EE/Ma 127b Error-Correcting Codes - Homework Assignment 4

Ling Li, ling@cs.caltech.edu

March 1, 2001

4.1 From the diagram, we get (note that $s_1(i+1) = u(i) + u(i) + s_2(i) = s_2(i)$)

$$\begin{aligned} \boldsymbol{s}(i+1) &= (s_2(i), s_1(i) + u(i)) = \boldsymbol{s}(i) \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + u(i) \begin{pmatrix} 0 & 1 \end{pmatrix}, \\ \boldsymbol{x}(i) &= (u(i), u(i) + s_2(i)) = \boldsymbol{s}(i) \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} + u(i) \begin{pmatrix} 1 & 1 \end{pmatrix}. \end{aligned}$$

(a) The state-space representation $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$ is

$$\mathcal{A} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \mathcal{B} = \begin{pmatrix} 0 & 1 \end{pmatrix}, \mathcal{C} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, \mathcal{D} = \begin{pmatrix} 1 & 1 \end{pmatrix}.$$

(b) $E(D) = \mathcal{B}(D^{-1}I_2 - \mathcal{A})^{-1} = \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} D^{-1} & -1 \\ -1 & D^{-1} \end{pmatrix}^{-1} = \begin{pmatrix} \frac{D^2}{1+D^2} & \frac{D}{1+D^2} \end{pmatrix}$. The generator matrix of this encoder is

$$\begin{aligned} G(D) &= \mathcal{D} + E(D)\mathcal{C} = \begin{pmatrix} 1 & 1 \end{pmatrix} + \begin{pmatrix} \frac{D^2}{1+D^2} & \frac{D}{1+D^2} \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 1 & \frac{1+D+D^2}{1+D^2} \end{pmatrix}. \end{aligned}$$

(c) The Laurent series of the input is $U(D) = 1$. Thus the Laurent series of the output is

$$\boldsymbol{X}(D) = U(D)G(D) = \begin{pmatrix} 1 & \frac{1+D+D^2}{1+D^2} \end{pmatrix}.$$

Notice that

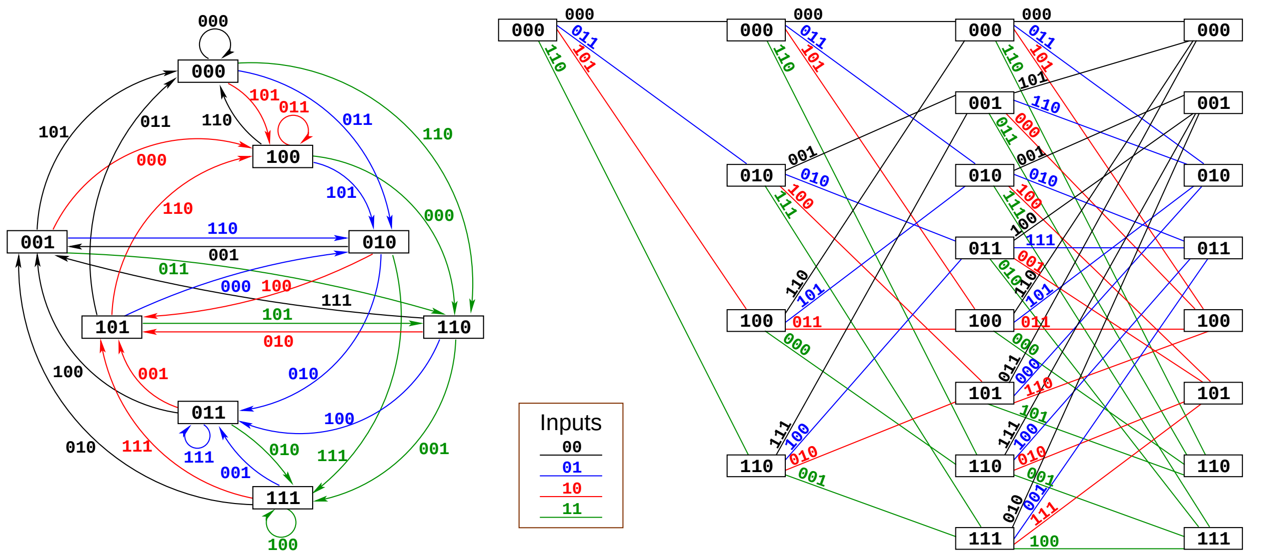
$$\frac{1+D+D^2}{1+D^2} = 1 + \frac{D}{1+D^2} = 1 + D(1 + D^2 + D^4 + \cdots) = 1 + D + D^3 + D^5 + \cdots.$$

So the output is $(11, 01, 00, 01, 00, 01, \ldots)$.

4.2 Let $\boldsymbol{s} = (s_1, s_2, s_3)$, where s_1 is the register for u_1 and s_2, s_3 are registers for u_2.

(a) From the circuit, we obtain

$$\begin{aligned} \boldsymbol{s}(t+1) &= (u_1(t), u_2(t), s_2(t)) = \boldsymbol{s}(t) \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} + \boldsymbol{u}(t) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \\ \boldsymbol{v}(t) &= (u_1(t) + s_1(t) + s_3(t), u_2(t) + s_1(t), u_1(t) + u_2(t) + s_2(t) + s_3(t)) \\ &= \boldsymbol{s}(t) \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} + \boldsymbol{u}(t) \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}. \end{aligned}$$



(a) State diagram (b) Trellis diagram

Figure 1: State and trellis diagrams of a binary rate $R = 2/3$ convolutional code. (Problem 2)

Thus we can get the state diagram and the trellis digram in Figure 1.

(b) By the G_0, G_1, G_2 given in the problem figure, the generator matrix is

$$\begin{aligned} G(D) &= G_0 + G_1 D + G_2 D^2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} D + \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} D^2 \\ &= \begin{pmatrix} 1+D & D & 1 \\ D^2 & 1 & 1+D+D^2 \end{pmatrix}. \end{aligned}$$

The "scalar" generator matrix G_{scalar} is

$$\begin{aligned} G_{\text{scalar}} &= \begin{pmatrix} G_0 & G_1 & G_2 & & \\ & G_0 & G_1 & G_2 & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & & & & \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & & & & \\ & & & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \\ & & & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & \\ & & & & & & & \ddots & & & \ddots & & \ddots \end{pmatrix} \end{aligned}$$

(c) $\boldsymbol{u} = (10, 11, 01, 10, 00, 00, \dots)$. From the trellis diagram, $\boldsymbol{v} = (101, 000, 100, 001, 011, 000, \dots)$.

4.3 *Bit metrics.* Let x stand for the transmitted symbol and y for the received symbol.

(a) The conditional probabilities and the logarithm of them are

$p(y\|x)$	$y=0$	$y=1$
$x=0$	0.9	0.1
$x=1$	0.3	0.7

$\log_2 p(y\|x)$	$y=0$	$y=1$
$x=0$	-0.1520	-3.3219
$x=1$	-1.1370	-0.5146

Using $M(y|x) = [4 \log_3 2 \cdot (\log_2 p(y|x) + \log_2 10)]$, where $[\cdot]$ is the round function, we get an efficient set of bit metrics

$M(y\|x)$	$y=0$	$y=1$
$x=0$	8	4
$x=1$	0	7

(b) Using $M(y|x) = [3(\log_2 p(y|x) + \log_2 10)]$, we get

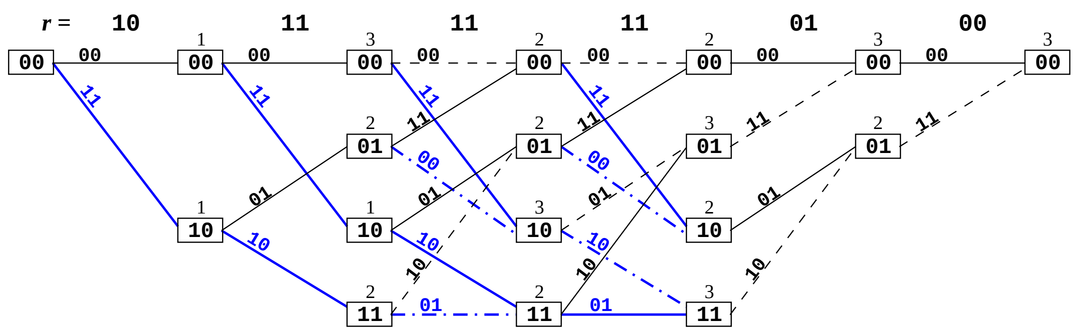
$p(y\|x)$	$y=0$	$y=E$	$y=1$
$x=0$	0.6	0.3	0.1
$x=1$	0.2	0.3	0.5

$M(y\|x)$	$y=0$	$y=E$	$y=1$
$x=0$	8	5	0
$x=1$	3	5	7

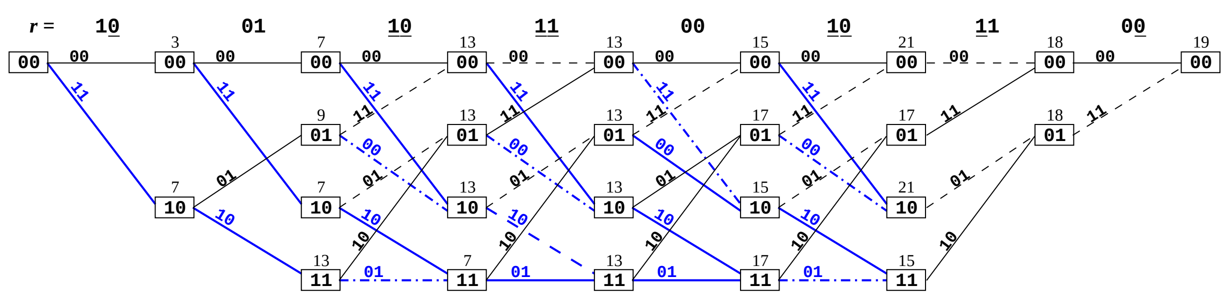
4.4 *Viterbi decoding.* Figure 2 shows the Viterbi decodings for this problem. We can find the ML path by starting at the rightmost state and tracing back along the surviving paths, and the corresponding codeword is the decoded codeword.

(a) From Figure 2(a), the decoded codeword is $(00, 11, 01, 11, 00, 00)$. 3 errors corrected.

(b) In Figure 2(b), two ties occur (those with two solid incoming lines) during the decoding. And thus there are two ML paths. The codeword with maximum likelyhood can be either $(00, 00, 00, 11, 10, 10, 11, 00)$ or $(00, 11, 10, 01, 01, 10, 11, 00)$.



(a) Hard-decision Viterbi decoding



(b) Soft-decision Viterbi decoding

Figure 2: Viterbi decodings. The thin (solid/dashed) lines stand for input 0 and the thick blue (solid/dash-dot) lines stand for input 1. Each node is labeled with its value computed by the Viterbi algorithm. Nonsurviving paths are denoted by dashed or dash-dot lines. (Problem 4)