ANALYSIS ON DECODING ALGORITHMS BASED ON SECTIONALIZED TRELLISES OF BLOCK CODES AND THEIR DUAL

Jing Cai, Marcel Adrian Ambroze
Digital Communications Research, University of Plymouth, UK
{J.Cai, M.Ambroze}@plymouth.ac.uk

Key words to describe this work: Decoding complexity, Sectionlization, Permutation.

Key results: The relationship between permutation-optimal trellis and sectionalization-optimal trellis is found. A new method to count the decoding complexity is provided. The sectionalization method is performed with different decoding algorithms.

How does the work advance the state-of-the-art?: The complexity of trellis-based decoding can be reduced by up to 50% with sectionalization.

Motivation (Problems addressed): To reduce the computational complexity and the size of memory storage required.

Introduction

A trellis T is an edge-labeled directed graph with the property that every state in T has a well-defined depth which can represent corresponding codes in coding theory. Currently, trellis-based algorithms are widely used, for example, Viterbi algorithm [7] and MAP algorithm [4]. There are many different factors that impose on the complexity of trellises, each decoding method has different level of complexity. In this paper, we briefly survey the complexity of the Viterbi decoding algorithm with Hamming codes and their dual.

A Review of Sectionalized Trellises

Consider an (n, k) linear block code C with a n-stage bit-level trellis T in which each branch represents a single code bit. The trellis can be sectionalized by any positive integer ν ranging from 1 to n, so the section boundary set is $\{h_0, h_1, \dots, h_\nu\}$, where $0 = h_0 < h_1 < \cdots < h_\nu = n$. An n-depth trellis has 2^{n-1} sectionalizations, for example, Hamming(7,4) has 2^6 different sectionalizations. The main idea in sectionalizing an original trellis is to amalgamate sections that involves two steps: [2].

1. deleting the states and branches between the initial section to the final section;
2. connecting states from the initial section to the final section with the combined labels.

According to[1], we can compute the number of branches $|B_j|$ and the number of states $|S_j|$ for any section from the generator matrix of C.

Viterbi decoding based on the Sectionalized Trellis

The Viterbi algorithm is a maximum likelihood decoding method which chooses a codeword having the *maximum likelihood metric*, or the *minimum distance metric*.

This decoding procedure consists of two major steps: computing the branch metrics and finding the survivor paths in the resulting trellis. The decoding implementation includes two metric: *branch metric* and *state metric*.

Vardy complexity algorithm [1] considers the complexity of computing the branch metric and the complexity of Viterbi decoding separately.

Branch metric complexity means computing the number of operations required in all the branches in one section. Decoding complexity $|D(T)|$ is the number of operations required to decode the trellis T. We denote the subcode of C as $C(T)$, and the dual subcode as $C(T)^\perp$. There are lots of pre-computations in every step, which can not be easily ignored in the real implementation. In the branch metrics, whether $C(T)$ is self-complementary need justifying. For an N-length block code, there are $NC_i = \sum_{i=1}^{N} i$ different $C(T)$ need to be judged. Assume in each section, there are m codewords. So every $C(T_j)$ need at most

$$J(T_j) = (m-1)\cdot m + \sum_{i=1}^{m-2} i \tag{1}$$

comparisons. If we pre-store the self-complementary table with the rows meaning the beginning boundary and the columns meaning the ending boundary, $N \times N$ matrix is required. For example, Hamming (7, 4) need $NC_i = 27$, $J(T_i) = 77$ and the size of memory storage, $M_C(T)$ is 7×7.

Also in the branch metrics, we need to consider whether $1 \in C(T)^\perp$ and the length of the section $l_i = 0 \mod 2$ or not. The dual code of $C(T)$ can be obtained from H matrix of $C(T)$. First, to find all the possibilites of $l_i = 0 \mod 2$, the number of the judgements is denoted by Y.

$$Y = \sum_{j=0}^{(\text{int})N/2} (N-1) - (2 \times j) \tag{2}$$

And at each section, there are at most m judgements required to determine whether $1 \in C(T)^\perp$ or not. The pre-table requires $N \times N$ matrix. We still use Hamming(7, 4) as the example, which need $m = 8$, $Y = 12$, and memory storage 7×7.

In the decoding part, all the subcodes of C are required to be considered. There are at most $\sum_{i=1}^{N} i \times m$ comparisons for each section. The results also can be pre-stored in an $N \times N$ matrix. The number of comparisons for the Hamming(7, 4) code is 216.

From the analysis above, we can conclude that the longer the code length, the more calculations and memory storage required, so the complexity of pre-determining calculations can not be simply ignored. So we provide the *Straightforward algorithm*, which trades complexity for implementation simplicity. Recalling the Viterbi decoder, suppose S_{h_i,p_j} in which h_i means the current depth and p_j means the position in this depth, D_n represents the state and O_n is the corresponding output. For example, the first stage of branch metric computation for Hamming (7, 4) trellis, we can get the equation as follows:

$$S'_{1,1} = S_{1,1} + (D_1 - O_1)^2 + (D_2 - O_2)^2 + (D_3 - O_3)^2$$
$$S'_{1,2} = S_{1,2} + (D'_1 - O_1)^2 + (D'_2 - O_2)^2 + (D'_3 - O_3)^2 \quad (3)$$

$$S_{2,1} = \min\{S'_{1,1}, S'_{1,2}\}. \quad (4)$$

As described in [3, 5], we consider the number of additions and the number of comparisons as the complexity of Viterbi decoding. Because of the linear property, the number of operations can be obtained section by section. For each section, the number of addition is equal to the number of branches in this section and the number of comparisons is $|B| - |S_{next}|$. The Vardy's algorithm is shown as follows:

$$D_v(T_{h,h'}) = 2 \times |B| - |S_1|. \quad (5)$$

Obviously, as the number of stages per section increases, the number of labels per branch will increase. We also need storage to save the labels. So we get the update metric as follows

$$D_v(T_{h,h'}) = 2 \times |B| - |S_1| + |B| \times |\sigma| \quad (6)$$

where σ is the number of labels per branch. Although the complexity is higher than Vardy's method, most criteria of Vardy's depend on lots of comparisons which are ignored in this algorithm, comparisons cannot be simply ignored in real implementation especially for long block codes.

Optimal sectionalization and Optimal Permutation

Given a code, there are very many different trellises that represent it, of widely varying complexity. When the permutation-optimal trellis is chosen, the decoding complexity can be minimized among all the representations. A permutation that yields the smallest state space dimension at every time of the code trellis and the smallest overall branch complexity is called an optimum permutation [6]. For Hamming codes, the permutation-optimal trellises can normally be obtained directly by natural lexicographic H matrix. In this case, we use Golay (24,12,8) and count the operations with the straightforward algorithm as the example shown in table 1.

Bit-level trellis operations	Optimum sectionalization	
	operations	boundary location
4472	2558	{0,8,12,14, 15,16,24}
89560	5630	{0,11,24}
120904	4478	{0,9,24}

Table 1: Comparison between permutation-optimal and sectionalization-optimal on Golay (24,12,8)

Conclusion

In this paper, the Viterbi algorithm has been modified with sectionalized trellises. Results of Hamming codes and their dual codes show that the sectionalization method can reduce the computing complexity and the memory storage. With the sectionaliztion algorithm, the computation complexity can be reduced by nearly 50%. Considering the large number of pre-calculations and large momery storage requirements by the previous metric, we investigated and provided the update metric: *Straightforward algorithm*, which can search out the optimal sectionalization of C more effieciently and quickly than Vardy's algorithm. And the relation between sectionalization and permutation is found in the paper. It turns out that the sectionalization-optimal code with the permutation-optimal mode can drastically change the number of the computational operations in decoding procedure, often by an exponential factor.

References

[1] A. Vardy: Trellis Structure of Codes, Handbook of Coding Theory, (edited by V. S. Pless, W. C. Huffman, and R. A. Brualdi), Elsevier Science Publishers, 1998.

[2] A. Lafourcade and A. Vardy, "Optimal Sectionalization of a Trellis," *IEEE Trans. on Information Theory*, 42, 3,pp. 689–703, May 1996.

[3] Robert J. McEliece, "On the BCJR Trellis for Linear Block Codes," *IEEE Trans. on Information Theory*, vol. 42, 4, pp. 1072–1091, July 1996.

[4] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimzing Symbol Error Rate," *IEEE Trans. Inform. Theory*, vol. 20, pp. 284–287, 1974.

[5] A. Vardy and F. R. Kschischang, "Proof of a conjecture of McEliece regarding tha expansion index of the minimal trellis,"*IEEE Tran. on Information Theory*, vol. 42, pp. 2027–2033,1996.

[6] S. Lin, T. Kasami, T. Fujiara, M. Fossorier,*Trellises and Trellis-Based Decoding Algorithm for Linear Block Codes.*Kluwer Academic Publisher, 1998.

[7] Viterbi, A. J., " Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm", *IEEE Trans. Inform. Theory* , **IT-13**, pp. 689–703, 1967.