A Quantized PSK 8-State Decoder for Spectrally Efficient Communications

Michael D. Ross
Frank Carden
Center for Telemetry Research
New Mexico State University
Las Cruces, New Mexico 88003

ABSTRACT

Trellis Coded Modulation [5] combines the Viterbi Algorithm [4] with PSK or QAM signalling to achieve a coding gain, using signal set expansion as an alternative to bandwidth expansion. Optimum detection of TCM requires the calculation of Euclidean distances in the signal set space. Circular Quantization of received signal vectors as an alternative to Euclidean distance calculation has been shown to result in minimal loss of performance when used with a 4-state trellis codes [1, 2, 3]. This paper investigates the effect of circular quantization on 2 different 8-state trellis codes. The 8-state codes showed a modest gain over the 4-state code, while the effect of circular quantization on the 8-state codes paralleled the effect on the 4-state code.

INTRODUCTION

In the decade of the 90’s, there will be a need for high quality, high data rate, spectrally efficient telemetering systems. This is especially true for systems requiring video instrumentation. The potential use of error correcting codes to improve the performance of satellite channels, which are power limited, was recognized in the early 70’s [11]. The rapid increase in demand for satellite communications has created a need for systems which are bandwidth efficient as well as power efficient [8]. Trellis Coded Modulation, pioneered by Ungerboeck [5, 10] is an answer to this need. TCM obtains the redundancy necessary for forward error correction by expanding the signal set, rather than by increasing the bandwidth. Additionally, TCM with PSK modulation may be used in applications requiring constant envelope signalling. The effectiveness of TCM is due to the fact that the performance of the error correcting code more than makes up for the increased density of the signal set, resulting in a net reduction of bit error rate.

TCM schemes employ a convolutional code, a type of code in which the current output of the encoder depends on previous inputs, as well as the current input. The encoding is accomplished by means of a shift register encoder, as shown in figures 1 & 2. The output codebits, C1 and C0 are modulo-2 sums of the tapped shift register cells. With these encoders, two codebits are generated per input data bit. To generate TCM, the
outputs of the convolutional encoder, and optionally, other data bits which bypass the convolutional encoder, select a vector from a PSK or QAM signal set, as shown in figure 3. Either the 8-state or the 4-state encoder may be used in this configuration. In this example, the signal vector is selected from the constellation of figure 4. For best results, the codebit mapping is assigned according to Ungerboeck’s set partitioning rules [5]. The 8-PSK encoder which generates this mapping is shown in Figure 5.

Decoding is accomplished using the Viterbi Algorithm [4]. In applying the Viterbi Algorithm, the convolutional encoder is considered to be a finite state machine, where the state is defined by the previous inputs retained in the shift register. The trellis diagrams of figures 6 & 7 show all possible state transitions of the 4 and 8-state encoders, respectively, with the 8-PSK symbols assigned to the transitions by the 8-PSK encoder. Either of two symbols may be associated with any transition, the choice depending on $x_0$, the data bit which bypasses the convolutional encoder in figure 3. The trellis diagram may be extended horizontally to any number of stages to depict the operation of the encoder during a given period of time. The Viterbi algorithm selects, on the basis of the received sequence, the path which the encoder is most likely to have taken through the trellis in generating the transmitted sequence. This is done by finding the distance (Hamming distance in the case of binary encoding, Euclidean distance in the case of TCM) between the received symbol and the symbol attached to each branch of the trellis in the corresponding stage. The associated distances along the various paths are summed, and the path of least total distance, or metric, is ultimately selected. Only the most likely path leading to each node of the trellis is retained by the decoder, thus the number of paths which must be retained in memory is equal to the number of states of the encoder used to generate the sequence.

**CIRCULAR QUANTIZATION**

The purpose of the Viterbi Algorithm is to maximize the probability of selecting the correct sequence when the encoded sequence is transmitted over an imperfect channel. The appropriate metric to use depends on the characteristics of the channel. For a two dimensional memoryless channel with additive white Gaussian noise, the optimal metric is the Euclidean distance. Because the metrics must be calculated for each incoming signal, the use of Euclidean distances could hinder the real time implementation of TCM.

Quantization, restricting the input vector to a finite number of points (placed between, and coincident with the signal set vectors) would allow the required metrics to be obtained from lookup tables rather than calculated in real time. Circular quantization, appropriate for use with PSK, places the quantization points on the circle outlined by the signal set vectors, as shown in figure 8. In this illustration, 24 quantization points are shown with the 8-PSK constellation, but the technique generalizes to other levels of M-ary PSK and different numbers of quantization points. Previous studies [1, 2, 4] have evaluated the performance of 16, 24, 32, & 48 sector quantization and shown positive potential for circular quantization using 24 or more points with 4-state codes.
DESIGN CONSIDERATIONS

The code of figure 6 is the optimal 4-state code found by Ungerboeck [11]. The 8-state codes of figures 7 & 9 were found using Ungerboeck’s set partitioning rules [5]. The codes of figures 6 & 7 are generated using the convolutional encoders and the 8-PSK encoder shown in figures 1, 2, 3, & 5. The Code of figure 9 was generated using the configuration of figure 13. The top level system diagrams of figures 10, 11, & 12 show the essential functions used in TCM decoding, and illustrate the slightly different requirements imposed by the three codes.

Implementation of the Viterbi Algorithm requires the calculation of some kind of metric, in this case, the Euclidean distance. In the parallel branch codes of figures 6 & 7 there are two symbols associated with each state transition, so that the more distant of each pair may be discarded, before any further decision making is performed. This is referred to as the outboard decision. This leaves only the four nearest symbols for metric calculation. With the other 8-state code, the four symbols allowed at any state are associated with transitions to four distinct states, which makes it necessary to calculate metrics for all eight symbols. This also increases the complexity of the path decision and path memory functions, so it is interesting to note that the performance of the two 8-state codes is essentially the same.

The path decision maker must select the minimum metric path leading into each node. In this function, the incremental metric for each branch is added to the cumulative metric at the preceding node. At each node, the converging branch with the least total metric is selected. The metric of the selected paths becomes the cumulative metric at that node, for the next stage of operation. When a code with parallel branches is used, the choice to be made is between the two branches left by the outboard decision. When the code without parallel branches is used, there is no outboard decision, and the choice is between four branches.

The information to be relayed from the path decision maker to the path memory consists of the selection made at each node, as well as the identification of the node which has the least cumulative metric. The memory operates in the pipeline fashion, whereby data flows from one end of the memory to the next. The oldest data in the path which currently has the least cumulative metric is assumed to be correct. In the 8-state systems, what flows out of the memory is the maximum likelihood sequence of 8-PSK symbols, so extra logic is needed to recover the original data bits. This depends on several consecutive symbols, not on the current symbol alone. In the 4-state system, a different approach was used, so that the data bits were obtained directly from the decoder memory and the outboard decision maker.

SIMULATION PROCEDURE

Block diagram models of Euclidean Distance Viterbi decoders were designed and simulated using the Block Oriented Systems Simulator, a commercially available software package. Separate systems were designed to implement the trellis codes of figures 6, 7, &
The performance of 16, 24, 32 and 48 sector 8-PSK was compared to that of unquantized 8-PSK for each of the 8-state codes. The two eight state codes were compared to each other and to the four state code using unquantized 8-PSK. A decoder path memory length of 32 was used for all systems, and results were based on trials of 250,000 8-PSK symbols. Optimal quadrature detection from a 2-dimensional channel is assumed. To simulate the effect of noise, a random Gaussian vector is added to each signal vector prior to quantization and Viterbi decoding. The required variance of the noise vector, found in Carlson [12] is $\sigma^2 = \frac{A_c^2 \nu}{2E}$ where $A_c$ is the carrier amplitude, $\nu$ is the one sided spectral noise density and $E$ is the symbol energy.

Without loss of generality, $A_c$ may be taken to be 1, so that

$$\sigma^2 = \frac{1}{2} \frac{\nu}{E} = \frac{1}{2} \frac{N_0}{E_s}$$

RESULTS

The plots of figures 14, 15, & 16 show the actual number of data bits missed, after complete decoding. Figure 14 shows the comparison of the 4-state and 8-state codes. Using a trial of 250,000 symbols, a statistically significant difference between the performance of the two 8-state codes was not found. At a bit error rate of $10^{-3}$, the 8-state codes begin to show a gain over the 4-state code, about 0.1dB. For the 8-state code without parallel branches (figure 16), the losses due to 48, 32, 24, and 16 sector quantization were approximately 0.45, 0.5, 0.6, & 0.85 dB, respectively. For the 8-state code with parallel branches, the losses were 0.25, 0.4, 0.55 and 0.85 respectively. This shows the practicality of circularly quantized 8-PSK systems employing 24 or 32 quantization points.

REFERENCES


FIGURE 1. 8-STATE CONVOLUTIONAL ENCODER

FIGURE 2. 4-STATE CONVOLUTIONAL ENCODER

FIGURE 3. GENERATION OF 4-STATE OR 8-STATE CODE WITH PARALLEL BRANCHES

FIGURE 4. 8-PSK SIGNAL CONSTELLATION WITH CODEBIT MAPPING

FIGURE 5. 8-PSK ENCODER
FIGURE 6. STATE TRELLIS

FIGURE 7. 8-STATE TRELLIS WITH PARALLEL BRANCHES
FIGURE 8. 8-PSK SIGNAL
CONSTELLATION WITH
24-SECTOR QUANTIZATION

FIGURE 9. 8-STATE TRELLIS
WITHOUT PARALLEL BRANCHES
FIGURE 10. SIMULATION DIAGRAM FOR 4-STATE CODE

FIGURE 11. SIMULATION DIAGRAM FOR 8-STATE CODE WITH PARALLEL BRANCHES
FIGURE 12. SIMULATION DIAGRAM FOR 8-STATE CODE WITHOUT PARALLEL BRANCHES

FIGURE 13. GENERATION OF 8-STATE CODE WITHOUT PARALLEL BRANCHES
FIGURE 14. 4-STATE VS 8-STATE CODES
FIGURE 15. 8-STATE CODE WITH PARALLEL BRANCHES
FIGURE 16. 8-STATE CODE WITHOUT PARALLEL BRANCHES