Reduction of Error-Trellises for Tail-Biting Convolutional Codes Using Shifted Error-Subsequences

Masato Tajima  
Graduate School of Sci. and Eng.  
University of Toyama  
3190 Gofuku, Toyama 930-8555, Japan  
Email: tajima@eng.u-toyama.ac.jp

Koji Okino  
Information Technology Center  
University of Toyama  
3190 Gofuku, Toyama 930-8555, Japan  
Email: okino@itc.u-toyama.ac.jp

Tatsuto Murayama  
Graduate School of Sci. and Eng.  
University of Toyama  
3190 Gofuku, Toyama 930-8555, Japan  
Email: murayama@eng.u-toyama.ac.jp

Abstract—In this paper, we discuss the reduction of error-trellises for tail-biting convolutional codes. In the case where some column of a parity-check matrix has a monomial factor $D^l$, we show that the associated tail-biting error-trellis can be reduced by cyclically shifting the corresponding error-subsequence by $l$ (i.e., the power of $D$) time units. We see that the resulting reduced error-trellis is again tail-biting. Moreover, we show that reduction is also possible using backward-shifted error-subsequences.

I. INTRODUCTION

Tail-biting is a technique by which a convolutional code can be used to construct a block code without any loss of rate [4], [6], [14]. Let $C_{tb}$ be a tail-biting convolutional code with an $N$-section code-trellis $T_{tb}^{(c)}$. The fundamental idea behind tail-biting is that the encoder starts and ends in the same state, i.e., $\beta_0 = \beta_N$ ($\beta_k$ is the encoder state at time $k$). Suppose that $T_{tb}^{(c)}$ has $\Sigma_0$ initial (or final) states, then it is composed of $\Sigma_0$ subtrellises, each having the same initial and final states. We call these subtrellises tail-biting code subtrellises. For example, a tail-biting code-trellis of length $N = 4$ based on the generator matrix

$$G_1(D) = (D + D^2, D^2, 1 + D)$$  

(1)

is shown in Fig.1. Since $\Sigma_0 = 4$, this tail-biting code-trellis is composed of 4 code subtrellises. In Fig.1, bold lines correspond to the code subtrellis with $\beta_0 = \beta_4 = (1, 1)$. On the other hand, it is reasonable to think that an error-trellis $T_{tb}^{(e)}$ for the tail-biting convolutional code $C_{tb}$ can equally be constructed. In this case, each error subtrellis should have the same initial and final states like a code subtrellis. In our previous works [11], [12], taking this property into consideration, we have presented an error-trellis construction for tail-biting convolutional codes. For example, consider the above case. The parity-check matrix $H_1(D)$ associated with $G_1(D)$ is given by

$$H_1(D) = \begin{pmatrix} 1 & 0 & D \\ D & 1 + D & 0 \end{pmatrix}.$$  

(2)

Let $z = z_1 z_2 z_3 z_4 = 110 101 101 011$ be the received data. In this case, using the method in [11], the tail-biting error-trellis corresponding to the code-trellis in Fig.1 can be constructed as is shown in Fig.2, where bold lines correspond to the error subtrellis with $\sigma_0 = \sigma_4 = (1, 0)$.

On the other hand [9], note that the third column of $H_1(D)$ has the monomial factor $D$. Let $e_k = (e_k^{(1)}, e_k^{(2)}, e_k^{(3)})$
and \( \zeta_k = (\zeta^{(1)}_k, \zeta^{(2)}_k) \) be the time-\( k \) error and syndrome, respectively. We have the following modification (\( T \) means transpose):

\[
\zeta_k = e_k H^T(D)
\]

\[
= (e^{(1)}_k, e^{(2)}_k, e^{(3)}_k)
\]

\[
= (e^{(1)}_k, e^{(2)}_k, e^{(3)}_k)
\]

\[
= (e^{(1)}_k, e^{(2)}_k, D e^{(3)}_k)
\]

where \( \tilde{e}_k = (e^{(1)}_k, e^{(2)}_k, e^{(3)}_k) \) and \( e^{(3)}_k \) is defined as \( D e^{(3)}_k = e^{(3)}_{k-1} \). Since the overall constraint length \( \tilde{\nu} \) of

\[
\tilde{H}_1(D) = \begin{pmatrix} 1 & 0 & 0 \\ D & 1 + D & 0 \end{pmatrix}
\]

is one, the above equation implies that the tail-biting error-trellis in Fig.2 can be reduced by shifting the subsequence \{\( e^{(3)}_k \)\} by the unit time.

In this paper, taking the above example into account, we discuss the reduction of error-trellises for tail-biting convolutional codes. It is assumed that some (\( j \)th) column of a parity-check matrix \( H(D) \) has a monomial factor \( D^j \). In this case, we show that the associated tail-biting error-trellis can be reduced by cyclically shifting the \( j \)th component \( e^{(2)}_k \) by \( l_j \) time units. We also show that the resulting reduced error-trellis is again tail-biting. We see that a kind of “periodicity” inherent in tail-biting trellises plays a key role in our discussion.

II. PRELIMINARIES

In this paper, we always assume that the underlying field is \( F = GF(2) \). Let \( G(D) \) be a generator matrix of an \((n, n-m)\) convolutional code \( C \). Let \( H(D) \) be a corresponding \( m \times n \) parity-check matrix of \( C \). Both \( G(D) \) and \( H(D) \) are assumed to be canonical [1], [5]. Denote by \( \nu \) the overall constraint length of \( H(D) \) and by \( M \) the memory length of \( H(D) \) (i.e., the maximum degree among the polynomials of \( H(D) \)). Then \( H(D) \) is expressed as

\[
H(D) = H_0 + H_1D + \cdots + H_mD^M.
\]

A. Adjoint-Obvious Realization of a Syndrome Former

Consider the adjoint-obvious realization (observer canonical form [2], [3]) of the syndrome former \( H^T(D) \). Let \( e_k = (e^{(1)}_k, e^{(2)}_k, \cdots, e^{(n)}_k) \) and \( \zeta_k = (\zeta^{(1)}_k, \zeta^{(2)}_k, \cdots, \zeta^{(m)}_k) \) be the input error and the corresponding output syndrome at time \( k \), respectively. Denote by \( e^{(q)}_{kp} \) the contents of the memory elements in the above realization. (If a memory element is missing, the corresponding \( e^{(q)}_{kp} \) is set to zero.) Using \( e^{(q)}_{kp} \), the syndrome-former state at time \( k \) is defined as

\[
\sigma_k \triangleq (\sigma^{(1)}_{k1}, \cdots, \sigma^{(m)}_{k1}, \cdots, \sigma^{(1)}_{km}, \cdots, \sigma^{(m)}_{km}).
\]

(Remark: The effective size of \( \sigma_k \) is equal to \( \nu \).)

For example, Fig.3 illustrates the adjoint-obvious realization of the syndrome former \( H^T_2(D) \) [1], where

\[
H_2(D) = \begin{pmatrix} D^2 + D^3 & D & 1 + D & D^2 & D^2 \\ D^2 & 1 + D + D^2 & D^2 & 1 & D \end{pmatrix}.
\]

Hence, we have

\[
\sigma_k = (\sigma^{(1)}_{k1}, \sigma^{(2)}_{k1}, \sigma^{(1)}_{k2}, \sigma^{(2)}_{k2}, \sigma^{(1)}_{k3}, 0).
\]

Note that the effective size of \( \sigma_k \) is \( \nu = 5 \).

Under the above conditions [7], [8], we have

\[
\sigma_k = (\sigma^{(2)}_{k-1}, \cdots, \sigma^{(M)}_{k-1}, 0) + e_k (H^T_1, H^T_2, \cdots, H^T_M). \]

Similarly, \( \zeta_k \) is expressed as

\[
\zeta_k = e_{k-M}H^T_M + e_{k-M+1}H^T_M + e_kH^T_0.
\]

B. Dual States

The encoder states can be labeled by the syndrome-former states (i.e., dual states [2]). The dual state \( \beta^*_k \) corresponding to an encoder state \( \beta_k \) is obtained by replacing \( e_k \) in \( \sigma_k \) by \( y_k = u_k G(D) \) (\( y_k \) is the code symbol at time \( k \) corresponding to the information symbol \( u_k \)). We have

\[
\beta^*_k = (y_{k-M+1}, \cdots, y_{k-1}, y_k)
\]

\[
\times \begin{pmatrix} H^T_M & 0 & 0 \\ H^T_2 & H^T_M & 0 \\ H^T_1 & H^T_{M-1} & H^T_M \end{pmatrix}.
\]

Example 1: Consider the parity-check matrix \( H_1(D) \). We have

\[
H_1(D) = \begin{pmatrix} 1 & 0 & 0 \\ D & 1 + D & 0 \end{pmatrix}
\]

\[
= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} D
\]

\[
\triangleq H_0 + H_1D.
\]
Then the dual state corresponding to an encoder state $\beta_k = (u_{k-1}, u_k)$ is obtained as follows.

$$\beta_k' = y_k H_1^T$$
$$= (y_k^{(1)}, y_k^{(2)}, y_k^{(3)}) \begin{pmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}$$
$$= (y_k^{(3)}, y_k^{(1)} + y_k^{(2)})$$
$$= (u_{k-1} + u_k, u_{k-1}). \quad (16)$$

C. Error-Trellises for Tail-Biting Convolutional Codes

Suppose that a tail-biting code-trellis based on $G(D)$ is defined in $[0, N]$, where $N \geq M$. In this case, the corresponding tail-biting error-trellis based on $H_T(D)$ is constructed as follows [11].

Step 1: Let $z = \{z_k\}_{k=1}^N$ be a received data. Denote by $\sigma_0$ the initial state of the syndrome former $H_T(D)$. Let $\sigma_{fin} (= \sigma_N)$ be the final syndrome-former state corresponding to the input $z$. Note that $\sigma_{fin}$ is independent of $\sigma_0$ and is uniquely determined only by $z$.

Step 2: Set $\sigma_0$ (i.e., the initial state of the syndrome former) to $\sigma_{fin}$ and input $z$ to the syndrome former again. Here, suppose that the syndrome sequence $\zeta = \{\zeta_k\}_{k=1}^N$ is obtained.

(Remark: $\zeta_k$ ($k \geq M + 1$) has been obtained in Step 1.)

Step 3: Concatenate the error-trellis modules corresponding to the syndromes $\zeta_k$. Then we have the tail-biting error-trellis.

Example 2: Consider the parity-check matrix $H_1(D)$. Let

$$z = z_1 \ z_2 \ z_3 \ z_4 = 110 \ 101 \ 101 \ 011 \quad (17)$$

be the received data. According to Step 1, let us input $z$ to the syndrome former $H_T^T(D)$. Then we have $\sigma_{fin} = (1, 1)$. Next, set $\sigma_0$ to $\sigma_{fin} = (1, 1)$ and input $z$ to the syndrome former again. In this case, the syndrome sequence

$$\zeta = \zeta_1 \ \zeta_2 \ \zeta_3 \ \zeta_4 = 00 \ 10 \ 01 \ 10 \quad (18)$$

is obtained. The tail-biting error-trellis is constructed by concatenating the error-trellis modules corresponding to $\zeta_k$. The resulting tail-biting error-trellis is shown in Fig.2.

With respect to the correspondence between tail-biting code subtrellises and tail-biting error subtrellises, we have the following [11], [12].

Proposition 1: Let $\beta_0 (= \beta_N) = \beta$ be the initial (final) state of a tail-biting code-trellis. Then the initial (final) state of the corresponding tail-biting error subtrellis is given by $\sigma_{fin} + \beta^*$. Example 2 (Continued): Consider the tail-biting error-trellis in Fig.2. In this example, we have $\sigma_{fin} = (1, 1)$. The corresponding tail-biting code-trellis based on $G_1(D)$ is shown in Fig.1. In Fig.1, take notice of the code subtrellis with initial (final) state $\beta = (1, 1)$ (bold lines). The dual state of $\beta = (1, 1)$ is calculated as $\beta^* = (u_{-1} + u_0, u_{-1}) = (1 + 1, 1) = (0, 1)$. Hence, the initial (final) state of the corresponding error subtrellis is given by $\sigma_{fin} + \beta^* = (1, 1) + (0, 1) = (1, 0)$ (bold lines in Fig.2).

III. REDUCTION OF TAIL-BITING ERROR-TRELLISES

A. Error-Trellis Reduction Using Shifted Error-Subsequences

Consider the example in Section I. Noting the relation $\beta_k^{(3)} = \alpha_k^{(3)}$, we cyclically shift the third component of each $z_k$ to the right by the unit time. Then we have the modified received data

$$z = \tilde{z}_1 \ \tilde{z}_2 \ \tilde{z}_3 \ \tilde{z}_4 = 111 \ 100 \ 101 \ 011. \quad (19)$$

Applying the method in Section II-C, we can construct a reduced tail-biting error-trellis. According to Step 1, let us input $\tilde{z}$ to the syndrome former $\tilde{H}_T^T(D)$. Then we have $\tilde{\sigma}_{fin} = (1)$. Next, set $\tilde{\sigma}_0$ to $\tilde{\sigma}_{fin} = (1)$ and input $\tilde{z}$ to the syndrome former again. In this case, the same syndrome sequence as the original one (i.e., $\zeta = 00 \ 10 \ 01 \ 10$) is obtained. The reduced tail-biting error-trellis is constructed by concatenating the reduced error-trellis modules corresponding to $\tilde{\zeta}_k$. The resulting tail-biting error-trellis is shown in Fig.4. Here let us examine how a tail-biting error subtrellis is embedded in the corresponding reduced error-trellis. For the purpose, take notice of the subtrellis with initial (final) state $(1, 0)$ (bold lines in Fig.2). First, consider where the state $(1, 0)$ is mapped to. In the original error-trellis, the final state $\sigma_N$ is expressed as

$$\sigma_N = e_N H_1^T = (e_N^{(3)}, e_N^{(1)} + e_N^{(2)}). \quad (20)$$

Using the relation $e_N^{(3)} = e_N^{(3)} + e_N^{(1)} + e_N^{(2)}$ is modified as $e_N^{(3)} + e_N^{(1)} + e_N^{(2)}$. Since the subscript $N+1$ is inappropriate for the state at time $N$, we have

$$\tilde{\sigma}_N = e_N^{(1)} + e_N^{(2)} = (e_N H_1^T). \quad (21)$$

(Remark: We have $\tilde{e}_N H_1^T = (0, e_N^{(1)} + e_N^{(2)})$. Hence, the first component can be deleted.) That is, state $\sigma_N = (1, 0)$ is mapped to $\tilde{\sigma}_N = (0)$. Next, consider an arbitrary error-path $e_p = e_1 \ e_2 \ e_3 \ e_4$ in the subtrellis with initial (final) state $(1, 0)$. Here, take notice of two sections from $t = 0$ to $t = 1$ and from $t = 3$ to $t = 4$. Note that these are adjacent sections in the circular error-trellis. From $\sigma_4 = (e_4^{(3)}, e_4^{(1)} + e_4^{(2)}) = (1, 0)$, we have $e_4^{(3)} = 1$. Since the third component of each $e_k$ is cyclically shifted to the right by the unit time, $e_4^{(3)}$ is replaced by $e_4^{(1)} = 1$. That is, the third label on the first branch of the error-path in the reduced trellis must be 1. By taking account of these conditions, we have four admissible error-paths:

$$\tilde{e}_{p_1} = 101 110 010 110$$
$$\tilde{e}_{p_2} = 101 110 111 001$$
$$\tilde{e}_{p_3} = 101 011 000 001$$
$$\tilde{e}_{p_4} = 101 011 101 110.$$
reduced tail-biting error subtrellis. The original error-paths are restored by noting the relation $e_k^{(3)} = e_{k+1}^{(3)}$. That is, we only need to cyclically shift the third component of each $\tilde{e}_k = (e_k^{(1)}, e_k^{(2)}, e_k^{(3)})$ to the left by the unit time. As a result, four error-paths

$$
\begin{align*}
\tilde{e}_{q_1} & = 100 \quad 110 \quad 010 \quad 111 \\
\tilde{e}_{q_2} & = 100 \quad 111 \quad 111 \quad 001 \\
\tilde{e}_{q_3} & = 101 \quad 010 \quad 001 \quad 001 \\
\tilde{e}_{q_4} & = 101 \quad 011 \quad 100 \quad 111 \\
\end{align*}
$$

are obtained. We see that these paths completely coincide with those in Fig. 2.

B. General Cases

The argument in the previous subsection, though it was presented in terms of a specific example, is entirely general. Hence, it can be directly extended to general cases. Suppose that a specific $(j)$th column of $H(D)$ has the form

$$
\begin{pmatrix}
D^{l_j} \hat{h}_{1j}(D) & D^{l_j} \hat{h}_{2j}(D) & \ldots & D^{l_j} \hat{h}_{mj}(D)
\end{pmatrix}^T,
$$

where $1 \leq l_j \leq M$. (Remark: A more general case where each column has the above form can also be treated.) Let $\hat{H}(D)$ be the modified version of $H(D)$ with the $j$th column being replaced by

$$
\begin{pmatrix}
\hat{h}_{1j}(D) & \hat{h}_{2j}(D) & \ldots & \hat{h}_{mj}(D)
\end{pmatrix}^T.
$$

$\hat{H}(D)$ is assumed to be canonical. In this case, the reduction of a tail-biting error-trellis is accomplished as follows.

(i) **Fundamental relation:** Denote by $e_k = (e_k^{(1)}, \ldots, e_k^{(n)})$ and $\zeta_k = (\zeta_k^{(1)}, \ldots, \zeta_k^{(m)})$ the time-$k$ error and syndrome, respectively. Also, let $\tilde{e}_k = (e_k^{(1)}, \ldots, e_k^{(n)}\sigma_k^{(m)})$, where $< t >$ denotes $t \mod N$. Then we have

$$
\zeta_k = \tilde{e}_k \hat{H}^T(D).
$$

(ii) **Construction of reduced tail-biting error-trellises:** Let

$$
z = \{z_k\}_{k=1}^N = \{(z_k^{(1)}, \ldots, z_k^{(j)}, \ldots, z_k^{(n)})\}_{k=1}^N
$$

be a received data. We construct the modified received data

$$
\tilde{z} = \{\tilde{z}_k\}_{k=1}^N = \{(z_k^{(1)}, \ldots, z_k^{(j)}, \ldots, z_k^{(n)})\}_{k=1}^N
$$

by cyclically shifting the $j$th component of each $z_k$ to the right by $l_j$ time units. By applying the method in Section II-C to the modified syndrome former $\hat{H}^T(D)$ and the modified received data $\tilde{z}$, a reduced tail-biting error-trellis is constructed. Note that the same syndrome sequence $\{\zeta_k\}$ as for the tail-biting error-trellis based on $H^T(D)$ is obtained.

(iii) **Reduced tail-biting error subtrellises:** Let $ST_{tb}^{(c)}(\sigma_N)$ be a tail-biting error subtrellis with initial (final) state $\sigma_N$. $\sigma_N$ can be expressed using $\{e_t\}_{t=0}^{N-M+1}$ (cf. (10)). Here replace each $e_t^{(j)} \quad (N - M + 1 \leq t \leq N)$ by $\sigma_t^{(j)}$ and delete those terms $e_t^{(j)}$ with subscript $t$ greater than $N$. Denote by $\sigma_N$ the resulting state expression. In this case, state $\sigma_N$ is mapped to state $\tilde{\sigma}_N$ in the reduced tail-biting error-trellis.

Consider the two trellis-sections from $t = 0$ to $t = l_j$ and from $t = N - l_j$ to $t = N$. Note that these form a continuous section of length $2l_j$ in the circular error-trellis. Now we can solve Eq. (10) ($k = \overline{N}$) given $\sigma_N$. (Remark: $\{e_t^{(j)}\}_{t=0}^{N-M+1}$ is uniquely determined under a moderate condition on $H(D)$.) Since the $j$th component of each $e_k$ is cyclically shifted to the right by $l_j$ time units, $e_t^{(j)} \quad (1 \leq t \leq l_j)$ is replaced by $e_{N-l_j+t}^{(j)}$. That is, the $j$th component $\sigma_t^{(j)}$ of the reduced path-segment $\tilde{e}_t \quad (1 \leq t \leq l_j)$ must be $e_{N-l_j+t}^{(j)}$. We call these segments “admissible”. Then $ST_{tb}^{(c)}$ is embedded in the reduced tail-biting error subtrellis with initial (final) state $\tilde{\sigma}_N$, where the path-segments in the first $l_j$ sections are restricted to admissible ones.

(iv) **Restoration of the original error-paths:** The original error-paths are restored by noting the relation $e_k^{(j)} = \sigma_k^{(j)}$. That is, for an error-path

$$
\tilde{e} = \{\tilde{e}_k\}_{k=1}^N = \{(e_k^{(1)}, \ldots, e_k^{(j)}, \ldots, e_k^{(n)})\}_{k=1}^N,
$$

we only need to cyclically shift the $j$th component of each $\tilde{e}_k$ to the left by $l_j$ time units.

We remark that $z$ has been periodically extended in both directions and this periodicity is fully used for tail-biting error-trellis construction. Now the relation $\zeta_k = e_k \hat{H}^T(D)$ is equivalently modified as $\zeta_k = \tilde{e}_k \hat{H}^T(D)$. Note that the correspondence between $\{e_k\}$ and $\{\tilde{e}_k\}$ is one-to-one $(\{e_k^{(j)}\}$ is cyclically shifted). Hence, the original error-path $e = \{e_k\}$ is indirectly represented using the reduced tail-biting error-trellis based on $\hat{H}(D)$. (Accordingly, the restoration in (iv) is required.) Notice that the overall constraint length $\sigma^\perp$ of $\hat{H}(D)$ is not more than $\nu^\perp$. Thus we have shown the following.

**Proposition 2:** Let $T_{tb}^{(c)}$ be a tail-biting error-trellis based on $H^T(D)$, where the $j$th column of $H(D)$ has a monomial factor $D^{l_j}$. Also, suppose that $\nu^\perp < \nu^\perp$. Then $T_{tb}^{(c)}$ can be reduced by cyclically shifting the $j$th subsequence of $\{e_k\}$ by $l_j$ time units. In this case, the reduced error-trellis $\tilde{T}_{tb}^{(c)}$ is again tail-biting.

C. Error-Trellis Reduction Using Backward-Shifted Error-Subsequences

A reduced tail-biting error-trellis can be constructed not only using forward-shifted error-subsequences but also using “backward-shifted” error-subsequences [9]. For example, con-
Simultaneous code/error-trellis reduction for convolutional codes.

Consider the parity-check matrix in (8):

$$H_2(D) = \begin{pmatrix} D^2 + D^3 & 1 + D & 1 \\ D^2 & 1 + D + D^2 & 1 \end{pmatrix}. $$

Since, the first column has the monomial factor $D^2$, $H_2(D)$ can be reduced to

$$\tilde{H}_2(D) = \begin{pmatrix} 1 + D & D \\ 1 & 1 + D + D^2 \end{pmatrix} \begin{pmatrix} D^2 \\ D \end{pmatrix} \begin{pmatrix} 1 \\ D \end{pmatrix}. $$

This matrix can be reduced to an equivalent canonical parity-check matrix $\tilde{H}_2(D)$. (Note that the first and second rows of $H_2(D)$ are just delayed versions of the first and second rows of $\tilde{H}_2(D)$.)

IV. REDUCTION OF TAIL-BITING CODE-TRELLIS

A code-trellis for a tail-biting convolutional code and the corresponding error-trellis are dual to each other. Hence, the reduction of tail-biting code-trellises is also possible. For example, consider the generator matrix $G_1(D)$ in (1). Observe that the first and second columns of $G_1(D)$ have the monomial factor $D$. This fact enables reduction of the original tail-bitting code-trellis. Let $u_k$ and $y_k = (y_k^{(1)}, y_k^{(2)}, y_k^{(3)})$ be the information and code symbol at time $k$, respectively. Then the relation $y_k = u_k \tilde{G}_1(D)$ is equivalently modified as $\tilde{y}_k = u_k \tilde{G}_1(D)$. Here,

$$\tilde{y}_k = (\tilde{y}_k^{(1)}, \tilde{y}_k^{(2)}, \tilde{y}_k^{(3)}) = \begin{pmatrix} y_k^{(1)} \\ y_k^{(2)} \\ y_k^{(3)} \end{pmatrix}. $$

and $\tilde{G}_1(D)$ is defined as

$$\tilde{G}_1(D) = (1 + D, D, 1 + D). $$

Using a similar argument as that in Section III-A, a reduced tail-biting code-trellis associated with the one in Fig.1 is constructed. The resulting reduced code-trellis is shown in Fig.5, where bold lines correspond to the original code subtrellis with $\beta_0 = \beta_4 = (1, 1)$. Note that the first two labels on each branch of the error-path are shifted to the left (i.e., backward-shifted) by the unit time. Accordingly, the path-segment from $t = 3$ to $t = 4$ is restricted to 010. We see that this specific example can be directly extended to general cases. We also remark that the reduction of a tail-biting code-trellis and that of the corresponding tail-biting error-trellis can be accomplished simultaneously, if reduction is possible (cf. [10]).

V. CONCLUSION

In this paper, we have discussed the reduction of error-trellises for tail-biting convolutional codes. In the case where a given parity-check matrix $H(D)$ has a monomial factor $D^j$ in some column, we have shown that the associated tail-biting error-trellis can be reduced by cyclically shifting the corresponding error-subsequence by $l$ time units. We have also shown that the obtained reduced error-trellis is again tail-biting. Moreover, we have shown that trellis-reduction is also accomplished using backward-shifted error-subsequences. The proposed method has been applied to concrete examples and it has been confirmed that each subtrellis is successfully embedded in the reduced tail-biting error-trellis. Finally, we have shown that the associated tail-biting code-trellis can equally be reduced using shifted code-subsequences. We remark that the convolutional code specified by a parity-check matrix $H(D)$ with the form discussed in the paper has a relatively poor distance property. We also remark that such parity-check matrices appear, for example, in [13].

REFERENCES

[1] M. Ariel and J. Snyders, “Error-trellises for convolutional codes—Part I: Construction,” IEEE Trans. Commun., vol. 46, no. 12, pp. 1592–1601, Dec. 1998.
[2] G. D. Forney, Jr., “Structural analysis of convolutional codes via dual syndromes,” IEEE Trans. Inform. Theory, vol. IT-19, no. 4, pp. 512–518, July 1973.
[3] R. Johannesson and K. S. Zigangirov, Fundamentals of Convolutional Coding. New York: IEEE Press, 1999.
[4] H. H. Ma and J. K. Wolf, “On tail biting convolutional codes,” IEEE Trans. Commun., vol. COM-34, no. 2, pp. 104–111, Feb. 1986.
[5] R. J. McEliece and W. Lin, “The trellis complexity of convolutional codes,” IEEE Trans. Inform. Theory, vol. 42, no. 6, pp. 1855–1864, Nov. 1996.
[6] R. Y. Shao, S. Lin, and M. P. C. Fossorier, “Two decoding algorithms for tailbiting codes,” IEEE Trans. Commun., vol. 51, no. 10, pp. 1658–1665, Oct. 2003.
[7] V. Sidorenko and V. Zyablov, “Decoding of convolutional codes using a syndrome trellis,” IEEE Trans. Inform. Theory, vol. 40, no. 5, pp. 1663–1666, Sept. 1994.
[8] M. Tajima, K. Okino, and T. Miyagoshi, “Minimal code(error)-trellis module construction for rate-k/n convolutional codes: Extension of Yamada-Harashima-Miyakawa’s construction,” IEICE Trans. Fundamentals, vol. E90-A, no. 11, pp. 2629–2634, Nov. 2007.
[9] M. Tajima, K. Okino, and T. Miyagoshi, “Minimal code(error)-trellis module construction for rate-k/n convolutional codes: Extension of Yamada-Harashima-Miyakawa’s construction,” IEICE Trans. Fundamentals, vol. E90-A, no. 11, pp. 2629–2634, Nov. 2007.
[10] M. Tajima, K. Okino, “Simultaneous code/error-trellis reduction for convolutional codes using shifted code/error-subsequences,” in Proc. 2011 IEEE Int. Symp. Inf. Theory, pp. 648–652, Oct. 2011.
[11] M. Tajima and K. Okino, “Error-trellis construction for tailbiting convolutional codes,” IEICE Technical Report, IT2011-59, pp. 85–90, March 2012.
[12] M. Tajima and K. Okino, “Error-trellises for tailbiting convolutional codes,” in Proc. ISITA2012, pp. 648–652, Oct. 2012.
[13] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja, and D. J. Costello, Jr., “LDPC block and convolutional codes based on circulant matrices,” IEEE Trans. Inform. Theory, vol. 50, no. 12, pp. 2966–2984, Dec. 2004.
[14] M. B. S. Tavare, K. S. Zigangirov, and G. P. Fettweis, “Tail-biting LDPC convolutional codes,” in Proc. 2007 IEEE Int. Symp. Inf. Theory, pp. 2341–2345, June 2007.