Dhirubhai Ambani
Institute of Information and Communication Technology

CT 216

INTRODUCTION TO COMMUNICATION SYSTEMS

Convolutional Code

Prof:

YASH M. VASAVADA

Group Members:

AADITYA PITHADIYA : 202201224
MITUL SUDANI : 202201244
HARSH BOSAMIYA : 202201243
KAVAN KANSODARIYA : 202201223
SMIT FEFAR : 202201253
HEMAL RAVRANI : 202201235
AASTHA BHAVSAR :202201259
RHYTHM PANCHAL : 202201230
FENIL AMIN : 202201236
MARMIK VASAVA : 202201252
YASH CHAUDHARI : 202201229

- **Probability of error for soft decision decoding:**

  In this section, we will try to estimate an upper bound for the error probability with soft decision Viterbi decoding.

  For the simplicity of the derivation, we assume that all zero sequences are transmitted and we determine the probability of error in deciding in favor of another sequence.

  We define $c_{jm}^{(i)}$ as the coded binary digit for the m^{th} bit for the j^th branch of the i^{th} path in the trellis for convolution code, and r_{jm} as the output of the demodulator for the corresponding m^{th} bit of the j^{th} branch, where r_{jm} is:

$$r_{jm} = \sqrt{\varepsilon_c}\,(2c_{jm} - 1) + n_{jm}$$

  Where n_{jm} is represented as AWGN with mean 0 and variance 1, and ε_c is the transmitted signal energy for each code bit.

  The branch metric is defined for the j^{th} branch of the i^{th} path through the trellis as the logarithm of the joint probability of the sequence $\{r_{jm}, m = 1, 2, 3\}$ conditioned on the transmitted sequence $\{c_{jm}, m = 1, 2, 3\}$ for the i^{th} path.

$$\mu_j^{(i)} = \log p\left(Y_j/C_j^{(i)}\right)$$

  Furthermore, a metric for the i^{th} path consisting of B branches through the trellis is defined as:

$$PM^{(i)} = \sum_{j=1}^{B} \mu_j^{(i)}$$

  For the soft decision decoding, the channel adds AWGN to the signal.Then the demodulator output is described statistically by the probability density function

$$P(r_{jm}/c_{jm}^{(i)}) = P(n_{jm} = r_{jm} - \sqrt{\varepsilon_c}(2c_{jm}^{(i)} - 1))$$

Probability of getting $r_{jm}^{(i)}$ given $c_{jm}^{(i)}$ is equal to probability of noise equal to r$_{jm} = \sqrt{\varepsilon_c}\,(2c_{jm} - 1) + n_{jm}$

$$P\left(r_{jm}/c_{jm}^{(i)}\right) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{\left[r_{jm} - \sqrt{\varepsilon_c}(2c_{jm}^{(i)} - 1)\right]^2}{2\sigma^2}\right)}$$

Since noise is normally distributed, we can directly compute it using the formula for the probability density function of a normally distributed variable.

- **correlation metric:**
  Correlation metric for path is defined as sum of branch metric of all branches of that path.

$$CM^{(i)} = \sum_{j=1}^{B} \mu_j^{(i)} = \sum_{j=1}^{B} \sum_{m=1}^{n} r_{jm} \left(2c_{jm}^{(i)} - 1\right)$$

  Suppose i=0 is all zero path and i=1 is another path that merge with all zero path after some transitions.

- **First event error probability:**
  We define the first-event error probability as the probability that another path merges with all zero path at node B has metric that exceeds the metric of the all zero path for the first time. Suppose the incorrect path call it i=1, that merges with the all-zero path differs from the all-zero path in d-bits.

  We define pairwise error probability as probability of getting as,

$$P_2(d) = P(CM^{(1)} \geq CM^{(0)})$$

$$P_2(d) = P(CM^{(1)} - CM^{(0)} \geq 0)$$

$$P_2(d) = P[2 \sum_{j=1}^{B} \sum_{m=1}^{n} r_{jm}(c_{jm}^{(1)} - c_{jm}^{(0)} \geq 0)]$$

  Since the coded bits in two paths are identical except in d positions only d terms will left

$$P_2(d) = P(\sum_{l=1}^{d} r_l^{'} \geq 0)$$

  The $r_l^{'}$ are the independent and identically distributed gaussian radnom variables with the mean $-\sqrt{\varepsilon_c}$ and variance $N_0/2$.

$$P_2(d) = P(X \geq 0)$$

  Let random variable,
  X= $\mathrm{r}_1^{'} + r_2^{'} + r_3^{'} + ... + r_d^{'}$
  $\mu(X) = \sum_{i=1}^{d} \mu(r_i^{'})$

$$\mu(X) = \sum_{i=1}^{d} \mu(\sqrt{\varepsilon_c}\,(2c_i - 1) + n_i)$$

$c_i = 0$

$$\mu(X) = \sum_{i=1}^{d} \mu(-\sqrt{\varepsilon_c} + n_i)$$

$$\mu(X) = \sum_{i=1}^{d} \mu(-\sqrt{\varepsilon_c}) + \sum_{i=1}^{d} \mu(n_i)$$

since mean of $n_i = 0$

$$\mu(X) = \sum_{i=1}^{d} \mu(-\sqrt{\varepsilon_c}) = -d\sqrt{\varepsilon_c}$$

$$\sigma^2(X) = \sigma^2(\sum_{i=1}^{d} r_i)$$

$$\sum_{i=1}^{d} \sigma^2(r_i) = d\sigma^2(r_i)$$

$$\sigma(X) = \sqrt{d\sigma^2(r_i)}$$

$$P_2(d) = P(X \geq 0)$$

$$P\left(\frac{X - \varepsilon_c}{\sigma} \geq -\frac{\varepsilon_c}{\sigma}\right) = P(Z \geq \frac{d\sqrt{\varepsilon_c}}{\sqrt{d}\sigma})$$

$$= P(Z \geq \sqrt{\frac{2d\varepsilon_c}{N_0}}) = P(Z \geq \sqrt{2d\gamma_b R_c})$$

$$P_2(d) = Q(\sqrt{2d\gamma_b R_c})$$

Where γ_b= ε_b/N_0 is received SNR per bit and R_c is code rate
There are many possible paths with different distances that merges with the all zero path at a given node B. We can sum the pairwise error probability over all paths to obtain an upper bound on the first-event error probability in the form

$$P_e \leq \sum_{d=d_{free}}^{\infty} a_d P_2(d)$$

$$P_e \leq \sum_{d=d_{free}}^{\infty} a_d Q(\sqrt{2\gamma_b R_c d})$$

Where a_d denotes the number of paths of distances d from the all-zero path that merge with the all-zero path for the first time.
We can upper bound Q function using Chernoff bound:

$$\begin{aligned}
P(X \geq a) &\leq e^{-at} M_x(t), t > 0 \\
&\leq e^{-at} e^{\mu t + \frac{\sigma^2 t^2}{2}} \\
&\leq e^{-at} e^{\frac{\sigma^2 t^2}{2}} \\
&\leq e^{-at} e^{\frac{t^2}{2}} \\
\mathbf{Q(a) = P(X \geq a)} &\leq \mathbf{e^{-\frac{a^2}{2}}}
\end{aligned}$$

$$Q(\sqrt{2\gamma_b R_c d}) \leq e^{-\gamma_b R_c d}$$

Now,

$$Q(\sqrt{2\gamma_b R_c d}) \leq e^{-\gamma_b R_c d} = D^d \Big|_{D=e^{-\gamma_b R_c}}$$

$$P_e < T(D) \Big|_{D=e^{-\gamma_b R_c}}$$

- **Bit error probability**
  If we multiply the pairwise error probability $P_2(d)$ by the number incorrectly decoded information bits for the incorrect path at the node where they merge we obtain the bit error rate for that path. We know that the exponent in the factor N contained in the transfer function T(D,N) indicate the number of information bit errors in the selecting an incorrect path that merges with all-zero path at some node B.

$$T(D,N) = \sum_{d=d_{free}}^{\infty} a_d D^d N^{f(d)}$$

$$\left.\frac{dT(D,N)}{dN}\right|_{N=1} = \sum_{d=d_{free}}^{\infty} a_d f(d) D^d$$

$$= \sum_{d=d_{free}}^{\infty} \beta_d D^d$$

$$P_b < \sum_{d=d_{free}}^{\infty} \beta_d P_2(d)$$

- **Hard decision decoding**

  We saw how to find first event error probability in SDD. Similarly we can find first event probability for Hard Decision Decoding.

- **Bitwise error probability**

  We assume that we transmit all-zero path.
  Suppose i=0 is all zero path and i=1 is another path that merge with all zero path after some transitions.

  Path i=1 contains d ones.
  So, if d is odd, path 1 will be chosen over path 0 if there are more than (d+1)/2 ones.

$$P_2(d) = \sum_{k=\frac{d+1}{2}}^{d} \begin{pmatrix} d \\ k \end{pmatrix} p^k (1-p)^{d-k}$$

  and if d is even, path 1 will be chosen over path 0 if there are more than d/2 ones and there is half probability that path 1 will be chosen when the number of 1's equal to d/2.

$$P_2(d) = \sum_{k=\frac{d}{2}+1}^{d} \begin{pmatrix} d \\ k \end{pmatrix} p^k (1-p)^{d-k} + \frac{1}{2} \begin{pmatrix} d \\ \frac{1}{2}d \end{pmatrix} p^{d/2} (1-p)^{d/2}$$

  There are many possible paths with different distances that merge with all-zero path at given node. Therefore, there is no simple exact expression for the first-event error probability. However, we can provide upper bound as we did in SDD.

$$P_e < \sum_{d=d_{free}}^{\infty} a_d P_2(d)$$

  We can provide upper bound of $\mathrm{P}_2(d)$:

$$P_2(d) = \sum_{e=\frac{d+1}{2}}^{d} \begin{pmatrix} d \\ e \end{pmatrix} p^e (1-p)^{d-e}$$

$$P_2(d) \leq \sum_{e=\frac{d+1}{2}}^{d} \begin{pmatrix} d \\ e \end{pmatrix} p^{\frac{d}{2}} (1-p)^{\frac{d}{2}}$$

$$P_2(d) = p^{\frac{d}{2}} (1-p)^{\frac{d}{2}} \sum_{e=\frac{d+1}{2}}^{d} \begin{pmatrix} d \\ e \end{pmatrix}$$

$$P_2(d) < p^{\frac{d}{2}}(1-p)^{\frac{d}{2}} \sum_{e=0}^{d} \begin{pmatrix} d \\ e \end{pmatrix}$$

$$= 2^d (p)^{\frac{d}{2}}(1-p)^{\frac{d}{2}}$$

$$P_2(d) < 2^d (p)^{\frac{d}{2}}(1-p)^{\frac{d}{2}}$$

$$P_2(d) < [4p(1-p)]^{\frac{d}{2}}$$

$$P_e < \sum_{d=d_{free}}^{\infty} a_d [4p(1-p)]^{\frac{d}{2}}$$

$$P_e < \sum_{d=d_{free}}^{\infty} T(D)\bigg|_{D=\sqrt{4p(1-p)}}$$

If we multiply the pairwise error probability P2(d) by the number incorrectly decoded information bits for the incorrect path at the node where they merge we obtain the bit error rate for that path. We will use the transfer function to get information about the number of incorrectly decoded bits.

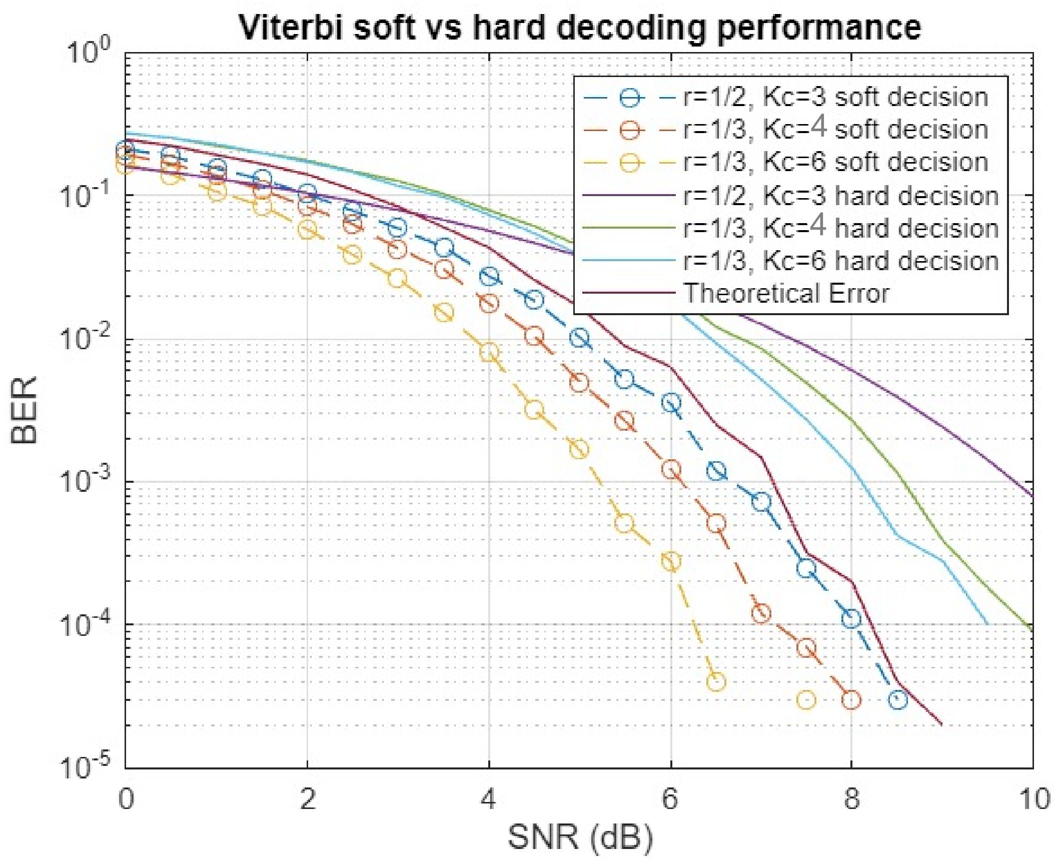
$$P_b < \sum_{d=d_{free}}^{\infty} \beta_d P_2(d)$$

$$P_b < \frac{dT(D,N)}{dN}\bigg|_{N=1, D=\sqrt{4p(1-p)}}$$

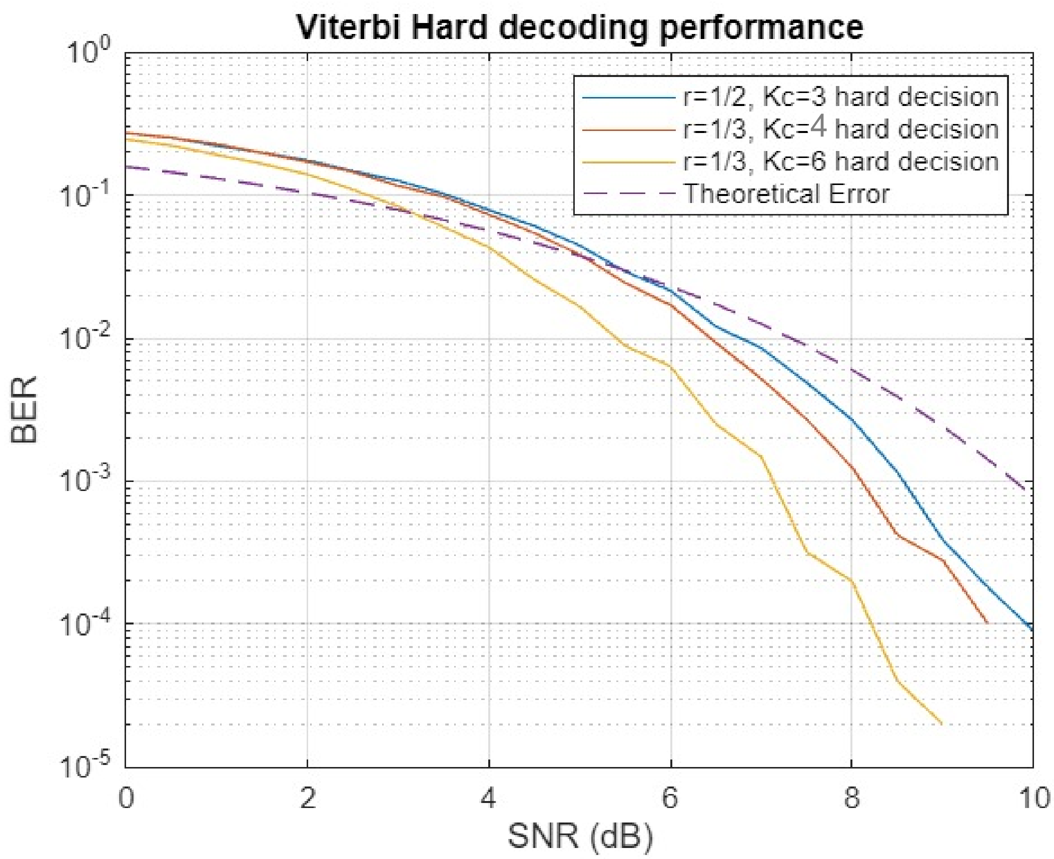
Conclusion

By increasing the number of output bits n, we introduce more redundancy in the encoded message and hence reduce the BER, but beyond a certain point, decreasing R (Code Rate) too much can lead to inefficiency in the use of the channel and bandwidth. As the memory of the register increases, more number of previous input bits is used to encode each new bit. This additional information increases the accuracy and decreases BER for a given value of code rate for the encoded data. With higher K_c, the decoder can better detect and correct errors introduced during transmission due to noise, leading to a lower BER at a given signal-to-noise ratio (SNR). As K_c increases, the complexity of both encoding and decoding processes also increases significantly. By taking the Euclidean distance between the expected and received voltages, we also take into account the information contained in the received bit where as in the hard decoding where we use a threshold voltage value to map received voltages to bits and therefore there is lose of information during the conversion. Therefore soft decision decoding provides better path metric and therefore is able to perform better than hard decoding for any given value of code rate and constraint length.

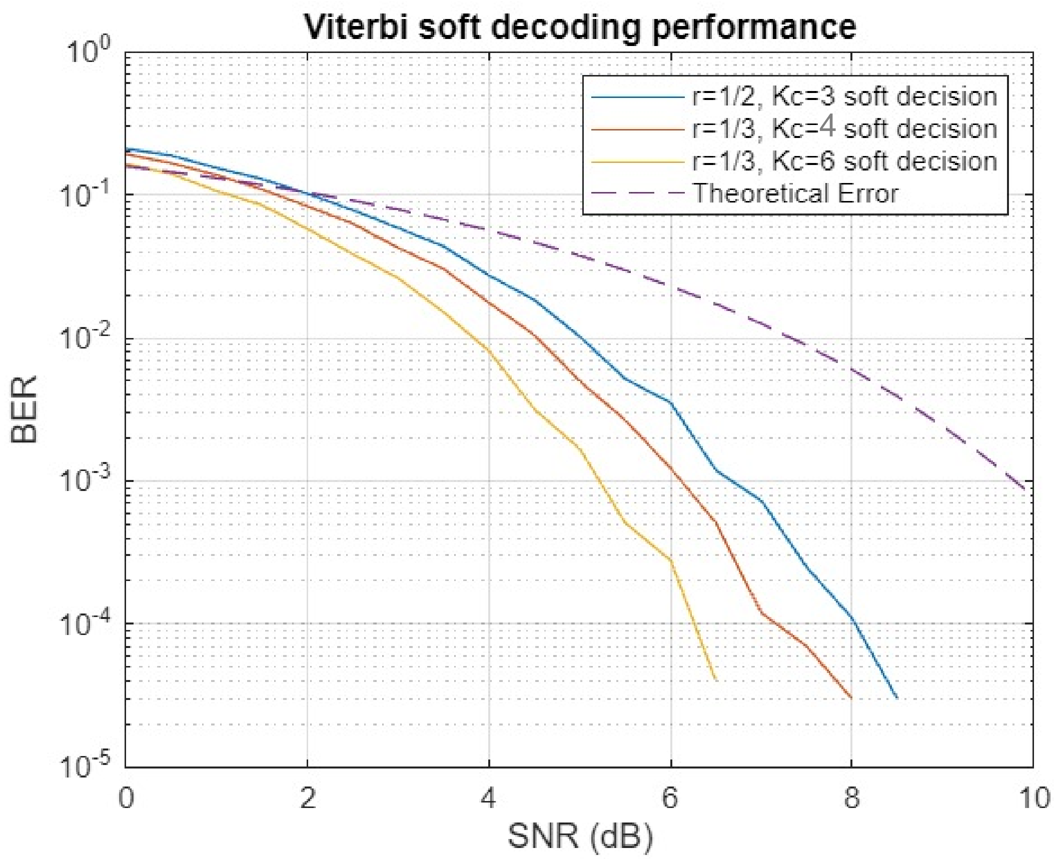
Simulations for the Viterbi soft vs hard decoding



Simulations for the Viterbi hard decoding



Simulations for the Viterbi soft decoding



Bibliography

[1] J. G. Proakis. Digital Communications. 4th edition. McGraw-Hill, 1995.

[2] Dasari Sai Bhargav. Viterbi Decoding for Convolutional Codes.

[3] MIT Opencourseware. PDF.

[4] NPTEL. Youtube videos.

[5] OpenAI. (January 2022). GPT (3.5) [Computer software].