Implementation of Encoder and Viterbi Decoder for Lossless Applications

TIRTHISSETTY YAMUNA NIHARIKA¹, D. RAVI NAYAK ²

¹Research Scholar, Pydah College of Engineering and Technology, AP, India, E-mail: niharika1612@gmail.com.
²Assistant Professor, Pydah College of Engineering and Technology, AP, India, E-mail: Ravinayak46@gmail.com.

Abstract: The best way of decoding against random errors is to compute the received sequence with every possible code sequence. This is called maximum likelihood (ML) decoding. The criterion for deciding between two paths is to select the one having the smaller metric. The rule maximizes the probability of a correct decision. The Andrew Viterbi proposed an efficient algorithm to find the minimum distance to received sequence in a trellis. It’s named after him as Viterbi algorithm (VA). This was recognized by Forney to be Max. Likelihood decoder. The Viterbi algorithm occupies large memory and computational resources. To address this problem Proposed Viterbi Algorithm is introduced. The Proposed Viterbi decoder functionally is same as the previous Viterbi decoder but it reduces memory and the hardware resources. The proposed block diagram checks every node for path metric value and eliminates the path that is found if it is not having minimum distance. This architecture is simulated, synthesized and implemented by VERILOG language using XILINX ISE Tool.

Keywords: Viterbi Decoder, CDMA, Register Exchange Method, Trellis Diagram, Finite State Machine, Hamming Distance.

I. INTRODUCTION

Convolutional coding has been used in communication systems including deep space communications and wireless communications. It offers an alternative to block codes for transmission over a noisy channel. An advantage of convolutional coding is that it can be applied to a continuous data stream as well as to blocks of data. IS-95, a wireless digital cellular standard for CDMA (code division multiple access), employs convolutional coding. A third generation wireless cellular standard, under preparation, plans to adopt turbo coding, which stems from convolutional coding. The Viterbi decoding algorithm, proposed in 1967 by Viterbi, is a decoding process for convolutional codes in memory-less noise. The algorithm can be applied to a host of problems encountered in the design of communication systems. The Viterbi decoding algorithm provides both a maximum-likelihood and a maximum a posteriori algorithm. A maximum a posteriori algorithm identifies a code word that maximizes the conditional probability of the received code word against the received code word, in contrast a maximum likelihood algorithm identifies a code word that maximizes the conditional probability of the received code word against the decoded code word. The two algorithms give the same results when the source information has a uniform distribution.

A viterbi decoder uses the Viterbi algorithm for decoding a bit stream that has been encoded using Forward error correction based on a Convolutional code. The Viterbi algorithm is commonly used in a wide range of communications and data storage applications. It is used for decoding convolutional codes, in baseband detection for wireless systems, and also for detection of recorded data in magnetic disk drives. The requirements for the Viterbi decoder or Viterbi detector, which is a processor that implements the Viterbi algorithm, depend on the applications where they are used. This results in very wide range of required data throughputs and power or area requirements. Viterbi detectors are used in cellular telephones with low data rates, of the order below 1Mb/s but with very low energy dissipation requirement. They are used for trellis code demodulation in telephone line modems, where the throughput is in the range of tens of kb/s, with restrictive limits in power dissipation and the area/cost of the chip. On the opposite end, very high speed Viterbi detectors are used in magnetic disk drive read channels, with throughputs over 600Mb/s. But at these high speeds, area and power are still limited. Convolutional coding has been used in communication systems including deep space communications & wireless communications. It offers an alternative to block codes for transmission over a noisy channel. An advantage of convolutional coding is that it can be applied to a continuous data stream as well as to blocks of data. IS-95, a wireless digital cellular standard for CDMA (code division multiple access), employs convolutional coding.

II. VITERBI DECODER ALGORITHM

The Viterbi decoding algorithm is a decoding process for convolutional codes for memory-less channel. Fig.1 depicts the normal flow of information over a noisy channel. For the purpose of error recovery, the encoder adds redundant information to the original Information, and the
output is transmitted through a channel. Input at receiver end (r) is the information with redundancy and possibly, noise. The receiver tries to extract the original information through a decoding algorithm and generates an estimate (e). A decoding algorithm that maximizes the probability p(r|e) is a maximum likelihood (ML) algorithm. An algorithm which maximizes the p(r|e) through the proper selection of the estimate (e) is called a maximum a posteriori (MAP) algorithm. The two algorithms have identical results when the source information i has a uniform distribution.

Fig.1. The Convolutional Decoding.

The Viterbi Algorithm was developed by Andrew J. Viterbi and first published in the IEEE transactions journal on Information theory in 1967 [1]. It is a maximum likelihood decoding algorithm for convolutional codes. This algorithm provides a method of finding the branch in the trellis diagram that has the highest probability of matching the actual transmitted sequence of bits. Since being discovered, it has become one of the most popular algorithms in use for convolutional decoding. Apart from being an efficient and robust error detection code, it has the advantage of having a fixed decoding time. This makes it suitable for hardware implementation.

A. Encoding Mechanism

Data is coded by using a convolutional encoder. It consists of a series of shift registers and an associated combinatorial logic. The combinatorial logic is usually a series of exclusive-or gates. The conventional encoder ½ K=7, (171,133) is used for the purpose of this project. The octal numbers 171 and 133 when represented in binary form correspond to the connection of the shift registers to the upper and lower exclusive-or gates respectively. Fig.2 represents this convolutional encoder that will be used for the project.

Fig.2. Rate=1/2 K=7, (171, 133) Convolutional Encoder.

B. Trace Back Unit

The global winner for the current state is received from Block 2. Its predecessor is selected in the manner. In this way, working backwards through the trellis, the path with the minimum accumulated path metric is selected. This path is known as the traceback path. A diagrammatic description will help visualize this process. The trellis diagram for a ½ K=3 (7, 5) coder with sample input taken as the received data as shown in Fig.3.

Fig.3. Selected minimum error path for a ½ K=3(7,5) coder.

The state having minimum accumulated error at the last time instant is State 10 and traceback is started here. Moving backwards through the trellis, the minimum error path out of the two possible predecessors from that state is selected. This path is marked in blue. The actual received data is described at the bottom while the expected data written in blue along the selected path. It is observed that at time slot three there was an error in received data (11). This was corrected to (10) by the decoder. Local winner information must be stored for five times the constraint length. For a K =7 decoder, this results in storing history for 7 x 5 = 35 time slots. The state of the decoder at the time instant 35 time slots prior can then be accurately determined. This state value is passed to Block 4. At the next time slot, all the trellis values are shifted left to the previous time slot. The path metric for the last received data and compute the minimum error path is then calculated. If the global winner at this stage is not a child of the previous global winner, the traceback path has to be updated accordingly until the traceback state is a child of the previous state as shown in Fig.4.

Fig.4. Block diagram of Trace back unit.

Multiple traceback paths are possible and it may be thought that traceback up to the first bit is necessary to correctly
determine the surviving path. However, it was found that all possible paths converge within a certain distance or depth of traceback. This information is useful as it allows the setting of a certain traceback depth beyond which it is neither necessary nor advantageous to store path metric and other information. This greatly reduces memory storage requirements and hence energy consumption of the decoder. Empirical observations showed that a depth of five times the constraint length was sufficient to ensure merging of paths [8]. Therefore, local winner information is stored for 35 slots (five times seven) in the decoder used for this project. Block 4. Data Input Determination Now going forwards through the traceback path, the state transitions at successive time intervals are studies and the data bit that would have caused this transition is determined. This represents the decoded output.

III. METHODS AND TYPES OF VITERBI DECODER

A. Register Exchange Method

The register exchange (RE) method is the simplest conceptually and a commonly used technique. Because of the large power consumption and large area required in VLSI implementations of the RE method, the trace back method (TB) method is the preferred method in the design of large constraint length, high performance Viterbi decoders. In the register exchange, a register assigned to each state contains information bits for the survivor path from the initial state to the current state. In fact, the register keeps the partially decoded output sequence along the path, as illustrated in Fig.5. The register of state S1 at t=3 contains ‘101’. This is the decoded output sequence along the hold path from the initial state.

![Fig.5. Register Exchange Method.](image)

The register-exchange method eliminates the need to trace back since the register of the final state contains the decoded output sequence. However, this method results in complex hardware due to the need to copy the contents of all the registers in a stage to the next stage. The survivor path information is applied to the least significant bit of each register, and all the registers perform a shift left operation at each stage to make room for the next bits. Hence, each register fills in the survivor path information from the least significant bit toward the most significant bit. The scheme is called shift update. The shift update method is simple in implementation but causes high switching activity due to the shift operation and, hence, results in high power dissipation.

B. Types of Viterbi Decoding

In order to realize a certain coding scheme a suitable measure of similarity or distance metric between two code words is vital. The two important metrics used to measure the distance between two code words are the Hamming distance and Euclidian distance adopted by the decoder depending on the code scheme, required accuracy, channel characteristics and demodulator type.

Hard Decision Viterbi Decoding: In the hard-decision decoding, the path through the trellis is determined using the Hamming distance measure. Thus, the most optimal path through the trellis is the path with the minimum Hamming distance. The Hamming distance can be defined as a number of bits that are different between the observed symbol at the decoder and the sent symbol from the encoder. Furthermore, the hard decision decoding applies one bit quantization on the received bits.

Soft Decision Viterbi Decoding: Soft-decision decoding is applied for the maximum likelihood decoding, when the data is transmitted over the Gaussian channel. On the contrary to the hard decision decoding, the soft-decision decoding uses multi-bit quantization for the received bits, and Euclidian distance as a distance measure instead of the hamming distance. The demodulator input is now an analog waveform and is usually quantized into different levels in order to help the decoder decide more easily. A 3-bit quantization results in an 8-array output.

![Fig.6. A recursive convolutional encoder.](image)

C. Trellis Diagram

A convolutional encoder is often seen as a finite state machine. Each state corresponds to some value of the encoder's register. Given the input bit value, from a certain state the encoder can move to two other states. These state transitions constitute a diagram which is called a trellis diagram. A trellis diagram for the code on the Fig.6 is depicted on the Fig.7. A solid line corresponds to input 0, a dotted line – to input 1 (note that encoder states are designated in such a way that the rightmost bit is the newest one). Each path on the trellis diagram corresponds to a valid
sequence from the encoder's output. Conversely, any valid sequence from the encoder's output can be represented as a path on the trellis diagram. One of the possible paths is denoted as red (as an example). Note that each state transition on the diagram corresponds to a pair of output bits. There are only two allowed transitions for every state, so there are two allowed pairs of output bits, and the two other pairs are forbidden. If an error occurs, it is very likely that the receiver will get a set of forbidden pairs, which don’t constitute a path on the trellis diagram. So, the task of the decoder is to find a path on the trellis diagram which is the closest match to the received sequence.

![Trellis Diagram](image)

**Fig. 7. A trellis diagram corresponding to the encoder on the Figure 6.**

Let’s define a free distance, as a minimal Hamming distance between two different allowed binary sequences (a Hamming distance is defined as a number of differing bits).

### IV. SYNTHESIS RESULTS

Synthesis and Simulation Results as shown in bellow Figs. 8 to 11.

**A. Schematic Diagram Of Viterbi Decoder**

![Schematic Diagram](image)

**Fig. 8. Schematic Diagram of Viterbi Decoder.**

**B. RTL Schematic**

![RTL Schematic](image)

**Fig. 9. RTL Schematic.**

**C. Device Utilization Summary**

<table>
<thead>
<tr>
<th>Device Utilization</th>
<th>Used</th>
<th>Available</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slices</td>
<td>3456</td>
<td>32448</td>
<td>10%</td>
</tr>
<tr>
<td>Number of Slice Flip Flops</td>
<td>4257</td>
<td>64096</td>
<td>6%</td>
</tr>
<tr>
<td>Number of Input LUTs</td>
<td>4955</td>
<td>64096</td>
<td>7%</td>
</tr>
<tr>
<td>Number of bonded IOBs</td>
<td>6</td>
<td>808</td>
<td>0%</td>
</tr>
<tr>
<td>Number of GCLKs</td>
<td>4</td>
<td>4</td>
<td>100%</td>
</tr>
</tbody>
</table>

**Timing Summary:**

- **Speed Grade:** -8
- **Minimum Period:** 53.950ns
- **Maximum Frequency:** 18.536MHz
- **Minimum input arrival time before clock:** 37.285ns
- **Maximum output required time after clock:** 5.783ns
D. Simulation Results

![Fig.11. Output Waveform.]

V. CONCLUSIONS

We have proposed a high speed VD design for TCM systems. The pre-computation architecture that incorporates T-algorithm efficiently reducing decoding speed appreciably. We have also analysed pre-computation algorithm where the optimal pre computation steps are calculated and discussed. This algorithm is suitable for TCM systems which always employ high rate convolution code. Finally we presented a design case. Both the ACSU and SMU are modified to correctly decode the signal. synthesis results show that VD could improving the maximum decoding speed. By using FPGA device and hybrid microprocessor the decoding benefits can be achieved in future. In future to improve the decoder performance the Viterbi algorithm is carried out in reconfigurable hardware. Power saving architecture can be designed for the above decoder which is executable in the mobile devices. Viterbi decoder can also be implemented using JAVA. Therefore in the future Viterbi algorithm may be used for various scenarios. So in the future the complexity can be greatly reduced.

VI. REFERENCES