Some Interesting Problems in Systems Theory which are of Fundamental Importance in Coding Theory

Joachim Rosenthal[1]
Department of Mathematics,
University of Notre Dame,
Notre Dame, Indiana 46556, USA
e-mail: `Rosenthal.1@nd.edu`

Abstract

The paper explains in a tutorial way some major research questions in the area of convolutional coding theory from a perspective of systems theory. We will show that those questions are related to some classical control problems like e.g. the quadratic regulator problem and the tracking problem. We also report on some code constructions which were derived by techniques from systems theory.

1 Introduction

This paper is tutorial and it is intended for an audience in systems theory with no or little knowledge in coding theory. The interested reader who wants to read more on the presented connection of coding theory with systems theory is referred to the PhD thesis of York [22]. Standard references on convolutional coding theory are [9, 18].

Since the late sixties it has been widely known that convolutional codes and linear systems which are defined over a finite field are essentially the same objects. Those connections were probably first worked out in the papers by Massey and Sain [14, 15] and we will say more about this in Section 2. More recently there has been a new and increased interest in the connection of linear systems theory and convolutional coding theory and we refer to the recent articles [3, 5, 6, 11, 12, 19, 20, 22, 23]. Since convolutional codes are easily implemented and have good error correcting properties, they are among the most widely used codes in data transmission. We refer to [9, Chapter 17] to underline this point.

[1] Supported in part by NSF grant DMS-96-10389.

Coding theory is concerned with the reliable transmission of information over a "noisy channel". Typically one seeks an encoding device which encodes a certain message word m into a code word c which is then transmitted. The design objective is an encoding map $\varphi : \mathcal{M} \to \mathcal{C},\ m \mapsto c$ from the total set of message words $\mathcal{M}$ to the total set of code words $\mathcal{C}$ which enables one to efficiently and correctly decode a transmitted code word $\tilde{c}$.

If the total number of message words is finite one often identifies the message words $\mathcal{M}$ with a subset of the vector space $\mathbb{F}^k$, where $\mathbb{F}$ is a finite field, and one uses as a encoding device an injective linear map

$$\varphi : \mathbb{F}^k \to \mathbb{F}^n,\ m \mapsto c.$$

If G is a $n \times k$ matrix which represents the linear map φ one says G is an encoder and

$$\mathcal{C} = \{c \in \mathbb{F}^n \mid \exists m \in \mathbb{F}^k : c = Gm\}$$

is a linear block code.

Examples of linear block codes include the ASCII code which is a code over the binary field $\mathbb{F} = \{0,1\}$ and the set of ISBN numbers which consists of a 9-dimensional linear subspace of $\mathbb{F}^{10}$, where $\mathbb{F} = \mathbb{F}_{11}$ is the Galois field of 11 elements.

In practice it is often necessary to transmit a whole sequence of message words $m_0, m_1, \ldots, m_N$. Introducing the delay operator z we can express such a message sequence conveniently through the polynomial vector $m(z) = \sum_{i=0}^{N} m_i z^i$. In this way we can identify finite, but arbitrary long message sequences with the polynomial vector $m(z) \in \mathbb{F}^k[z]$. If G is an encoder of a linear block code then φ extends to a module homomorphism

$$\hat{\varphi} : \mathbb{F}^k[z] \mapsto \mathbb{F}^n[z],\ m(z) \mapsto c(z) = Gm(z). \quad (1)$$

2 Convolutional codes and the relation to linear systems

In 1955 Elias [2] proposed to replace in the encoding scheme (1) the encoding matrix G by a polynomial matrix $G(z)$ introducing in this way the concept of a convolutional code. Formally we can define a convolutional code through:

$$\mathcal{C} = \{c \in \mathbb{F}^n[z] \mid \exists m(z) \in \mathbb{F}^k[z] : c(z) = G(z)m(z)\}.$$

From a mathematical point of view $\mathcal{C}$ represents simply a $\mathbb{F}[z]$ submodule of $\mathbb{F}^n[z]$ and every submodule of $\mathbb{F}^n[z]$ is of this form.

We would like to remark that our definition imposes that the code words have a finite support. This condition was also used in the work of Valcher and Fornasini [21] but it is slightly different from a definition used by Forney [4]. This difference will be however of little importance for all practical applications.

If c is the largest degree of the $k \times k$ full size minors we will say that $\mathcal{C}$ has complexity c. Note that this definition does not depend on the particular choice of the encoder $G(z)$. Based on the earlier remarks we can view linear block codes as convolutional codes of complexity zero.

In the later sixties the coding theorist Massey and the systems theorist Sain explained in two important papers [14, 15] the connection between systems theory and convolutional coding theory. They pointed out that convolutional codes are conveniently described through a "linear sequential circuit" (LSC) and that every convolutional code has a so called feedforward realization. This realization can be obtained in the following way: Consider a polynomial, column proper encoder matrix $G(z)$ and assume that the matrices A, B, C, D form a minimal state space realization of the strictly proper transfer function $G(z^{-1})$, i.e.

$$G(z^{-1}) = C(zI - A)^{-1}B + D. \qquad (2)$$

Then the matrices A, B, C, D describe the linear sequential circuit

$$x_{t+1} = Ax_t + Bm_t, \;\; c_t = Cx_t + Dm_t, \;\; x_0 = 0. \quad (3)$$

Note that the transfer function $G(z^{-1})$ has McMillan degree c, the complexity of the encoder $G(z)$. If $G(z)$ has complexity zero then (3) reduces to a time-independent linear constraint at each code block c_t and this constraint is expressed through $y_t = Du_t$.

Example 2.1 ([20]) Let $\mathcal{C}$ be the convolutional code with generator matrix $G(z) = (z^2 + z + 1, z^2 + 1)^t$. The linear sequential circuit has then the form:

$$\begin{array}{ccccccc}
 & & & \oplus & \longrightarrow & c_t^{(1)} & \\
 & & \nearrow & \uparrow & \nwarrow & & \\
\longrightarrow & m_t & \longrightarrow & \boxed{x_t^{(1)}} & \rightarrow & \boxed{x_t^{(2)}} & . \\
 & & \searrow & & \swarrow & & \\
 & & & \oplus & \longrightarrow & c_t^{(2)} &
\end{array}$$

In terms of matrices the dynamics is given through:

$$\begin{array}{rcl}
x_{t+1} & = & \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} x_t + \begin{pmatrix} 1 \\ 0 \end{pmatrix} m_t \\
c_t & = & \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} x_t + \begin{pmatrix} 1 \\ 1 \end{pmatrix} m_t.
\end{array} \qquad (4)$$

Note that indeed the relation (2) holds.

The first order representation (3) is very natural but for the purpose of constructing good convolutional codes this representation is not optimal. For this we would like to note that a matrix tuple A, B, C, D which realizes the transfer function $G(z^{-1})$ is of a very special form. Indeed one can show e.g. that the matrix A has necessarily to be nilpotent.

Besides the dynamical system (3) there is a second natural linear system related to the convolutional code $\mathcal{C}$. This linear system can be obtained by duality as we explain in a moment.

Identify an infinite sequence space $w_0, w_1, w_2, \ldots$ of vectors in $\mathbb{F}^n$ with the formal power series $w(z) = \sum_i w_i z^i \in \mathbb{F}^n[[z]]$ and consider the bilinear form

$$\begin{array}{rrcl}
(\ ,\ ): & \mathbb{F}^n[[z]] \times \mathbb{F}^n[z] & \longrightarrow & \mathbb{F} \\
 & (w(z), c(z)) & \mapsto & \sum_{i=0}^{\infty} \langle w_i, c_i \rangle.
\end{array} \qquad (5)$$

As explained in detail in [19] the bilinear form $(\ ,\ )$ induces a duality between submodules of $\mathbb{F}^n[z]$ on one side and linear, left shift invariant and complete behaviors $\mathcal{B} \subset \mathbb{F}^n[[z]]$ on the other side. This duality is a special instance of a broad range of dualities between modules and shift spaces as they were studied by Oberst [16].

In terms of the encoding matrix $G(z)$ this duality amounts to the following: Partition the $n \times k$ matrix $G(z)$ into

$$G(z) = \begin{pmatrix} P(z) \\ Q(z) \end{pmatrix},$$

where $Q(z)$ is a square matrix of size $k \times k$. Correspondingly partition the code sequence $c(z)$ into

$$c_t := \begin{bmatrix} y_t \\ u_t \end{bmatrix}, \; t = 0, 1, 2, \dots$$

In the situation where $\deg \det Q(z) = c$, the complexity of $G(z)$, then there exists a controllable state space representation having the form

$$x_{t+1} = Ax_t + Bu_t, \;\; y_t = Cx_t + Du_t, \;\; x_0 = 0. \quad (6)$$

The associated transfer function $C(zI - A)^{-1}B + D$ is then equal to $P(z)Q(z)^{-1}$ and every controllable representation (6) describes a convolutional code. Moreover if (6) is an observable representation then the associated polynomial encoder $G(z)$ is right coprime and hence *non-catasrophic.* (See e.g. [18]).

In the coding literature [9] an encoder of the form

$$\hat{G}(z) = \begin{pmatrix} P(z)Q(z)^{-1} \\ I_k \end{pmatrix}$$

is referred to as a *systematic encoder.* In Example 2.1 one readily computes the realization of the transfer function $\frac{z^2+z+1}{z^2+1}$ as:

$$\begin{array}{rcl} x_{t+1} & = & \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} x_t + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u_t \\ y_t & = & \begin{pmatrix} 1 & 0 \end{pmatrix} x_t + (1) u_t. \end{array} \quad (7)$$

3 Code constructions and the free distance problem

So far we described the connection between systems theory and coding theory. In the rest of this paper we describe the design objectives and we report on some recent results.

An obvious design objective for a convolutional code $\mathcal{C}$ is the possibility to correct a large number of errors which might occur during the transmission. In order that this objective can be achieved it will be necessary that the codewords $\{c_t\}_{t \geq 0}$ are pairwise 'far apart'. Of course there is the question about a suitable metric. Unfortunately $\mathbb{F}^n$ is not an inner product space and hence there is no Euclidean metric available.

In part because of this reason coding theorist typically work with the *Hamming distance* and the *Hamming weight.* Formally those notions are defined in the following way.

If $a \in \mathbb{F}^n$ is any vector one defines the Hamming weight of a as the number of nonzero components of the n-vector a. We will denote the Hamming weight of a by $\mathrm{Ham}\,(a)$.

If a, b are two vectors in $\mathbb{F}^n$ one defines the Hamming distance $\mathrm{dist}\,(a, b)$ through the formula $\mathrm{dist}\,(a, b) := \mathrm{Ham}\,(a - b)$. The Hamming distance satisfies all axioms of a metric similarly to the Euclidean metric $||a - b||$, $a, b \in \mathbb{R}^n$.

If $\{c_t\}_{t \geq 0} = \left\{ \binom{u_t}{y_t} \right\}_{t \geq 0}$ is a particular code word we define its Hamming weight through:

$$\mathrm{Ham}\,(\{c_t\}_{t \geq 0}) := \sum_{t=0}^{\infty} \left(\mathrm{Ham}\,(u_t) + \mathrm{Ham}\,(y_t) \right).$$

Note that $\mathrm{Ham}\,(\{c_t\}_{t \geq 0})$ is by definition a positive integer. Finally if $\mathcal{C}$ is a convolutional code then we define the free distance $d_{free}(\mathcal{C})$ of $\mathcal{C}$ through:

$$\min \left\{ \mathrm{Ham}\,(\{c_t\}_{t \geq 0}) \mid \{c_t\}_{t \geq 0} \in \mathcal{C}, \{c_t\}_{t \geq 0} \neq 0 \right\}.$$

The *free distance problem* asks for the construction of a controllable and observable convolutional code which has maximal distance among all convolutional codes with a certain fixed rate and a certain complexity.

The reader certainly observes that the computation of the free distance is formally related to the linear quadratic regulator problem and this was also observed in [20]. Different from the Euclidean norm $||c_t||$, the Hamming weight $\mathrm{Ham}\,(c_t)$ seems not to be induced by a positive definite bilinear form and hence standard techniques from systems theory cannot be readily applied.

The free distance problem is a major design problem. If we require that the complexity of the code is zero then the free distance problem asks for the construction of a matrix D such that the $(n-k) \times n$ 'parity check matrix'

$$H := [I_{n-k} \mid -D]$$

has the property that any $d - 1$ columns of H are linearly independent. Indeed if this is the case any nonzero code-word will necessarily have weight at least d and the free distance of the associated code is therefore $d_{free} \geq d$. In the coding literature the free distance problem for convolutional codes of complexity zero is often referred to as the *main* *linear coding problem.* See e.g. [8, 10, 13] for introductory material on linear block codes. Contrary to the situation when the complexity is zero there are however very few algebraic constructions of convolutional codes. In Section 5 we show how to 'lift' one of the major constructions of linear block codes to the general situation of convolutional codes.

4 The decoding problem, a discrete tracking problem

On the side of the construction of codes with large free distance there is another important design objective. Indeed it should be possible to efficiently decode a received signal and to correct in this way the errors which occurred during the transmission. This is called the *decoding problem.*

Nowadays most convolutional codes are decoded using the so called Viterbi decoding algorithm. In order to explain this process we will assume that a certain code word $\{c_t\}_{t\geq 0} = \left\{\binom{u_t}{y_t}\right\}_{t\geq 0}$ was sent and the message word $\{\hat{c}_t\}_{t\geq 0} = \left\{\binom{\hat{u}_t}{\hat{y}_t}\right\}_{t\geq 0}$ has been received. The decoding problem then asks for the minimization of the error

$$\begin{aligned} \text{error} \;&:=\; \min_{\{c_t\}\in\mathcal{C}} \sum_{t=0}^{\infty} \text{dist}\,(c_t, \hat{c}_t) \qquad (8) \\ &=\; \min\left(\sum_{t=0}^{\infty}(\text{dist}\,(u_t,\hat{u}_t) + \text{dist}\,(y_t,\hat{y}_t))\right). \end{aligned}$$

If no transmission error did occur then $\{\hat{c}_t\}_{t\geq 0}$ is a valid trajectory and the error value in (8) is zero. If there was a transmission error the Viterbi decoding algorithm solves equation (8) applying the principle of dynamic programming. If fewer than d_{free} errors did occur it is guaranteed that an error can be detected and if fewer than $\frac{d_{free}-1}{2}$ errors did occur it is guaranteed that the original message $\{c_t\}_{t\geq 0}$ can be correctly decoded.

The decoding problem as we described it above has two interesting systems theoretic interpretations: First we can view the decoding problem as a 'tracking problem', where the decoder is supposed to track the incoming signal by the 'nearest trajectory'. Secondly we can view the decoding problem as a 'filtering problem' where some noise has to be filtered out. There exists a large systems literature both about tracking and filtering (deconvolution!) (see e.g. [7]). The main difficulty lies again in the fact that the underlying metric is not Euclidean but rather the Hamming metric.

The decoding problem is in general a highly complex problem and the Viterbi algorithm becomes in general infeasible if the complexity of the convolutional code is large.

Major classes of block codes like e.g. Reed Solomon codes, BCH codes and algebraic geometric codes are attractive since one has very efficient algebraic decoding algorithms available, which are in computational complexity superior over the Viterbi decoding algorithm.

In the area of convolutional codes there are very few classes of codes who come with decoding algorithms which are less complex than the Viterbi decoding algorithm. One remarkable exception is a new class of codes called *Turbo codes.* This class of codes was introduced in 1993 by Berrou, Glavieux and Thitimajshima [1]. In this paper the authors describe an encoding device consisting of two convolutional codes concatenated by a large 'interleaver'. The remarkable property is the fact that the encoding device performs near the channel capacity computed by Shannon and that the transmitted code words can be decoded with a simple iterative scheme.

A closer systems theoretic examination reveals that Turbo codes are equivalent to convolutional codes having complexity proportional to the interleaver size. An interleaver size of 50,000 is typical.

Turbo codes are currently under intense investigation and the recent paper [17] lists 43 papers and reports on Turbo codes which all were published in the last 4 years.

Despite this large effort it is our belief that a systems theoretic investigation of Turbo codes would be a worthwhile task. Indeed the fundamental question mentioned in [17] if the original decoding scheme presented in [1] necessarily converges to the optimal solution seems to be unresolved.

5 An algebraic code construction

In this paragraph we describe a convolutional code construction as it has been derived in [19] using systems theoretic ideas.

First we would like to review the classical Reed

Solomon construction for block codes. For this assume that q, the number of field elements of $\mathbb{F}$ is larger than k. Let α be a primitive of $\mathbb{F}$ and define

$$D := \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha & \alpha^2 & \cdots & \alpha^k \\ \alpha^2 & \alpha^4 & \cdots & \alpha^{2k} \\ \vdots & \vdots & & \vdots \\ \alpha^{n-k-1} & \alpha^{2(n-k-1)} & \cdots & \alpha^{k(n-k-1)} \end{pmatrix}.$$

A simple calculation shows that the $(n-k)\times n$ parity check matrix $H := [I_{n-k} \mid -D]$ has the property that any $n-k$ columns of H are linearly independent. This means that the linear block code $y_t = Du_t$ has (free) distance $n-k+1$. This is the maximum possible distance for a rate k/n linear block code. In the coding literature such a code is referred to as a Reed Solomon code. Reed Solomon codes are maximum distance separable (MDS) codes. We refer to [8, 10, 13] where the reader will find details.

Our assumption that $q \geq k$ is in general too restrictive although there are examples of block codes (like e.g. the international standard book numbers ISBN where $q = 11$ and $k = 9$) where this assumption does hold. If one wants to construct e.g. binary ($q = 2$) codes then one constructs first a block code over an extension field $\mathbb{F}_{q^m}$ of $\mathbb{F}_q$ and by rewriting this extended code over the base field $\mathbb{F}_q$ one obtains a code which is often referred to as a BCH code.

In the sequel we outline that the Reed Solomon construction generalizes to convolutional codes, whose complexity is $c > 0$.

Theorem 5.1 *Given a convolutional code $\mathcal{C}$ represented through* (6). *Let* $\mathbf{I}$ *be the* $(\gamma+1)(n-k)\times(\gamma+1)(n-k)$ *identity matrix and let* $c_i = \begin{bmatrix} y_i \\ u_i \end{bmatrix}$. *Then* $c(z) = c_0 + c_1 z + \ldots + c_\gamma z^\gamma \in \mathcal{C}$ *if and only if the vector* $\left(y_0^t, y_1^t, \ldots, y_\gamma^t, u_0^t, u_1^t, \ldots, u_\gamma^t\right)^t$ *is in the kernel of the matrix*

$$\left(\begin{array}{c|cccccc} 0 & A^\gamma B & A^{\gamma-1}B & \cdots & & AB & B \\ \hline & D & & & & & \\ & CB & D & & & & \\ -\mathbf{I} & CAB & \ddots & \ddots & & & \\ & \vdots & \ddots & CB & & D & \\ & CA^{\gamma-1}B & CA^{\gamma-2}B & \cdots & & CB & D \end{array}\right). \tag{9}$$

Note that the matrix appearing above takes the place of the usual sliding block matrix as it can be found in the coding theory literature (see e.g. [9]). Also note that the structure of this matrix is rather different from the sliding block form. By making particular choices of the matrices A, B, C, and D in (9) we can now attempt to find convolutional codes with good distance properties. We propose the following; let $c, n, k \in \mathbb{Z}_+$ with $n > k$. Choose a primitive α of the field $\mathbb{F}_q$, where $q \geq k(c^2+1)$, and define

$$A := \begin{pmatrix} \alpha^r & 0 & \cdots & 0 \\ 0 & \alpha^{2r} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \alpha^{cr} \end{pmatrix},$$

$$B := \begin{pmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{k-1} \\ 1 & \alpha^2 & \alpha^4 & \cdots & \alpha^{2(k-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha^c & \alpha^{2c} & \cdots & \alpha^{c(k-1)} \end{pmatrix},$$

$$C := \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha & \alpha^2 & \cdots & \alpha^c \\ \alpha^2 & \alpha^4 & \cdots & \alpha^{2c} \\ \vdots & \vdots & & \vdots \\ \alpha^{n-k-1} & \alpha^{2(n-k-1)} & \cdots & \alpha^{c(n-k-1)} \end{pmatrix}.$$

Furthermore, let D be any $(n-k)\times k$ matrix over $\mathbb{F}_q$. With this choice of matrices we have the following:

Theorem 5.2 *([19]) The matrices A, B, C and D define an observable, rate $\frac{k}{n}$, complexity c convolutional code $\mathcal{C}$ with free distance*

$$d_{free} \geq c+1.$$

We will call the value $\delta = c+1$ the designed distance of $\mathcal{C}$. It is easy to see that in the above matrices the matrix pair (A, B) is controllable and the matrix pair (A, C) is observable. The distance property of the code follows in part from the fact that our choice of parameters actually implies that the controllability matrix

$$\left[B \; AB \; \ldots \; A^{c^2-1}B\right], \tag{10}$$

is a Vandermond matrix. By examining (9), one can see that (10) being Vandermond implies the following: for all code words $v(z) \in \mathcal{C}$ with $\gamma = \deg(v(z)) < c^2$, we must have $\mathrm{Ham}(v(z)) > \delta$. For the case when $\deg(v(z)) \geq c^2$ we note that the observability of the pair (A, C) implies that the input and output of (6) has weight greater than δ.

In summary, the key steps for being able to construct codes with a designed distance δ for systems of the form (6) are that any choice of $\delta - 1$ columns of (9) must be linearly independent. For the complete proof of Theorem 5.2 we refer the reader to [19].

As in the classical Reed Solomon construction we will have to impose the restrictive assumption that the number of field elements q is larger than $k(c^2 + 1)$. If this assumption is not satisfied it is again possible to work over an extension field and to construct in this way 'BCH convolutional codes'. Details of this construction can be found in [22].

References

[1] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon limit error-correcting coding and decoding: Turbo-codes. In *Proc. of IEEE Int. Conference on Communication*, pages 1064–1070, Geneva, Switzerland, May 1993.

[2] P. Elias. Coding for noisy channels. *IRE Conv. Rec.*, 4:37–46, 1955.

[3] E. Fornasini and M.E. Valcher. Observability and extendability of finite support nD behaviors. In *Proc. of the 34th IEEE Conference on Decision and Control*, pages 3277–3282, New Orleans, Louisiana, 1995.

[4] G. D. Forney. Convolutional codes I: Algebraic structure. *IEEE Trans. Inform. Theory*, IT-16(5):720–738, 1970.

[5] G. D. Forney, B. Marcus, N. T. Sindhushayana, and M. Trott. A multilingual dictionary: System theory, coding theory, symbolic dynamics and automata theory. In *Different Aspects of Coding Theory*, number 50 in Proceedings of Symposia in Applied Mathematics, pages 109–138. American Mathematical Society, 1995.

[6] G. D. Forney and M. D. Trott. Controllability, observability, and duality in behavioral group systems. In *Proc. of the 34th IEEE Conference on Decision and Control*, pages 3259–3264, New Orleans, Louisiana, 1995.

[7] C. G. Goodwin and K. S. Sin. *Adaptive Filtering Prediction and Control.* Prentice Hall, 1984.

[8] R. Hill. *A First Course in Coding Theory.* Oxford Applied Math. & Comp. Science Series, 1988.

[9] S. Lin and D. Costello. *Error Control Coding: Fundamentals and Applications.* Prentice-Hall, Englewood Cliffs, NJ, 1983.

[10] J. H. van Lint. *Introduction to Coding Theory.* Springer Verlag, Berlin, New York, 1982.

[11] H. A. Loeliger, G. D. Forney, T. Mittelholzer, and M. D. Trott. Minimality and observability of group systems. *Linear Algebra Appl.*, 205/206:937–963, 1994.

[12] H. A. Loeliger and T. Mittelholzer. Convolution codes over groups. *IEEE Trans. Inform. Theory*, 42(6):1660–1686, 1996.

[13] F. J. MacWilliams and N. J.A. Sloane. *The Theory of Error-Correcting Codes.* North Holland, Amsterdam, 1977.

[14] J. L. Massey and M. K. Sain. Codes, automata, and continuous systems: Explicit interconnections. *IEEE Trans. Automat. Contr.*, AC-12(6):644–650, 1967.

[15] J. L. Massey and M. K. Sain. Inverse of linear sequential circuits. *IEEE Trans. on Computers*, C-17(4):330–337, 1968.

[16] U. Oberst. Multidimensional constant linear systems. *Acta Appl. Math*, 20:1–175, 1990.

[17] L. C. Perez, J. Seghers, and D. J. Costello, Jr. A distance spectrum interpretation of turbo codes. *IEEE Trans. Inform. Theory*, 42(6):1698–1709, 1996.

[18] Ph. Piret. *Convolutional Codes, an Algebraic Approach.* MIT Press, Cambridge, MA, 1988.

[19] J. Rosenthal, J. M. Schumacher, and E.V. York. On behaviors and convolutional codes. *IEEE Trans. Inform. Theory*, 42(6):1881–1891, 1996.

[20] J. Rosenthal and E.V. York. Linear systems defined over a finite field, dynamic programming and convolutional codes. In *Proc. of the IFAC Conference on System Structure and Control*, pages 466–471, Nantes, France, July 1995.

[21] M.E. Valcher and E. Fornasini. On 2D finite support convolutional codes: an algebraic approach. *Multidim. Sys. and Sign. Proc.*, 5:231–243, 1994.

[22] E.V. York. *Algebraic Description and Construction of Error Correcting Codes, a Systems Theory Point of View.* PhD thesis, University of Notre Dame, 1997. Available at http://www.nd.edu/~rosen/preprints.html.

[23] E.V. York, J. Rosenthal, and J. M. Schumacher. On the relationship between algebraic systems theory and coding theory: Representations of codes. In *Proc. of the 34th IEEE Conference on Decision and Control*, pages 3271–3276, New Orleans, Louisiana, 1995.