On a Technique to Calculate the Exact Performance of a Convolutional Code

Peter C. Fisherburn,A. R. Calderbank,Member, IEEE,

and Daniel J. Costello, Jr., Fellow, IEEE

IEEE TRANSACTIONS ON INFORMATION THEORY,

VOL 41, NO. 2,PP. 441-447, March 1995

Prepared by

Chun-Yi Wu

g895609

Department of Institute of Communications Engineering

National Tsing Hua University, Taiwan 300, R.O.C.

JAN 8, 2001

Outline

- Introduction
- Key idea
- Derivation
- Simulation
- Conclusion

Introduction

Convolutional code is a set of code sequences generated by finite-state machine whose states defined a trellis and allow efficient maximum-likelihood decoding techniques such as the Viterbi algorithm.

- Convolutional codes were first introduced by Elias in 1955. He showed that redundancy can be introduced into a data stream through the use of linear shifter.He also showed that the resulting codes were very good when randomly chosen, This result was very interesting, for it correlated with Shanoon's more theoretical work showing that there exist randomly selectrf codes that, on the average, provide arbitrarily high levels of reliability given data transmission at a rate less than the channel capacity.

add the structure of convolutional code

Key idea

- A fundamental question of any error control scheme is its probability of error. The probability of error determines the performances of the coding scheme. For convolutional codes, there are several ways of measuring the probability of error. The easiest to understand is the bit-error probability.
- The performance of a convolutional code can be evaluated by computing.
- A Markovian technique is described to calculate the exact performance of the Viterbi algorithm algorithm used as either a channel decoder or a source encoder for a convolutional code.

Derivation

The associated Markov chain consists of the one-step trajectories of the Viterbi algorithm. Its states are the possible metric vectors.
A potential problem is encountered by the Viterbi algorithm when a tie occurs at a given vertex, that is, when two or more paths produces the same metrics at a given time unit.

We consider three type of fair tie-breaking rules.

C1) lexicographic tie -breaker

C2) anti-lexicographic tie-breaker

C3) the coin-fair tie-breaker

The probability is called Markovian tie-breaking rule that only considers the proceeding vertex.

The Viterbi Algorithm's Markov Chain

- Hamming distance: d_H
- be the received sequence of length N: R^N
- vertex: γ
- be the maximum-likelihood path: A_γ^N
- metric: D_γ^N

$$D_\gamma^N = d_H(A_\gamma^N, R^N) \tag{1}$$

- relative metric: $\bar{D}_{\gamma}^{N}$

$$\bar{D}_{\gamma}^{N} = D_{\gamma}^{N} - \underbrace{min}(D_{\delta}^{N}) \tag{2}$$

The metric vector at time unit N is

$$\bar{D}^{N} = (\bar{D}_{0}^{N}, \bar{D}_{1}^{N}, \cdots, \bar{D}_{2^{\upsilon}-1}^{N}) \tag{3}$$

C1) Fig 2. shows the trellis diagram of the rate 1/2, 2-state convolutional code with generator matrix $[1, 1+D]$.

C2) For this code, the Hamming distance between a branch label and a channel output is at most 2, so that the possible metric vectors or states of the Markov chain are $(2,0),(1,0),(0,0,),(0,1),(0,2)$

C3) The conditional probability of this received signal defined the transition probability matrix T for the Markov chain.

$$T = \begin{bmatrix} (1-p)^2 & 0 & 2p(1-p) & 0 & p^2 \\ (1-p)^2 & 0 & 2p(1-p) & 0 & p^2 \\ 0 & 1-p & 0 & p & 0 \\ p(1-p) & 0 & p^2+(1-p)^2 & 0 & p(1-p) \\ p(1-p) & 0 & p^2+(1-p)^2 & 0 & p(1-p) \end{bmatrix}$$

put the chart....Markovian chart state diagrams

$$\pi = \frac{1}{1+3p^2-2p^3}\begin{pmatrix} 1-4p+8p^2-7p^3+2p^4 \\ 2p-5p^2+5p^3-2p^4 \\ 2p-3p^2+2p^3 \\ 2p^2-3p^3+2p^4 \\ p^2+p^3-2p^4 \end{pmatrix}$$

Exact Calculation Of Information Bit Error Probability

let $P(\gamma^l, \bar{D}) = i/k$, If i information bits are decoded incorrectly, $i = 0, 1, \cdots, k$. Averaging over all metric states and received sequences, the probability of error can then be expressed as

$$P_e = \sum_{\bar{D},\gamma^l} \pi_{\bar{D}} q_{\gamma^l} P(\gamma^l, \bar{D}) \tag{4}$$

where q_{γ^l} is the probability of receiving sequence γ^l.

If we use the lexicographic tie-breaker for rate 1/2, 2-state code..

$$P_e = 2p(1-p)\pi_2 + \pi 3 + \pi 4 = \frac{7p^2 - 12p^3 + 10p^4 - 4p^5}{1 + 3p^2 - 2p^3} \tag{5}$$

If use anti-lexicographic tie-breaker...

$$P_e = p(1-p)\pi 1 + 2p(1-p)\pi_2 + (4p^3 - 7p^2 + 4p)\pi 3 + \pi 4 \tag{6}$$

$$= \frac{p^2(7 - 8p - 8p^2 + 26p^3 - 24p^4 + 8p^5)}{1 + 3p^2 - 2p^3} \tag{7}$$

If use coin-fair tie-breaker...

$$
\begin{aligned}
P(1) &= \frac{p(1-p)}{2} &(8)\\
P(2) &= \frac{4p(1-p)}{2-p+4p^2-4p^3} &(9)\\
P(3) &= \frac{2+7p-12p^2+13p^3-12p^4+4p^5}{2(2-p+4p^2-4p^3)} &(10)\\
P_e &= P(1)\pi 1+P(2)\pi 2+P(3)\pi 3+\pi 4 &(11)\\
&= \frac{p^2(14-23p+16p^2+2p^3-16p^4+8p^5)}{(1+3p^2-2p^3)(2-p+4p^2-4p^3)} &(12)
\end{aligned}
$$

Simulation

plot the chart

Conclusion

- Markovian approach has limitations, especially for larger constraint length.
- code searches are hard. With convolutional codes, there does not appear to be a way of easily finding the code with minimizes the required E_b/N_O