Codes, Trellis Representations and the Interplay of System Theory and Coding Theory

Heide Gluesing-Luerssen

***Abstract*— We will present state realizations (trellises) for convolutional codes and block codes and discuss their system theoretic properties. For convolutional codes we will give an application of such realizations by presenting a MacWilliams Identity which relates the weight distribution of a code to the weight distribution of the dual code. For block codes, state realizations are based on the interpretation of the code (of length n, say) as a set of admissible trajectories (codewords) on the time axis $0,\dots,n-1$. After discussing conventional trellis realizations, we will turn to tail-biting trellises in which the time axis is considered circular and time index arithmetic is performed modulo n. In this case the future and past of trajectories are intertwined. One particular consequence is that minimal tail-biting trellises are not unique. We will show how to obtain all minimal tail-biting trellises for a given code and discuss further properties.**

I. INTRODUCTION

Block codes and convolutional codes are the most widely used types of codes in communication systems. In this paper both classes of codes will be discussed. We will take the viewpoint of regarding these codes as behaviors (where the codewords become the admissible trajectories) and will discuss various state realizations. For both types of codes these realizations (trellis representations) play an important role. They do not only reveal information about the code structure but also lend themselves to the efficient Viterbi decoding algorithm [21]. Originally, this algorithm has been derived for convolutional codes, but by now has found, due to its optimality (maximum likelihood decoder) widespread applications for other codes as well. We will briefly sketch the ideas of the Viterbi algorithm in Section II.A. Since the complexity of this algorithm depends on the size of the trellis, one is highly interested in finding *minimal trellis realizations* with respect to various measures. When measuring the size of a trellis in terms of its state space this simply becomes the familiar system theoretic quest of finding minimal realizations.

We will present two types of minimal trellis representations for convolutional codes. As an application we will show a MacWilliams Identity Theorem which relates the weight distribution of a code to the weight distribution of the dual code. Here the weight distribution is a matrix, indexed by the set of all ordered state pairs, and where each entry contains the weight distribution of all outputs (codeword block) associated with the corresponding state transition.

This work was partially supported by by National Science Foundation grant #DMS-0908379

University of Kentucky, Department of Mathematics, 715 Patterson Office Tower, Lexington, KY 40506, USA; heidegl@ms.uky.edu

Thereafter we will turn to trellis representations for block codes. Vectors in $\mathbb{F}^n$ will be interpreted as trajectories of length n so that block codes (subspaces of $\mathbb{F}^n$) may be regarded as behaviors over a time axis of length n. In this paper we will restrict to two types of trellis realizations, conventional and tail-biting ones. For conventional trellises the state space at time zero is the zero space, whereas for tail-biting trellises this space may be non-trivial and one regards the time axis circular so that admissible trajectories correspond to cycles through the trellis (rather than paths).

For conventional trellises it is by now well-known that for a given linear block code (and with a fixed coordinate ordering) there exists a unique minimal trellis. This unique minimal trellis simultaneously minimizes the state space dimension at each time as well as any other conceivable complexity measure. In Section III.A we will introduce this minimal trellis via the past-future induced minimal realization, as it has been constructed by Forney [3].

For tail-biting trellises the situation is considerably more difficult. Due to an additional degree of freedom (the state space at time 0) minimal tail-biting trellises are not unique and this applies to minimality with respect to various measures [2], [12]. As opposed to conventional trellises, minimality is not equivalent to non-mergeability anymore (where a trellis is called non-mergeable if at each time the only sets of states that can be lumped are singletons). We will present a construction from [12] which allows to deduce all minimal tail-biting trellises from it. Moreover, we will introduce a construction of trellises going back to [17], and which may be regarded in some sense as an observer-canonical form. We will study properties of these constructions. As an application we will finally address a question of duality, which in some sense amounts to how to dualize a given tail-biting trellis in order to obtain a trellis representing the dual code.

We close the introduction with briefly recalling the basic notions of block codes (and coding theory), while convolutional codes will be introduced in the next section. Throughout, let $\mathbb{F}$ be a finite field. A *block code* of length n is a subspace of $\mathbb{F}^n$. Using row vector notation, we may write a k-dimensional block code as $\mathcal{C} = \operatorname{im} G := \{uG \mid u \in \mathbb{F}^k\}$ for some $G \in \mathbb{F}^{k\times n}$. The matrix G is called an encoder matrix of the code: it encodes the *message* $u \in \mathbb{F}^k$ into the associated *codeword* $v := uG \in \mathbb{F}^n$. The encoding process amounts to adding redundancy to the message, which then will (hopefully) lead to the possibility of error correction when the codeword is being sent over a transmission channel. In order to measure errors one usually endows the vector space $\mathbb{F}^n$ with the *Hamming metric*

ISBN 978-963-311-370-7

$d\big((v_1,\dots,v_n),(w_1,\dots,w_n)\big) = |\{i \mid v_i \neq w_i\}|$. The theoretical error-correcting capability of the code is determined by its *minimum distance* $\mathrm{dist}(\mathcal{C}) = \min\{d(v,w) \mid v, w \in \mathcal{C}, v \neq w\}$. Indeed, over a symmetric and memoryless transmission channel the codeword that most likely has been sent given that $\hat{v} \in \mathbb{F}^n$ has been received is the codeword in $\mathcal{C}$ closest to $\hat{v}$ with respect to the Hamming distance. (Over other channels a different metric or a probability measure may be necessary). In this sense, the sent codeword can be reconstructed via a best approximation to the received word. This will indeed reconstruct the sent codeword $v \in \mathcal{C}$ if no more than $(\mathrm{dist}(\mathcal{C})-1)/2$ errors occurred during the transmission. The main challenge of block code theory is the design of block codes with large minimum distance along with efficient decoding algorithms (finding a best approximation). The system theoretic interpretation of block codes as behaviors will be introduced in Section III.

II. Convolutional Codes

From a systems theoretic point of view a convolutional code is a linear time-invariant discrete-time system over a finite field. Consider the field $\mathbb{F}(\!(D)\!) = \{\sum_{t\geq T} a_t D^t \mid T \in \mathbb{Z},\, a_t \in \mathbb{F}\}$ of Laurent series in the indeterminate D. A convolutional code of length n is a $\mathbb{F}(\!(D)\!)$-subspace of $\mathbb{F}(\!(D)\!)^n$. Its elements are the codewords, or, in behavioral language, the admissible trajectories. If the code is k-dimensional, we may represent it as $\mathcal{C} = \mathrm{im}\, G := \{uG \mid u \in \mathbb{F}(\!(D)\!)^k\}$ for some $G \in \mathbb{F}(\!(D)\!)^{k\times n}$. Hence G consists of an $\mathbb{F}(\!(D)\!)$-basis of $\mathcal{C}$. In this notation, messages $u = \sum_{t\geq T} u_t D^t \in \mathbb{F}(\!(D)\!)^k$ are understood as the D-transforms of the sequence of message blocks $u_t \in \mathbb{F}^k$. The matrix G then consists of the D-transforms of the k impulse responses at time 0. In other words, it may be regarded as the transfer matrix mapping input messages $u \in \mathbb{F}(\!(D)\!)^k$ to output codewords $uG \in \mathbb{F}(\!(D)\!)^n$. Just like in the behavioral setting, the system (or code) is the set of trajectories $\mathcal{C} = \mathrm{im}\, G$, rather than the transfer function. This gives rise to the task of finding particularly nice encoder matrices for a given code. Using that $\mathbb{F}(\!(D)\!)$ is a field, the encoder G can always be chosen *polynomial*, that is, $G \in \mathbb{F}[D]^{k\times n}$. In other words, the encoder map can be made time-causal [1]. Moreover, each code has a polynomial encoder G with one or more of the following properties.

[1]One should observe that for classical discrete-time linear systems one uses the z-transform instead of the D-transform, where $z = D^{-1}$. This time-reversal turns a polynomial encoder matrix into a proper rational matrix, which is the more familiar version of causality.

(a) G is *delay-free* if $\mathrm{rk}\, G(0) = k$.

(b) G is *non-catastrophic* if the greatest common divisor of all k-minors of G is of the form D^m for some $m \in \mathbb{N}_0$. In other words, $\mathrm{rk}\, G(\alpha) = k$ for all $\alpha \in \overline{\mathbb{F}}\backslash\{0\}$, where $\overline{\mathbb{F}}$ is an algebraic closure of $\mathbb{F}$. This is equivalent to saying that infinite message strings are never encoded into finite codeword strings.

(c) G is *row-proper* if the row-wise highest coefficient matrix of G has rank k.

In Section III we will encounter a block code analogue of delay-free, non-catastrophic, and row-proper encoders.

A. Trellis Representations of Convolutional Codes

Being a discrete-time linear system, a code $\mathcal{C} \subseteq \mathbb{F}(\!(D)\!)^n$ can be realized as a state-space system. There are naturally two different notions of realizations that lend themselves in this particular case. Let us begin with what we call the driving variable realization. In this case we choose an encoder $G \in \mathbb{F}[D]^{k\times n}$ which we consider as the transfer matrix (recall footnote 1) so that the message is the input stream and the associated codeword is the output. Then there exists some $\delta \in \mathbb{N}_0$ and matrices $(A, B, C, E) \in \mathbb{F}^{\delta\times\delta+k\times\delta+\delta\times n+k\times n}$ such that $u = \sum_{t\geq T} u_t D^t \in \mathbb{F}(\!(D)\!)^k$ and $v = \sum_{t\geq T} v_t D^t \in \mathbb{F}(\!(D)\!)^n$ satisfy

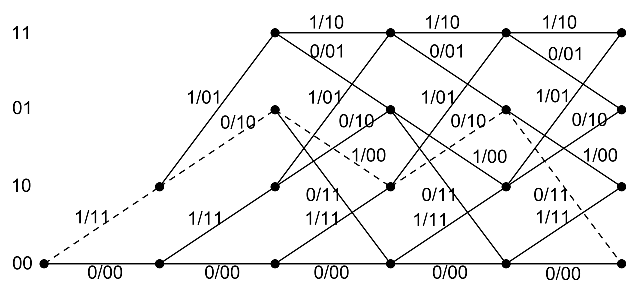
$$v = uG \Longleftrightarrow \left\{ \begin{array}{rcl} x_{t+1} &=& x_t A + u_t B \\ v_t &=& x_t C + u_t E \end{array} \text{ and } x_T = 0. \right\} \quad (1)$$

The McMillan degree, the minimal δ for which such a *driving variable realization* exists, is called the *overall constraint length* of the code. It coincides with the sum of the row degrees in a row-proper, delay-free, and non-catastrophic encoder matrix. Evidently, the realization depends on the choice of the encoder and in fact, it can be shown [8, Thm. 2.6] that for a given code $\mathcal{C}$ the driving variable realization is unique up to the full state feedback group.

Example 1 Let $\mathcal{C} = \mathrm{im}\, G \subseteq \mathbb{F}(\!(D)\!)^2$ where $\mathbb{F} = \mathbb{F}_2$ and $G = (1 + D + D^2,\, 1 + D^2)$. Using the controller canonical form one easily obtains the minimal driving variable realization

$$x_{t+1} = x_t \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} + u_t \begin{pmatrix} 1 & 0 \end{pmatrix}, \;\; v_t = x_t \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} + u_t \begin{pmatrix} 1 & 1 \end{pmatrix}.$$

Since the state space is given by $\mathbb{F}^2$ and thus consists of only 4 elements (just like for any convolutional code, the system can be regarded as a finite-state machine), there exist only finitely many state transitions at each time which allows



us to display all codewords as trajectories in a trellis diagram given at the bottom of the previous page. Here the four states are listed on the vertical axis. Moreover, we restricted the system to starting time zero. A state transition $x_{t+1} = x_tA + u_tB$, $v_t = x_tC + u_tE$ is represented by an edge from x_t to x_{t+1} with the input u_t as first label and the output v_t as the second label. The codewords correspond to the paths through the trellis starting at time 0 and the finite (that is, polynomial) codewords correspond to those paths that end in the zero state at some finite time. The corresponding input sequence can be read off via the first labels following the path. In this trellis the dashed path represents the polynomial codeword $v = (1 + D^2)G = (1 + D + D^3 + D^4, 1 + D^4)$. The repetitive pattern of the trellis diagram after time $t = 2$ simply reflects the time-invariance of the linear system.

It is exactly this type of trellis representation of a convolutional code that explains best the famous Viterbi decoding algorithm [21], which reshaped the whole area of information technology. Indeed, given the received word $\hat{v} = \sum_{t\geq T} \hat{v}_tD^t$, the Viterbi algorithm follows, in an efficient way, all paths through the trellis (starting at time T at the zero state) and singles out the one that is closest to the received word, that is, a codeword $v = \sum_{t\geq T} v_tD^t \in \mathcal{C}$ such that the overall Hamming distance $\sum_{t\geq T} d(v_t, \hat{v}_t)$ is minimized among all codewords. The efficiency is obtained by comparing for each time t and each state at that time all paths leading into the state and choosing the one closest (so far) to the received word. One can then inductively proceed by only storing those survivors for each state at that time. Instead of minimizing the overall Hamming distance $\sum_{t\geq T} d(v_t, \hat{v}_t)$ any other metric or probability measure may be used as well, which makes this algorithm available for various types of transmission channels. We refer to the vast literature for detailed descriptions of the algorithm, its exciting history, and its widespread applications.

Let us now briefly present a different state-space realization for convolutional codes that turns out to be very useful as well. Following again behavioral systems theory a code $\mathcal{C} = \operatorname{im} G$ may be regarded as an image representation describing the relations between the manifest variables. One may now investigate whether some of these variables can be declared as inputs while the others act as outputs. Aiming at a time-causal relationship between inputs and outputs and switching to the usual system-theoretic D^{-1}-transform where messages and codewords have entries in $\mathbb{F}(\!(D^{-1})\!)$, this amounts to the familiar partitioning of the encoder matrix into $G = (Q,\, P)$ such that $Q \in \mathbb{F}[D]^{k\times k}$ is regular, $P \in \mathbb{F}[D]^{k\times(n-k)}$, and $Q^{-1}P \in \mathbb{F}(D)^{k\times(n-k)}$ is proper. Then $\mathcal{C} = \operatorname{im}(I,\, Q^{-1}P)$ and thus a vector $v = (u,\, y) \in \mathbb{F}(\!(D^{-1})\!)^n$ is a codeword if and only if $y = Q^{-1}Pu$. The encoder $(I,\, Q^{-1}P)$ is called *systematic* because the message $u \in \mathbb{F}(\!(D^{-1})\!)^k$ appears unaltered as the first k entries of the associated codeword. A state-space realization of the transfer matrix $Q^{-1}P$ now results in the *input/state/output-realization*

$$x_{t+1} = x_t\tilde{A} + u_t\tilde{B},\ y_t = x_t\tilde{C} + u_t\tilde{E} \qquad (2)$$

with constant matrices $(\tilde{A}, \tilde{B}, \tilde{C}, \tilde{E})$ of suitable sizes. In this realization the codeword consists of the input and output components, that is, is given by the sequence $v_t = (u_t,\, y_t)^{\mathsf{T}}$. Of course, a minimal i/s/o-realization is uniquely determined by the given code up to state space isomorphism. Again, the system may be illustrated by a trellis diagram.

B. Weight Enumeration and Duality

While the distance of a given (block or convolutional) code gives basic information about its error-correcting quality, much more detailed information is contained in the weight distribution, that is, the cardinality of codewords of any given weight. One of the most celebrated results in block coding theory is the MacWilliams Identity. It relates the (Hamming) weight distribution of a code $\mathcal{C} \subseteq \mathbb{F}^n$ to that of its dual $\mathcal{C}^\perp := \{w \in \mathbb{F}^n \mid wv^{\mathsf{T}} = 0 \text{ for all } v \in \mathcal{C}\}$. Define the *weight enumerator* of a subset $S \subseteq \mathbb{F}^n$ as $\operatorname{we}(S) := \sum_{j=0}^n \alpha_j W^j \in \mathbb{C}[W]_{\leq n}$ where $\alpha_j := |\{a \in S \mid \operatorname{wt}(a) = j\}|$. Then the classical MacWilliams Identity for block codes states that if $\mathcal{C} \subseteq \mathbb{F}^n$ is a k-dimensional code and $\mathbb{F}$ is a field with q elements, then $\operatorname{we}(\mathcal{C}^\perp) = q^{-k}\mathbf{H}\big(\operatorname{we}(\mathcal{C})\big)$ with $\mathbf{H} : \mathbb{C}[W]_{\leq n} \to \mathbb{C}[W]_{\leq n}$ being the MacWilliams transform

$$\mathbf{H}(f)(W) := (1 + (q-1)W)^n f\big(\tfrac{1-W}{1+(q-1)W}\big). \qquad (3)$$

This result has been generalizes to convolutional codes in [9], [10], see also [5]. It has been known for a long time [19] that such a MacWilliams Identity for convolutional codes is not as straightforward as one would first expect. Indeed, one needs a very detailed weight enumerating invariant. In the sequel we will briefly sketch this object and the resulting identity.

Let $G \in \mathbb{F}[D]^{k\times n}$ be a reduced encoder with minimal driving variable realization (A, B, C, E) as in (1). The *weight adjacency matrix* (WAM) of the realization is defined to be the matrix $\Lambda := \Lambda(G) \in \mathbb{C}[W]^{q^\delta\times q^\delta}$ indexed by $(X, Y) \in \mathbb{F}^\delta \times \mathbb{F}^\delta$ with the entries

$$\Lambda_{X,Y} := \operatorname{we}\{XC + uE \mid u \in \mathbb{F}^k : Y = XA + uB\}.$$

In other words, the WAM enumerates the output weights for each pair of state-transitions (due to time-invariance the time index will not play a role in these considerations). The WAM carries detailed information about the error-correcting performance of the code; for details we refer to [7, Sec. 2]. It is important to note that the WAM is not an invariant of the code but rather depends on the choice of the encoder and its realization. Precisely, if Λ is a particular WAM of $\mathcal{C}$, then the orbit $[\Lambda] := \{\Lambda' \mid \exists\ P \in GL_\delta(\mathbb{F}) :\ \Lambda'_{X,Y} = \Lambda_{XP,YP} \text{ for all } X, Y \in \mathbb{F}^\delta\}$ forms an invariant of $\mathcal{C}$, called the *generalized weight adjacency matrix* of $\mathcal{C}$.

In order to formulate the MacWilliams Identity we need a notion of duality for convolutional codes. There exist several notions in the literature, all of which are closely related to each other. We will choose the dual of $\mathcal{C} \subseteq \mathbb{F}(\!(D)\!)^n$ defined as $\mathcal{C}^\perp = \{w \in \mathbb{F}(\!(D)\!)^n \mid wv^{\mathsf{T}} = 0\}$, where wv^{T} refers to the usual inner product in the $\mathbb{F}(\!(D)\!)$-vector space $\mathbb{F}(\!(D)\!)^n$. Now we can present our main result, proven in [9], [10].

Theorem 2 *Let $\mathbb{F}$ be a field with q elements and let $\mathcal{C} \subseteq \mathbb{F}(\!(D)\!)^n$ be a convolutional code with Λ being a particular WAM of $\mathcal{C}$. Furthermore, let $\widehat{\Lambda}$ be a particular WAM of the dual code $\mathcal{C}^\perp$. Then there exists a matrix $P \in GL_\delta(\mathbb{F})$ such that*

$$\widehat{\Lambda}_{X,Y} = q^{-k}\mathbf{H}\big((\mathcal{H}\Lambda^\mathsf{T}\mathcal{H}^{-1})_{XP,YP}\big) \text{ for all } X,\, Y \in \mathbb{F}^\delta, \quad (5)$$

where $\mathbf{H}$ is the MacWilliams transform as in (3) and the (Fourier) matrix $\mathcal{H}$ is defined as

$$\mathcal{H} := q^{-\frac{\delta}{2}}\big(\chi(XY^\mathsf{T})\big)_{X,Y\in\mathbb{F}^\delta} \in \mathbb{C}^{q^\delta \times q^\delta}$$

with $\chi : \mathbb{F} \longrightarrow \mathbb{C}^*$ being a non-trivial character on $\mathbb{F}$. As a consequence, the generalized weight adjacency matrices $[\Lambda]$ and $[\widehat{\Lambda}]$ of the codes $\mathcal{C}$ and $\widehat{\mathcal{C}}$ satisfy $[\widehat{\Lambda}] = q^{-k}\mathbf{H}(\mathcal{H}[\Lambda]^\mathsf{T}\mathcal{H}^{-1})$.

The proof of this result in [9], [10] is, just like the exposition given here, based on a driving variable realization (1) of the code. A more elegant and easier proof has been presented recently by Forney in [5]. It makes use of the i/s/o realization in (2). The advantage of that approach is that the i/s/o realization directly gives rise to a dual realization. Given the realization (2) for $\mathcal{C} = \mathrm{im}\,(Q,P)$, the quadruple $(\tilde{A}^\mathsf{T}, -\tilde{C}^\mathsf{T}, \tilde{B}^\mathsf{T}, -\tilde{E}^\mathsf{T})$ results in an i/s/o realization of the dual code $\mathcal{C}^\perp$ with respect to the dual partition into inputs and outputs (and for a different, but closely related notion of duality). This in turn allows to avoid the difficulties of the state space isomorphisms (the matrix P in the theorem above) and the MacWilliams Identity (5) is derived much more straightforwardly. In either approach the main tool for proving the identity is, just like for the block code version, the Poisson formula for a suitably defined weight generating function and its Fourier transform.

III. Trellis Representations of Block Codes

In this section we return to block codes $\mathcal{C} \subseteq \mathbb{F}^n$. We will now consider such codes as behaviors (sets of admissible trajectories) on the time axis $\mathcal{I} := \{0, 1, \ldots, n-1\}$. This will allow us to present the code via a state-space system (trellis) similar to those for convolutional codes, only that now the labels will be the entries of the codewords.[2] As explained in the introduction, the foremost motivation for such trellis representations is their usefulness for the Viterbi decoding algorithm and therefore one aims at *minimal trellis realizations.*

[2] One may also consider other sectionalizations, where one bundles, say, two consecutive, entries of the codewords resulting in trellises over a time axis of length $n/2$ for codes of even length n.

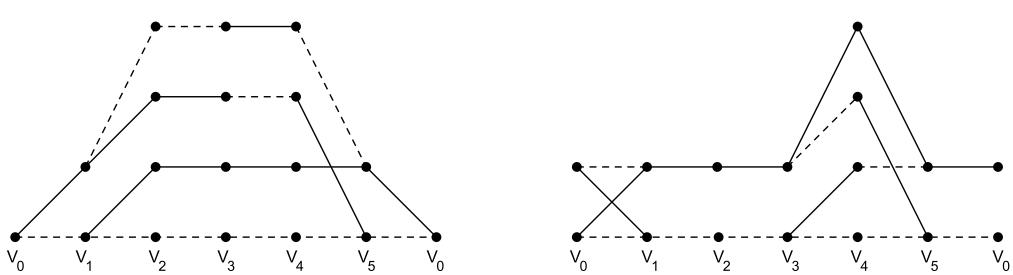
We need to fix some notation. Throughout this section, a (tail-biting) trellis $T = (\cup_{i=0}^{n-1} V_i, \cup_{i=0}^{n-1} E_i)$ is a labeled graph with vertex sets V_i and edge sets E_i for $i \in \mathcal{I}$ and with edge labels from $\mathbb{F}$. Edges may only connect vertices at consecutive times, thus $E_i \subseteq V_i \times \mathbb{F} \times V_{i+1}$. We will compute modulo n on the time axis $\mathcal{I}$. This will become relevant only later on when dealing with non-conventional tail-biting trellises. Hence $V_n := V_0$ and $E_{n-1} \subseteq V_{n-1} \times \mathbb{F} \times V_0$. All trellises throughout this section will be linear in the sense of [12, Def. 4.1]. In particular, V_i and E_i are linear spaces. The vertices in V_i will be called states (at time i) and the edges represent the state transitions and we will call the edge spaces E_i the *local behaviors*. As we will see in the next section all this is consistent with the system theoretic usage of these notion. We say that a trellis T *represents the code* $\mathcal{C} \subseteq \mathbb{F}^n$ if $\mathcal{C}(T) = \mathcal{C}$, where $\mathcal{C}(T)$ is defined as the set of all vectors $c = (c_0, \ldots, c_{n-1}) \in \mathbb{F}^n$ for which there exists a cycle in T (a path starting and ending in the same state at time zero) having c as its edge-label sequence. The trellis is called one-to-one if different cycles have different edge-label sequences. The trellis is called *conventional* if $V_0 = \{0\}$ (hence all paths are cycles) and *tail-biting* in general. The trellis $T = (V, E)$ is called *minimal* if there exists no trellis $T' = (V', E')$ representing $\mathcal{C}$ such that $|V_i'| \le |V_i|$ for all $i \in \mathcal{I}$ and $|V_j'| < |V_j|$ for some $j \in \mathcal{I}$ (there are various notions of measuring the size of trellises, this one being one of the most general ones). Moreover, T is called *non-mergeable* if for any $i \in \mathcal{I}$ merging any two distinct vertices $v_1, v_2 \in V_i$ results in a trellis $\hat{T}$ that does not represent $\mathcal{C}$ (that is, merging causes new cycles that do not represent codewords). We will also consider the *state complexity profile* (SCP) defined as $(s_0, \ldots, s_{n-1})$, where $s_i = \dim V_i$, and the *edge complexity profile* (ECP) defined as $(e_0, \ldots, e_{n-1})$, where $e_i = \dim E_i$.

Example 3 Consider the binary code

$$\mathcal{C} = \mathrm{im}\begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \subseteq \mathbb{F}_2^6.$$

At the bottom of this page two minimal trellis representations of this code are given, the first one being conventional. The edge label 0 is indicated by a dashed line, while solid lines denote the edge label 1. In both cases there exist 4 cycles through the trellis and their edge-label sequences are the 4 codewords of $\mathcal{C}$.

From the viewpoint of behavioral systems theory a trellis T is a dynamical system with latent variables in the sense of [18, Def. 1.3.4], where the latent variables represent the



(4)

state sequence which indeed satisfies the axiom of state at each time, see [22, Def. 1.3]. Notice that the system is, in general, highly time-varying since even the state space dimensions depend on time. The label code, consisting of all sequences of edge labels *and* state labels, is the full behavior while the edge-label code $\mathcal{C}(T)$ is the manifest behavior.

The very well understood theory of conventional trellises will be presented briefly in the next subsection. The theory of tail-biting trellises is still much less developed and we will give an account of some of the latest developments in Subsection B.

Before starting with the details of trellis representations for block codes, let us also introduce the following. Due to the cyclic structure of the time axis $\mathcal{I}$, the following interval notation has proven to be very convenient. For $a,\, b \in \mathcal{I}$ we define $[a,\, b] := \{a, a+1, \ldots, b\}$ if $a \leq b$ and $[a,\, b] := \{a, a+1, \ldots, n-1, 0, 1, \ldots, b\}$ if $a > b$. Moreover, we define the semi-open interval $(a,\, b] = [a, b]\backslash\{a\}$. We call the intervals $(a,\, b]$ and $[a,\, b]$ *conventional* if $a \leq b$ and *circular* else. For a nonzero vector $c = (c_0, \ldots, c_{n-1}) \in \mathbb{F}^n$ we call the interval $(a,\, b]$ a *span* of c if $c_a \neq 0 \neq c_b$ and $c_j = 0$ for $j \notin [a,\, b]$. Notice that every nonzero vector has a unique conventional span. It might appear surprising to call the half-open interval $(a, b]$ the span of c rather than the closed interval $[a, b]$. While this is simply convenient for later usage, the reason is actually that in a trellis realization one has two (interlaced) time axes, one for the states and one for the symbols (the edge labels). This can also be seen from Figure (4). Consequently, it would be more consistent to make these two time axes explicit, as it is done in [4], [6]. Our choice not to introduce this extra layer of notation leads to the definition above for the span of a vector.

Finally, for a subset $\mathcal{A} \subseteq \mathcal{I}$ we define the indicator function $I^{\mathcal{A}} \in \mathbb{F}^n$ as $I_i^{\mathcal{A}} = 1$ if $i \in \mathcal{A}$ and $I_i^{\mathcal{A}} = 0$ else.

For the rest of this paper, let $\mathcal{C} = \operatorname{im} G = \ker H^{\mathsf{T}} := \{v \in \mathbb{F}^n \mid vH^{\mathsf{T}} = 0\} \subseteq \mathbb{F}^n$ denote a k-dimensional code of length n, where the encoder matrix $G \in \mathbb{F}^{k\times n}$ and the parity check matrix H are specified as

$$G = (G_{lj})_{\substack{l=1,\ldots,k \\ j=0,\ldots,n-1}} = \begin{pmatrix} G_0 & \ldots & G_{n-1} \end{pmatrix},$$
$$H = \begin{pmatrix} H_0 & \ldots & H_{n-1} \end{pmatrix} \in \mathbb{F}^{(n-k)\times n}.$$

For simplicity we will also assume that the support of $\mathcal{C}$ is $\mathcal{I} = \{0, \ldots, n-1\}$, that is, no column of G is zero.

A. Conventional Trellises

Recall that a trellis $T = (V, E)$ is called conventional if $V_0 = \{0\}$. A conventional trellis representation of $\mathcal{C}$ is simply a state-space realization along the time axis $\mathcal{I}$ in the classical sense of systems theory. Indeed, since in a conventional trellis all paths are cycles the vertices can be regarded as states in the sense that any past trajectory leading into that vertex can be concatenated with any future trajectory; see also the first trellis in (4).

Such a state-space realization can be obtained by the usual methods of systems theory, for instance, the past-future induced realization, presented first by Forney [3].

Put $\mathcal{C}_0 = \mathcal{C}$ and for $i = 1, \ldots, n$ put

$$\mathcal{C}_i := \{(c_0, \ldots, c_{n-1}) \in \mathcal{C} \mid \textstyle\sum_{j=0}^{i-1} c_j H_j = 0\} \tag{6}$$

and

$$\hat{V}_i := \mathcal{C}/\mathcal{C}_i = \{[c]_i \mid c \in \mathcal{C}\}, \text{ where } [c]_i := c + \mathcal{C}_i,$$
$$\hat{E}_i := \{([c]_i, c_i, [c]_{i+1}) \mid c \in \mathcal{C}\}.$$

Then the trellis $\hat{T} = (\hat{V}, \hat{E})$ is a minimal conventional trellis of $\mathcal{C}$. Moreover, $\hat{T}$ is non-mergeable and every conventional one-to-one trellis $T = (V, E)$ of $\mathcal{C}$ can be merged into $\hat{T}$ (the maps $\varphi_i : V_i \to \hat{V}_i,\, v \mapsto [c]_i$, where $c \in \mathcal{C}$ is any codeword whose path in T passes through v, are well-defined and linear and thus $V_i/\ker\varphi_i \cong \hat{V}_i$, which furnishes the merging). As a consequence, for conventional trellises non-mergeability is equivalent to minimality. For details on all this we refer to [16], [14], [20].

A concrete way of creating this minimal trellis (or actually computing the subcodes $\mathcal{C}_i$) is given by a particular choice of encoder matrix of the code. Indeed, using basic matrix algebra one can show that $\mathcal{C}$ has an encoder matrix G such that the conventional spans $(a_l, b_l]$, $l = 1, \ldots, k$, of the rows of G have distinct starting points $a_1, \ldots, a_k$ and distinct end points $b_1, \ldots, b_k$. Such an encoder is called a minimal-span generator matrix (MSGM). Evidently, if G is an MSGM then the conventional span of any nonzero codeword $v = (u_1, \ldots, u_k)G$ is given by $(a,\, b]$ where $a = \min\{a_l \mid u_l \neq 0\}$ and $b = \max\{b_l \mid u_l \neq 0\}$. In other words, the MSGM satisfies a predictable span property. This makes the MSGM the block code counterpart to the delay-free, non-catastrophic and row-proper polynomial encoder matrix for convolutional codes. The predictability of the starting point a of the span corresponds to the delay-freeness of a polynomial encoder (preventing the cancelation of constant terms in a linear combination) while, likewise, the end point predictability corresponds to the linear independence of the highest coefficient vectors in a polynomial matrix (the row-properness). The spans of an MSGM are uniquely defined by the code and, moreover, the sum of the lengths of the conventional spans of an MSGM is minimal among all encoders of the code; see [14, Sec. VI]. Again, this is in analogy to the notion of row-properness for polynomial delay-free and non-catastrophic matrices, where the row degrees are uniquely determined by the code and their sum (the McMillan degree) is minimal among all polynomial encoders. A unifying approach covering both instances of "shortest generators" for a given code can be found in [6]. Due to the predictable span property it is easy to determine the spaces $\mathcal{C}_i$ defined in (6) from an MSGM. Indeed, one always has $\mathcal{C}_i = \mathcal{P}_i \oplus \mathcal{F}_i$ where

$$\mathcal{P}_i = \{c \in \mathcal{C} \mid c_j = 0 \text{ for } j = i, \ldots, n-1\},$$
$$\mathcal{F}_i = \{c \in \mathcal{C} \mid c_j = 0 \text{ for } j = 0, \ldots, i-1\}$$

and these two spaces can be determined by inspection from an MSGM of $\mathcal{C}$. These past and future subcodes $\mathcal{P}_i$ and $\mathcal{F}_i$ (with respect to time i) show that the above construction is simply the past-future induced canonical realization of the behavior $\mathcal{C}$ in the sense of [22, Sec. 2.2]. All this should also have made clear that this trellis representation is a minimal realization in which the vertices satisfy the axiom of state.

B. Tail-Biting Trellises

As the trellises in (4) show, there exist tail-biting trellises with less states (or edges) than the minimal conventional trellis. This has led to the quest of constructing minimal tail-biting trellises for a given code. Several constructions of tail-biting trellises (not necessarily minimal) are known. We begin with product trellises (these trellises may be regarded as the product of certain elementary trellises, thus the name).

Theorem 4 ([13], [12], [11, Thm. 3.3, Prop. 3.5]) *Let $\mathcal{S} := \big[(a_l, b_l],\ l = 1,\dots,k\big]$ be a span list for G, that is, $(a_l, b_l]$ is a span (conventional or circular) for the l-th row of G for $l = 1,\dots,k$. Define the product trellis $T_{G,\mathcal{S}}$ as the trellis with state spaces $V_i = \operatorname{im} M_i$, where*

$$M_i = \begin{pmatrix} I_i^{(a_1,b_1]} & & \\ & \ddots & \\ & & I_i^{(a_k,b_k]} \end{pmatrix} \in \mathbb{F}^{k\times k},$$

and local behaviors (edge spaces) $E_i = \operatorname{im}(M_i, G_i, M_{i+1})$. Then the trellis $T_{G,\mathcal{S}}$ is one-to-one, linear and represents the code $\mathcal{C}$. Moreover, the SCP is given by $s = (s_0,\dots,s_{n-1})$, where $s_i = |\{l \mid i \in (a_l, b_l]\}|$. Put $\mathcal{A} := \{a_1,\dots,a_k\}$ and $\mathcal{B} = \{b_1,\dots,b_k\}$. Then the ECP $(e_0,\dots,e_{n-1})$ of $T_{G,\mathcal{S}}$ satisfies $e = s + I^{\mathcal{A}}$ if $a_1,\dots,a_k$ are distinct and $e = (s_1,\dots,s_{n-1},s_0) + I^{\mathcal{B}}$ if $b_1,\dots,b_k$ are distinct.

A main result of [12] is the construction of a particular characteristic pair from which all minimal trellises can be derived. We will define a characteristic pair more general, namely based on certain properties rather than on the outcome of a particular algorithm. This will facilitate later considerations. The existence of this invariant will then follow from the algorithm derived in [12].

Definition 5 A *characteristic pair* of $\mathcal{C}$ is defined to be a pair $(X, \mathcal{T})$, where

$$X \in \mathbb{F}^{n\times n},\ \mathcal{T} = \big[(a_l, b_l],\ l = 1,\dots,n\big]$$

have the following properties.

(i) $\operatorname{im} X = \mathcal{C}$, that is, the rows $x_1,\dots,x_n$ of X generate $\mathcal{C}$.

(ii) $(a_l,\ b_l]$ is a span of x_l for $l = 1,\dots,n$.

(iii) $a_1,\dots,a_n$ are distinct and $b_1,\dots,b_n$ are distinct.

(iv) For all $j \in \mathcal{I}$ there exist exactly $n-k$ row indices $l_1,\dots,l_{n-k}$ such that $j \in (a_{l_i},\ b_{l_i}]$ for $i = 1,\dots,n-k$.

We call X a *characteristic matrix* of $\mathcal{C}$ and $\mathcal{T}$ the *characteristic span list.*

By part (iv) there exist exactly k conventional spans in the span list $\mathcal{T}$ (notice that $(a,b]$ is conventional iff $0 \notin (a,b]$). Therefore and due to (iii), the corresponding rows of X form an MSGM of $\mathcal{C}$ in the sense of the previous subsection. Likewise, one can show that for each $j = 0,\dots,n-1$ a characteristic matrix contains a j-fold shifted MSGM for the j-fold shifted code. In this sense, a characteristic matrix forms a generalization of the conventional MSGM to the tail-biting version and its rows may be called a set of "shortest generators" of $\mathcal{C}$ (and all its cyclic shifts) in the sense of [6].

The existence of characteristic pairs for a given code, along with the uniqueness of the span list, has been proven in [12] and an algorithm has been given. All this leads to the following result which is true for our slightly more general notion as well.

Theorem 6 *Let $\mathcal{C} \subseteq \mathbb{F}^n$ be a k-dimensional code with support $\mathcal{I}$. Then $\mathcal{C}$ has a characteristic pair and the characteristic span list is, up to ordering, uniquely determined by $\mathcal{C}$.*

The importance of the characteristic pairs is that all minimal trellises of the code can be retrieved from them. This is one of the main results in [12]. Let us first introduce the following notion.

Definition 7 Let $(X, \mathcal{T})$ be a characteristic pair of $\mathcal{C}$. A *$\mathrm{KV}_{(X,\mathcal{T})}$-trellis* of $\mathcal{C}$ is a product trellis $T_{G,\mathcal{S}}$, where $G \in \mathbb{F}^{k\times n}$ is a submatrix of X of rank k (hence an encoder of $\mathcal{C}$) and $\mathcal{S}$ is the sublist of $\mathcal{T}$ consisting of the characteristic spans of the rows in G. Every trellis that is a $\mathrm{KV}_{(X,\mathcal{T})}$-trellis for some characteristic pair $(X, \mathcal{T})$ is called a KV-trellis of $\mathcal{C}$.

Notice that every KV-trellis is one-to-one and its SCP and ECP may be computed as in Theorem 4.

The next example shows that the set of $\mathrm{KV}_{(X,\mathcal{T})}$-trellises depends on the choice of the characteristic pair.

Example 8 Let $\mathcal{C} = \{(0000), (1100), (0111), (1011)\} \subseteq \mathbb{F}_2^4$. The two pairs $(X, \mathcal{T})$ and $(X', \mathcal{T})$, where

$$X = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix},\ X' = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}$$

and $\mathcal{T} = \big[(0,1],\ (1,3],\ (2,0],\ (3,2]\big]$, are both characteristic pairs of $\mathcal{C}$. The last two rows of X are linearly independent and lead to a $\mathrm{KV}_{(X,\mathcal{T})}$-trellis with SCP $(2,1,1,1)$. But since the last two rows of X' are linearly dependent, this SCP does not appear for any $\mathrm{KV}_{(X',\mathcal{T})}$-trellis of $\mathcal{C}$ as one can see directly from their common span list using Theorem 4. Examining the four codewords of $\mathcal{C}$ it is easy to see that the two matrices X and X' are the only two characteristic matrices for the code $\mathcal{C}$, up to ordering of the rows.

Let us now state the main result of [12]. Here and in the following we call two trellises isomorphic if they differ only by the labeling of the states (and relabeling is a isomorphism on the state space).

Theorem 9 ([12, Thm. 5.5]) *Let $\mathcal{C} \subseteq \mathbb{F}^n$ have support $\mathcal{I}$ and let T be a minimal trellis of $\mathcal{C}$. Then there exists a characteristic pair $(X, \mathcal{T})$ of $\mathcal{C}$ such that T is (isomorphic to) a $\mathrm{KV}_{(X,\mathcal{T})}$-trellis.*

One should bear in mind that, firstly, not all KV-trellises are minimal (examples are given in [12, Ex. 6.1] or can be constructed using Example 8) and, secondly, not all minimal trellises of $\mathcal{C}$ arise from the same characteristic matrix (see [11, Ex. 3.15]). Finally, as shown in [12, Thm. 5.6], the same result as in Theorem 9 applies to many other notions of minimality for tail-biting trellises.

Let us now turn to a different way of producing tail-biting trellises for a given code. These trellises were introduced in [17] and generalize the conventional BCJR-construction of minimal trellises as introduced by [1], see also [14].

The following can easily be shown, see [17], [11].

Theorem 10 ([17, Sec. III]) *Let $N_0 \in \mathbb{F}^{k\times(n-k)}$ be any matrix and for $i = 1,\ldots,n-1$ define the matrices*

$$N_i = N_{i-1} + G_{i-1}H_{i-1}^{\mathsf{T}}$$

Then the trellis $T_{(G,H,N_0)}$ with state spaces $V_i := \operatorname{im} N_i \subseteq \mathbb{F}^{n-k}$, and local behaviors $E_i = \operatorname{im}(N_i, G_i, N_{i+1})$ is linear and represents the code $\mathcal{C}$, that is, $\mathcal{C}(T) = \mathcal{C}$. We call N_0 the displacement matrix for the trellis $T_{(G,H,N_0)}$.

The displacement matrix N_0 may be interpreted as capturing the (circular) past at time zero of each cycle in $T_{(G,H,N_0)}$. If $N_0 = 0$, each cycle starts with zero history and the trellis $T_{(G,H,N_0)}$ is conventional. In this case the trellis can be regarded as the observer-canonical realization because the state v_i at time i through which the trajectory $c \in \mathcal{C}$ passes is given by $v_i = \sum_{j=0}^{i-1} c_j H_j^{\mathsf{T}} = -\sum_{j=i}^{n-1} c_j H_j^{\mathsf{T}}$, which shows that the state v_i is, by design, observable from the strict past as well as the future of the trajectory.

One of the nice properties of the trellis construction in Theorem 10 is that it respects duality.

Proposition 11 ([17, Lemma 12]) *If $T_{(G,H,N_0)}$ is as in Thm. 10, then the trellis $T_{(H,G,N_0^{\mathsf{T}})}$ represents the dual code $\mathcal{C}^\perp$. Moreover, $T_{(G,H,N_0)}$ and $T_{(H,G,N_0^{\mathsf{T}})}$ have the same SCP.*

A particular instance of the trellis $T_{(G,H,N_0)}$ is obtained by choosing the displacement matrix based on given spans of the generator matrix G, see [17, Def. 9]. This will relate this construction to product trellises.

Definition 12 Let $\mathcal{S} := \big[(a_l, b_l], l = 1,\ldots,k\big]$ be a span list of G. Then the trellis $T_{(G,H,\mathcal{S})}$ defined as $T_{(G,H,N_0)}$, where the l-th row of N_0 is defined as

$$\sum_{j=a_l}^{n-1} G_{lj}H_j^{\mathsf{T}},$$

is called a *(tail-biting) BCJR-trellis* of $\mathcal{C}$.

Notice that if all spans are conventional, then the identity $GH^{\mathsf{T}} = 0$ implies that N_0 is the zero matrix and the resulting trellis is the conventional BCJR-trellis, see [14, Sec. IV] and [1]. It should be added that we define a BCJR-trellis different than Nori/Shankar in [17].

Now we can state our main results about product trellises and BCJR-trellises. It extends [17, Thm. 3.1].

Theorem 13 ([11, Sec. 4]) *Let $\mathcal{S} := \big[(a_l, b_l],\ l = 1,\ldots,k\big]$ and $T_{G,\mathcal{S}}$ and $T_{(G,H,\mathcal{S})}$ be as in Thm. 4 and Def. 12. Then*

(1) $T_{(G,H,\mathcal{S})}$ is non-mergeable.

(2) The product trellis can be merged to the BCJR-trellis by taking suitable quotients. Precisely, let $\hat{V}_i := \operatorname{im} M_i$ and V_i be the state spaces of $T_{G,\mathcal{S}}$ and $T_{(G,H,\mathcal{S})}$, respectively. Then there exist subspaces $W_i \subseteq \hat{V}_i$ such that the quotient trellis T' with state spaces $V_i' = \hat{V}_i/W_i$ and local behaviors $E_i' = \{(\alpha M_i + W_i,\ \alpha G_i,\ \alpha M_{i+1} + W_{i+1}) \mid \alpha \in \mathbb{F}^k\}$ is isomorphic to the BCJR-trellis $T_{(G,H,\mathcal{S})}$.

If $T_{(G,H,\mathcal{S})}$ and $T_{G,\mathcal{S}}$ have the same SCP, that is, $\dim V_i = \dim \hat{V}_i$ for all $i \in \mathcal{I}$, then one has the following.

(3) $T_{(G,H,\mathcal{S})}$ and $T_{G,\mathcal{S}}$ are isomorphic.

(4) $a_1,\ldots,a_k$ are distinct and so are $b_1,\ldots,b_k$. Thus, the formulas for the ECP in Theorem 4 apply.

The above along with the specific properties of characteristic pairs lead to the following results for KV-trellises.

Theorem 14 ([11, Thm. 4.12]) *Each KV-trellis $T_{G,\mathcal{S}}$ is isomorphic to its corresponding BCJR-trellis $T_{(G,H,\mathcal{S})}$. As a consequence, KV-trellises are non-mergeable.*

C. On a Duality Conjecture by Koetter/Vardy

Let us finally address the question whether a given trellis for $\mathcal{C}$ may be dualized such that it turns into a trellis for the dual code. While Proposition 11 answers this question in the affirmative for trellises defined as in Theorem 10, and in particular for BCJR-trellises, the resulting dual trellises do not always have nice properties. Another dualization technique for trellises amounts to dualizing the local behaviors with respect to a particular bilinear form and goes back to Mittelholzer [15] and Forney [4, Sec. VII.D and Thm. 8.4]. Finally for KV-trellises, Koetter/Vardy suggested yet another way of dualization. In [12, Sec. 5] they have shown that if $\mathcal{C} \subseteq \mathbb{F}^n$ and $\mathcal{C}^\perp$ both are codes with support $\mathcal{I}$, and the characteristic span list of $\mathcal{C}$ is given by $\mathcal{T} = \big[(a_l, b_l],\ l = 1,\ldots,n\big]$, then the characteristic span list of $\mathcal{C}^\perp$ is given by $\big[(b_l, a_l],\ l = 1,\ldots,n\big]$. Moreover they proved the following: Fix a characteristic matrix of $\mathcal{C}$, select k linearly independent rows with spans, say, $(a_{l_i}, b_{l_i}],\ i = 1,\ldots,k$, and construct the resulting KV-trellis T. Pick the $n-k$ rows of a characteristic matrix of $\mathcal{C}^\perp$ that do *not* have spans $(b_{l_i}, a_{l_i}],\ i = 1,\ldots,k$, and construct the resulting product trellis $\hat{T}$. Then the trellises T and $\hat{T}$ share the same SCP. However, while the trellis T for $\mathcal{C}$ is, by construction, a KV-trellis in the sense of Definition 7, the trellis $\hat{T}$ might not even represent $\mathcal{C}^\perp$ because it is not a priori clear whether the underlying $n-k$ rows are linearly independent (and thus span $\mathcal{C}^\perp$). Koetter/Vardy conjecture that those rows are indeed linearly independent and thus the trellis $\hat{T}$ is a KV-trellis for $\mathcal{C}^\perp$. As it turns out, the conjecture is not true in this generality, but rather depends on the choice of the characteristic matrix for $\mathcal{C}^\perp$. An example can be created using the code in Example 8 and its dual, see [11, Ex. 5.2].

Notice that if the dual row selection is linearly independent then this process leads to a dualization that turns (minimal) KV-trellises for $\mathcal{C}$ into (minimal) KV-trellises for $\mathcal{C}^\perp$. The statement about minimal trellises follows from the very definitions of minimality and the SCP.

We close the paper with presenting our main contribution to Koetter/Vardy's conjecture. It states that the conjecture holds true for minimal KV-trellises and for *some* characteristic matrix for $\mathcal{C}^\perp$. Moreover, the proof shows that in this case this dualization technique coincides with the other two techniques mentioned above.

Theorem 15 ([11, Thm. 5.3]) *Let $\mathcal{C}, \mathcal{C}^\perp \subseteq \mathbb{F}^n$ both have support $\mathcal{I}$ and let $(X, \mathcal{T})$, $\mathcal{T} = [(a_l, b_l], l = 1, \ldots, n]$, be a characteristic pair for $\mathcal{C}$. Suppose the rows of X with indices $l_1, \ldots, l_k$ are linearly independent and give rise to a minimal KV-trellis of $\mathcal{C}$. Then there exists a characteristic matrix $\hat{X}$ of $\mathcal{C}^\perp$ such that the $n-k$ rows of $\hat{X}$ that do not have spans $(b_{l_i}, a_{l_i}], i = 1 \ldots, k$, are linearly independent and give rise to a minimal KV-trellis of $\mathcal{C}^\perp$.*

We strongly believe that this theorem holds true for all KV-trellises.

REFERENCES

[1] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv. Optimal decoding of linear codes for minimizing symbol error rate. *IEEE Trans. Inform. Theory*, IT-20:284–287, 1974.

[2] A. R. Calderbank, G. D. Forney, Jr., and A. Vardy. Minimal tail-biting trellises: The Golay code and more. *IEEE Trans. Inform. Theory*, IT-45:1435–1455, 1999.

[3] G. D. Forney, Jr. Coset codes II: Binary lattices and related codes. *IEEE Trans. Inform. Theory*, IT-34:1152–1187, 1988.

[4] G. D. Forney, Jr. Codes on graphs: Normal realizations. *IEEE Trans. Inform. Theory*, IT-47:520–548, 2001.

[5] G. D. Forney, Jr. Codes on graphs: MacWilliams identites. Preprint 2009. ArXiv: 0911.5508v2.

[6] G. D. Forney, Jr. Minimal realizations of linear systems: The "shortest Basis" approach. Preprint 2009. ArXiv: 0910.4336.

[7] H. Gluesing-Luerssen. On isometries for convolutional codes. *Advances in Mathematics of Communications*, 3:179–203, 2009.

[8] H. Gluesing-Luerssen and G. Schneider. State space realizations and monomial equivalence for convolutional codes. *Linear Algebra and its Applications*, 425:518–533, 2007.

[9] H. Gluesing-Luerssen and G. Schneider. On the MacWilliams identity for convolutional codes. *IEEE Trans. Inform. Theory*, IT-54:1536–1550, 2008.

[10] H. Gluesing-Luerssen and G. Schneider. A MacWilliams identity for convolutional codes: The general case. *IEEE Trans. Inform. Theory*, IT-55:2920–2930, 2009.

[11] H. Gluesing-Luerssen and E. Weaver. Linear tail-biting trellises: Characteristic generators and the BCJR-construction. Preprint 2010. Available at http://front.math.ucdavis.edu/1003.4539.

[12] R. Koetter and A. Vardy. The structure of tail-biting trellises: Minimality and basic principles. *IEEE Trans. Inform. Theory*, IT-49:2081–2105, 2003.

[13] F. R. Kschischang and V. Sorokine. On the trellis structure of block codes. *IEEE Trans. Inform. Theory*, IT-41:1924–1937, 1995.

[14] R. J. McEliece. On the BCJR Trellis for linear block codes. *IEEE Trans. Inform. Theory*, IT-42:1072–1092, 1996.

[15] T. Mittelholzer. Convolutional codes over groups: A pragmatic approach. In *Proc. of the 33rd Allerton Conference on Communications, Control, and Computing*, pages 380–381, 1995.

[16] D. J. Muder. Minimal trellises for block codes. *IEEE Trans. Inform. Theory*, IT-34:1049–1053, 1988.

[17] A. V. Nori and P. Shankar. Unifying views of tail-biting trellis constructions for linear block codes. *IEEE Trans. Inform. Theory*, IT-52:4431–4443, 2006.

[18] J. W. Polderman and J. C. Willems. *Introduction to Mathematical Systems Theory; A behavioral approach*. Springer, New York, 1998.

[19] J. B. Shearer and R. J. McEliece. There is no MacWilliams identity for convolutional codes. *IEEE Trans. Inform. Theory*, IT-23:775–776, 1977.

[20] A. Vardy. Trellis structure of codes. In V. S. Pless and W. C. Huffman, editors, *Handbook of Coding Theory, Vol. 2*, pages 1989–2117. Elsevier, Amsterdam, 1998.

[21] A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Inform. Theory*, IT-13:260–269, 1967.

[22] J. C. Willems. Models for dynamics. *Dynamics Reported*, 2:171–269, 1989.