A Novel and Fast Way to Decode VLEC

Richa Gupta, BhuDev Sharma, Vishal Jain

Abstract*: *Joint Source Channel Coding (JSCC) is an emerging area of Information Theory. It jointly optimises source code and channel code both. Variable Length Error Correcting Codes (VLECs) are the tools of JSCC and they are very important as there are a lot of applications of VLECs in the transmission of multimedia files. There are several decoding algorithms of VLECs, one of them is Maximum Likelihood (ML) decoding. This uses trellis/tree structure to decode the received bitstream. The problem with the trellis approach is its complexity which increases exponentially with the length of the received vector. This paper proposes a rather novel and fast tabular decoding method for ML decoding of VLECs. In place of the trellis or tree structure, if tabular method is used, the complexity of the decoding algorithm reduces a lot.

Index Terms: Variable length codes, trellis structure, maximum likelihood decoding, Viterbi algorithm and Hamming distance.

I. INTRODUCTION

Variable length codes have been widely used for data compression (source coding) but not for error correction. Consider the eight symbol memory-less source given in Table 1. If this source is encoded using the brute-force method, keeping the length of the codeword same, then we would require a minimum of three bits per symbol. However, since some symbols are more probable than others, we could find a way to encode these eight symbols into words of variable lengths, using fewer average number of bits per symbol which is less than 3. An optimum way to achieve this was given by Huffman [1]. The resulting Huffman code for this source is also shown in Table 1. Using this code, the average number of bits required per symbol is 2.5 bits, and hence we have achieved compression. The only problem with such a scheme is that should the data be subjected to errors, then at the decoder we would have error propagation due to loss of synchronization. The other type of codes are channel codes. These, to avoid the synchronization problem which may arise due to errors, are the constant length codes. Constant length coding, while serve a great purpose and has richly employed abstract algebra tool, compromises on efficiency. All constant length codes as length increases are rather bad. The channel coding is a technique which introduces the redundancy in the information before the transmission of the information across any given channel. This redundancy is exploited by the receiver in order to detect and correct the errors introduced by the noisy channels. The channel codes are also known as error-correcting codes. The study of channel coding began with the pioneered work of Hamming and Shannon [2]. Efficiency considerations demand developing variable length codes for noisy channels, known as Variable Length Error Correcting-codes (VLECs) [3].

Source	Probability	Huffman
A	0.35	0
B	0.30	10
C	0.10	1100
D	0.10	1101
E	0.05	1110
F	0.05	11110
G	0.03	111110
H	0.02	111111

Table 1: Huffman Codes

VLEC, being as a type of JSCC, was first introduced by Hartnett in 1974 [4]. Hartnett presented the basic structure of VLECs. Later on, several researchers worked on VLECs. Progress on VLEC suffered in the absence of algebraic tools for it and high demand for error correcting codes met by constant length codes. There is quite some interest and scope in error coding through variable length code words. Sharma & Bernard's study has provided a lead by initiating combinatorial bounds on the average codeword length of variable length error correcting codes [5]. Sharma & Bernard's study also includes the necessary and sufficient conditions for the generation of variable length error correcting codes [5], [6]. An improved combinatorial bound on average codeword length was given by Bhudev Sharma and Richa in 2011 [7]. An efficient algorithm to construct variable length error correcting codes using constant length error correcting codes has been proposed by Bhudev Sharma and Richa in 2012 [8][9]. As designing a good decoding algorithm is another challenge, active research is also going on in this direction [10][11][12][13]. This paper presents a rather novel, fast and efficient way to decode the variable length error correcting coded data by 'maximum likelihood algorithm,' which is similar to well-known Viterbi algorithm for convolution codes, that we call here as modified Viterbi algorithm. The paper is organized as follows. Section I gives a brief overview of Variable Length Error Correcting Codes.

Revised Manuscript Received on March 20, 2019.

Richa Gupta, Electronics and Communication Engineering, JIIT, Noida, India

BhuDev Sharma, Mathematic Department, Formerly at JIIT, Noida (India); Clark Atlanta University, Atlanta, FL (USA).

Vishal Jain, Computer Applications and Management, Bharati Vidyapeeth's Institute of Computer Applications and Management, N. Delhi.

The Maximum Likelihood Decoding Algorithm for variable length error correcting is explained in II section. Section III describes the proposed novel algorithm to decode VLECs using maximum likelihood decoding. Results are discussed in Section IV followed by conclusion in Section V.

II. MAXIMUM LIKELIHOOD DECODING OF VARIABLE LENGTH ERROR CORRECTING CODES

Let us begin by realizing the difficulty in decoding variable-length codes under noisy conditions. Because of variable lengths the decoder cannot readily determine where one word ends and the other begins. It may be remarked that in our scheme, the errors are only substitution errors, the digits are neither dropped nor added due to noise. Hence an optimum decoding strategy would be to choose that sequence of codewords of length n bits which minimises the (Hamming) distance between this sequence and the received n bits. The following example illustrates how this is done. Consider a simple VLEC code with code-words C1 = [00], C2 = [111]. A tree diagram may be constructed for such a code that takes into account the number of bits emitted from the encoder. This is shown in Fig 1. The tree is constructed such that all nodes that emit the same number of output bits are placed on the same vertical line.

Refer Fig 1 (given at the end of the paper)

Next let the transmitted sequence be “aaba”. Corresponding binary sequence will be of 9 bits as 000011100. Suppose that channel noise introduces an error in the received sequence, shown bold in the third position of the sequence: 00**1**011100. Using the tree diagram shown in Fig 1, it can be noted that there are 5 possible codeword sequences giving rise to 9 bits, as given in Table 2. The sequence giving the minimum Hamming distance with the received sequence will be the decoded sequence.

Symbol sequence	Codeword sequence	Hamming distance
aaab	000000111	5
aaba	000011100	1
abaa	001110000	3
baaa	111000000	5
bbb	111111111	5

Table 2: All possible binary sequence of 9 bits for the considered VLEC

As proposed by Victor Buttigieg in 1995 [14][15], this decoding can be mathematically done using tree structure. But the problem with this tree representation is that it grows exponentially with the number of received bits. Another way to decode a sequence is by using trellis diagram, which is done by combining all nodes in the tree at the same bit position into one state in the trellis diagram. It also suffers by the same problem of exponential growth of trellis with the received sequence length [16][17]. The trellis diagram for the code given Example above is shown in Fig 2. Note that the states are labelled by the bit position represented (starting from 0).

Refer Fig 2 (given at the end of the paper)

To overcome the problem of exponential growth of either tree or trellis, a modified version of the Viterbi decoding algorithm [2-3] may be used. The problem in applying the Viterbi algorithm directly to the above trellis is the fact that the state transitions involve a variable number of bits. Therefore, it is necessary to keep track of the position from where to take the next bits in the input sequence. This is quite simple to do. Each state in the trellis diagram is associated with a different, unique, position. Therefore, the next transitions emitting out of that state must be compared to the input bit sequence starting at this position. Decoding the sequence using trellis diagram can be made much easier using a table in place of state diagram. The tabular approached has been proposed for the same purpose. The details of the tabular approach are explained in the next section.

III. TABULAR WAY TO IMPLETMENT MAXIMUM LIKELIHOOD DECODING OF VARIABLE LENGTH ERROR CORRECTING CODES

Maximum likelihood decoding involves the probabilistic estimation of transmitted vector based on the hamming distance calculations. Consider the same example: a code consisting of 2 codewords 00, 111. If the received vector is 00110, it is more likely that the transmitted vector was 00111 rather than 11100. Trellis Diagram for received vector 00110 is shown in Fig 3.

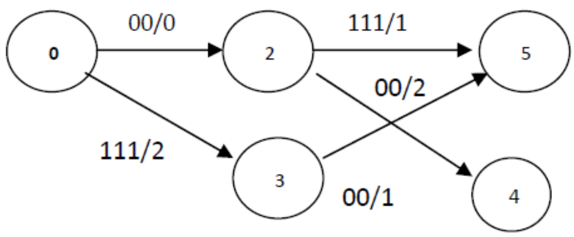

Fig 3: Trellis Diagram for the received vector

Using the trellis diagram, we can see that the hamming distance for path (0-2-5) is 1 unit whereas for the path (0-3-5) it is 3 units. Therefore, it is more likely that the vector transmitted was 00 (0-2) and 111 (2-5), that is 00111.

Let us now try to decode the same received vector using the other approach: tabular decoding. First step is to draw the trellis matrix, which is of size (2n+1) rows and m columns, where n is the number of codewords of the code and m is the length of received vector. First row indicates the state numbers, which is equal to the length of the received vector. Next n rows indicate possible states achievable through 1st, 2nd, …., nth codewords respectively. And the remaining n rows indicate hamming weights respectively corresponding to the first, second and nth code-words through all possible intermediate states from which state transition may occur.

Refer Fig 4 (given at the end of the paper)

For the considered example, the trellis matrix will as of size 5 X 5, as shown in Fig 4. The row description of trellis matrix is as follow:

Row 1: State Numbers

Row 2: Possible states achievable through codeword 1 - initialized with a dummy value (999)

Row 3: Possible states achievable through codeword 2 - initialized with 999

Row 4: Hamming weight for transition through codeword 1 - initialized with 111

Row 5: Hamming weight for transition through codeword 2- initialized with 111

Hamming weights of the trellis matrix are calculated and stored in the matrix, a decision is arrived at seeing the minimum weight of the path. The state giving the minimum Hamming state at the final state will be used to generate the ML decoded sequence by back-tracing, as shown in Fig 5.

Refer Fig 5 (given at the end of the paper)

The simulation results have been attached below for different variable length error correcting codes. Let us consider the first example as shown in Table 3.

Refer Table 3 (given at the end of the paper)

Refer Fig 6 (given at the end of the paper)

Fig 6 shows the code-words and the received vector. The trellis diagram's entries are computed and stored in the trellis array as shown below in Fig 7 and trellis diagram with cumulative sum of Hamming weights is shown in Fig 8.

Refer Fig 7, 8 (given at the end of the paper)

To aid decision making it is important that the cumulative Hamming distance is calculated. The final ML decoded sequence can be obtained by back tracing the sequence giving the minimum Hamming weight with the received sequence. For the example considered, the ML decoded sequence is as shown in Fig 9.

Refer Fig 9 (given at the end of the paper)

Let us consider another example with 3 code-words. The code words are given in Table 4 along with considered received sequence.

Codeword	Symbol
C_1	[0 0 0]
C_2	[1 1 1 0 0 0]
C_3	[1 1 1 1 1 1]
Received Vector	[0 0 1 . 1 1 1 1 1 0 . 1 1 0 1 1 1 . 1 1 1 0 1 0 . 1 1 1 0 0 1]

Table 4: VLEC code 2 and the considered received vector

Fig 10 shows the code-words and the received vector. The trellis diagram's entries are computed and stored in the trellis array as shown below in Fig 11 and trellis diagram with cumulative sum of Hamming weights is shown in Fig 12.

Refer Fig 10, 11, 12 (given at the end of the paper)

To aid decision making it is important that the cumulative Hamming distance is calculated. The final ML decoded sequence can be obtained by back tracing the sequence giving the minimum Hamming weight with the received sequence. For the example considered, the ML decoded sequence is as shown in Fig 13.

Refer Fig 13 (given at the end of the paper)

Implementation of VLEC decoding using tabular approach is easier and less complex as compared to simple trellis implementation as adding a codeword in the codebook only increases two rows at a time, thus involved less storage and retrieval time. This technique is also known as modified viterbi algorithm as the algorithm is similar to Viterbi algorithm but in place of constant lengths, here we have variable lengths.

IV. RESULT

A tabular method is proposed to decode variable length error correcting codes using maximum likelihood decoding. It has been observed that the complexity and computational time reduces as compared to earlier used maximally likelihood decoding. The proposed method is easy to implement and provide a much faster way of decoding for a bit-stream of any size. Trellis decoding method has the drawback of exponential increase with the length of received bit sequence to be decoded. Another problem with trellis approach is the way it represents all nodes in the form of the state, so backtracking is another issue in that. Our proposed method simplifies this approach by creating a table of size n+1 rows and m columns, where n is the number of codewords and m is the length of received bit-stream.

V. CONCLUSION

A tabular method to decode variable length error-correcting codes using maximum likelihood decoding has been developed. The proposed decoding method, which is based on a tabular search, is much less complex than Buttigieg's trellis decoding method which is based on an exponential search. Due to the reduction in complexity, less computation time is required to decode the received vector.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

REFERENCES

1. D. Huffman, "A method for the Construction of Minimum Redundancy Codes", Proceedings of IRE, 40, 1098-1101, 1962.
2. R. W. Hamming, "Error detecting and error correcting codes", Bell System Tech. Journal, 29, 147-160, 1950.
3. M. Bernard, "Error Correcting Codes with Variable Lengths and Non-Uniform Errors", PhD. Thesis, Department of Mathematics, Faculty of Natural Sciences, St. Augustine, 1987.
4. W. E. Hartnett, "Foundation of Coding Theory", D. Reidling Publishing Co., Dordietcht, Holland, 1974.
5. M. Bernard and B. D. Sharma, "Some Combinatorial Results on Variable Length Error Correcting Codes", ARS Combinatoria 25 B, 181 -194, 1988.
6. M. Bernard and B. D. Sharma, "A Lower Bound on Average Codeword Length of Variable Length Error-Correcting Codes", IEEE, Transactions in Information Theory 36, 1474-1475, 1990.
7. R. Gupta and B. D. Sharma, "Improved Combinatorial Bound on Variable Length Error Correcting Codes", presented in IMST FIM –XX, Concordia University, Montreal, Canada, July 2011.
8. R. Gupta and B. D. Sharma, "Generation of Variable Length Error Correcting Codes using Constant Length Error Correcting Codes", International Journal of Emerging Trends in Engineering and Development, ISSN NO: 2249-6149, Issue 2, Vol. 1, January-2012, pp 269 - 279.
9. B. D. Sharma and R. Gupta, "Directions in Optimal Error Correction Coding: Variable Length Error Coding, a class of Distances and Reversible Variable Length Codes", Proceedings of the National Symposium on Mathematical Methods and Applications 2009, Indian Institute of Technology Madras, Chennai, pp 1-12, 2009.
10. R. Gupta and B. D. Sharma, "A Comparative analysis of the Variable Length Error Correcting Code Generating Algorithms" Proceedings of Twenty First International Conference of Forum for Interdisciplinary Mathematics on Interdisciplinary Mathematics, Statistics and Computational Techniques, IMSCT 2012- FIM XXI, Chandigarh, Dec 15-17, 2012
11. T. Y. Wu, P. N. Chen, F. Alajaji, Y. S. Han, "On the Design of Variable-Length Error-Correcting Codes", Communications IEEE Transactions on, vol. 61, no. 9, pp. 3553-3565, 2013.
12. T. Y. Wu, P. N. Chen, F. Alajaji, Y. S. Han, "On the construction and MAP decoding of optimal variable-length error-correcting codes", Information Theory Proceedings (ISIT) 2011 IEEE International Symposium on, pp. 2223-2227, 2011.
13. R. l Prévost, M. Coulon, D. Bonacci, J. LeMaitre, J. Millerioux, J. Y. Tourneret, "Extended constrained Viterbi algorithm for AIS signals received by satellite", Satellite Telecommunications (ESTEL) 2012 IEEE First AESS European Conference on, pp. 1-6, 2012.
14. V. Buttigieg, "Variable Length Error Correcting Codes", PhD Thesis, Department of Electrical Engineering, University of Manchester, 1995.
15. V. Buttigieg and P. G. Farrell, "Variable Length Error Correcting Codes", IEEE Proc.-Communications., Vol 147, No. 4, August 2000.
16. V. Buttigieg, N. Farrugia, "Improved bit error rate performance of convolutional codes with synchronization errors", Communications (ICC) 2015 IEEE International Conference on, pp. 4077-4082, 2015.
17. V. Buttigieg, J. A. Briffa, "Codebook and marker sequence design for synchronization-correcting codes", Information Theory Proceedings (ISIT) 2011 IEEE International Symposium on, pp. 1579-1583, 2011.

AUTHORS PROFILE

Richa Gupta received her Ph.D. Degree from JIIT, Noida, India in the year 2013 and M.Tech in Information Systems from IIT, Kanpur, India in 2005. Since August 2007, she has been working as Assistant Professor in the Department of Electronics and Communications Engineering, JIIT Noida. Prior to this, she has worked with a USA based research company 'ATC Labs, Noida' for 2 years. She has published several research papers in International Journals and presented many in International Conferences. She is a Fellow member of IETE, a Life member of IEI and a Life member of ISTE. She has been associated with research and development in the field of Digital Signal Processing, Communications, Digital Electronics, Digital Image Processing, Multimedia Communications, Information Theory and Applications.

BhuDev Sharma is a former Professor of Mathematics at JIIT, Noida (India); Clark Atlanta University, Atlanta, FL, (USA); Xavier University of Louisiana, New Orleans, (USA); University of the West Indies, Trinidad & Tobago; University of Delhi, (India). He has received Visionary Eduleader of India, by Former President of India, Sri Pranab Mukherjee, Fellow International Academy of Physical Sciences; Life Time Achievement Award by Indian Association for Reliability & Statistics; Outstanding Contribution Award – Waseda University, Japan; Life Time Service award, Panjab University, Chandigarh, 'Outstanding Contributions in Mathematical Sciences', University of South Alabama, Mobile, AL, USA; Best Contributions in Information Theory Award - Indian Society of Information Theory and Its Applications; Ten year celebration award - Founder President Mathematics Olympiads of Trinidad & Tobago. He is the president of Forum for Interdisciplinary Mathematics, Vice President of Academy of Discrete Mathematics & Applications, Mysore; Elected Foundation Fellow of 'The Institute of Combinatorics and its Applications', Manitoba, Canada' Vice President: Calcutta Mathematical Society. He has 156 papers mostly in International journals. He successfully supervised Ph.D. theses of 28 persons, who are working in different universities of India, USA, Canada, West Indies and Brazil.

Vishal Jain is currently working as Associate Professor with Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi. He has completed Ph.D (CSE) from Lingaya's University, Faridabad, M.Tech (CSE) from GGSIPU, New Delhi. He has received Young Active Member award for the year 2012 – 13 from Computer Society of India. He has worked as a Delhi State Students' Coordinator, Delhi Chapter, Computer Society of India (2014 – 2016), member of Managing Committee (2016-2017).

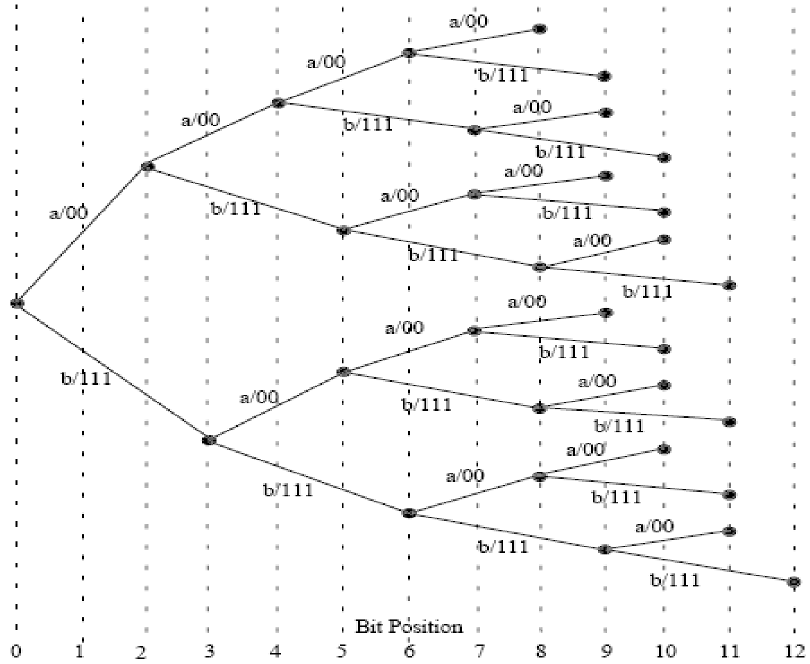

Fig 1: Tree diagram for VLEC code

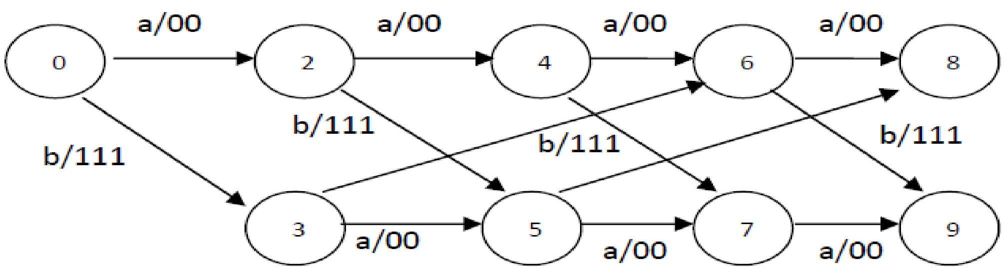

Fig 2: Trellis diagram for the received vector

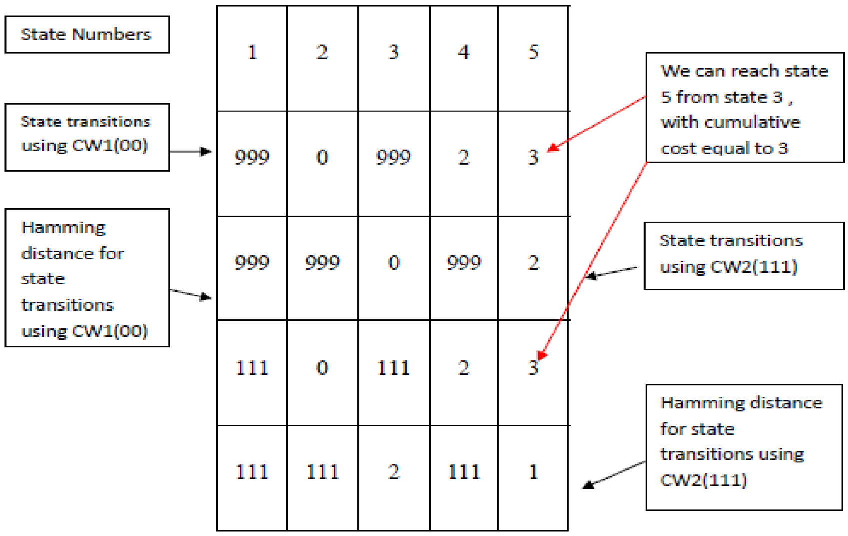

Fig 4: Trellis matrix for the considered example

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

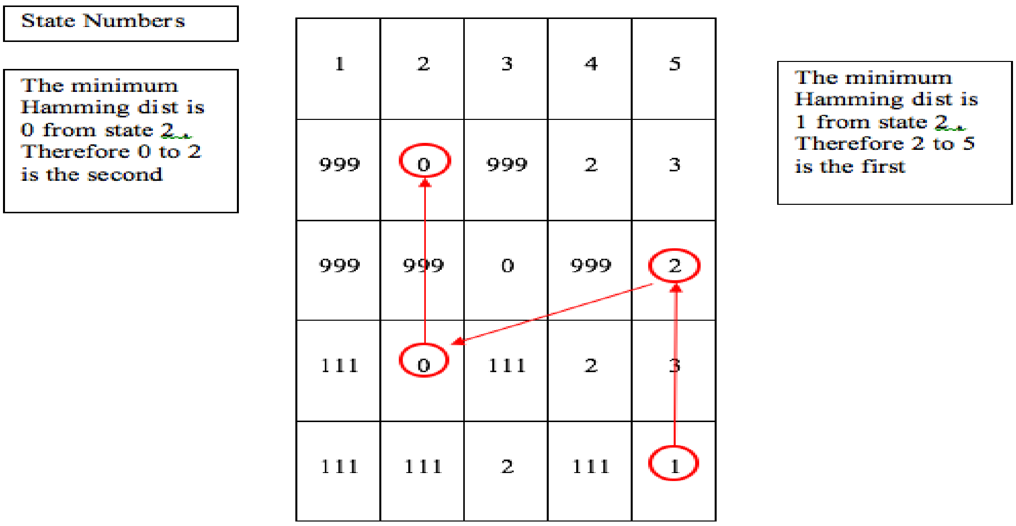

Fig 5: Trellis matrix explanation

Codeword	Symbol
C_1	[0 0]
C_2	[1 1 1]
Received Vector	01.01.110.111.00.101.00.110

Table 3: VLEC code 1 and the considered received vector

```
enter the number of codewords2
enter codeword1[0 0]
enter codeword2[1 1 1]
enter the recieved vector[0 1 0 1 1 1 0 1 1 1 0 0 1 0 1 0 0 1 1 0]
```

Fig 6: Code words and the received vector

```
mld_trellis =

     1     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17    18    19    20
   999     0   999     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17    18
   999   999     0   999     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17
   111     1   111     1     2     2     1     1     2     2     1     0     1     1     1     1     0     1     2     1
   111   111     2   111     1     0     1     1     1     0     1     2     2     2     1     2     2     2     1     1
```

Fig 7: Trellis Diagram

```
mld_trellis =

     1     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17    18    19    20
   999     0   999     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17    18
   999   999     0   999     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17
   111     1   111     2     4     4     3     3     5     5     4     3     5     4     6     5     4     6     6     7
   111   111     2   111     2     2     3     3     3     3     4     5     5     6     4     7     6     6     6     5
```

Fig 8: Trellis Diagram with cumulative sum

```
ans =

     0     0     0     0     1     1     1     1     1     1     0     0     1     1     1     0     0     1     1     1
```

Fig 9: Decoded vector

```
enter the number of codewords3
enter codeword1[0 0 0]
enter codeword2[1 1 1 0 0 0]
enter codeword3[1 1 1 1 1 1]
enter the recieved vector[0 0 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 0 1 0 1 1 1 0 0 1]
```

Fig 10: Code words and the received vector

```
mld_trellis =

  Columns 1 through 15

     1     2     3     4     5     6     7     8     9    10    11    12    13    14    15
   999   999     0   999   999     3   999   999     6   999   999     9   999   999    12
   999   999   999   999   999     0   999   999     3   999   999     6   999   999     9
   999   999   999   999   999     0   999   999     3   999   999     6   999   999     9
   999   999     1   111   111     3   111   111     2   111   111     2   111   111     3
   999   999   111   111   111     5   111   111     2   111   111     3   111   111     4
   999   999   111   111   111     2   111   111     1   111   111     2   111   111     1

  Columns 16 through 27

    16    17    18    19    20    21    22    23    24    25    26    27
   999   999    15   999   999    18   999   999    21   999   999    24
   999   999    12   999   999    15   999   999    18   999   999    21
   999   999    12   999   999    15   999   999    18   999   999    21
   111   111     3   111   111     1   111   111     3   111   111     1
   111   111     3   111   111     1   111   111     5   111   111     1
   111   111     0   111   111     2   111   111     2   111   111     2
```

Figure 11: Trellis Diagram

```
mld_trellis =

  Columns 1 through 15

     1     2     3     4     5     6     7     8     9    10    11    12    13    14    15
   999   999     0   999   999     3   999   999     6   999   999     9   999   999    12
   999   999   999   999   999     0   999   999     3   999   999     6   999   999     9
   999   999   999   999   999     0   999   999     3   999   999     6   999   999     9
   999   999     1   111   111     4   111   111     4   111   111     4   111   111     7
   999   999   111   111   111     5   111   111     3   111   111     5   111   111     6
   999   999   111   111   111     2   111   111     2   111   111     4   111   111     3

  Columns 16 through 27

    16    17    18    19    20    21    22    23    24    25    26    27
   999   999    15   999   999    18   999   999    21   999   999    24
   999   999    12   999   999    15   999   999    18   999   999    21
   999   999    12   999   999    15   999   999    18   999   999    21
   111   111     6   111   111     5   111   111     7   111   111     7
   111   111     7   111   111     4   111   111     9   111   111     5
   111   111     4   111   111     5   111   111     6   111   111     6
```

Fig 12: Trellis Diagram with cumulative sum

```
ans =

  Columns 1 through 15

     0     0     0     1     1     1     1     1     1     1     1     1     1     1     1

  Columns 16 through 27

     1     1     1     0     0     0     1     1     1     0     0     0
```

Fig. 13: The decoded vector

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication