

Parallel concatenated convolutional codes from linear systems theory viewpoint

Joan-Josep Climent^a, Victoria Herranz^b, Carmen Perea^b

^a*Departament de Matemàtiques, Universitat d'Alacant
Campus de Sant Vicent del Raspeig
Apartat de correus 99, E-03080 Alacant. Spain*

^b*Centro de Investigación Operativa, Departamento de Estadística, Matemáticas e Informática
Universidad Miguel Hernández
Av. Universidad s/n, E-03202 Elche, Spain*

Abstract

The aim of this work is to characterize two models of concatenated convolutional codes based on the theory of linear systems. The problem we consider can be viewed as the study of composite linear system from the classical control theory or as the interconnection from the behavioral system viewpoint. In this paper we provide an input-state-output representation of both models and introduce some conditions for such representations to be both controllable and observable. We also introduce a lower bound on their free distances and the column distances.

Keywords: Convolutional code, concatenated convolutional code, free distance, column distances, observable pair, controllable pair, observable convolutional code, input-state-output representation, minimal representation.

1. Introduction

The link between convolutional codes and the theory of linear systems has already been established by several authors. Massey and Sain [24, 25] were the first ones to tackle theoretically the systems analysis of convolutional codes and encoders. Omura [28] considered Viterbi decoding and its relationship to dynamic programming and later [29] the applications of control theory to optimal receiver design for convolutional codes. Forney in his landmark papers [13, 14] introduce the foundation for the algebraic structure of convolutional codes.

Other authors have also shown the existence of a close connection between linear systems over finite fields and convolutional codes. Rosenthal [31] provides a survey of the different points of view about convolutional codes. He reviews their definition in the context of symbolic dynamics and studies different first-order representations in each of the different viewpoints. These representations have been applied to generating optimal convolutional codes (see, for example, [33, 35, 37]). Again, Rosenthal [31] reviews the class of time-invariant, complete linear behaviors in the sense of Willems. Willems [39, 40, 41] had already introduced the behavioral approach to linear systems and shown that the systems theory and the convolutional coding theory were interconnected. Willems does not assume the input/output point of view in his description of system/environment interaction. He in fact, focuses on the set of systems trajectories, its behavior, that is, on the mathematical model to describe the relations among all systems variables.

Concatenated codes form a class of error-correcting codes that are derived from combining two or more codes. The general idea can be tracked back to Elia's product code construction [11], developed later by Forney [12] as a solution to the problem of finding a code that has both exponentially decreasing error probability with increasing block length and polynomial-time decoding complexity. Concatenated codes became widely used in space communications

Email addresses: jcliment@ua.es (Joan-Josep Climent), mavi.herranz@umh.es (Victoria Herranz), perea@umh.es (Carmen Perea)

in the 1970s [7]. In fact, thanks to deep-space communications, both convolutional and concatenated codes found their first practical use in the Voyager and Mariner missions in the late sixties [1, 27].

Berrou, Glavieux and Thitimajshima [5] introduced an interleaver between the two encoders of a serial concatenation, which provides the correction of error burst from the inner encoder by the outer encoder. The result was called “turbo codes”. Turbo coding combines multiple simple codes into single powerful codes that operate close to Shannon capacity (see [5]). While their performance resembles that of random codes, the availability of iterative decoding algorithms makes the implementation of turbo codes practical (see, for example [9, 10, 16]).

A major parameter of a convolutional code is its free distance [21, 30] since it determines the decoding capability of a code under maximum likelihood decoding. It depends on the collection of code sequences, not on the specific device that is used to generate those sequences. So, to obtain convolutional codes with good distance properties one has to work with finite representations of convolutional codes. In fact, McEliece [26, Section 2] points out that finite-weight codewords are the only ones that can occur in engineering practice.

This paper is structured as follows. In Section 2, we review convolutional codes and we highlight their definition in terms of linear systems. We also explain some recent advances in systems theory in the context of convolutional codes defined over a Galois field. In Section 3, we study two models of parallel concatenated codes from the linear systems point of view, giving their input-state-output representations, and we supply conditions to get a minimal input-state-output representation for observable concatenated codes. Finally, in Section 4, we provide some lower bounds on the column distances and free distance of the second model of concatenation.

2. Preliminary results

In this paper, we denote by $\mathbb{F} = GF(q)$ the Galois field of q elements, $\mathbb{F}[z]$ the polynomial ring on the variable z with coefficients in \mathbb{F} and $\overline{\mathbb{F}}$ the algebraic closure of \mathbb{F} .

Following [31, 32], we define a convolutional code as a submodule C of $\mathbb{F}^n[z]$. Since $\mathbb{F}[z]$ is a principal ideal domain, and C is a submodule of the free submodule $\mathbb{F}^n[z]$, the convolutional code C is free and it has a well defined rank k (with $k \leq n$). Assume that the columns of $G(z) \in \mathbb{F}^{n \times k}[z]$ form a basis of C , then C is defined as

$$C = \left\{ \mathbf{v}(z) \in \mathbb{F}^n[z] \mid \mathbf{v}(z) = G(z)\mathbf{u}(z) \quad \text{with} \quad \mathbf{u}(z) \in \mathbb{F}^k[z] \right\}.$$

In that case, we say that C has rate k/n , and that $G(z)$ is a *generator matrix* of C . The *free distance* of C is given by

$$d_{\text{free}}(C) = \min \{ \text{wt}(\mathbf{v}(z)) \mid \mathbf{v}(z) \in C \text{ with } \mathbf{v}(z) \neq \mathbf{0} \}$$

where $\text{wt}(\mathbf{v}(z)) = \sum_{i=0}^l \text{wt}(\mathbf{v}_i)$ with $\text{wt}(\mathbf{v}_i)$ the Hamming weight of $\mathbf{v}_i \in \mathbb{F}^n$, i.e., the number of nonzero components of \mathbf{v}_i ,

$$\text{and } \mathbf{v}(z) = \sum_{i=0}^l \mathbf{v}_i z^i \in \mathbb{F}^n[z].$$

A matrix $G(z) \in \mathbb{F}^{n \times k}[z]$ is called *basic* if it has a polynomial left inverse (see for example [20, 21, 36]). The *degree* or *complexity* δ of a rate k/n convolutional code C is the maximum degree of the $k \times k$ minors of any generator matrix (see [26, 30]). A generator matrix $G(z) = (g_{ij}(z)) \in \mathbb{F}^{n \times k}[z]$ of C is called *minimal* if the number $\sum_{j=1}^k \delta_j$ attains the minimal value among all generator matrices of C , where $\delta_j = \max_{1 \leq i \leq n} \deg(g_{ij}(z))$, for $j = 1, 2, \dots, k$ (see for example

[15, 36]); in that case, $\delta = \sum_{j=1}^k \delta_j$, where δ is the degree of C .

Following [32], we consider a convolutional code to be *observable* if it has a basic generator matrix; basic matrices are commonly known as noncatastrophic matrices (see, for instance, [25],[30, Chapter 2],[36, Definition 1.1]).

Assume that $A \in \mathbb{F}^{\delta \times \delta}$, $B \in \mathbb{F}^{\delta \times k}$, $C \in \mathbb{F}^{(n-k) \times \delta}$, and $D \in \mathbb{F}^{(n-k) \times k}$. A rate k/n convolutional code of degree δ can be described by the linear system of equations (see [35])

$$\left. \begin{aligned} \mathbf{x}_{t+1} &= A\mathbf{x}_t + B\mathbf{u}_t, \\ \mathbf{y}_t &= C\mathbf{x}_t + D\mathbf{u}_t, \end{aligned} \right\}, \quad \mathbf{v}_t = \begin{pmatrix} \mathbf{y}_t \\ \mathbf{u}_t \end{pmatrix}, \quad t = 0, 1, 2, \dots, \quad \mathbf{x}_0 = \mathbf{0}. \quad (1)$$

For each instant t , we call $\mathbf{x}_t \in \mathbb{F}^\delta$ the *state vector*, $\mathbf{u}_t \in \mathbb{F}^k$ the *information vector*, $\mathbf{y}_t \in \mathbb{F}^{n-k}$ the *parity vector*, and $\mathbf{v}_t \in \mathbb{F}^n$ the *code vector*. We assume that $\{\mathbf{v}_t\}_{t \geq 0}$ in expression (1) is a finite-weight codeword (see [35]), i.e., expression (1) is satisfied for all $t = 0, 1, 2, \dots$ and there is an integer γ such that $\mathbf{x}_{t+1} = \mathbf{0}$, $\mathbf{u}_t = \mathbf{0}$, and therefore $\mathbf{y}_t = \mathbf{0}$, for $t \geq \gamma + 1$; this means that, beginning from the zero state, after a finite number of steps, we return to the zero state. If

$$\mathbf{u}(z) = \sum_{t=0}^{\gamma} \mathbf{u}_t z^{\gamma-t}, \mathbf{y}(z) = \sum_{t=0}^{\gamma} \mathbf{y}_t z^{\gamma-t}, \text{ and } \mathbf{v}(z) = \begin{pmatrix} \mathbf{y}(z) \\ \mathbf{u}(z) \end{pmatrix},$$

then the set of finite-weight codewords has a natural module structure over the polynomial ring $\mathbb{F}[z]$ (see [35]). We call this module a finite-weight convolutional code C generated by matrices (A, B, C, D) or that (A, B, C, D) is an *input-state-output* representation of C . From now on we only consider finite-weight convolutional codes.

It is possible to describe the code C using an input-state-output representation (A', B', C', D') with $A' \in \mathbb{F}^{\delta' \times \delta'}$, $B' \in \mathbb{F}^{\delta' \times k}$, $C' \in \mathbb{F}^{(n-k) \times \delta'}$, and $D' \in \mathbb{F}^{(n-k) \times k}$ with $\delta' \geq \delta$. But if C has complexity δ , then it is possible (see [23, 35]) to choose the matrices A, B, C , and D of sizes $\delta \times \delta$, $\delta \times k$, $(n-k) \times \delta$, and $(n-k) \times k$, respectively. In this case, we say that (A, B, C, D) is a *minimal* input-state-output representation of C , and a condition for that to happen is that the pair (A, B) is controllable (see [35]), i.e.,

$$\text{rank} \begin{pmatrix} B & AB & \cdots & A^{\delta-1}B \end{pmatrix} = \delta$$

or equivalently (see [17]),

$$\text{rank} \begin{pmatrix} zI - A & B \end{pmatrix} = \delta, \quad \text{for all } z \in \overline{\mathbb{F}}.$$

Note that if $\text{rank}(B) = \delta$, then the pair (A, B) is controllable.

On the other hand, we say that the pair (A, C) is *observable* if the pair (A^T, C^T) is controllable. Furthermore, if $\text{rank}(C) = \delta$, then the pair (A, C) is observable.

If the pair (A, B) is controllable, it means that, by an appropriate choice of input vectors, it is possible to drive a given state vector to any other state vector in finite time. Analogously, the observability of the pair (A, C) means that it is possible to determine the state vector at a given time t_0 by observing the output vectors for a finite number of time steps beginning with t_0 (see, for example, [18, 19, 32, 35]).

From now on, we adopt the notation used by McEliece [26] and we call a convolutional code of rate k/n and degree δ an (n, k, δ) -convolutional code.

The following result, that we quote for further references, characterizes the observable convolutional codes.

Lemma 1 (Lemma 2.11 of [35]): *Assume that (A, B, C, D) is an input-state-output representation of an (n, k, δ) -convolutional code C . Assume also that the pair (A, B) is controllable. Then the convolutional code C represents an observable convolutional code if and only if the pair (A, C) is observable.*

Once we have a minimal input-state-output representation of an observable convolutional code, we analyze the characterization of the free distance in terms of linear systems. It is well known that the free distance determines the decoding capability of a convolutional code, especially when Viterbi decoding [38] or sequential decoding [6] are used.

In terms of the input-state-output representation (1), the free distance of a convolutional code C can be characterized (see [18]) as

$$d_{\text{free}}(C) = \lim_{j \rightarrow \infty} d_j^c(C) \tag{2}$$

where

$$d_j^c(C) = \min_{\mathbf{u}_0 \neq \mathbf{0}} \left\{ \sum_{t=0}^j \text{wt}(\mathbf{u}_t) + \sum_{t=0}^j \text{wt}(\mathbf{y}_t) \right\}$$

is the j th column distance of C , for $j = 0, 1, 2, \dots$

Finally, the free distance of an (n, k, δ) -convolutional code C is always upper-bounded (see [34]) by the generalized Singleton bound

$$d_{\text{free}}(C) \leq (n-k) \left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1 \right) + \delta + 1.$$

Moreover, the convolutional code C is called *maximum distance separable (MDS)* if its free distance is equal to the generalized Singleton bound [34].

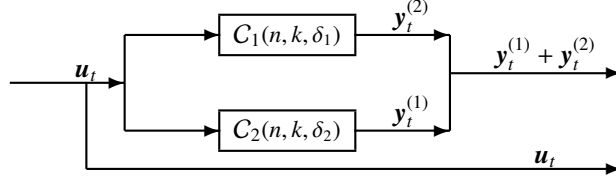


Figure 1: Concatenated convolutional code $\mathcal{PC}^{(1)}$

Depending on the value of the column distances, we can obtain the following classification of convolutional codes. A convolutional code C has a *maximum distance profile*, and we say that it is an *MDP code* (see [18]), if its column distances are maximal, that is,

$$d_j^c(C) = (n - k)(j + 1) + 1, \quad \text{for } j = 0, 1, \dots, L, \quad \text{where } L = \left\lfloor \frac{\delta}{k} \right\rfloor + \left\lfloor \frac{\delta}{n - k} \right\rfloor.$$

In addition, a convolutional code C is called a *strongly MDS code* (see [18]) if the sequence $\{d_j^c(C)\}_{j \geq 0}$ attains the generalized Singleton bound at the earliest possible step, i.e.,

$$d_M^c(C) = (n - k) \left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1 \right) + \delta + 1, \quad \text{for } M = \left\lfloor \frac{\delta}{k} \right\rfloor + \left\lceil \frac{\delta}{n - k} \right\rceil.$$

3. Input-state-output representation of parallel concatenated convolutional codes

In this section, we analyze two models of parallel concatenated convolutional codes from the point of view of linear systems theory.

For $l = 1, 2$, let (A_l, B_l, C_l, D_l) be an input-state-output representation of the convolutional code C_l . For each instant t , let $\mathbf{x}_t^{(l)}$, $\mathbf{u}_t^{(l)}$, $\mathbf{y}_t^{(l)}$ and $\mathbf{v}_t^{(l)}$ be the state vector, the information vector, the parity vector and the code vector of C_l , for $l = 1, 2$. Recall that the code vector $\mathbf{v}_t^{(l)}$ of C_l , is given by

$$\mathbf{v}_t^{(l)} = \begin{pmatrix} \mathbf{y}_t^{(l)} \\ \mathbf{u}_t^{(l)} \end{pmatrix}, \quad \text{for } l = 1, 2. \quad (3)$$

The first model of parallel concatenation which we study is the one used in the state space theory (see, for instance, [22]). In this model, the convolutional codes C_1 and C_2 of rates k/n are concatenated as Figure 1 shows. The input information \mathbf{u}_t is encoded by the two convolutional codes so

$$\mathbf{u}_t^{(1)} = \mathbf{u}_t \quad \text{and} \quad \mathbf{u}_t^{(2)} = \mathbf{u}_t,$$

and the parity vector of the concatenation is given by the sum of the parity vectors of the constituent codes, i.e.,

$$\mathbf{y}_t = \mathbf{y}_t^{(1)} + \mathbf{y}_t^{(2)}.$$

We denote by $\mathcal{PC}^{(1)}$ the corresponding concatenated convolutional code. For each instant t , the vector state \mathbf{x}_t , and the code vector \mathbf{v}_t of $\mathcal{PC}^{(1)}$ are given by

$$\mathbf{x}_t = \begin{pmatrix} \mathbf{x}_t^{(2)} \\ \mathbf{x}_t^{(1)} \end{pmatrix} \quad \text{and} \quad \mathbf{v}_t = \begin{pmatrix} \mathbf{y}_t \\ \mathbf{u}_t \end{pmatrix} = \begin{pmatrix} \mathbf{y}_t^{(1)} + \mathbf{y}_t^{(2)} \\ \mathbf{u}_t \end{pmatrix}.$$

The next theorem introduces an input-state-output representation of the concatenated convolutional code $\mathcal{PC}^{(1)}$ from input-state-output representations of the constituent codes (see [2]). From now on, we denote by O the zero matrix of the appropriate size.

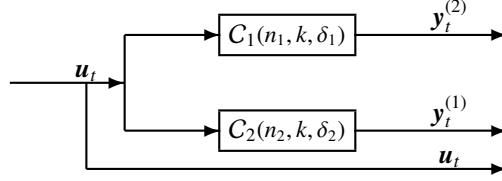


Figure 2: Concatenated convolutional code $\mathcal{PC}^{(2)}$

Theorem 1: For $l = 1, 2$, let C_l be an (n_l, k, δ_l) -convolutional code with input-state-output representation (A_l, B_l, C_l, D_l) . Then, an input-state-output representation (A, B, C, D) for the rate k/n concatenated convolutional code $\mathcal{PC}^{(1)}$ is given by expression (1), where

$$A = \begin{pmatrix} A_2 & O \\ 0 & A_1 \end{pmatrix}, \quad B = \begin{pmatrix} B_2 \\ B_1 \end{pmatrix}, \quad C = (C_2 \quad C_1) \quad \text{and} \quad D = D_1 + D_2. \quad (4)$$

The second model of parallel concatenation is used, for example, in some constructions of turbo codes [3, 4], and it can be represented as in Figure 2. In this model, the information vector \mathbf{u}_t is also encoded by both codes C_1 and C_2 , so

$$\mathbf{u}_t^{(1)} = \mathbf{u}_t \quad \text{and} \quad \mathbf{u}_t^{(2)} = \mathbf{u}_t, \quad (5)$$

but in this case, the parity vector \mathbf{y}_t of the concatenation is given by the parity vectors of the constituent codes, i.e.,

$$\mathbf{y}_t = \begin{pmatrix} \mathbf{y}_t^{(2)} \\ \mathbf{y}_t^{(1)} \end{pmatrix}. \quad (6)$$

We denote by $\mathcal{PC}^{(2)}$ the corresponding concatenated convolutional code. Note that the vector state \mathbf{x}_t is given, as in the previous model, by

$$\mathbf{x}_t = \begin{pmatrix} \mathbf{x}_t^{(2)} \\ \mathbf{x}_t^{(1)} \end{pmatrix}.$$

But in this case, the code vector \mathbf{v}_t of $\mathcal{PC}^{(2)}$ is given by

$$\mathbf{v}_t = \begin{pmatrix} \mathbf{y}_t \\ \mathbf{u}_t \end{pmatrix} = \begin{pmatrix} \mathbf{y}_t^{(2)} \\ \mathbf{y}_t^{(1)} \\ \mathbf{u}_t \end{pmatrix} = \begin{pmatrix} \mathbf{y}_t^{(2)} \\ \mathbf{v}_t^{(1)} \end{pmatrix}. \quad (7)$$

As in the previous case, our next theorem introduces an input-state-output representation of the concatenated convolutional code $\mathcal{PC}^{(2)}$ from input-state-output representations of the constituent convolutional codes C_1 and C_2 .

Theorem 2: For $l = 1, 2$, let C_l be an (n_l, k, δ_l) -convolutional code with input-state-output representation (A_l, B_l, C_l, D_l) . Then, an input-state-output representation (A, B, C, D) for the rate $k/(n_1 + n_2 - k)$ concatenated convolutional code $\mathcal{PC}^{(2)}$ is given by expression (1), where

$$A = \begin{pmatrix} A_2 & O \\ 0 & A_1 \end{pmatrix}, \quad B = \begin{pmatrix} B_2 \\ B_1 \end{pmatrix}, \quad C = \begin{pmatrix} C_2 & O \\ O & C_1 \end{pmatrix} \quad \text{and} \quad D = \begin{pmatrix} D_2 \\ D_1 \end{pmatrix}. \quad (8)$$

Once we have obtained an input-state-output representation of the concatenated convolutional code, we are interested in the conditions on the matrices A_l , B_l , C_l , and D_l of the convolutional codes C_l , for $l = 1, 2$, so that the concatenated convolutional code is observable and has a minimal input-state-output representation. However, the results we introduce below are valid for matrices A , B , C , and D with the same structure as the ones given by expressions (4) and (8), not necessarily obtained from a concatenation of convolutional codes. Later, we apply these results to the two models of concatenation described previously.

We begin with the study of the controllability of the pair (A, B) , where A and B are matrices given by expressions (4) or (8). Note that these matrices have the same structure in both cases, so the results developed here will be applicable to both models of concatenation.

For $l = 1, 2$, let A_l and B_l be matrices of sizes $\delta_l \times \delta_l$ and $\delta_l \times k$, respectively, and assume that A and B are the matrices of sizes $(\delta_1 + \delta_2) \times (\delta_1 + \delta_2)$ and $(\delta_1 + \delta_2) \times k$, respectively, described by expressions (4) or (8). The next example shows that it is not sufficient that the pair (A_l, B_l) is controllable, for $l = 1, 2$, to get a controllable pair (A, B) .

Example 1: Let α be a primitive element of the Galois field $\mathbb{F} = GF(9)$, with $\alpha^2 + \alpha + 2 = 0$. Consider the controllable pair (A_l, B_l) , for $l = 1, 2$, given by

$$A_1 = (\alpha), \quad B_1 = (1 \ 0 \ 0 \ 0), \quad A_2 = \begin{pmatrix} \alpha^6 & \alpha^5 \\ \alpha^5 & \alpha^3 \end{pmatrix} \quad \text{and} \quad B_2 = \begin{pmatrix} \alpha & \alpha^3 & \alpha^7 & 1 \\ \alpha^3 & \alpha^4 & 1 & \alpha \end{pmatrix}.$$

Let A and B be the matrices given by (4) and (8), i.e.,

$$A = \begin{pmatrix} \alpha^6 & \alpha^5 & 0 \\ \alpha^5 & \alpha^3 & 0 \\ 0 & 0 & \alpha \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} \alpha & \alpha^3 & \alpha^7 & 1 \\ \alpha^3 & \alpha^4 & 1 & \alpha \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

It follows then that (A, B) is not a controllable pair because $\text{rank}(\alpha I - A \ B) = 2 \neq 3$. □

However, if the matrices A_1 and A_2 do not have common eigenvalues, we have the following result. The proof can be found in [8, Chapter 30]. Here, we denote by $\sigma(A_l)$ the set of eigenvalues of A_l , for $l = 1, 2$.

Theorem 3: For $l = 1, 2$, let A_l and B_l be matrices of sizes $\delta_l \times \delta_l$ and $\delta_l \times k$, respectively, such that the pair (A_l, B_l) is controllable. Assume that A and B are the matrices given by expressions (4) or (8). If $\sigma(A_1) \cap \sigma(A_2) = \emptyset$, then the pair (A, B) is controllable.

Now, in the particular case where $k = 1$, we can ensure that the condition of Theorem 3 is a necessary and sufficient condition. The proof can be found in [8, Chapter 30].

Theorem 4: For $l = 1, 2$, let A_l and B_l be matrices of sizes $\delta_l \times \delta_l$ and $\delta_l \times 1$, respectively. Assume that A and B are matrices given by expressions (4) or (8). Then, the pair (A, B) is controllable if and only if $\sigma(A_1) \cap \sigma(A_2) = \emptyset$.

As a consequence of Theorem 3 and the fact that $\text{rank } B = \delta_1 + \delta_2$ implies that the pair (A, B) is controllable, we obtain the following result for the particular case where $\delta_1 = \delta_2 = 1$.

Corollary 1: For $l = 1, 2$, let A_l and B_l be matrices of sizes 1×1 and $1 \times k$, respectively. Assume that A and B are matrices given by expressions (4) or (8). Then the pair (A, B) is controllable if and only if $A_1 \neq A_2$ or $\text{rank}(B) = 2$.

The above results can be applied concretely to both convolutional codes $\mathcal{PC}^{(1)}$ and $\mathcal{PC}^{(2)}$, which can be summarized in the following result. As we mentioned earlier, matrices A and B of both concatenated convolutional codes $\mathcal{PC}^{(1)}$ and $\mathcal{PC}^{(2)}$ have the same structure. Since we develop results for both types of concatenation, we denote by \mathcal{PC} the concatenated convolutional code $\mathcal{PC}^{(1)}$ described by expression (4) as well as the concatenated convolutional code $\mathcal{PC}^{(2)}$ described by expression (8). So we consider an (n_1, k, δ_1) -convolutional code C_1 and an (n_2, k, δ_2) -convolutional code C_2 , taking into account that for the convolutional code $\mathcal{PC}^{(1)}$, they must satisfy the additional condition $n_1 = n_2$.

Corollary 2: For $l = 1, 2$, let C_l be an (n_l, k, δ_l) -convolutional code described by matrices (A_l, B_l, C_l, D_l) , such that the pair (A_l, B_l) is controllable and let \mathcal{PC} be the parallel concatenated convolutional code described by the matrices (A, B, C, D) given by expression (4) or (8).

1. If $\sigma(A_1) \cap \sigma(A_2) = \emptyset$, then (A, B, C, D) is a minimal representation of \mathcal{PC} with complexity $\delta_1 + \delta_2$.
2. If $k = 1$, then (A, B, C, D) is a minimal representation of \mathcal{PC} with complexity $\delta_1 + \delta_2$ if and only if $\sigma(A_1) \cap \sigma(A_2) = \emptyset$.
3. If $\delta_1 = \delta_2 = 1$, then (A, B, C, D) is a minimal representation of \mathcal{PC} if and only if $A_1 \neq A_2$, or $\text{rank}(B) = 2$.

Now, we are interested in the conditions the matrices defining the constituent convolutional codes C_1 and C_2 must satisfy so that the pair (A, C) of the parallel concatenation is observable. Although the matrices A and B are the same in both cases, note that matrix C has different expressions depending on the model of concatenation. So we may have different results for the observability of each concatenated convolutional code.

As for the controllability case, we begin with some results on control theory.

For $l = 1, 2$, let A_l and C_l be matrices of sizes $\delta_l \times \delta_l$ and $(n - k) \times \delta_l$, respectively, such that the pair (A_l, C_l) is observable. Assume that A and C are the matrices of sizes $(\delta_1 + \delta_2) \times (\delta_1 + \delta_2)$ and $(n - k) \times (\delta_1 + \delta_2)$, respectively, described by expression (4). Then, (A, C) is not necessarily an observable pair, as the next example shows.

Example 2: As in Example 1, let α be a primitive element of $\mathbb{F} = GF(9)$, such that $\alpha^2 + \alpha + 2 = 0$. Consider the observable pair (A_l, C_l) , for $l = 1, 2$, given by

$$A_1 = (\alpha), \quad C_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad A_2 = \begin{pmatrix} \alpha^6 & \alpha^5 \\ \alpha^5 & \alpha^3 \end{pmatrix} \quad \text{and} \quad C_2 = \begin{pmatrix} \alpha^6 & 1 \\ \alpha^2 & \alpha^3 \\ \alpha^6 & \alpha^7 \\ 1 & \alpha \end{pmatrix}$$

and let A and C be the matrices given by expression (4)

$$A = \begin{pmatrix} \alpha^6 & \alpha^5 & 0 \\ \alpha^5 & \alpha^3 & 0 \\ 0 & 0 & \alpha \end{pmatrix} \quad \text{and} \quad C = \begin{pmatrix} \alpha^6 & 1 & 1 \\ \alpha^2 & \alpha^3 & 0 \\ \alpha^6 & \alpha^7 & 0 \\ 1 & \alpha & 0 \end{pmatrix}.$$

Now, since

$$\text{rank} \begin{pmatrix} \alpha I - A \\ C \end{pmatrix} = \text{rank} \begin{pmatrix} 1 & \alpha & 0 \\ \alpha & \alpha^2 & 0 \\ 0 & 0 & 0 \\ \alpha^6 & 1 & 1 \\ \alpha^2 & \alpha^3 & 0 \\ \alpha^6 & \alpha^7 & 0 \\ 1 & \alpha & 0 \end{pmatrix} = 2 \neq 3,$$

we can ensure that the pair (A, C) is not observable. □

From now on, our aim is to give conditions on the matrices A_l and C_l , for $l = 1, 2$ so that the pair (A, C) described by expression (4) is observable.

Recall that if the matrices A_1 and A_2 do not have common eigenvalues, Theorem 3 says that the pair (A, B) described by expression (4) is controllable. Now, as a consequence of the structure of matrices A and C , it follows that the pair (A, C) is observable. So, we can state the following result.

Theorem 5: For $l = 1, 2$, let A_l and C_l be matrices of sizes $\delta_l \times \delta_l$ and $(n - k) \times \delta_l$, respectively. Assume that A and C are matrices given by expression (4). If $\sigma(A_1) \cap \sigma(A_2) = \emptyset$, then the pair (A, C) is observable.

Proof: Taking into account that the pair (A, C) is observable if and only if (A^T, C^T) is controllable, we obtain the observability of (A, C) by following a similar argument as the one used in Theorem 3. ◇

Now, for the particular case where $n = k + 1$, we can ensure that the condition of Theorem 5 is a necessary and sufficient condition.

Theorem 6: For $l = 1, 2$, let A_l and C_l be matrices of sizes $\delta_l \times \delta_l$ and $1 \times \delta_l$, respectively. Assume that A and C are matrices given by expression (4). Then the pair (A, C) is observable if and only if $\sigma(A_1) \cap \sigma(A_2) = \emptyset$.

Proof: If $\sigma(A_1) \cap \sigma(A_2) = \emptyset$, we know from Theorem 5 that the pair (A, C) is observable.

Assume that the pair (A, C) is observable. Then, the pair (A^T, C^T) is controllable. Note that the matrix C^T is of size $\delta \times 1$, so by following a similar argument as in the proof of Theorem 4, we get $\sigma(A_1) \cap \sigma(A_2) = \emptyset$. ◇

As a consequence of Theorem 5 and the fact that $\text{rank } C = \delta_1 + \delta_2$ implies that the pair (A, C) is observable, we obtain the following result for the particular case where $\delta_1 = \delta_2 = 1$.

Corollary 3: For $l = 1, 2$, let A_l and C_l be matrices of sizes 1×1 and $(n - k) \times 1$, respectively. Assume that A and C are matrices given by expression (4). Then the pair (A, C) is observable if and only if $A_1 \neq A_2$ or $\text{rank}(C) = 2$.

The above results can also be applied to the concatenation $\mathcal{PC}^{(1)}$. We summarized these results in the corollary below. Remember that a minimal representation (A, B, C, D) of the convolutional code C is characterized through the condition that the pair (A, B) is controllable and, from Lemma 1, if the pairs (A, B) and (A, C) are controllable and observable, respectively, then C is an observable convolutional code.

Corollary 4: For $l = 1, 2$, let C_l be an (n, k, δ_l) -observable convolutional code with minimal input-state-output representation (A_l, B_l, C_l, D_l) . Let $\mathcal{PC}^{(1)}$ be the rate k/n concatenated convolutional code described by matrices (A, B, C, D) in expression (4).

1. If $\sigma(A_1) \cap \sigma(A_2) = \emptyset$, then (A, B, C, D) is a minimal and observable representation of $\mathcal{PC}^{(1)}$ with complexity $\delta_1 + \delta_2$.
2. If $k = 1$ and $n = 2$, then (A, B, C, D) is a minimal and observable representation of $\mathcal{PC}^{(1)}$ with complexity $\delta_1 + \delta_2$ if and only if $\sigma(A_1) \cap \sigma(A_2) = \emptyset$.
3. If $\delta_1 = \delta_2 = 1$, then (A, B, C, D) is a minimal and observable representation of $\mathcal{PC}^{(1)}$ with complexity $\delta = 2$ if and only if $A_1 \neq A_2$ or $\text{rank}(C) = \text{rank}(B) = 2$.

Example 2 shows that if the pair (A_l, C_l) is observable, for $l = 1, 2$, then the pair (A, C) described by expression (4) is not necessarily observable. Nevertheless, for matrices A and C described by expression (8), we do not have the same situation.

Theorem 7: For $l = 1, 2$, let A_l and C_l be matrices of sizes $\delta_l \times \delta_l$ and $(n_l - k) \times \delta_l$, respectively, so that the pair (A_l, C_l) is observable. Assume that A and C are matrices given by expression (8). Then, the pair (A, C) is observable.

Proof: Since (A_l, C_l) is observable, for $l = 1, 2$, we have that

$$\text{rank} \begin{pmatrix} zI - A_l \\ C_l \end{pmatrix} = \delta_l \quad \text{for all } z \in \overline{\mathbb{F}},$$

so, from expression (8),

$$\text{rank} \begin{pmatrix} zI - A \\ C \end{pmatrix} = \text{rank} \begin{pmatrix} zI_{\delta_2} - A_2 & O \\ O & zI_{\delta_1} - A_1 \\ C_2 & O \\ O & C_1 \end{pmatrix} = \text{rank} \begin{pmatrix} zI_{\delta_2} - A_2 & O \\ C_2 & O \\ O & zI_{\delta_1} - A_1 \\ O & C_1 \end{pmatrix} = \delta_1 + \delta_2,$$

for all $z \in \overline{\mathbb{F}}$. We conclude then that the pair (A, C) is observable. \diamond

4. Column distances and free distance

Note that the parity vector \mathbf{y}_t of the input-state-output representation of the concatenated convolutional code $\mathcal{PC}^{(1)}$ given by expression (4) is the sum of the parity vectors $\mathbf{y}_t^{(1)}$ and $\mathbf{y}_t^{(2)}$ of the constituent codes. This means that, in some cases, the above parallel concatenation leads to a convolutional code with poor free distance, as we can see in the example below.

In the next examples, we compute the column distances of all codes using an appropriate software and then we obtain the free distances of these codes according to expression (2).

Example 3: Let $\mathbb{F} = GF(2)$ and let C_1 be the $(4, 2, 1)$ -convolutional code with free distance $d_{\text{free}}(C_1) = 2$, described by the matrices

$$A_1 = (1), \quad B_1 = (1 \ 1), \quad C_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad D_1 = (1 \ 1).$$

Let C_2 be the $(4, 2, 1)$ -convolutional code with free distance $d_{\text{free}}(C_2) = 3$, described by the matrices

$$A_2 = (1), \quad B_2 = (1 \ 0), \quad C_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{and} \quad D_2 = (0 \ 1).$$

Then, the matrices (A, B, C, D) of expression (4) are

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad D = (1 \ 0).$$

Since $\text{rank}(B) = \text{rank}(C) = 2$, by applying Corollary 4, it follows that the above matrices are a minimal input-state-output representation of the $(4, 2, 2)$ -observable concatenated convolutional code $\mathcal{PC}^{(1)}$. Furthermore, we obtain that it has free distance

$$d_{\text{free}}(\mathcal{PC}^{(1)}) = 3 = d_{\text{free}}(C_2).$$

That is, we obtain a parallel concatenated convolutional code whose free distance is not greater than the free distances of the constituent codes. Note that the free distance of $\mathcal{PC}^{(1)}$ is far from the generalized Singleton bound, which in this case is 7. \square

However, in some cases, we can obtain an optimal concatenated convolutional code from a non MDS convolutional code and an MDS convolutional code, as we show in the following example.

Example 4: Let α be a primitive element of the Galois field $\mathbb{F} = GF(8)$ with $\alpha^3 + \alpha + 1 = 0$. Let C_1 be the $(2, 1, 1)$ -MDS convolutional code with $d_{\text{free}}(C_1) = 4$, described by the matrices

$$A_1 = (\alpha^2), \quad B_1 = (\alpha), \quad C_1 = (\alpha^4) \quad \text{and} \quad D_1 = (1).$$

Let C_2 be the $(2, 1, 1)$ -convolutional code with free distance $d_{\text{free}}(C_2) = 3$ (non MDS code), described by the matrices

$$A_2 = (\alpha), \quad B_2 = (1), \quad C_2 = (\alpha^3) \quad \text{and} \quad D_2 = (0).$$

By applying Theorem 1, the matrices (A, B, C, D) with

$$A = \begin{pmatrix} \alpha & 0 \\ 0 & \alpha^2 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ \alpha \end{pmatrix}, \quad C = (\alpha^3 \ \alpha^4) \quad \text{and} \quad D = (1)$$

are an input-state-output representation of the $(2, 1, 2)$ -concatenated convolutional code $\mathcal{PC}^{(1)}$.

Furthermore, $d_{\text{free}}(\mathcal{PC}^{(1)}) = 6$, which is the generalized Singleton bound in this case. So $\mathcal{PC}^{(1)}$ is an MDS convolutional code. \square

As we show in the previous examples, the free distance of the concatenated convolutional code $\mathcal{PC}^{(1)}$ can either attain the generalized Singleton bound or be far away from it. Furthermore, the expression of the parity vector as the sum of the parity vectors of the constituent codes makes it difficult to obtain a relation between the free distance of the convolutional code $\mathcal{PC}^{(1)}$ and the free distances of the convolutional codes of the concatenation. However, note that, according to expression (7), the code vector \mathbf{v}_t of the convolutional code $\mathcal{PC}^{(2)}$ is given by

$$\mathbf{v}_t = \begin{pmatrix} \mathbf{y}_t^{(2)} \\ \mathbf{y}_t^{(1)} \\ \mathbf{u}_t \end{pmatrix}$$

that is, it contains the code vector $\mathbf{y}_t^{(1)} = \begin{pmatrix} \mathbf{y}_t^{(1)} \\ \mathbf{u}_t \end{pmatrix}$ of C_1 and the code vector $\mathbf{y}_t^{(2)} = \begin{pmatrix} \mathbf{y}_t^{(2)} \\ \mathbf{u}_t \end{pmatrix}$ of C_2 (see expression (3)). This fact allows us to obtain a lower bound on the free distance of $\mathcal{PC}^{(2)}$ in terms of the free distances of the constituent codes. Firstly, we obtain a lower bound on the column distances of $\mathcal{PC}^{(2)}$ in terms of the column distances of C_1 and C_2 .

Lemma 2: Let $\mathcal{PC}^{(2)}$ be the concatenated convolutional code obtained by Theorem 2 from the convolutional codes C_1 and C_2 . Then

$$d_j^c(\mathcal{PC}^{(2)}) \geq \max \{d_j^c(C_1), d_j^c(C_2)\} \quad \text{for } j = 0, 1, 2, \dots \quad (9)$$

Proof: Taking into account the relationship between the vectors $\mathbf{y}_t, \mathbf{y}_t^{(1)}, \mathbf{y}_t^{(2)}$; $\mathbf{u}_t, \mathbf{u}_t^{(1)}, \mathbf{u}_t^{(2)}$ and $\mathbf{v}_t, \mathbf{v}_t^{(1)}, \mathbf{v}_t^{(2)}$ given by expressions (5), (6) and (7), we obtain

$$d_j^c(\mathcal{PC}^{(2)}) = \min_{\mathbf{u}_0 \neq \mathbf{0}} \left\{ \sum_{t=0}^j \text{wt}(\mathbf{v}_t) \right\} \geq \min_{\mathbf{u}_0^{(1)} \neq \mathbf{0}} \left\{ \sum_{t=0}^j \text{wt}(\mathbf{v}_t^{(1)}) \right\} = d_j^c(C_1), \quad (10)$$

$$d_j^c(\mathcal{PC}^{(2)}) = \min_{\mathbf{u}_0 \neq \mathbf{0}} \left\{ \sum_{t=0}^j \text{wt}(\mathbf{v}_t) \right\} \geq \min_{\mathbf{u}_0^{(2)} \neq \mathbf{0}} \left\{ \sum_{t=0}^j \text{wt}(\mathbf{v}_t^{(2)}) \right\} = d_j^c(C_2). \quad (11)$$

So, inequality (9) follows from the above inequalities. \diamond

Now, if $\text{rank}(D_l) = k$, for some $l = 1, 2$, we have a refinement on the bound given in Lemma 2, as the following result shows.

Lemma 3: Let $\mathcal{PC}^{(2)}$ be the concatenated convolutional code obtained by Theorem 2 from the convolutional codes C_1 and C_2 .

1. If $\text{rank}(D_1) = k$, then

$$d_j^c(\mathcal{PC}^{(2)}) \geq \max \{d_j^c(C_1), d_j^c(C_2) + 1\}, \quad \text{for } j = 0, 1, 2, \dots \quad (12)$$

2. If $\text{rank}(D_2) = k$, then

$$d_j^c(\mathcal{PC}^{(2)}) \geq \max \{d_j^c(C_1) + 1, d_j^c(C_2)\}, \quad \text{for } j = 0, 1, 2, \dots \quad (13)$$

3. If $\text{rank}(D_1) = \text{rank}(D_2) = k$, then

$$d_j^c(\mathcal{PC}^{(2)}) \geq \max \{d_j^c(C_1) + 1, d_j^c(C_2) + 1\}, \quad \text{for } j = 0, 1, 2, \dots \quad (14)$$

Proof: 1. Since $\mathbf{y}_0^{(1)} = D_1 \mathbf{u}_0^{(1)}$ and $\text{rank}(D_1) = k$, then $\mathbf{y}_0^{(1)} \neq \mathbf{0}$ if and only if $\mathbf{u}_0^{(1)} = \mathbf{u}_0 \neq \mathbf{0}$. Moreover, since $\mathbf{u}_0 = \mathbf{u}_0^{(2)}$, we obtain that

$$\begin{aligned} d_j^c(\mathcal{PC}^{(2)}) &= \min_{\mathbf{u}_0 \neq \mathbf{0}} \left\{ \sum_{t=0}^j \text{wt}(\mathbf{u}_t) + \sum_{t=0}^j \text{wt}(\mathbf{y}_t) \right\} \\ &\geq \min_{\mathbf{u}_0^{(2)} \neq \mathbf{0}} \left\{ \sum_{t=0}^j \text{wt}(\mathbf{u}_t^{(2)}) + \sum_{t=0}^j \text{wt}(\mathbf{y}_t^{(2)}) + \text{wt}(\mathbf{y}_0^{(1)}) \right\} \geq d_j^c(C_2) + 1. \end{aligned} \quad (15)$$

Then, from expressions (10) and (15), we obtain inequality (12).

2. Following a similar argument as in part 1, we obtain inequality (13) from expression (11) and condition $\text{rank}(D_2) = k$.

3. Inequality (14) follows as a direct consequence of expressions (12) and (13). \diamond

Now, as an immediate consequence of expression (2) and the above lemmas, we obtain the following result.

Theorem 8: Let $\mathcal{PC}^{(2)}$ be the concatenated convolutional code obtained by Theorem 2 from the convolutional codes C_1 and C_2 . Then

1. $d_{\text{free}}(\mathcal{PC}^{(2)}) \geq \max \{d_{\text{free}}(C_1), d_{\text{free}}(C_2)\}$.
2. If $\text{rank}(D_2) = k$, then $d_{\text{free}}(\mathcal{PC}^{(2)}) \geq \max \{d_{\text{free}}(C_1) + 1, d_{\text{free}}(C_2)\}$.
3. If $\text{rank}(D_1) = k$, then $d_{\text{free}}(\mathcal{PC}^{(2)}) \geq \max \{d_{\text{free}}(C_1), d_{\text{free}}(C_2) + 1\}$.
4. If $\text{rank}(D_1) = \text{rank}(D_2) = k$, then $d_{\text{free}}(\mathcal{PC}^{(2)}) \geq \max \{d_{\text{free}}(C_1) + 1, d_{\text{free}}(C_2) + 1\}$.

Applying expression (16) in Lemma 4 for $j = M = \max\{M_1, M_2\}$, we obtain

$$d_M^c(\mathcal{PC}^{(2)}) \geq d_M^c(C_1) + d_M^c(C_2) - k(M + 1) = d_{\text{free}}(C_1) + d_{\text{free}}(C_2) - k(M + 1.) \quad (21)$$

Now, since $\{d_j^c(\mathcal{PC}^{(2)})\}_{j \geq 0}$ is a nondecreasing sequence,

$$d_j^c(\mathcal{PC}^{(2)}) \geq d_M^c(\mathcal{PC}^{(2)}) \quad \text{for } j \geq M. \quad (22)$$

Expression (19) follows now from expressions (21) and (22). Finally, taking into account that $d_{\text{free}}(\mathcal{PC}^{(2)}) \geq d_j^c(\mathcal{PC}^{(2)})$ for $j \geq 0$, we obtain expression (20). \diamond

5. Conclusions

In this paper we introduced two models of parallel concatenation of convolutional codes from linear systems viewpoint. We introduce conditions on the constituent codes of the concatenation in order to get a minimal input-state-output representation of the observable concatenated code. The main result of this paper is a series of lower bounds on the column distances and free distance of the second model of concatenation. In our future work, we want to give concrete constructions of the constituent codes in order to get a concatenated MDP convolutional code or a concatenated strongly MDS convolutional code.

Acknowledgment

The authors are very grateful to the anonymous reviewers for their comments and suggestions which led to significant improvements.

References

- [1] Kenneth S. Andrews, Dariush Divsalar, Sam Dolinar, Jon Hamnkins, Christopher R. Jones, and Fabrizio Pollara. The development of turbo codes and LDPC codes for deep-space applications. *Proceedings of the IEEE*, 95(11):2142–2156, 2007.
- [2] Panos J. Antsaklis and Anthony N. Michel. *Linear Systems*. Birkhäuser, Boston, MA, 2006.
- [3] Sergio Benedetto and Guido Montorsi. Design of parallel concatenated convolutional codes. *IEEE Transactions on Communications*, 44(5):591–600, 1996.
- [4] Sergio Benedetto and Guido Montorsi. Unveiling turbo codes: some results on parallel concatenated coding schemes. *IEEE Transactions on Information Theory*, 42(2):409–428, 1996.
- [5] Claude Berrou, Alain Glavieux, and Punya Thitimajshima. Near Shannon limit error-correcting coding and decoding: turbo codes (1). In *Proceedings of the IEEE International Conference on Communication (ICC'93)*, pages 1064–1070, Geneva, Switzerland, 1993. IEEE.
- [6] Pierre R. Chevillat and Daniel J. Costello, Jr. Distance and computation in sequential decoding. *IEEE Transactions on Communications*, 24(4):440–447, 1976.
- [7] Giovanni Emanuele Corazza, editor. *Digital Satellite Communications*. Springer, New York, NY, 2007.
- [8] Mohammed Dahleh, Munther A. Dahleh, and George Verghese. *Lectures on Dynamic Systems and Control*. MIT OpenCourseWare, Boston, MA, 2011.
- [9] Catherine Douillard and Claude Berrou. Turbo codes with rate- $m/(m+1)$ constituent convolutional codes. *IEEE Transactions on Computers*, 53(10):1630–1638, 2005.
- [10] Catherine Douillard, Michel Jézéquel, Claude Berrou, Nathalie Brengarth, Jacky Tusch, and Nghia Pham. The turbo code standard for DVB-RCS. In Keattisak Sripimanwat, editor, *Proceedings of the 2nd International Symposium on Turbo Codes and Iterative Information Processing*, pages 535–538, Brest, France, September 2000.
- [11] Peter Elias. Error free coding. *Transactions of the IRE Professional Group on Information Theory*, 4(4):29–37, 1954.
- [12] G. David Forney, Jr. Concatenated codes. Technical Report 440, Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA, December 1965.
- [13] G. David Forney, Jr. Convolutional codes I: algebraic structure. *IEEE Transactions on Information Theory*, 16(6):720–738, 1970.
- [14] G. David Forney, Jr. Structural analysis of convolutional codes via dual codes. *IEEE Transactions on Information Theory*, 19(4):512–518, 1973.
- [15] G. David Forney, Jr. Minimal bases of rational vector spaces, with applications to multivariable linear systems. *SIAM Journal on Control*, 13(3):493–520, 1975.
- [16] Ken Gracie and Marie-Hélène Hamon. Turbo and turbo-like codes: principles and applications in telecommunications. *Proceedings of the IEEE*, 95(6):1228–1254, 2007.
- [17] M. L. J. Hautus. Controllability and observability condition for linear autonomous systems. *Proceedings of Nederlandse Akademie voor Wetenschappen (Series A)*, 72:443–448, 1969.

- [18] Ryan Hutchinson, Joachim Rosenthal, and Roxana Smarandache. Convolutional codes with maximum distance profile. *Systems & Control Letters*, 54(1):53–63, 2005.
- [19] Ryan D. Hutchinson. *Generic Properties of Convolutional Codes*. PhD thesis, Department of Mathematics, University of Notre Dame, Indiana, USA, March 2006.
- [20] Rolf Johannesson and Zhe-Xian Wan. A linear algebra approach to minimal convolutional encoders. *IEEE Transactions on Information Theory*, 39(4):1219–1233, 1993.
- [21] Rolf Johannesson and Kamil Sh. Zigangirov. *Fundamentals of Convolutional Coding*. IEEE Press, New York, NY, 1999.
- [22] Thomas Kailath. *Linear Systems*. Prentice-Hall, Upper Saddle River, NJ, 1980.
- [23] Rudolf Emil Kalman, Peter L. Falb, and Michael A. Arbib. *Topics in Mathematical System Theory*. McGraw-Hill, New York, NY, 1969.
- [24] James L. Massey and Michael K. Sain. Codes, automata, and continuous systems: explicit interconnections. *IEEE Transactions on Automatic Control*, 12(6):644–650, 1967.
- [25] James L. Massey and Michael K. Sain. Inverses of linear sequential circuits. *IEEE Transactions on Computers*, 17(4):330–337, 1968.
- [26] Robert J. McEliece. The algebraic theory of convolutional codes. In V. S. Pless and W. C. Huffman, editors, *Handbook of Coding Theory*, pages 1065–1138. Elsevier, North-Holland, 1998.
- [27] Robert J. McEliece and Laif Swanson. Reed-Solomon codes and the exploration on the solar system. In Stephen B. Wicker and Vijay K. Bhargava, editors, *Reed-Solomon Codes and Their Applications*, chapter 3, pages 25–40. IEEE Press, Piscataway, NY, 1994.
- [28] Jim K. Omura. On the Viterbi decoding algorithm. *IEEE Transactions on Information Theory*, 15(1):177–179, 1969.
- [29] Jim K. Omura. Optimal receiver design for convolutional codes and channels with memory via control theoretical concepts. *Information Sciences*, 3(3):243–266, 1971.
- [30] Philippe Piret. *Convolutional Codes, an Algebraic Approach*. MIT Press, Boston, MA, 1988.
- [31] Joachim Rosenthal. Connections between linear systems and convolutional codes. In Brian Marcus and Joachim Rosenthal, editors, *Codes, Systems and Graphical Models*, volume 123 of *The IMA Volumes in Mathematics and its Applications*, pages 39–66. Springer-Verlag, New York, 2001.
- [32] Joachim Rosenthal, J. M. Schumacher, and Eric V. York. On behaviors and convolutional codes. *IEEE Transactions on Information Theory*, 42(6):1881–1891, 1996.
- [33] Joachim Rosenthal and Roxana Smarandache. Construction of convolutional codes using methods from linear systems theory. In *Proceedings of the 35th Allerton Conference on Communications, Control and Computing*, pages 953–960, Allerton House, Monticello, IL, September 1997.
- [34] Joachim Rosenthal and Roxana Smarandache. Maximum distance separable convolutional codes. *Applicable Algebra in Engineering, Communication and Computing*, 10(1):15–32, 1999.
- [35] Joachim Rosenthal and Eric V. York. BCH convolutional codes. *IEEE Transactions on Information Theory*, 45(6):1833–1844, 1999.
- [36] Roxana Smarandache, Heide Gluesing-Luerssen, and Joachim Rosenthal. Constructions of MDS-convolutional codes. *IEEE Transactions on Information Theory*, 47(5):2045–2049, 2001.
- [37] Roxana Smarandache and Joachim Rosenthal. A state space approach for constructing MDS rate $1/n$ convolutional codes. In *Proceedings of the 1998 IEEE Information Theory Workshop (ITW 1998)*, pages 116–117, Killarney, Kerry, Ireland, June 1998.
- [38] Andrew J. Viterbi. Convolutional codes and their performance in communications systems. *IEEE Transactions on Communication Technology*, 19(5):751–772, 1971.
- [39] Jan C. Willems. Systems theoretic models for the analysis of physical systems. *Riserve di Automatica*, 10:71–106, 1979.
- [40] Jan C. Willems. From time series to linear system – Part I. finite dimensional linear time invariant systems. *Automatica*, 22(5):561–580, 1986.
- [41] Jan C. Willems. Models for dynamics. In U. Kirchgraber and H. O. Walthert, editors, *Dynamics Reported*, volume 2, pages 171–269. John Wiley & Sons Ltd. and B.G. Teubner, 1989.