An Algorithm for Identifying Rate $(n-1)/n$ Catastrophic Punctured Convolutional Encoders

Feng-Wen Sun, *Member, IEEE*, and
Adrianus J. Vinck, *Senior Member, IEEE*

Abstract—**It is known that both Viterbi and sequential decoding of convolutional codes can be greatly simplified by employing punctured convolutional codes, which are obtained by periodically deleting a part of the bits of a low-rate convolutional code. Even if the original low-rate convolutional code is noncatastrophic, some deleting maps may result in rate $(n-1)/n$ catastrophic punctured encoders. An algorithm is presented to identify such encoders when the original rate $1/b$ encoder is antipodal. The major part of the algorithm solves a linear equation of $\nu+1$ variables, where ν is the constraint length of the original rate $1/b$ code.**

Index Terms—**Convolutional codes, catastrophic encoders, puncturing, snydrome former.**

I. Introduction

Both Viterbi decoding and sequential decoding of high-rate convolutional codes are greatly simplified by employing the class of punctured convolutional codes, which are obtained by periodically deleting a part of the bits of a low-rate code [1]–[4]. The simple structure of the low-rate code can be utilized to encode and decode the high-rate code.

Good punctured convolutional codes are generally obtained by computer searches. During the searching procedure, catastrophic encoders, which result in infinite number of decoded errors from finite channel errors, must be eliminated. This appears to be a nontrivial problem since some deleting maps may result in catastrophic encoders even if the original code is noncatastrophic.

For example, consider the rate $1/2$ code with generator matrix

$$(1+D^2,\, 1+D+D^2). \tag{1}$$

This is a minimum encoder with constraint length 2. Its trellis diagram is shown in Fig. 1. By periodically deleting the last bit of every other branches as shown in Fig. 2, a rate $2/3$ punctured convolutional encoder can be obtained. Note that the state transitions

$$01 \longrightarrow 10 \longrightarrow 01$$

give an all-zero output. This simply means that the punctured convolutional encoder is catastrophic [5]!

Good punctured convolutional codes are found by examining a great deal of punctured encoders. Therefore, in order to speed up searching procedures, an efficient algorithm to eliminate catastrophic encoders is highly desirable.

Manuscript received July 22, 1994; revised October 25, 1995. The material in the correspondence was presented at the IEEE International Symposium on Information Theory, Whistler, BC, Canada, September 17–22, 1995.

F.-W. Sun was with the Department of Electrical Engineering, McGill University, Montreal, PQ H3A 2A7, Canada. He is now with the Advanced Development Group, Hughes Network Systems, Germantown, MD 20876 USA.

A. J. Vinck is with the Institute of Experimental Mathematics, Essen University, 45326, Essen, Germany.

Publisher Item Identifier S 0018-9448(96)02929-X.

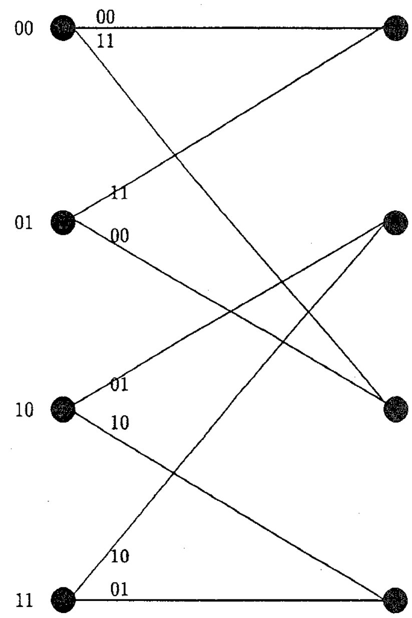

Fig. 1. Trellis diagram of (1).

An algorithm was previously proposed by Hole [5] when the punctured encoder is obtained from a rate $1/b$, $b<n$ antipodal encoder. A rate $1/b$ encoder of constraint length ν is called *antipodal* if each generator polynomial has degree ν and the constant term of each generator polynomial is equal to one.

Hole's algorithm searches for a zero-weight cycle in the punctured diagram of the original encoder. Although his algorithm is quite efficient for codes with short constraint length, its complexity grows exponentially with the constraint length of the original encoder. Thus it cannot be applied to punctured encoders with large constraint length, for instance, to those punctured codes for sequential decoding [3].

In this work, we propose an algorithm to eliminate catastrophic encoders of rate $(n-1)/n$ punctured codes when the original encoder is antipodal. The algorithm is computationally efficient for both large and small constraint lengths.

From [6], we know that a punctured convolutional encoder obtained from an antipodal encoder is noncatastrophic if and only if it is minimum. The algorithm to be presented first finds a nonzero codeword of the dual of the punctured rate $(n-1)/n$ code. Since the dual code is a rate $1/n$ code, its minimum encoder can be easily found from any nonzero codeword. Thus the overall constraint length of a minimum encoder of the dual code is determined. The constraint length of a minimum encoder is always equal to that of the minimum encoder of its dual [7]. In this way, the minimality of the punctured encoder, thus the catastrophic property, is determined.

We first review some standard definitions on punctured convolutional codes. We assume that a rate $(n-1)/n$ punctured convolutional code is obtained by puncturing a rate $1/b$ code with a $b\times(n-1)$ deleting matrix whose first column contains two 1's and the remainder of the columns contains one 1 each and all the remaining elements are zero. A 0 in the deleting matrix indicates that the corresponding coded bit is to be deleted. The columns of the deleting matrix are

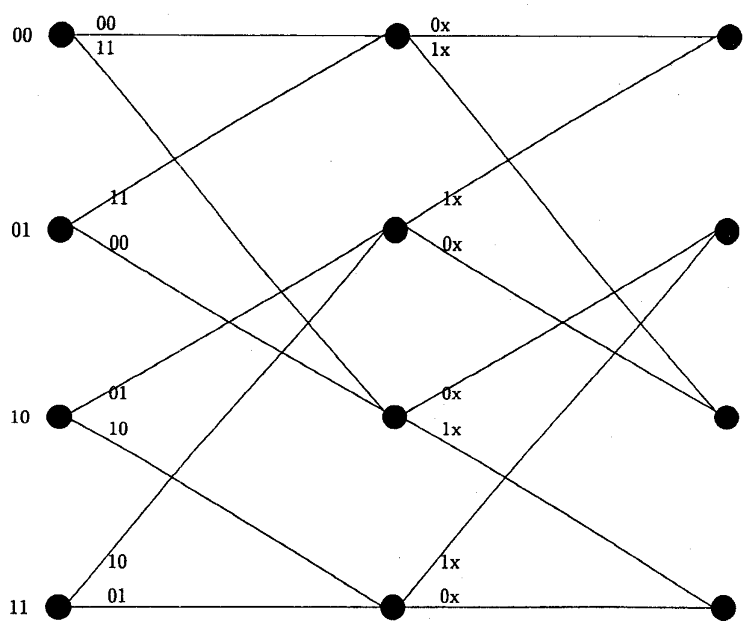

Fig. 2. A catastrophic punctured encoder from the antipodal encoder of (1).

applied to the output of the original rate $1/b$ successively. When the last column of the deleting matrix is reached, the deleting matrix is used once again. For instance, the deleting matrix of Fig. 2 is

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}. \tag{2}$$

In this way, a rate $(n-1)/n$ time-invariant convolutional code is obtained. The original rate $1/b$ code is defined by b generator polynomials

$$G^j(D) = 1 + g_1^j D + \cdots + g_{\nu-1}^j D^{\nu-1} + D^\nu \quad (1 \le j \le b)$$

where g_i^j is a binary digit and ν is the constraint length of the code.

II. The Algorithm

The major part of the algorithm looks for a nonzero finite-weight codeword of the dual of the punctured rate $(n-1)/n$ code. We shall show that this can be done by using only the original rate $1/b$ encoder and the deleting matrix.

Given a $b(n-1)$-dimensional binary vector x, an n-dimensional binary vector $pun(x)$ can be obtained by applying the deleting matrix to x column by column.

For example, for the deleting matrix

$$\begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{3}$$

and a $3 \times 2 = 6$-dimensional vector $x = (101\ 011)$, $pun(x)$ can be obtained as follows:

$$\begin{aligned} x &= (101\ 011) \longrightarrow (10x\ xx1) \longrightarrow (101) \\ &= pun(x) \end{aligned} \tag{4}$$

where "x" indicates that the corresponding bit is deleted.

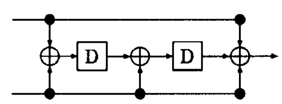

Fig. 3. The syndrome former of the dual code of (1).

Given an n-dimensional vector x, $y = ext(x)$ is defined as a $b(n-1)$-dimensional vector such that $pun(y) = x$ and the deleted bits of y are equal to zero. For example, for the deleting matrix of (3)

$$\begin{aligned} x &= (101) \longrightarrow (100\ 001) \\ &= ext(x). \end{aligned}$$

These definitions can be extended to a sequence of vectors in a straightforward way. For $\underline{x} = (\cdots, x_0, x_1, \cdots, x_k, \cdots)$, where x_i is an n-dimensional vector, $ext(\underline{x})$ is defined as

$$[\cdots, ext(x_0), ext(x_1), \cdots, ext(x_k), \cdots].$$

Similarly, for a sequence $\underline{y} = (\cdots, y_0, y_1, \cdots, y_k, \cdots)$ with components over the $b(n-1)$-dimensional binary vector space, $pun(\underline{y})$ is defined as

$$[\cdots, pun(y_0), pun(y_1), \cdots, pun(y_k), \cdots].$$

The following lemma reveals the relationship of the dual of the punctured convolutional code with the dual of the original rate $1/b$ code. It is actually an analog of a theorem in [8].

Lemma 1: A finite-weight sequence $\underline{x}$ of n-dimensional vectors is in the dual of the punctured convolutional code if and only if $ext(\underline{x})$ is in the dual of the original rate $1/b$ code.

Proof: By definition, the componentwise inner product $\underline{x} \odot pun(c)$ is equal to zero for any c in the original rate $1/b$ code. Clearly

$$\underline{x} \odot pun(c) = ext(\underline{x}) \odot c.$$

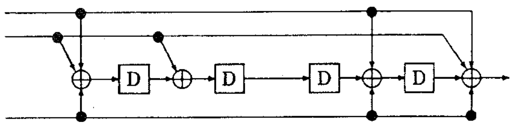

Fig. 4. Syndrome former of the dual code of (5).

Thus $ext(\underline{x}) \odot c$ is equal to zero for any codeword c in the original rate $1/b$ code. Hence, $ext(\underline{x})$ is in the dual of the original code. □

Let G be a minimum encoder of the original rate $1/b$ code with syndrome former H^T and G^T be the syndrome former of the code generated by H. The syndrome former G^T can be realized in an adjoint-obvious realization [9], involving a shift register of length ν, the constraint length of the original minimum encoder. Fig. 3 gives the syndrome former of the dual of the code generated by (1) in an adjoint-obvious realization, and Fig. 4 illustrates an adjoint-obvious realization of the transpose of the generator matrix

$$(1 + D + D^4,\, 1 + D^3 + D^4,\, 1 + D + D^4). \tag{5}$$

It was shown [9] that in its adjoint-obvious realization, the physical states of a syndrome former shadow the trellis states of a minimum dual encoder, so that the syndrome former can be used as a state-track as well.

Starting from any state, we want to find an n-dimensional vector such that $ext(x)$ is a *valid* input of the syndrome former. This means that taking $ext(x)$ as an input, and starting from this particular state, the syndrome former produces $n - 1$ zeros and transfers to another state.

For example, if the deleting map of (3) is used, and the syndrome former in Fig. 4 starts at the state (0001), we want to find a three-dimensional binary vector $a = (a_1, a_2, a_3)$ such that $ext(a) = (a_1 a_2 0\ 00 a_3)$ is a valid input of the syndrome former. Since the rightmost cell is equal to 1, $a_1 + a_2$ must be equal to one in order to produce a zero output. We may choose $a_1 = 1$, $a_2 = 0$. After $(a_1 a_2 0)$ is fed into the syndrome former, its state transfers to (1001). Clearly, a_3 must be equal to one to produce a zero output and the syndrome former transfers to the state (1101). Thus one of the choices of a is (101), which causes the syndrome former transfers from state (0001) to state (1101) via the intermediate state (1001). This intermediate state is of no interest to us since it shadows an intermediate state of the punctured convolutional code rather than a "true" trellis state.

The general principle is formulated in the following lemma.

Lemma 2: For any state of the syndrome former of the dual of an antipodal rate $1/b$ convolutional code , there exist two n-dimensional vectors, say x, x', such that when the syndrome former starts from this state, the input $ext(x)$ $[ext(x')]$ causes the syndrome former to transfer to another state with the *all-zero output*. Any one of the two vectors can be found in no more than $n(\nu + 1)$ binary operations.

Proof: At each time instant, an n-dimensional vector is fed into the syndrome former. By the antipodal property, any input bit can immediately affect the output. In order to get a zero output, the binary sum of the input n-dimensional vector at any instant must be equal to the content of the rightmost cell of the shift register. For any $a = (a_1, a_2, \cdots, a_n)$, of an n-dimensional vector, $ext(a)$ can be fed into the syndrome former in $(n - 1)$ time instants. The output of the syndrome former at the first time instant is equal to the sum of the content of the rightmost cell of the shift register and $a_1 + a_2$, Thus there are only two possible choices for (a_1, a_2). At the remaining time instants, the output is the sum of a_i and the content of the rightmost cell of the shift register. Thus a_i $(i > 2)$ and the next state are completely determined by the initial state, and (a_1, a_2).

At the first time instant, in no more than $2(\nu+1)$ binary operations, we can find the next states since there are at most two nonzero input bits. For the remaining time instant, there is at most one nonzero input bit. Thus the state transition can be found in $\nu + 1$ binary operations. In total, no more than

$$2(\nu + 1) + (n - 2)(\nu + 1) = n(\nu + 1)$$

binary operations are required to find $ext(a)$ and the next state. □

Therefore, from the all-zero state, in no more than $n(\nu + 1)^2$ binary operations we can find a sequence of n-dimensional vectors $(x_1, x_2, \cdots, x_{\nu+1})$ such that $x_1 \neq 0$ and $[ext(x_1), ext(x_2), \cdots, ext(x_{\nu+1})]$ is a valid input of the syndrome former. Assume that the syndrome former is at state S_i after $ext(x_i)$ $(1 \le i \le \nu + 1)$ has been fed into the syndrome former. Since S_i's are ν-dimensional vectors, the equation

$$\sum_{i=1}^{\nu+1} b_i S_i = 0 \tag{6}$$

must have nontrivial solutions, where b_i's are binary digits. Let $(b_1^*, \cdots, b_{\nu+1}^*)$ be such a solution.

Since

$$[ext(x_1), ext(x_2), \cdots, ext(x_{\nu+1})]$$

is a valid input sequence of the syndrome former with state transitions $(0, S_1, \cdots, S_{\nu+1})$, the sequence

$$[(0, \cdots, 0, ext(x_1), \cdots, ext(x_i)] \quad (1 \le i \le \nu + 1)$$

is also valid input sequence with state transitions $(0, \cdots, 0, S_1, \cdots, S_i)$. From the linearity of the syndrome former, the input sequence

$$\sum_{i=1}^{\nu+1} b_i^* [0, \cdots, 0, ext(x_1), \cdots, ext(x_i)] \tag{7}$$

is also a valid input sequence with state transitions

$$\sum_{i=1}^{\nu+1} b_i^* (0, \cdots, 0, S_1, \cdots, S_i).$$

By the definition of b_i^*'s, we know that this state sequence ends up at the all-zero state. Therefore, (7) is actually a codeword of the dual of the original rate $1/b$ code. In view of Lemma 1, this implies that

$$\sum_{i=1}^{\nu+1} b_i^* (0, \cdots, 0, x_1, \cdots, x_i)$$

is a *nonzero* codeword of the dual of the punctured rate $(n - 1)/n$ code. We can represent this codeword in n polynomials, say $[c_1(D), \cdots, c_n(D)]$. If $\deg(c_i(D)) < \nu$ for all i or the degree of the greatest common divisor of $c_i(D)$ is no less than 1, the overall constraint length of the minimum encoder of the dual of the punctured convolutional code, thus that of the minimum encoder of the punctured convolutional code itself, is less than ν. From [6], this means that it is catastrophic.

We summarize the algorithm as follows.

III. The Algorithm

1) Initialize the adjoint-obvious realization [9] of G^T as the all-zero state.
2) Find a sequence of n-dimensional vectors $(x_1, x_2, \cdots, x_{\nu+1})$ such that $x_1 \neq 0$ and

$$[ext(x_1), ext(x_2), \cdots, ext(x_{\nu+1})]$$

is a valid input sequence of the syndrome former with state transitions $(0, S_1, S_2, \cdots, S_{\nu+1})$.

3) Find a nontrivial solution $(b_1^*, \cdots, b_{\nu+1}^*)$ of the equation

$$\sum_{i=1}^{\nu+1} b_i S_i = 0.$$

4) Calculate the sum

$$\underline{y} = \sum_{i=1}^{\nu+1} b_i^*(0, \cdots, 0, x_1, \cdots, x_i).$$

5) Represent $\underline{y}$ in n polynomials.

6) If all the degrees of the n polynomials are less than ν or the degree of their greatest common divisor is larger than one, the punctured convolutional encoder is catastrophic.

END

Example 1: Consider the rate $2/3$ punctured convolutional code obtained from (5) with deleting matrix (3). The following sequence can be easily found by the procedure described in the proof of Lemma 2:

$$[ext(111)\ ext(100)\ ext(001)\ ext(000)\ ext(001)].$$

This is a valid input of the syndrome former realized in Fig. 4 when it starts from the all-zero state. The corresponding state transitions (omitting the intermediate states) are

$$\begin{aligned}(0000) &\longrightarrow (1011) \longrightarrow (0110) \\ &\longrightarrow (1000) \longrightarrow (0010) \longrightarrow (1001).\end{aligned}$$

Clearly, the equation

$$b_1(1011) + b_2(0110) + b_3(1000) + b_4(0010) + b_5(1001) = 0$$

has the nontrivial solution (10011). Therefore, the sum

$$\begin{array}{cccccc} & 000 & 000 & 000 & 000 & 111 \\ + & 000 & 111 & 100 & 001 & 000 \\ + & 111 & 100 & 001 & 000 & 001 \\ \hline = & 111 & 011 & 101 & 001 & 110 \end{array}$$

is a nonzero codeword of the dual of the rate $2/3$ punctured code. In terms of polynomials, this codeword is

$$\begin{aligned}(111) + (011)D + (101)D^2 + (001)D^3 + (110)D^4 \\ = (1 + D^2 + D^4,\ 1 + D + D^4,\ 1 + D + D^2 + D^3).\end{aligned}$$

Note that $1 + D + D^2 + D^3 = (1 + D)^3$ and none of the first and the second polynomial can be divided by $(1 + D)$. Thus there is no nontrivial common divisor among the three polynomials. This proves that the rate $2/3$ punctured encoder is not catastrophic.

Example 2: This example considers the rate $2/3$ punctured code obtained from (1) with deleting matrix (2). The syndrome former is presented in Fig. 3. Clearly, the sequence $[ext(111)\ ext(001)]$ is a valid input sequence for the all-zero state with state transitions $00 \to 10 \to 10$. Thus

$$(111\,001) + (000\,111) = (111\,110)$$

is a nonzero codeword of the dual code of the punctured code. The polynomial representation of the codeword is

$$(111) + (110)D = (1 + D,\ 1 + D,\ 1).$$

The highest degree of the polynomials is less than 2 so that the punctured encoder is catastrophic.

In order to get good codes, the constraint length of the original encoder ν should not be less than $(n - 1)$. Otherwise, the minimum distance of the punctured convolutional code will be equal to one. Therefore, the complexity of the algorithm is dominated by that of solving the linear equation of (6), which requires $O(\nu^3)$ binary operations when Gaussian elimination is used. This compares very favorably with that of [5], which requires $O(n2^\nu)$ binary operations. As mentioned in [5], We can also find a $(n - 1) \times n$ polynomial encoder for the punctured convolutional code, which has the same encoding mapping as the punctured encoder. Then the classical technique [10] can be applied. Namely, all the determinants of distinct $(n - 1)$ by $(n - 1)$ submatrix of the generator are calculated. If the greatest common divisor of these determinants is not of the form D^i for some $i \geq 0$, the encoder is catastrophic. The current approach is advantageous since the catastrophic property can be determined directly from the generator matrix of the original encoder.

References

[1] J. B. Cain, G. C. Clark, Jr., and J. M. Geist, "Punctured convolutional codes of rate $(n-1)/n$ and simplified maximum likelihood decoding," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 97–100, Jan. 1979.

[2] Y. Yasuda, K. Kashiki, and Y. Hirata, "High-rate punctured convolutional codes for soft decision Viterbi decoding," *IEEE Trans. Commun.*, vol. COM-32, pp. 315–319, Mar. 1984.

[3] D. Haccoun and G. Begin, "High-rate punctured convolutional codes for Viterbi and sequential decoding," *IEEE Trans. Commun.*, vol. 37, pp. 1113–1125, Nov. 1989.

[4] J. Hagenauer, "Rate compatible punctured convolutional codes and their applications," *IEEE Trans. Commun.*, vol. 36, pp. 389–400, Apr. 1988.

[5] K. J. Hole, "An algorithm for determining if a rate $(n-1)/n$ punctured convolutional encoder is catastrophic," *IEEE Trans. Commun.*, vol. 39, pp. 386–389, Mar. 1991.

[6] ——, "Rate $k/(k+1)$ minimum punctured convolutional encoders," *IEEE Trans. Inform. Theory*, vol. 37, pp. 653–655, May 1991.

[7] G. D. Forney Jr., "Convolutional codes I: Algebraic structure," *IEEE Trans. Inform. Theory*, vol. IT-16, pp. 720–738, Nov. 1970.

[8] ——, "Dimension/length profiles and trellis complexity of linear block codes," to appear in *IEEE Trans. Inform. Theory.*

[9] ——, "Structural analysis of convolutional codes via dual codes," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 512–518, July 1973.

[10] J. L. Massey and M. K. Sain, "Inverses of linear sequential circuits," *IEEE Trans. Comput.*, vol. C-17, pp. 330–337, Apr. 1968.

[11] D. Lavoie and D. Haccoun, "On the determination of catastrophic convolutional encoders," in *Proc. 1994 Int. Symp. on Information Theory* (Trondheim, Norway, June 1994), p. 20.