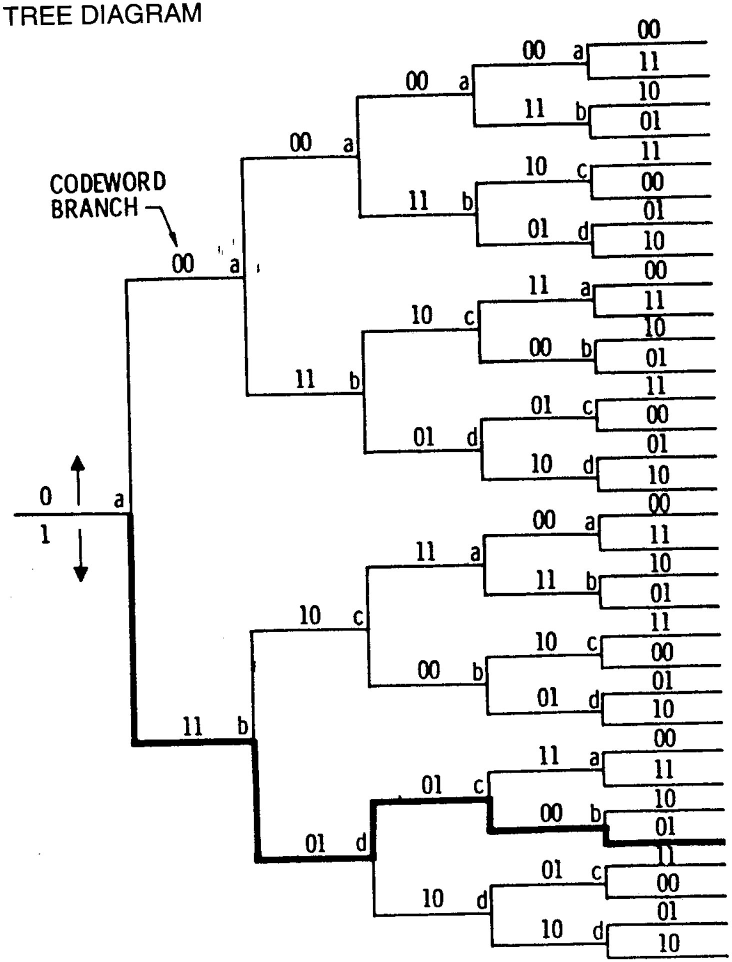
$$m(X)g_1(X) = (1 + X^2)(1 + X + X^2) = 1 + X + X^3 + X^4$$
$$m(X)g_2(X) = (1 + X^2)(1 + X^2) = 1 + X^4$$

RE-WRITE:

$$m(X)g_1(X) = 1 + X + 0X^2 + X^3 + X^4$$
$$m(X)g_2(X) = 1 + 0X + 0X^2 + 0X^3 + X^4$$
$$U(X) = (1,1) + (1,0)X + (0,0)X^2 + (1,0)X^3 + (1,1)X^4$$

OR $U = 11 \quad 10 \quad 00 \quad 10 \quad 11$

4. TREE DIAGRAM



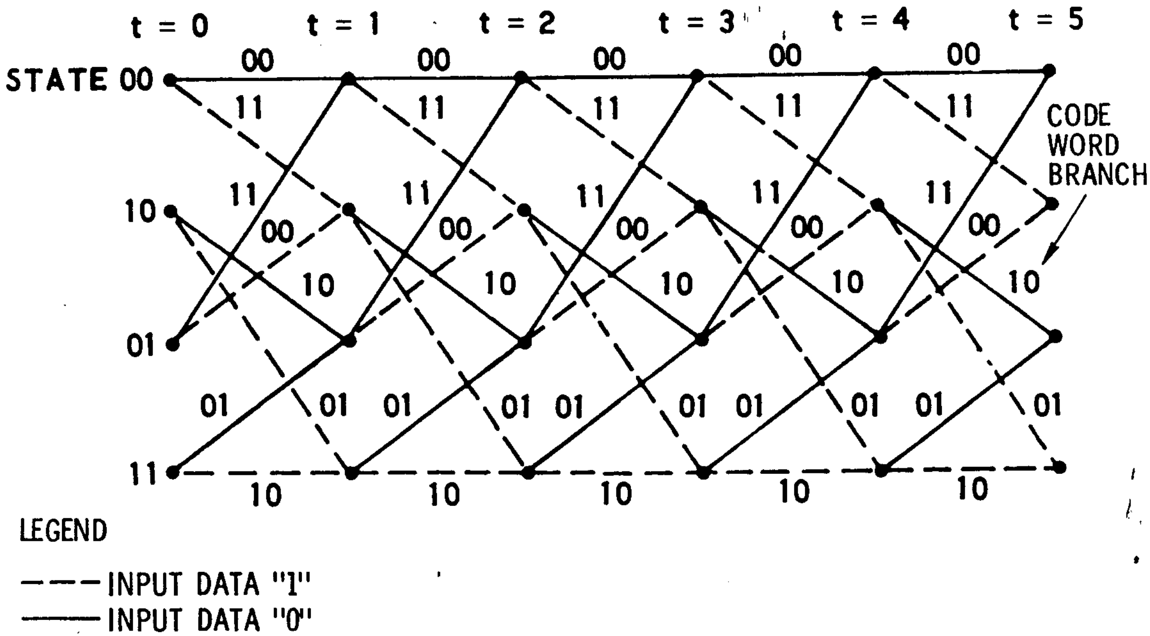
Tree Representation of Encoder

THE INPUT DATA SEQUENCE m = 11011 TRACES THE HEAVY-LINE PATH, HAVING THE CODEWORD SEQUENCE

U = 11 01 01 00 01

5. TRELLIS DIAGRAM

Encoder Trellis Diagram
Shows all Possible Transitions at Each Time Unit



Problem of Decoding

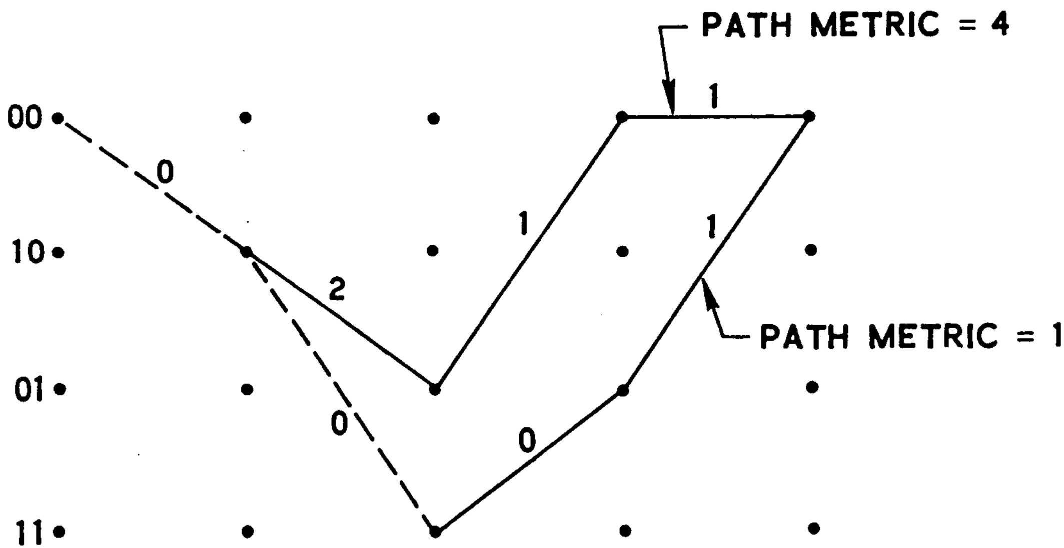
Maximum Likelihood Decoding

- COMPARING THE RECEIVED SEQUENCE WITH ALL POSSIBLE TRANSMITTED SEQUENCE BEFORE MAKING A DECISION
- ALL THE POSSIBLE TRANSMITTED SEQUENCE CAN BE OBTAINED FROM THE TREE STRUCTURE OF THE ENCODER
- HENCE, OPTIMAL DECODING OF AN L-BIT BINARY SEQUENCE REQUIRES COMPARING THE 2^L ACCUMULATED METRICS $\longrightarrow$ EXPONENTIAL INCREASE OF DECODING EFFORT WITH THE LENGTH L OF THE SEQUENCE.

Viterbi Decoding

- SUBOPTIMAL DECODING
- IN TREE STRUCTURE DECODING ABOVE, THE PATHS THAT REMERGE IS ENTIRELY IGNORED
- VITERBI SUGGESTED TO USE TRELLIS STRUCTURE TO CONSIDER THE REMERGING PATHS:
  WHEN 2 PATHS MERGE, THE PATH WITH THE LARGER METRIC CAN BE ELIMINATED
- HENCE, FOR L SEQUENCE REQUIRES ONLY $L2^{K-1}$ OPERATIONS

If Two Paths Merge, One of Them Can be Eliminated



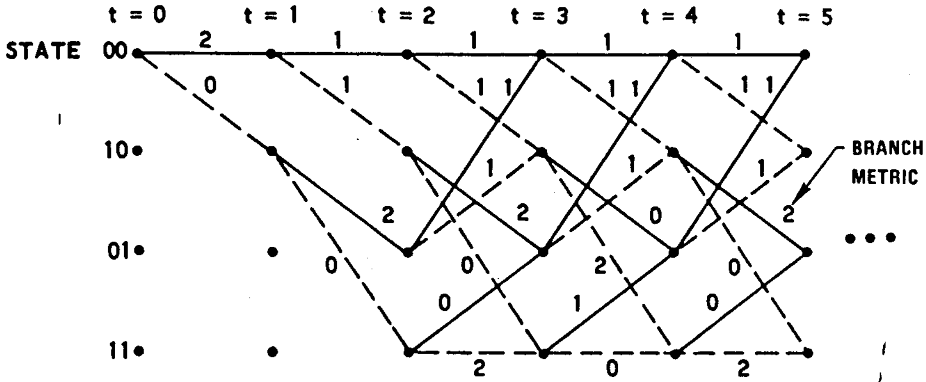
PATH METRICS FOR TWO REMERGING PATHS
AT t = 4

Viterbi Decoding

- FOR SIMPLICITY, ASSUME BINARY SYMMETRIC CHANNEL
- ENCODER HAS CONSTRAINT LENGTH 3, AND RATE 1/2
- A TRELLIS REPRESENTS THE DECODER (similar to the encoder trellis)
- THE TRELLIS TRANSITIONS ARE LABELLED WITH BRANCH METRICS (hamming distance between branch code word and rcvd code word)
- IF ANY 2 TRELLIS PATHS MERGE TO A SINGLE STATE, THE PATH WITH THE LARGER METRIC IS ELIMINATED

Viterbi Decoding Example

INPUT DATA SEQUENCE	m:	1	1	0	1	1	• • •
TRANSMITTED CODEWORD	U:	11	01	01	00	01	• • •
RECEIVED CODEWORD	Z:	11	01	01	10	01	• • •

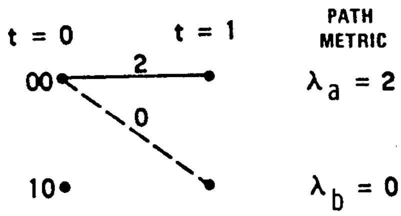


ASSUME

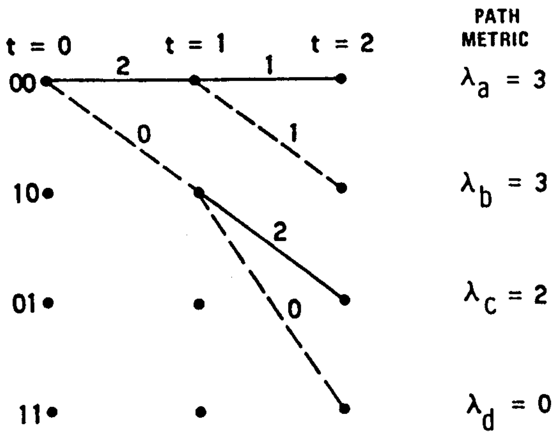
DECODER KNOWS CORRECT INITIAL STATE OF THE TRELLIS

DECODER TRELLIS DIAGRAM

Selection of Survivor Paths

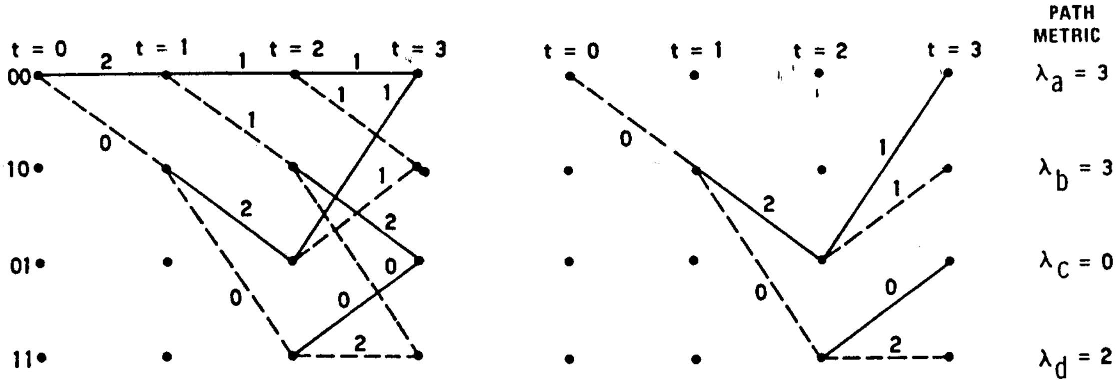


SURVIVORS AT t = 1



SURVIVORS AT t = 2

Selection of Survivor Paths (cont'd)



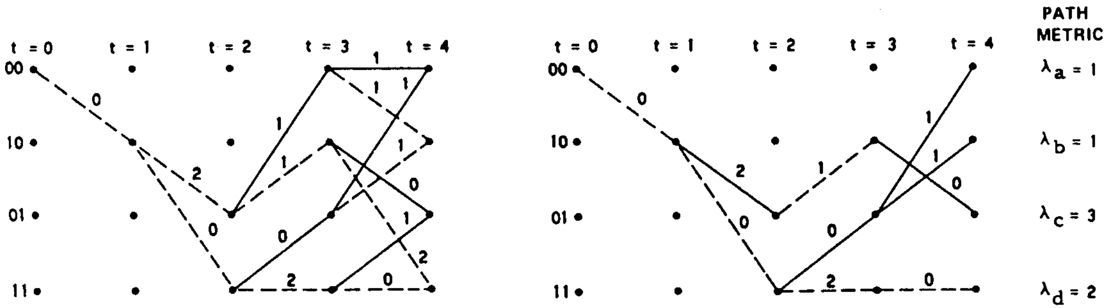
METRIC COMPARISONS AT t = 3

SURVIVORS AT t = 3

ONE PATH ENTERING EACH STATE CAN BE ELIMINATED

DECODER DECIDES THAT TRANSITION FROM t = 0 TO t = 1, WAS PRODUCED BY DATA BIT "1"

Selection of Survivor Paths (cont'd)

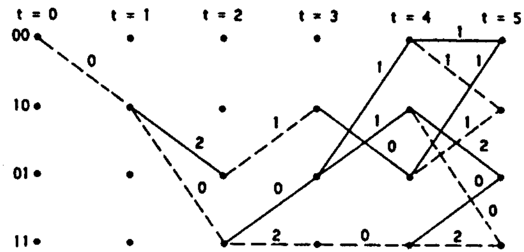


METRIC COMPARISONS AT t = 4

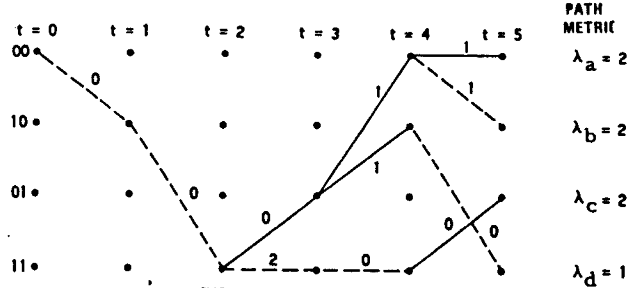
SURVIVORS AT t = 4

AGAIN, ONE OF TWO PATHS ENTERING
SAME STATE CAN BE ELIMINATED

Selection of Survivor Paths (concluded)



METRIC COMPARISONS AT t = 5



SURVIVORS AT t = 5

DECODER DECIDES, THE SECOND DATA BIT = "1"

Path Memory

☞ PROBLEM: IF A L-BIT LONG BINARY SEQUENCE (L USUALLY IS A LARGE NUMBER) IS TRANSMITTED, SHOULD WE WAIT UNTIL THE RECEIVER RECEIVED ALL THE L BITS, COMPARED AND DECODED THEM, THEN WE START TO OUTPUT THE DECODED SEQUENCE?

☞ IF WE DO:
⇨ WE HAVE RECEIVER WITH VERY LONG DELAY
⇨ RECEIVER REQUIRES VERY LARGE MEMORY STORAGE ($= L2^{K-1}$)

☞ PRACTICAL DECODER IS NEEDED WHICH DOES NOT HAVE TO WAIT TILL ALL THE L BITS ARE RECEIVED BEFORE IT CAN OUTPUT THE DECODED BIT

☞ FROM THE PREVIOUS EXAMPLE, WE SEE:
- AT t_4 WE CAN OUTPUT BIT DECODED AT t_1
- BUT BIT DECODED AT t_2 CANNOT BE OUTPUT UNTIL t_6, SO THERE IS DISCONTINUITY
- ALSO, IF MANY ERROR BITS ARE RECEIVED, IT IS IMPOSSIBLE TO HAVE A SINGLE PATH SURVIVED AT t_1 WHEN DECODING DEPTH HAS REACHED TO t_4 AS THE ABOVE-MENTIONED

☞ SIMULATION RESEARCH HAS FOUND THAT THE PATH STORAGE REQUIRED IS

$$U = h.2^{K-1}$$

WHERE h = 4 OR 5 TIMES K

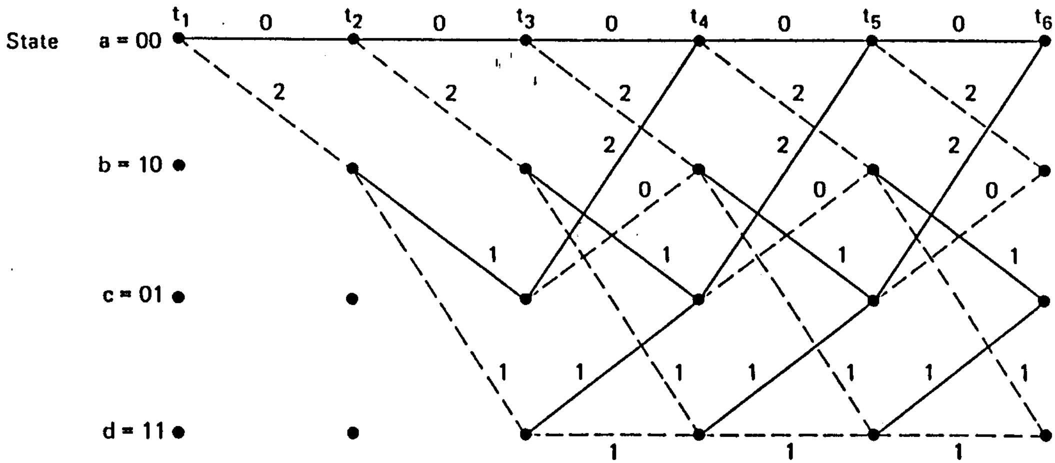
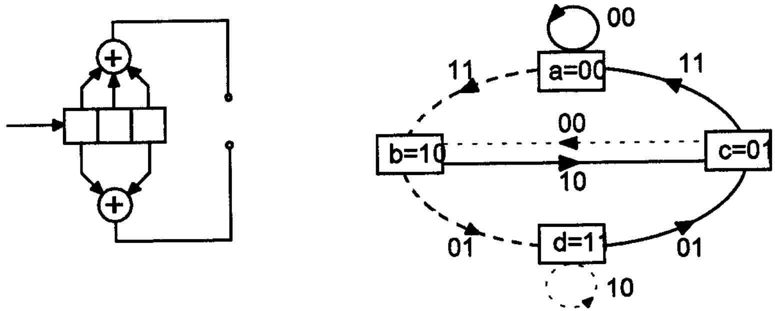


Figure 6.13 Trellis diagram, labeled with distances from the all-zeros path.

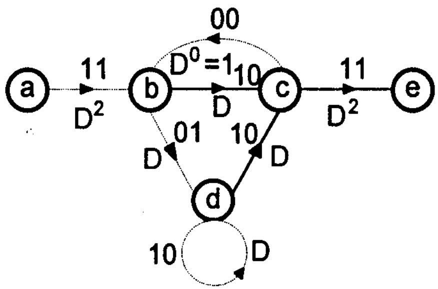
For linear code, can assume all-zeros path is the correct transmitted path. Compare all possible paths with this all-zeros path.
The minimum distance from the all-zeros path is the minimum free distance of the code (a to b to c to a). The minimum free distance of the above code is 2+1+2=5.

Transfer Function



MODIFY STATE DIAGRAM INTO INPUT-OUTPUT FORM:

- LABEL THE BRANCHES AS EITHER D^0 (=1), D^1, D^2, ..., WHERE THE EXPONENT OF D DENOTES THE HAMMING DISTANCE FROM THE BRANCH WORD TO THE ALL-ZERO BRANCH
- ELIMINATE THE SELF-LOOP AT NODE a SINCE IT CONTRIBUTE NOTHING TO THE DISTANCE PROPERTY
- SPLIT NODE a INTO 2 NODES: ONE INPUT, THE OTHER OUTPUT
- DENOTE THE PARTIAL PATHS TO THE INTERMEDIATE NODES a, b, c, d AND e WITH DUMMY VARIABLES X_a, X_b, X_c, X_d AND X_e



$$X_b = D^2 X_a + X_c \quad \text{.........(1)}$$
$$X_c = DX_b + DX_d \quad \text{....... (2)}$$
$$X_d = DX_b + DX_d \quad \text{.........(3)}$$
$$X_e = D^2 X_c \quad \text{.........(4)}$$

(2) - (3): $X_c = X_d$

(2): $X_c = DX_b + DX_c \quad \Rightarrow \quad X_b = \left(\frac{1-D}{D}\right)X_c$

(1):

$$\frac{1-D}{D}\Bigg)X_c = D^2 X_a + X_c$$

$$X_c = \left(\frac{D^3}{1-2D}\right)X_a$$

(4):

$$X_e = D^2\left(\frac{D^3}{1-2D}\right)X_a$$

$$\frac{X_e}{X_a} = \frac{D^5}{1-2D}$$

HENCE,

$$T(D) = \frac{D^5}{1-2D}$$

NOTE THAT FOR INFINITE GP, THE SUM OF THE SERIES:

$$S_\infty = \frac{a}{1-r} = a + ar + ar^2 + \ldots\ldots$$

SO LET $a = D^5$ AND $r = 2D$:

$$T(D) = \frac{D^5}{1-2D}$$

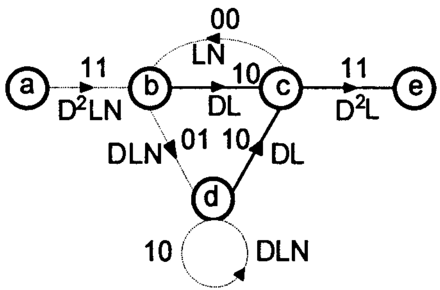
GIVE $d_f = 5$

$$= D^5 + 2D^6 + 4D^7 + \ldots\ldots + 2^\ell D^{\ell+5} + \ldots\ldots\ldots$$

one path of distance 5 from all-zero path	two paths of distance 6 from all-zero path	in general, there are 2^ℓ paths of distance $\ell+5$ from all-zero path

MORE DETAILED INFORMATION CAN BE PROVIDED BY INTRODUCING FACTORS:

- L: INTO ALL THE BRANCHES. THE EXPONENT OF L INDICATES THE NUMBER OF BRANCHES IN ANY GIVEN PATH FROM STATE a = 00 TO STATE e = 00
- N: ONLY INTO THE BRANCH TRANSITIONS CAUSED BY THE INPUT BIT ONE. THE EXPONENT OF N INDICATES HOW MANY INPUT BITS THIS PATH DIFFERS FROM THE ALL-ZEROS PATH



$X_b = D^2LN\,X_a + LN\,X_c$ ....(1)

$X_c = DL\,X_b + DL\,X_d$ .... (2)

$X_d = DLN\,X_b + DLN\,X_d$ ....(3)

$X_e = D^2L\,X_c$ ....(4)

THE TRANSFER FUNCTION BECOMES:

$$T(D,L,N) = \frac{D^5L^3N}{1-DL(1+L)N}$$

$$= D^5L^3N + D^6L^4(1+L)N^2 + D^7L^5(1+L)^2N^3 + \ldots\ldots + D^{\ell+5}L^{\ell+3}N^{\ell+1} + \ldots\ldots$$

Bit Error Probability

THE UPPER BOUND OF THE CONVOLUTIONAL CODE PROBABILITY OF BIT ERROR IS

$$P_B \le \left.\frac{dT(D,N)}{dN}\right|_{N=1,D=2\sqrt{p(1-p)}}$$

WHERE p = THE PROBABILITY OF CHANNEL SYMBOL ERROR

EXAMPLE:

USE THE $T(D,L,N)$ OBTAINED PREVIOUSLY, AND SET L = 1:

$$T(D,N) = \frac{D^5 N}{1-2DN}$$

$$\Rightarrow \left.\frac{dT(D,N)}{dN}\right|_{N=1} = \frac{D^5}{(1-2D)^2}$$

$$P_B \le \frac{\{2[p(1-p)]^{1/2}\}^5}{\{1-4[p(1-p)]^{1/2}\}^2}$$

Catastrophic Error

- CATASTROPHIC ERROR = AN EVENT WHEREBY A *FINITE* NUMBER OF CODE SYMBOL ERRORS CAUSE AN *INFINITE* NUMBER OF DECODED DATA BIT ERRORS
- SOME ENCODER CONNECTION DESIGNS CAN PRODUCE CATASTROPHIC ERROR
- FOR RATE 1/n ENCODER, THE NECESSARY AND SUFFICIENT CONDITION FOR CONVOLUTIONAL CODES TO DISPLAY CATASTROPHIC ERROR IS

THE GENERATORS HAVE A COMMON POLYNOMIAL FACTOR

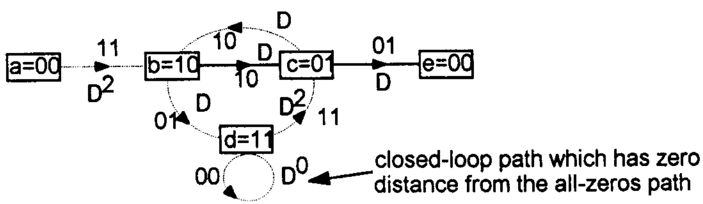
EXAMPLE: $g_1(X) = 1 + X$

$g_2(X) = 1 + X^2 = (1 + X)(1 + X)$

THE COMMON POLYNOMIAL FACTOR IS $1 + X$

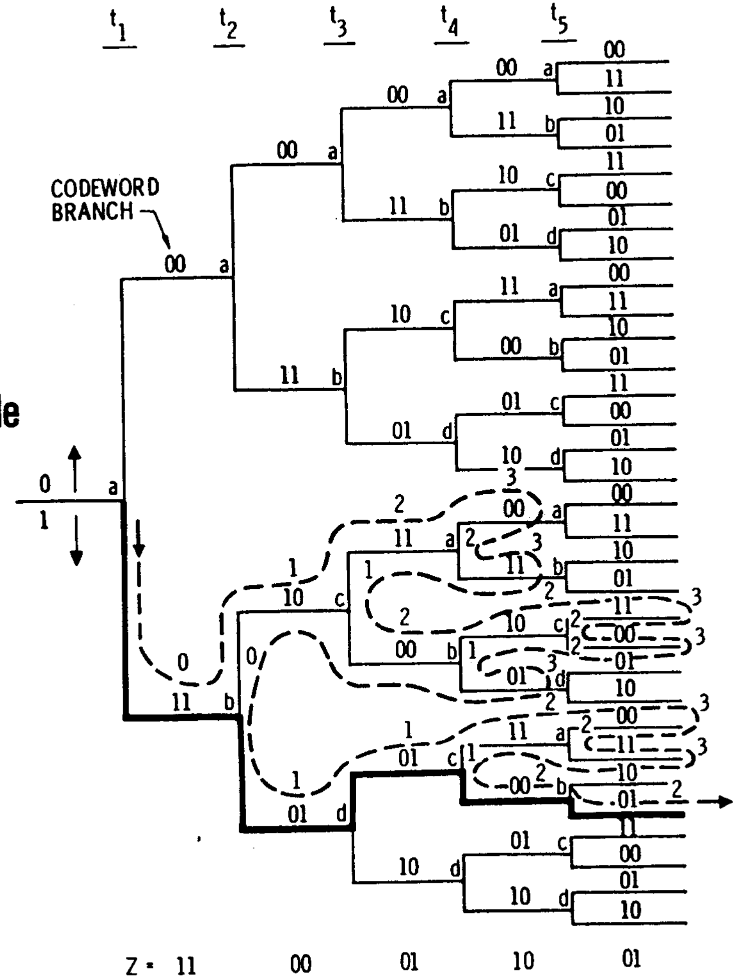
- IN TERMS OF STATE DIAGRAM IS

IF ONLY IF, ANY CLOSED-LOOP PATH IN THE DIAGRAM HAS ZERO DISTANCE FROM THE ALL-ZERO PATH.



Sequential Decoding Example

(FANO ALGORITHM)



SET THRESHOLD = 3

SEQUENTIAL DECODER (STACK ALGORITHM)

☞ COMPARISON WITH FANO ALGORITHM:

- FANO ALGORITHM IS MORE PRACTICAL BECAUSE IT USES LESS MEMORY STORAGE
- STACK ALGORITHM HAS ADVANTAGE IN SPEED OF COMPUTATION
- STACK ALGORITHM IS EASIER TO UNDERSTAND

☞ THE STEPS INVOLVED:

- BRANCH METRIC CALCULATION (FOR HARD-DECISION, BSC CHANNEL)

$$\mu_{jm}^{(r)} = \begin{cases} \log_2 2(1-p) - R_c & \text{if } \tilde{y}_{jm} = c_{jm}^{(r)} \\ \log_2 2p - R_c & \text{if } \tilde{y}_{jm} \neq c_{jm}^{(r)} \end{cases}$$

WHERE μ = THE METRIC FOR rth PATH, jth BRANCH, AND mth CODE BIT

p = BSC CHANNEL TRANSITION ERROR PROBABILITY

R_C = CODE RATE

y_{jm} = THE mth RECEIVED BIT

c_{jm} = THE mth CODED BIT

- PLACE THE MOST PROBABLE PATHS, I.E. PATH WITH THE LARGEST METRIC AT THE TOP OF THE STACK, FOLLOWED BY THE NEXT LARGEST AND SO ON TO THE LEAST AT THE BOTTOM.
- AT EACH STEP, ONLY THE PATH AT THE TOP OF THE STACK IS EXTENDED BY ONE BRANCH.
- RE-ORDER THE VALUES IN THE STACK AT EVERY OF THESE STEPS WITH THE LARGEST ON TOP THE LEAST ON THE BOTTOM.
- REPEAT THE PROCESS OF EXTENDING OF THE PATH.

EXAMPLE OF STACK ALGORITHM

WE FOLLOW THE EXAMPLE FOR FANO ALGORITHM.

- TAKE THE BSC TRANSITION PROBABILITY, p = 0.1.

- CALCULATE THE BRANCH METRIC FOR EACH BIT:

$$\mu_{jm}^{(r)} = \begin{cases} \log_2 2(1-0.1) - 1/2 = 0.35 & \text{if } \tilde{y}_{jm} = c_{jm}^{(r)} \\ \log_2 2(0.1) - 1/2 = -2.82 & \text{if } \tilde{y}_{jm} \neq c_{jm}^{(r)} \end{cases}$$

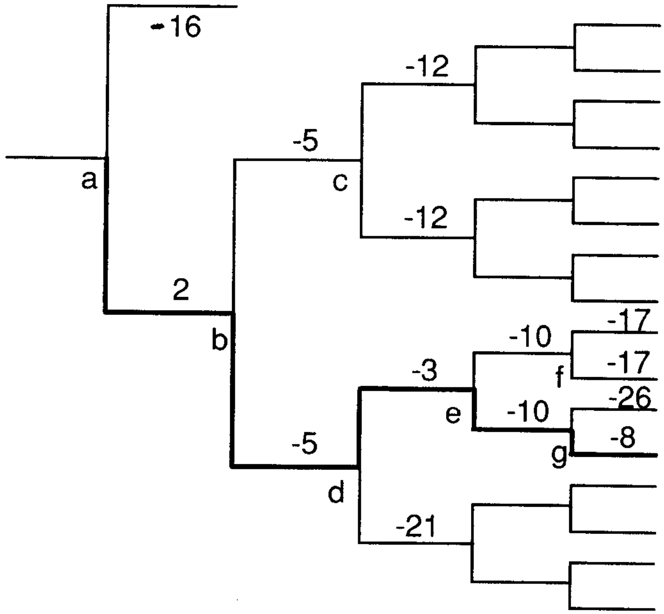
TO SIMPLIFY CALCULATION, NORMALISE THE ABOVE VALUES:

$$\mu_{jm}^{(r)} = \begin{cases} 1 & \text{if } \tilde{y}_{jm} = c_{jm}^{(r)} \\ -8 & \text{if } \tilde{y}_{jm} \neq c_{jm}^{(r)} \end{cases}$$

HENCE, FOR TWO OUTPUT BITS, THE BRANCH METRIC IS

$$\begin{aligned} \mu_j^{(r)} &= 2 - d + (-8)d \\ &= 2 - 9d \end{aligned}$$

- CALCULATE THE ACCUMULATIVE BRANCH METRICS ON THE TREE DIAGRAM AND RE-ORDER THE VALUES ON THE STACK:



Step a	Step b	Step c	Step d	Step e	Step f	Step g
2	-5	-5	-3	-10	-10	-8
-16	-5	-12	-12	-10	-12	-12
	-16	-12	-12	-12	-12	-12
		-16	-16	-12	-16	-16
			-21	-16	-17	-17
				-21	-17	-17
					-21	-21
						-26

Comparison - Viterbi And Sequential Decoding

Viterbi Decoding

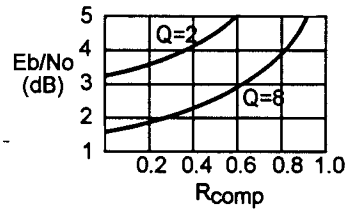
- DECODER COMPLEXITY GROWS EXPONENTIALLY WITH CONSTRAINT LENGTH
- COMPUTATIONAL EFFORT IS FIXED FOR PARTICULAR DECODER INDEPENDENT OF CHANNEL CHARACTERISTICS
- DUE TO THE FIXED NATURE OF COMPUTATIONAL EFFORT, THE STORAGE PATH HISTORY AND DECODING SPEED REQUIREMENTS MAY DEFINITELY BE SET FOR A PARTICULAR APPLICATION
- VITERBI DECODER ARE LIMITED TO SMALL ($K \leq 10$) CONSTRAINT-LENGTH CODES; ARE USED WHERE A MODERATE ERROR-PROBABILITY IS SUFFICIENT (10^{-4} TO 10^{-5}); AND VERY HIGH DATA RATES (10s OF MBPS)

Sequential Decoding

- DECODER COMPLEXITY IS INDEPENDENT OF CONSTRAINT LENGTH
- COMPUTATIONAL EFFORT (COMPLEXITY) IS VERY SENSITIVE TO CHANNEL CHARACTERISTICS AND IS RANDOM VARIABLE

☞ DUE TO THE RANDOMNESS AND SENSITIVITY OF COMPUTATION EFFORT TO CHANNEL CHARACTERISTICS:

1. SEQUENTIAL DECODERS ARE LIMITED TO CODES WITH CODE RATE BELOW R_{COMP} (= COMPUTATIONAL CUT-OFF RATE)

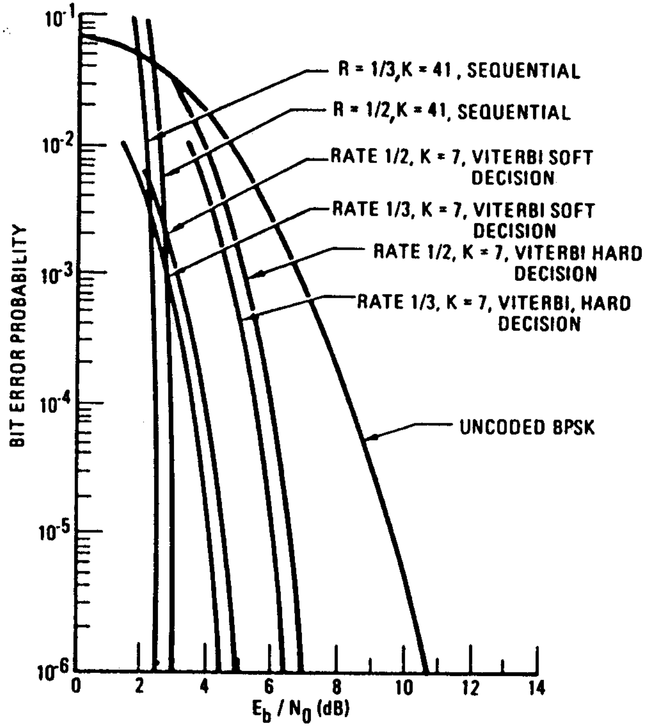


2. BUFFER TO STORE INCOMING DATA TO DECODER AND SPEED OF DECODER ARE ALSO RANDOM AND CANNOT BE SET FOR PARTICULAR APPLICATION. SETTING THEM FIXED WILL CAUSE A LARGE VARIABILITY OF DECODING PERFORMANCE (WITH JUST A VERY SLIGHT CHANGE IN Eb/No). THIS IS THE REASON FOR THE VERY STEEP GRAPH FOR THE ERROR PERFORMANCE OF SEQUENTIAL DECODERS

3. VERY SENSITIVE TO BURSTY NOISE PATTERN

☞ SEQUENTIAL DECODERS ARE USED IF HIGH PERFORMANCE (Pb < 10-5) IS REQUIRED. HIGH PERFORMANCE CAN BE ACHIEVED BY USING LARGE CONSTRAINT-LENGTH CODES. THEIR USED ARE RESTRICTED TO WELL-BEHAVED MEMORYLESS CHANNELS SUCH AS THE SPACE AND SATELLITE CHANNELS AT MODERATE SPEED (~ 5 MBPS)

Viterbi Decoder Performance Rate 1/2 vs Rate 1/3 K = 7 Hard vs Soft



Summary On Convolutional Codes

☞ ENCODER REPRESENTATION AND CODEWORDS GENERATION:

1. CONNECTION PICTORIAL
2. STATE DIAGRAM
3. CONNECTION VECTORS OR POLYNOMIALS
4. TREE DIAGRAM
5. TRELLIS DIAGRAM

☞ DECODING TECHNIQUES:

- RELATIONSHIP BETWEEN MAXIMUM LIKELIHOOD DECODING AND HAMMING DISTANCE
- HARD AND SOFT DECISION CHANNEL MODELS
- VITERBI DECODING ALGORITHM:
  - ⇨ CUMMULATIVE HAMMING DISTANCE OF A PATH: - PATH METRIC
  - ⇨ SELECTION OF SURVIVOR PATH FROM THE TWO PATHS WHICH MERGED
  - ⇨ PATH MEMORY: $U = h.2^{K-1}$
- SEQUENTIAL DECODING: FANO AND STACK ALGORITHMS
- ADVANTAGES AND DISADVANTAGES BETWEEN VITERBI AND SEQUENTIAL DECODINGS

☞ PROPERTIES OF CONVOLUTIONAL CODES:

- DISTANCE PROPERTIES AND TRANSFER FUNCTION
- CATASTROPHIC ERROR