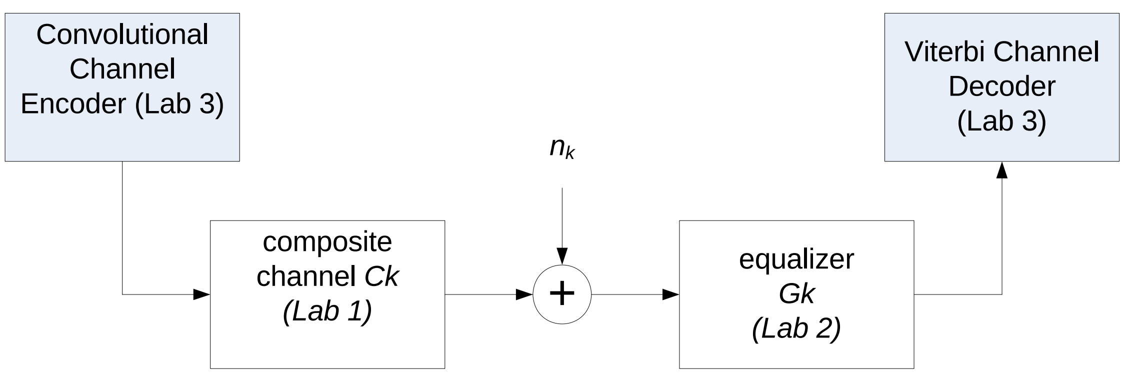
1 Convolutional Channel Coding

References:

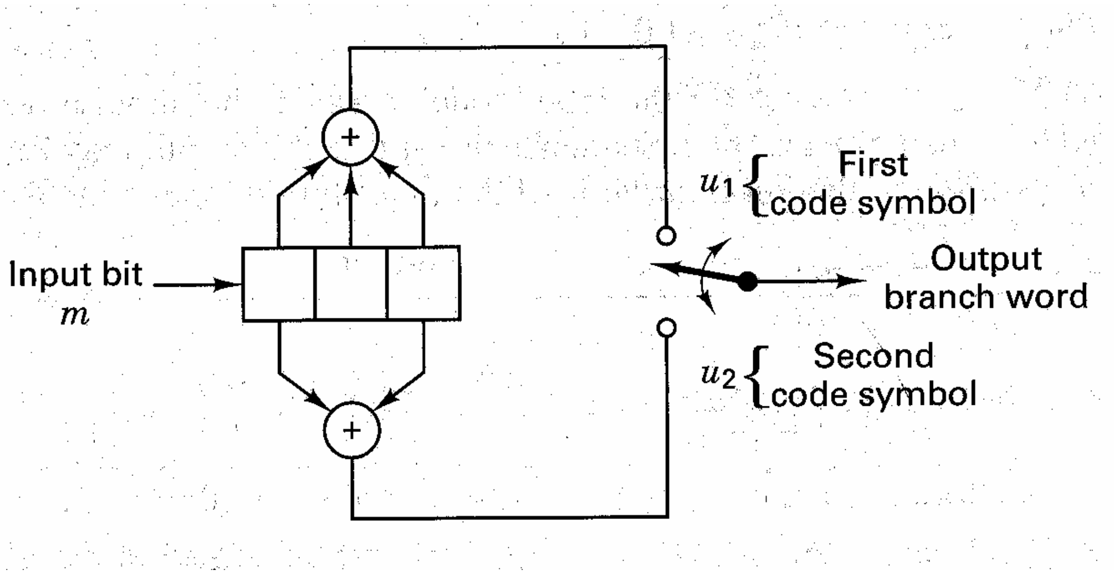
- Sklar, *Digital Communications*, Chapter 7, Sections 7.1 - 7.3
- Proakis, *Contemporary Communication Systems*, Chapter 8, Section 8.3.2

1.1 The diagram



- Channel coding provides additional protection against ISI and fading
- Code rate, k/n, measures the amount of added redundancy
- We shall consider only the most commonly used binary convolutional encoders with $k = 1$.

1.2 Channel encoder and impulse response

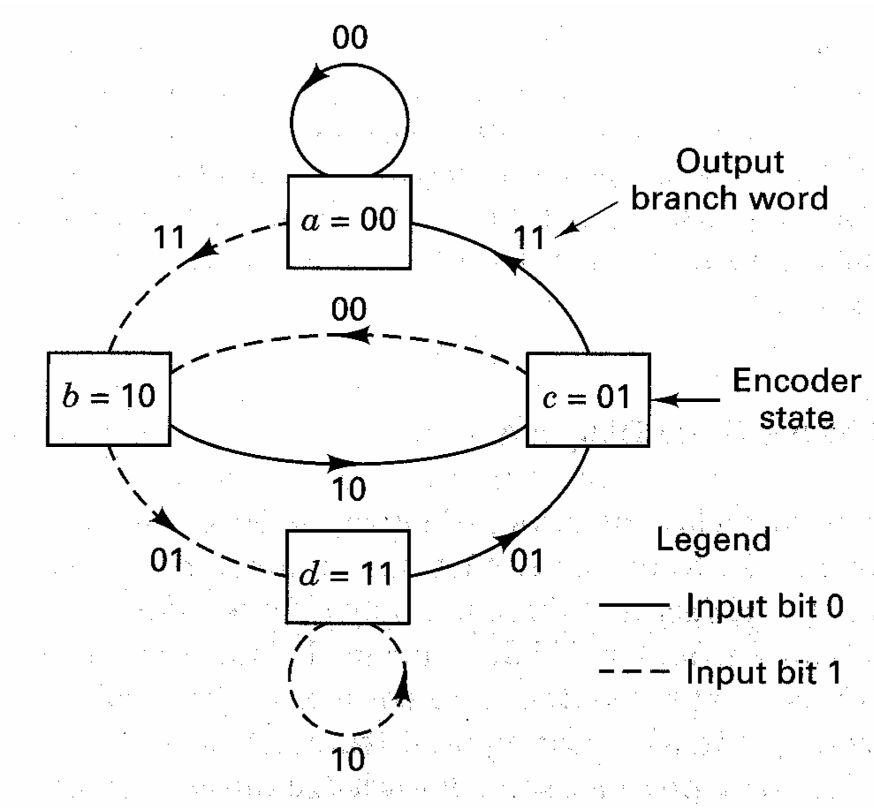


Rate 1/2 convolutional encoder

- $k = 1, n = 2, K = 3$
- K: constraint length (filter length); each output bit is not only a function of the current input bit, but also a function of the $K - 1$ bits that precede it.
- exclusive modulo-2 operators
- no block size in convolutional code (unlike block code)
- Connection vectors: $\mathbf{g}_1 = [\,1\ \ 1\ \ 1\,]$, and $\mathbf{g}_2 = [\,1\ \ 0\ \ 1\,]$
- impulse response: the response of the encoder to a single 1-bit that moves through it.

Example

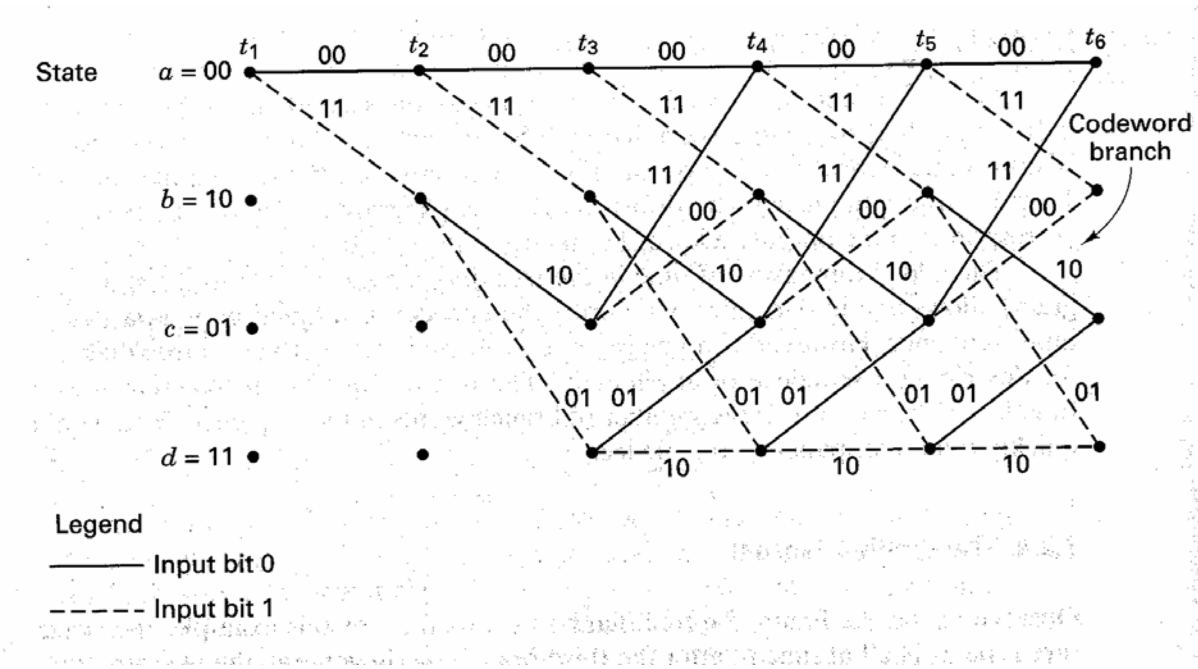
1.3 State diagram



Rate 1/2 encoder state diagram

- State: the contents of the rightmost $K-1$ stages (thus 2^{K-1} states)
- Knowledge of the current state together with knowledge of the next input determine the next output
- Two transitions, corresponding to the two possible input bits, from each state
- each transition generates n output bits.
- transitions are not arbitrary

1.4 Tree diagram and trellis diagram



Encoder trellis diagram

- Trellis and tree diagram provides the dimension of time to the state diagram
- A solid line denotes the output generated by an input bit 0, and a dashed line denotes the output generated by an input bit 1.
- Each note of the trellis characterizes a state - there are 2^{K-1} states at each unit of time
- Each of the states can be entered from either of two preceding states; each of the states can transition to one of two states
- output words (coded bits) appear as labels of the trellis branches

1.5 Maximum likelihood decoding

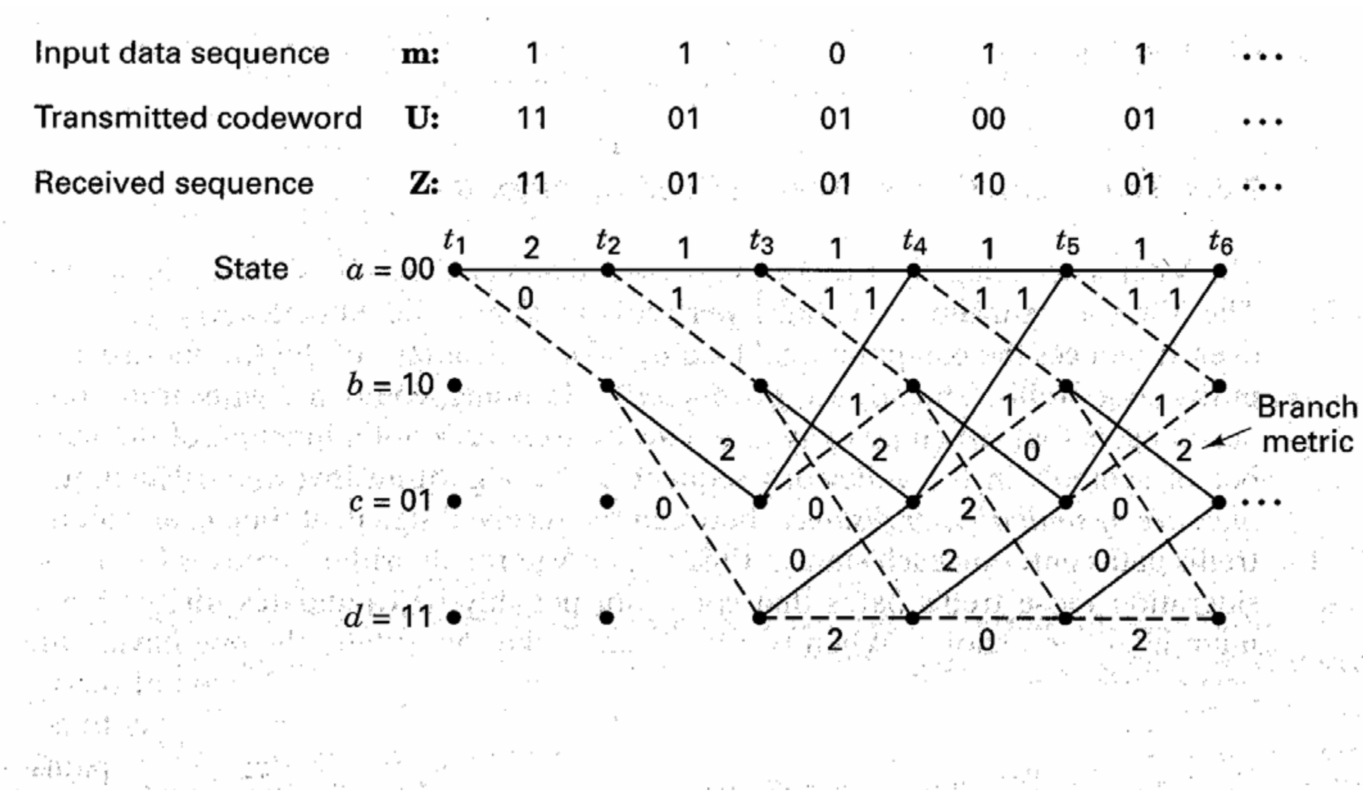
$$\text{Encoding} \; : \; m \Longrightarrow \mathbf{U}$$
$$\text{Received sequence (binary, after detection)} \; : \; \mathbf{Z} = \mathbf{U} + n$$

- Since the encoded bits are no longer isolated (i.e., the convolutional code has memory), the maximum likelihood decoder maximizes the likelihood function over the entire sequence:

$$m' = \arg_m \max P(\mathbf{Z}|\mathbf{U}^{(m)})$$

- MA decoding = finding the codeword $\mathbf{U}^{(m')}$ that is the closest in *Hamming distance* to $\mathbf{Z}$.
- However, there are 2^L possible sequences in an L branch word!

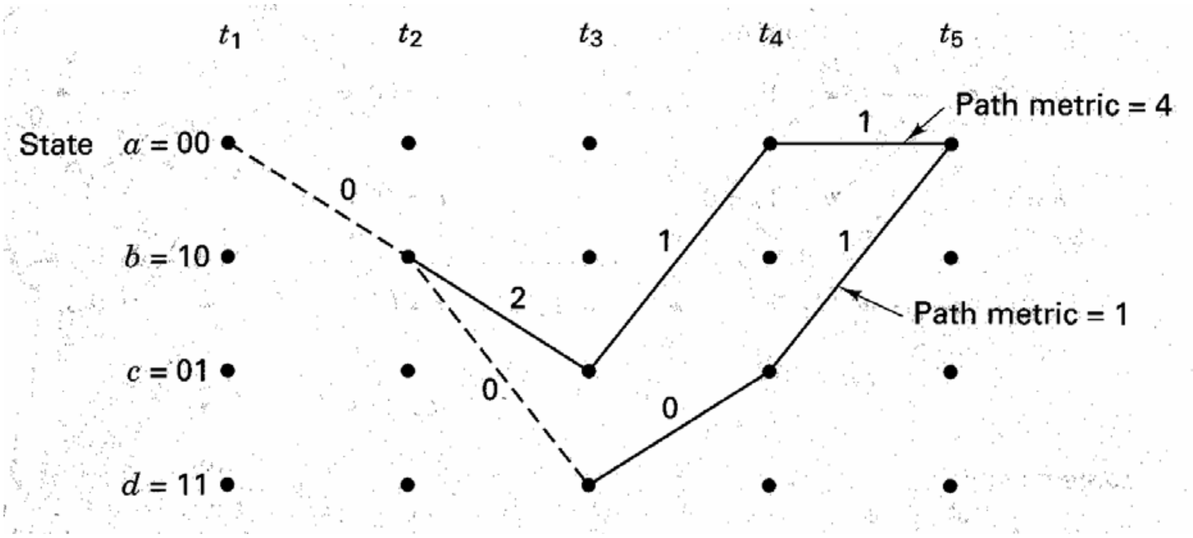
1.6 The Viterbi algorithm



Decoder trellis diagram

- The Viterbi algorithm reduces the decoding cost by taking advantage of the trellis structure
- Complexity reduced from $O(2^L)$ to $O(L)$
- When two paths enter the same state, the one having the best metric is chosen (the *surviving path*)
- The selection of surviving paths is performed for all states (2^{K-1}), at each time
- The Viterbi algorithm is a true ML decoder

1.7 Path metric



Merging paths

- Each branch of the decoder trellis is labeled with the *Hamming distance* between the received code symbols and the branch code word for that time interval.
- The *cumulative Hamming path metric* at a given time is the sum of the branch Hamming distance along that path up to time t_i
- When two paths merge to a single state, the one with a higher cumulative Hamming path metric is eliminated (why?)
- The winning path metric for each state is the *state metric* for that state at time t_i.There are no more winning paths than the number of states at each time.
- If equal value, one arbitrary path is eliminated.
- Decoding delay can be five times the constraint length: $5K$

Example