Digital Communications I: Modulation and Coding Course

Term 3 - 2008

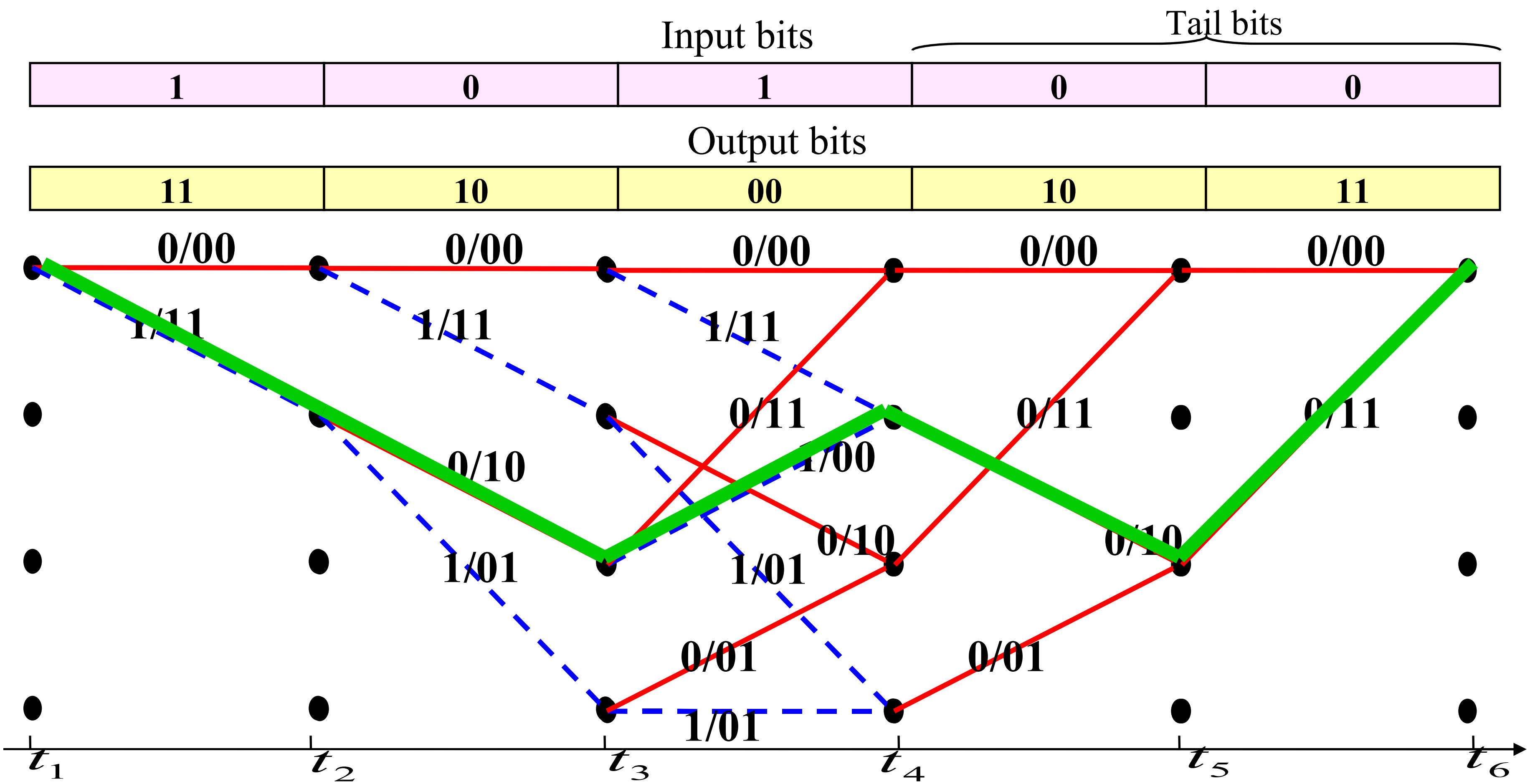
Catharina Logothetis

Lecture 12

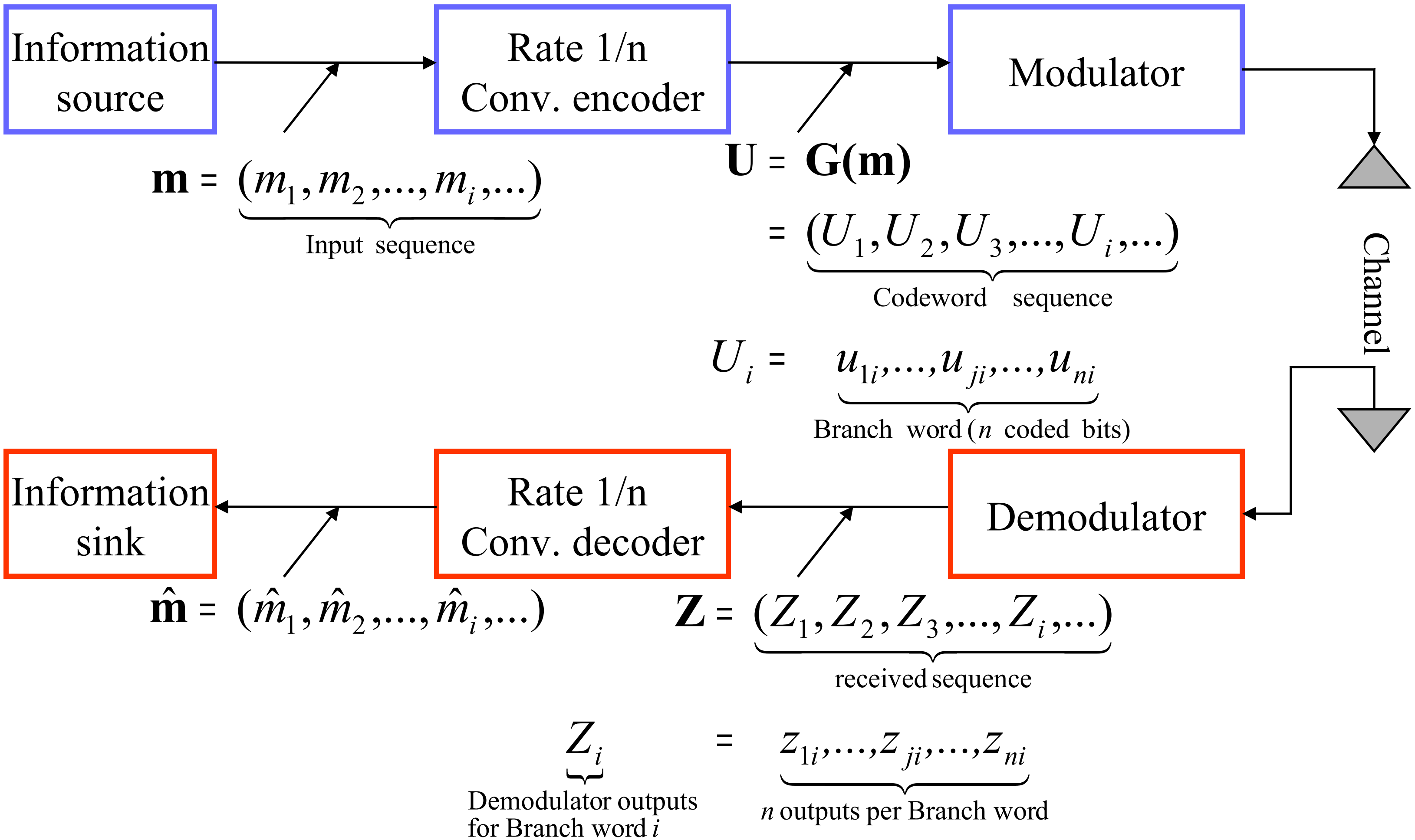
Last time, we talked about:

- How decoding is performed for Convolutional codes?
- What is a Maximum likelihood decoder?
- What are soft decisions and hard decisions?
- How does the Viterbi algorithm work?

Trellis of an example ½ Conv. code



Block diagram of the DCS



Soft and hard decision decoding

- In hard decision:
  - The demodulator makes a firm or hard decision whether one or zero was transmitted and provides no other information for the decoder such as how reliable the decision is.

- In Soft decision:
  - The demodulator provides the decoder with some side information together with the decision. The side information provides the decoder with a measure of confidence for the decision.

Soft and hard decision decoding …

- ML soft-decisions decoding rule:
  - Choose the path in the trellis with minimum Euclidean distance from the received sequence

- ML hard-decisions decoding rule:
  - Choose the path in the trellis with minimum Hamming distance from the received sequence

The Viterbi algorithm

- The Viterbi algorithm performs Maximum likelihood decoding.

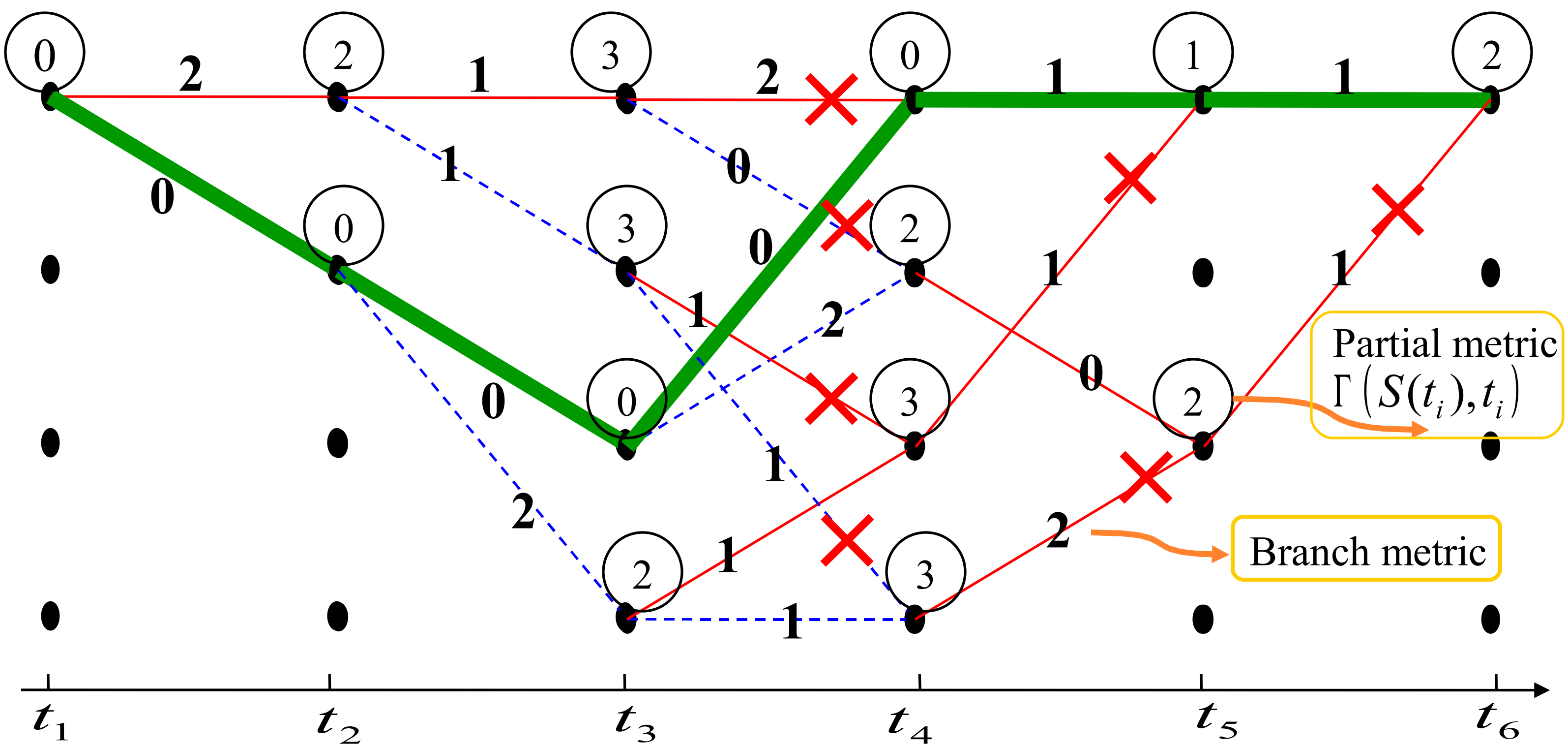
- It finds a path through trellis with the largest metric (maximum correlation or minimum distance).
  - At each step in the trellis, it compares the partial metric of all paths entering each state, and keeps only the path with the largest metric, called the survivor, together with its metric.

Example of hard-decision Viterbi decoding

$\mathbf{Z} = (11\ \ 10\ \ 11\ \ 10\ \ 01)$

$\hat{\mathbf{m}} = (100)$
$\hat{\mathbf{U}} = (11\ \ 10\ \ 11\ \ 00\ \ 11)$

$\mathbf{m} = (101)$
$\mathbf{U} = (11\ \ 10\ \ 00\ \ 10\ \ 11)$

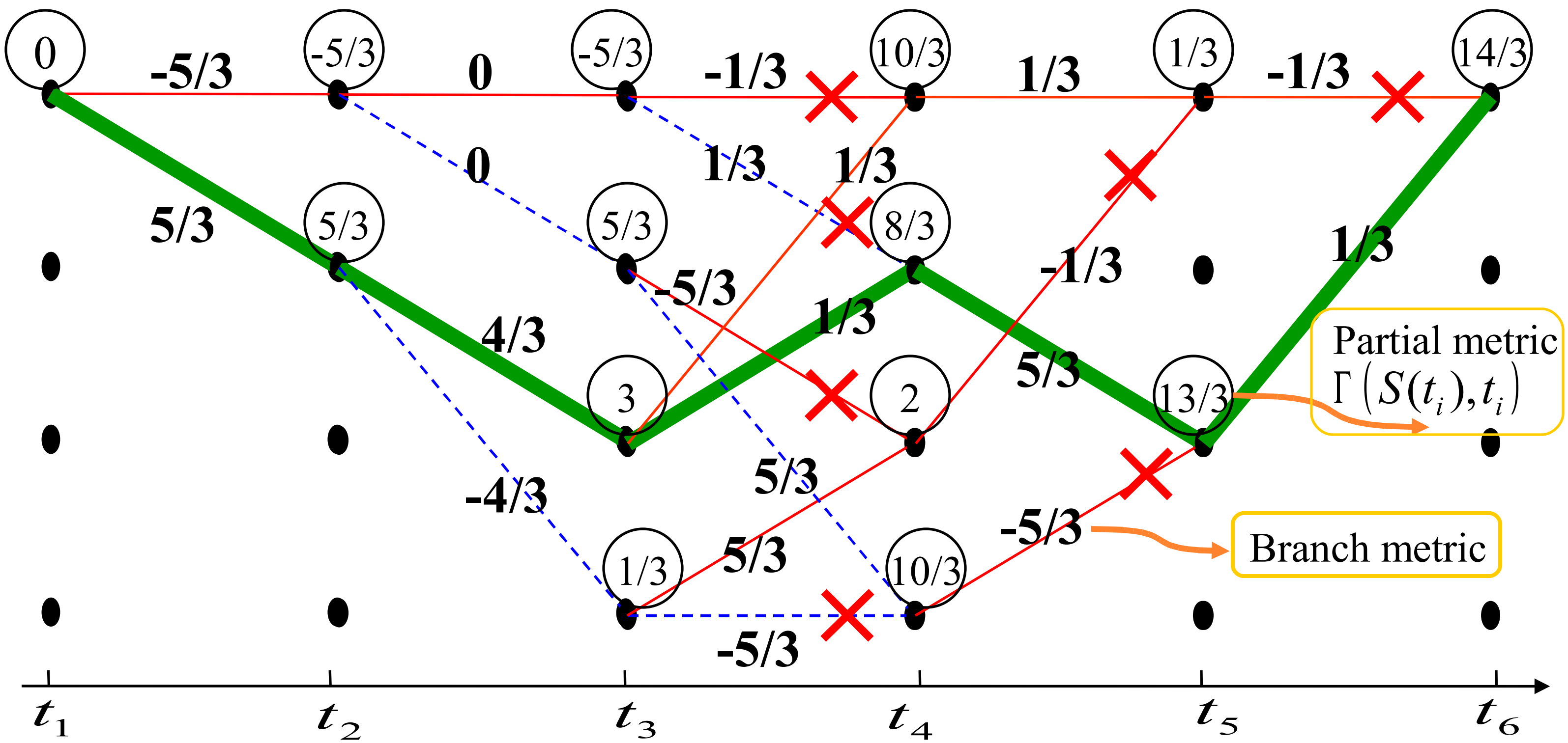


Example of soft-decision Viterbi decoding

$$\mathbf{Z} = \left(1, \frac{2}{3}, \frac{2}{3}, \frac{-2}{3}, \frac{-2}{3}, 1, \frac{2}{3}, -1, \frac{-2}{3}, 1\right)$$

$$\hat{\mathbf{m}} = (101)$$
$$\hat{\mathbf{U}} = (11\ \ 10\ \ 00\ \ 10\ \ 11)$$

$$\mathbf{m} = (101)$$
$$\mathbf{U} = (11\ \ 10\ \ 00\ \ 10\ \ 11)$$



Today, we are going to talk about:

- The properties of Convolutional codes:
  - Free distance
  - Transfer function
  - Systematic Conv. codes
  - Catastrophic Conv. codes
  - Error performance
- Interleaving
- Concatenated codes
- Error correction scheme in Compact disc

Free distance of Convolutional codes

- Distance properties:
  - Since a Convolutional encoder generates codewords with various sizes (as opposite to the block codes), the following approach is used to find the minimum distance between all pairs of codewords:
    - Since the code is linear, the minimum distance of the code is the minimum distance between each of the codewords and the all-zero codeword.
    - This is the minimum distance in the set of all arbitrary long paths along the trellis that diverge and re-merge to the all-zero path.
    - It is called the minimum free distance or the free distance of the code, denoted by d_{free} or d_f

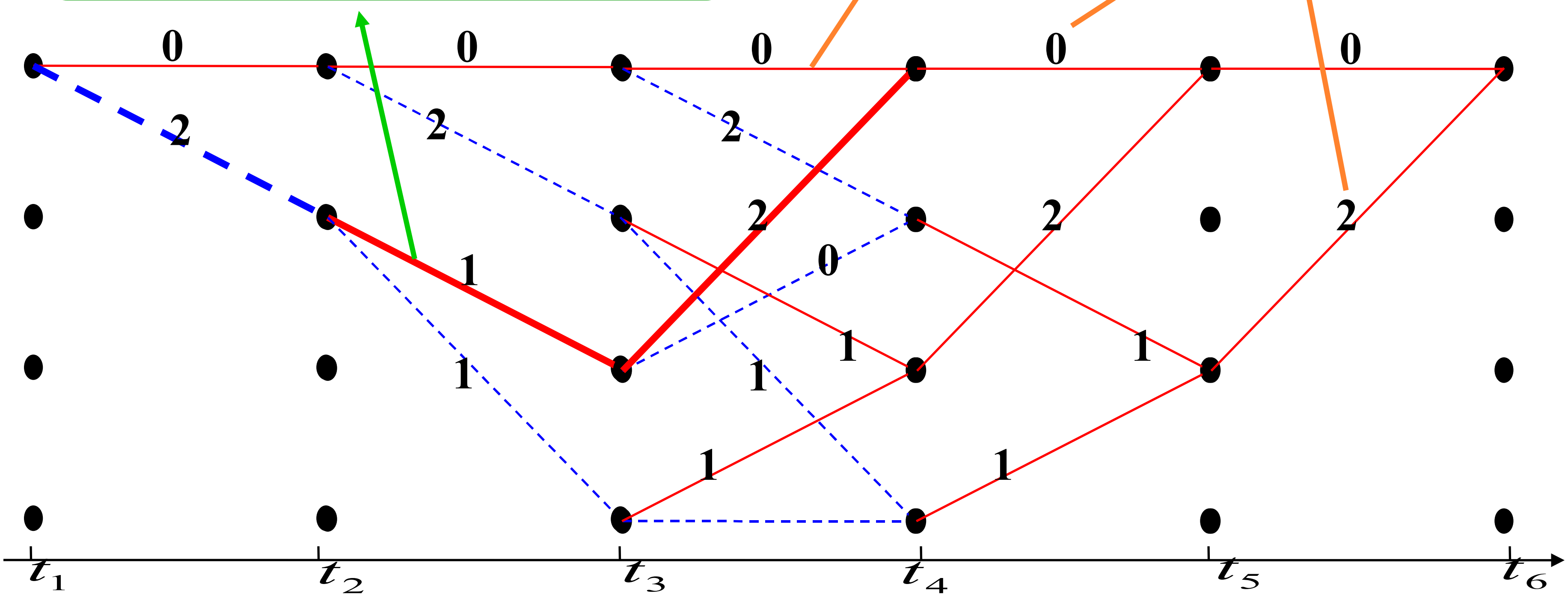
Free distance …

The path diverging and re-merging to the all-zero path w. minimum weight

$\Rightarrow$ $d_f = 5$

All-zero path

Hamming weight of the branch



Transfer function of Convolutional codes

- Transfer function:
  - The transfer function of the generating function is a tool which provides information about the weight distribution of the codewords.
    - The weight distribution specifies weights of different paths in the trellis (codewords) with their corresponding lengths and amount of information.

$$\boxed{T(D,L,N) = \sum_{i=d_f} \sum_{j=K} \sum_{l=1} D^i L^j N^l}$$

$D, L, N:$ place holders

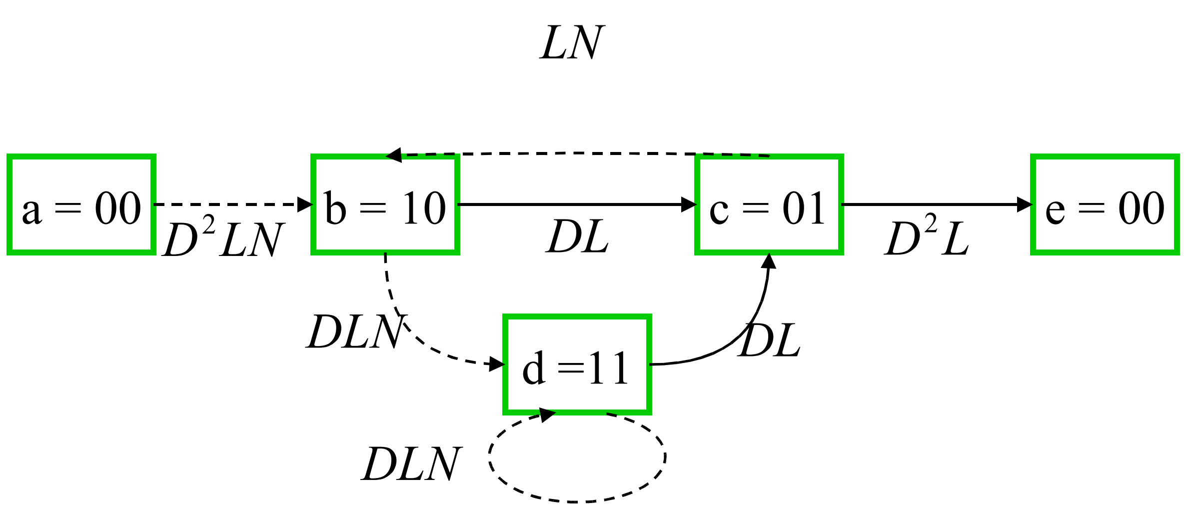
$i:$ distance of the path from the all-zero path

$j:$ number of branches that the path takes until it remerges to the all-zero path

$l:$ weight of the information bits corresponding to the path

Transfer function …

- Example of transfer function for the rate ½ Convolutional code.
  1. Redraw the state diagram such that the zero state is split into two nodes, the starting and ending nodes.
  2. Label each branch by the corresponding $D^i L^j N^l$



Transfer function …

- Write the state equations ($X_a, \ldots, X_e$ dummy variables)

$$\begin{cases} X_b = D^2 L N X_a + L N X_c \\ X_c = D L X_b + D L X_d \\ X_d = D L N X_b + D L N X_d \\ X_e = D^2 L X_c \end{cases}$$

- Solve $T(D, L, N) = X_e / X_a$

$$T(D, L, N) = \frac{D^5 L^3 N}{1 - DL(1 + L)N} = D^5 L^3 N + D^6 L^4 N^2 + D^6 L^5 N^2 + \ldots.$$

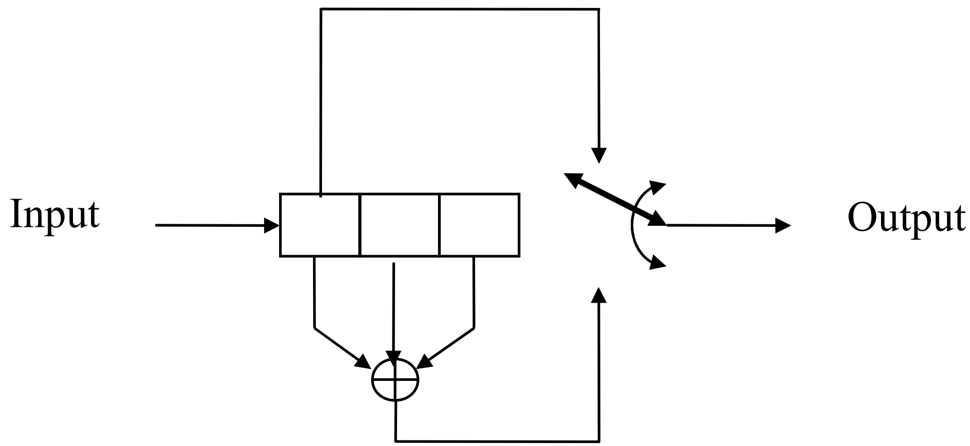
One path with weight 5, length 3 and data weight of 1

One path with weight 6, length 4 and data weight of 2

One path with weight 5, length 5 and data weight of 2

Systematic Convolutional codes

- A Conv. Coder at rate k/n is systematic if the k-input bits appear as part of the n-bits branch word.



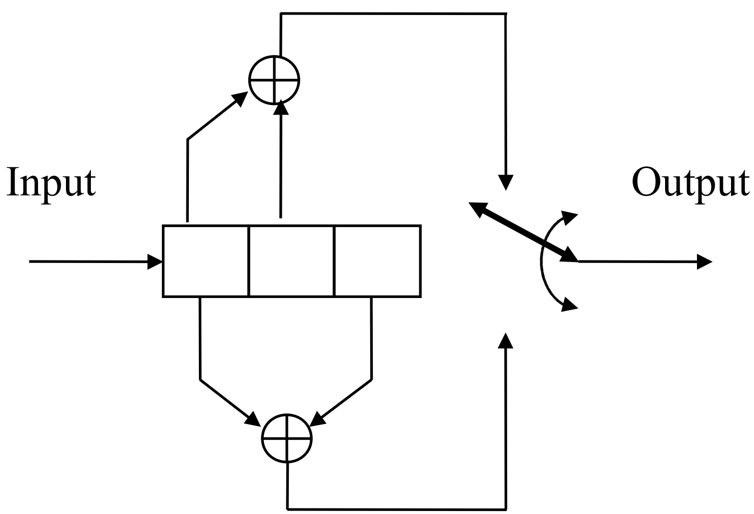
- Systematic codes in general have smaller free distance than non-systematic codes.

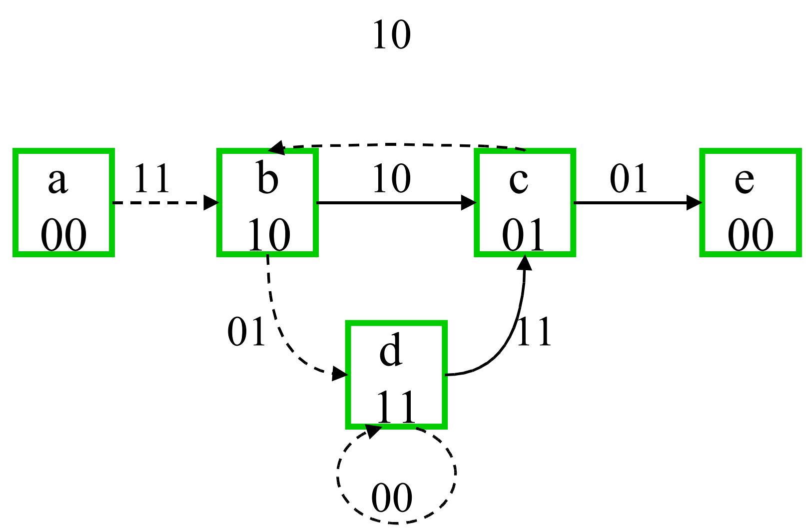
Catastrophic Convolutional codes

- Catastrophic error propagations in Conv. code:
  - A finite number of errors in the coded bits cause an infinite number of errors in the decoded data bits.
- A Convolutional code is catastrophic if there is a closed loop in the state diagram with zero weight.
- Systematic codes are not catastrophic:
  - At least one branch of output word is generated by input bits.
- Small fraction of non-systematic codes are catastrophic.

Catastrophic Conv. …

- Example of a catastrophic Conv. code:
  - Assume all-zero codeword is transmitted.
  - Three errors happens on the coded bits such that the decoder takes the wrong path abdd…ddce.
  - This path has 6 ones, no matter how many times stays in the loop at node d.
  - It results in many erroneous decoded data bits.



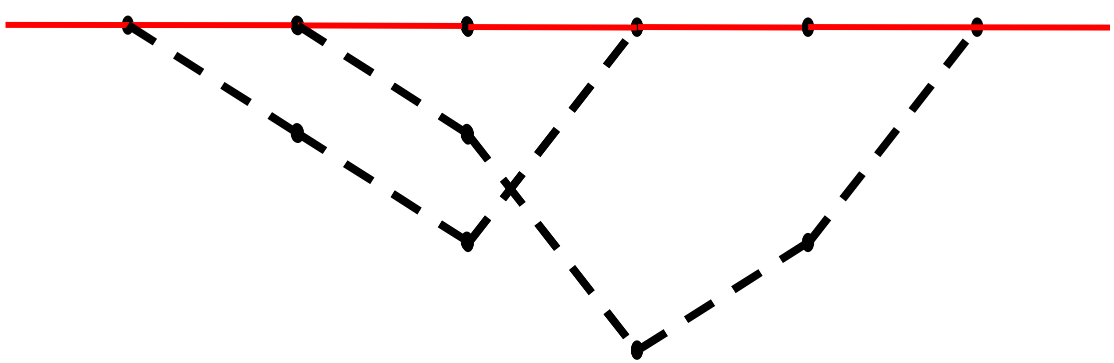


Performance bounds for Conv. codes

- Error performance of the Conv. codes is analyzed based on the average bit error probability (not the average codeword error probability), because
  - Codewords have variable sizes due to different sizes of the input.
  - For large blocks, codeword error probability may converge to one bit but the bit error probability may remain constant.
  - ….

Performance bounds …

- Analysis is based on:
  - Assuming the all-zero codeword is transmitted
  - Evaluating the probability of an “**error event**” (usually using bounds such as union bound).
    - An “error event” occurs at a time instant in the trellis if a non-zero path leaves the all-zero path and re-merges to it at a later time.



Performance bounds …

- Bounds on bit error probability for memoryless channels:
  - Hard-decision decoding:

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

  - Soft decision decoding on AWGN channels using BPSK

$$P_B \le Q\left(\sqrt{2d_f \frac{E_c}{N_0}}\right)\exp\left(d_f \frac{E_c}{N_0}\right)\left.\frac{dT(D,L,N)}{dN}\right|_{N=1,L=1,D=\exp(-E_c/N_0)}$$

Performance bounds …

- Error correction capability of Convolutional codes, given by $t = \lfloor (d_f - 1)/2 \rfloor$, depends on
  - If the decoding is performed long enough (within 3 to 5 times of the constraint length)
  - How the errors are distributed (bursty or random)
- For a given code rate, increasing the constraint length, usually increases the free distance.
- For a given constraint length, decreasing the coding rate, usually increases the free distance.
- The coding gain is upper bounded

$$\boxed{\text{coding gain} \leq 10 \log_{10}(R_c d_f)}$$

Performance bounds ...

- Basic coding gain (dB) for soft-decision Viterbi decoding

Uncoded E_b/N_0 (dB)	Code rate		1/3		1/2	
	P_B	K	7	8	6	7
6.8	10^{-3}		4.2	4.4	3.5	3.8
9.6	10^{-5}		5.7	5.9	4.6	5.1
11.3	10^{-7}		6.2	6.5	5.3	5.8
Upper bound			7.0	7.3	6.0	7.0

Interleaving

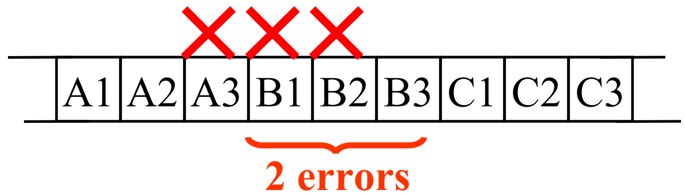
- Convolutional codes are suitable for memoryless channels with random error events.
- Some errors have bursty nature:
  - Statistical dependence among successive error events (time-correlation) due to the channel memory.
    - Like errors in multipath fading channels in wireless communications, errors due to the switching noise, …
- "Interleaving" makes the channel looks like as a memoryless channel at the decoder.

Interleaving …

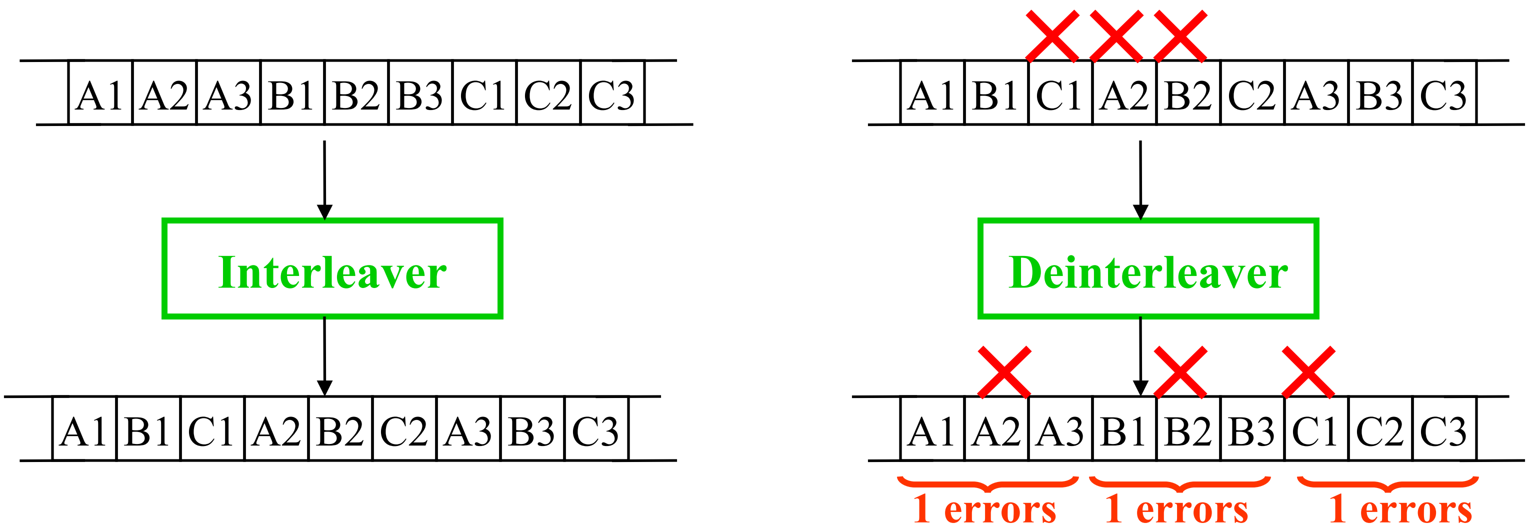
- Interleaving is achieved by spreading the coded symbols in time (interleaving) before transmission.
- The reverse in done at the receiver by deinterleaving the received sequence.
- “Interleaving” makes bursty errors look like random. Hence, Conv. codes can be used.
- Types of interleaving:
  - Block interleaving
  - Convolutional or cross interleaving

Interleaving …

- Consider a code with t=1 and 3 coded bits.
- A burst error of length 3 can not be corrected.

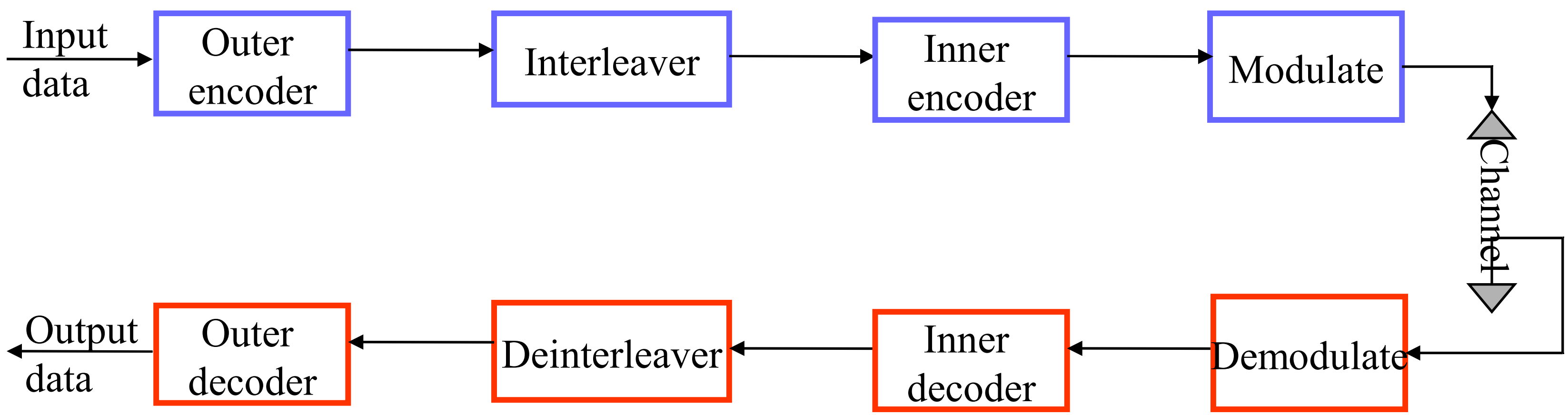


- Let us use a block interleaver 3X3



Concatenated codes

- A concatenated code uses two levels on coding, an inner code and an outer code (higher rate).
  - Popular concatenated codes: Convolutional codes with Viterbi decoding as the inner code and Reed-Solomon codes as the outer code
- The purpose is to reduce the overall complexity, yet achieving the required error performance.



Practical example: Compact disc

> "Without error correcting codes, digital audio would not be technically feasible."

- The channel in a CD playback system consists of a transmitting laser, a recorded disc and a photo-detector.
- Sources of errors are manufacturing damages, fingerprints or scratches
- Errors have bursty like nature.
- Error correction and concealment is achieved by using a concatenated error control scheme, called cross-interleaver Reed-Solomon code (CIRC).

Compact disc – cont'd

- CIRC encoder and decoder: