On the weight distribution of convolutional codes

Gluesing-Luerssen, H

Published in:
Linear Algebra and Its Applications

DOI:
10.1016/j.laa.2005.06.023

Publication date:
2005

Available online at www.sciencedirect.com

SCIENCE DIRECT®

LINEAR ALGEBRA AND ITS APPLICATIONS

ELSEVIER Linear Algebra and its Applications 408 (2005) 298–326

www.elsevier.com/locate/laa

On the weight distribution of convolutional codes

Heide Gluesing-Luerssen

University of Groningen, Department of Mathematics, P.O. Box 800,
9700 AV Groningen, The Netherlands

Received 15 December 2004; accepted 19 June 2005

Submitted by J. Rosenthal

Abstract

Detailed information about the weight distribution of a convolutional code is given by the adjacency matrix of the state diagram associated with a minimal realization of the code. We will show that this matrix is an invariant of the code. Moreover, it will be proven that codes with the same adjacency matrix have the same dimension and the same Forney indices and finally that for one-dimensional binary convolutional codes the adjacency matrix determines the code uniquely up to monomial equivalence.

AMS classification: 94B05; 94B10; 93B15

Keywords: Convolutional coding theory; Controller canonical form; Weight distribution; Monomial equivalence; MacWilliams duality

1. Introduction

The weight distribution of a code forms an important parameter containing a lot of information about the code for practical as well as theoretical purposes. A lot of research in block code theory has been devoted to the investigation of the weight distribution and to weight preserving maps. The most famous results in this area are certainly the Duality Theorem of MacWilliams which presents a transformation between the weight distributions of a given code and its dual [12, Theorem 3.5.3]

E-mail address: gluesing@math.rug.nl

0024-3795/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.laa.2005.06.023

as well as MacWilliams' Equivalence Theorem stating that two isometric codes are monomially equivalent [8, Theorem 7.9.4].

In this paper we will address the issue of MacWilliams duality for convolutional codes. It is worth mentioning that even though convolutional codes can be regarded as block codes over the ring $\mathbb{F}[z]$ endowed with a certain weight function, it is not possible to apply any results about block codes over rings since they always need that the ring be finite. In [23] it has been shown that there cannot exist a duality theorem for the weight distribution of convolutional codes and their duals, if the weight distribution is defined as the enumerator of the atomic codewords. The authors present two binary one-dimensional codes having the same weight distribution while their duals have different ones. We will present this example at the end of Section 5. It indicates that the thus defined weight distribution contains too little information about the code when it comes to duality. In this paper we will concentrate on a different type of weight distribution containing considerably more information. This is the adjacency matrix of the state diagram associated with a minimal realization of a minimal encoder. For block codes this matrix reduces to the usual weight distribution. The adjacency matrix depends on the choice of encoder and realization. As we will show in Section 4 this dependence can easily be described by an equivalence relation, and by factoring it out we obtain an invariant of the code, the generalized adjacency matrix. In Section 5 we will investigate as to how much information about the code this matrix contains. We will present two results. Firstly, codes with the same generalized adjacency matrix have the same dimension and the same Forney indices. Secondly, one-dimensional binary codes with the same generalized adjacency matrix are monomially equivalent. One should bear in mind that this result cannot be expected for general codes, since it is not even true for higher-dimensional binary block codes (notice that the assumption of having identical weight distribution is much weaker than the assumption of being isometric in MacWilliams' Equivalence Theorem). However, as a consequence we obtain that if two binary one-dimensional convolutional codes share the same generalized adjacency matrix then so do their dual codes. This shows in particular that the example in [23] mentioned above does not apply if we consider the generalized adjacency matrix rather than the classical weight distribution. This in turn tempts us to conjecture that there might exist a MacWilliams Duality for convolutional codes based on the generalized adjacency matrix. This is indeed true for codes with overall constraint length one. We will present a transformation for the generalized adjacency matrices of such a code and its dual at the end of Section 5. It has been derived in [1] in a totally different form and can be written in closed form as given in (5.3). Proving or disproving the existence of a MacWilliams transformation for general codes, however, appears to be quite a difficult problem and has to remain open for future research.

In the next two sections we will introduce the material as necessary for deriving our results. We will discuss the controller canonical form as well as other minimal realizations of an encoder matrix and will introduce the associated state diagram along with its adjacency matrix. Since the results as we will need them later on are somewhat spread over the literature and proofs are not always easily accessible, we think it is worthwhile presenting the material of the next two sections as self-contained as possible, even though most of the results are not new. Furthermore, as opposed to the existing literature we will give a purely matrix theoretic approach. Following McEliece [16] we will introduce the notions of atomic and molecular codewords and show how the corresponding weight distribution can be derived, theoretically, from the adjacency matrix of the state diagram. Sections 4 and 5 contain the new results as described in the previous paragraph. We will close the paper with some open problems in Section 6.

We end the introduction with presenting the basic notions of convolutional coding theory. Throughout the paper the symbol $\mathbb{F}$ stands for any finite field while $\mathbb{F}_q$ always denotes a field with q elements. We will define convolutional codes purely in the polynomial context. Thus, all messages and codewords are finite sequences of data blocks. This differs in parts from the literature where mostly also infinite sequences (Laurent series) are considered. For our purposes this does not make a difference since generator matrices of convolutional codes are polynomial in any setting, see also [3,15]. In [20] the issue of finite versus infinite sequences has been discussed in detail. For the coding theoretic relevance of the notions defined below we refer to [9, Chapter 2].

Definition 1.1. A k-dimensional convolutional code is a submodule $\mathscr{C}$ of $\mathbb{F}[z]^n$ of the form $\mathscr{C} = \operatorname{im} G := \{uG \mid u \in \mathbb{F}[z]^k\}$ where G is a *basic* matrix in $\mathbb{F}[z]^{k\times n}$, i.e. there exists some matrix $\widetilde{G} \in \mathbb{F}[z]^{n\times k}$ such that $G\widetilde{G} = I_k$. We call G a *generator matrix* or *encoder* and the number $\gamma := \max\{\deg g \mid g \text{ is a } k\text{-minor of } G\}$ is said to be the *overall constraint length* of the code $\mathscr{C}$.

For various characterizations of basicness we refer to [15, Theorem A.1]. It is well-known [3, Theorem 5] or [5, p. 495] that each code has a minimal generator matrix in the sense defined next. For a polynomial vector $v \in \mathbb{F}[z]^n$ we define $\deg v$ to be the maximum degree of its entries and, as usual, we put $\deg 0 = -\infty$.

Definition 1.2. Let $G \in \mathbb{F}[z]^{k\times n}$ be a basic matrix with overall constraint length γ and let $\nu_1, \ldots, \nu_k$ be the row degrees of G. We say that G is *minimal* if $\gamma = \sum_{i=1}^{k} \nu_i$. In this case, the row degrees of G are uniquely determined by the code $\mathscr{C} := \operatorname{im} G$. They are called the *Forney indices* of $\mathscr{C}$. The maximal Forney index is called the *memory* of the code.

The notion minimality stems from the fact that for an arbitrary matrix G one has $\gamma \leqslant \sum_{i=1}^{k} \nu_i$. For characterizations of minimality see, e.g., [5, Main Theorem], [15, Theorem A.2], or [9, Theorem 2.22]. From the above it follows that a convolutional code has a constant generator matrix if and only if the overall constraint length is zero. In that case the code can be regarded as a block code.

The definition of weight and distance in convolutional coding theory is straightforward. For a polynomial vector $v = \sum_{j=0}^{N} v_j z^j \in \mathbb{F}[z]^n$ we define the *weight* of v

to be $\operatorname{wt}(v)=\sum_{j=0}^{N}\operatorname{wt}(v_j)$ where, as usual, $\operatorname{wt}(v_j)$ denotes the Hamming weight of $v_j\in\mathbb{F}^n$. The *(free) distance* of a code $\mathscr{C}\subseteq\mathbb{F}[z]^n$ is given as $\operatorname{dist}(\mathscr{C}):=\min\{\operatorname{wt}(v)\,|\,v\in\mathscr{C},\ v\neq 0\}$.

2. Controller canonical form and state diagram

In this section we introduce the matrix representation of the controller canonical form for a given encoder matrix G. The interpretation of this form has been discussed in detail in [4, Section II, 3, 9, Section 2.1]. The relation to the encoding process will be made clear below. The results of this section are in essence well-known from the references above and other coding literature. However, since we were not able to find detailed references including proofs for all results we think it is worthwhile to summarize them with strict matrix theoretical proofs. We will make heavy use of these results in later sections.

Proposition 2.1. *Let $G=(g_{ij})\in\mathbb{F}[z]^{k\times n}$ be a generator matrix with row degrees $\gamma_1,\ldots,\gamma_k$ and let $g_{ij}=\sum_{\nu=0}^{\gamma_i}g_{ij}^{(\nu)}z^\nu$. Put $\gamma=\sum_{i=1}^{k}\gamma_i$ and assume $\gamma>0$. For $i=1,\ldots,k$ define*

$$A_i=\begin{pmatrix}0&1&&\\&&\ddots&\\&&&1\\&&&0\end{pmatrix}\in\mathbb{F}^{\gamma_i\times\gamma_i},\quad B_i=\begin{pmatrix}1&0&\cdots&0\end{pmatrix}\in\mathbb{F}^{\gamma_i},$$

$$C_i=\begin{pmatrix}g_{i1}^{(1)}&\cdots&g_{in}^{(1)}\\\vdots&&\vdots\\g_{i1}^{(\gamma_i)}&\cdots&g_{in}^{(\gamma_i)}\end{pmatrix}\in\mathbb{F}^{\gamma_i\times n}$$

and put

$$A=\begin{pmatrix}A_1&&\\&\ddots&\\&&A_k\end{pmatrix}\in\mathbb{F}^{\gamma\times\gamma},\quad B=\begin{pmatrix}B_1&&\\&\ddots&\\&&B_k\end{pmatrix}\in\mathbb{F}^{k\times\gamma},$$

$$C=\begin{pmatrix}C_1\\\vdots\\C_k\end{pmatrix}\in\mathbb{F}^{\gamma\times n}$$

as well as $D=(g_{ij}^{(0)})\in\mathbb{F}^{k\times n}$. In case $\gamma_i=0$ for some i, the corresponding block is missing and in B a zero row occurs. Then

$$G=B(z^{-1}I-A)^{-1}C+D=zB(I-zA)^{-1}C+D. \tag{2.1}$$

Moreover,

$$\operatorname{rank}\begin{pmatrix} B \\ BA \\ \vdots \\ BA^{\gamma-1}\end{pmatrix} = \gamma \quad \textit{and}$$

$$[\operatorname{rank}(C, AC, A^2C, \ldots, A^{\gamma-1}C) = \gamma \Longleftrightarrow G \textit{ minimal}]. \tag{2.2}$$

We call (A, B, C, D) *the controller canonical form of the code* $\mathscr{C} = \operatorname{im} G$.

Proof. It is easy to see that

$$B(z^{-1}I - A)^{-1} = \begin{pmatrix} z & z^2 & \cdots & z^{\gamma_1} & & & & & & & & \\ & & & & z & z^2 & \cdots & z^{\gamma_2} & & & & \\ & & & & & & & & \ddots & & & \\ & & & & & & & & & z & z^2 & \cdots & z^{\gamma_k} \end{pmatrix}, \tag{2.3}$$

where a zero row occurs if $\gamma_i = 0$. From this and the definition of all matrices involved we obtain $B(z^{-1}I - A)^{-1}C = (\sum_{\nu=1}^{\gamma_i} g_{ij}^{(\nu)} z^{\nu}) = G - D$. This shows the first identity of (2.1). If $\gamma_i = 0$, then the assertion is correct, too. The second equality of (2.1) follows easily from the first one. The first of the rank statements (2.2) is directly seen from the matrices. As for the second one one may argue as follows. That rank condition is equivalent to $\operatorname{rank}(\lambda I - A, C) = \gamma$ for all $\lambda \in \overline{\mathbb{F}}$, an algebraic closure of $\mathbb{F}$, see [24, Lemma 3.3.7]. Due to the nilpotent Jordan form of A the latter is true if and only if it is true for $\lambda = 0$ and this in turn is easily seen to be equivalent to $\operatorname{rank}(g_{ij}^{(\nu_i)}) = k$. But this is exactly the minimality of G, see [5, p. 495]. □

Identities of the form (2.1) have been well studied in system theory. In general there exist many matrix quadruples (A, B, C, D) satisfying (2.1). By slight abuse of system theoretic notions, we call any matrix quadruple $(A', B', C', D') \in \mathbb{F}^{\gamma'\times\gamma' + k\times\gamma' + \gamma'\times n + k\times n}$ a realization of G if $G = B'(z^{-1}I - A')^{-1}C' + D'$. The conditions (2.2) are known as controllability and observability criteria. Moreover, the following is known.

Proposition 2.2 [24, Section 5.5, Theorem 20]. *Let* $G \in \mathbb{F}[z]^{k\times n}$ *be a minimal generator matrix with overall constraint length* $\gamma > 0$ *and let* $(A', B', C', D') \in \mathbb{F}^{\gamma'\times\gamma' + k\times\gamma' + \gamma'\times n + k\times n}$ *be a realization of* G. *Then* $\gamma' \geqslant \gamma$. *If* $\gamma' = \gamma$, *we call* (A', B', C', D') *a minimal realization. In this case there exists a matrix* $T \in \mathrm{Gl}_\gamma(\mathbb{F})$ *such that* $(A', B', C', D') = (TAT^{-1}, BT^{-1}, TC, D)$, *where* (A, B, C, D) *is the controller canonical form. Conversely, for each* $T \in \mathrm{Gl}_\gamma(\mathbb{F})$ *the quadruple* $(TAT^{-1}, BT^{-1}, TC, D)$ *is a realization of* G.

Since $G(z^{-1}) = B(zI - A)^{-1}C + D$ for any realization the above implies that γ is the McMillan degree of the proper rational function $G(z^{-1})$ in the system theoretic sense, see [24, p. 228].

It is well-known that the encoding process of the convolutional code generated by G can be described by a linear shift register. The matrix version of this is exactly the controller canonical form along with the corresponding dynamical system as given in part (2) of the next theorem. We will give the interpretation right after the proof.

Theorem 2.3. *Let $G \in \mathbb{F}[z]^{k\times n}$ be a minimal generator matrix and (A, B, C, D) be any minimal realization of G. Let $u \in \mathbb{F}[z]^k$ and $v \in \mathbb{F}[z]^n$ and define $x = uB(z^{-1}I - A)^{-1}$. Then*

(1) *$x \in z\mathbb{F}[z]^\gamma$ (that is, x is polynomial and has zero constant term). Furthermore, writing $u = (u^{(1)}, \ldots, u^{(k)})$ one has $\deg x = \max\{\gamma_i + \deg u^{(i)} \mid i = 1, \ldots, k, \gamma_i \neq 0\}$ and $\deg(uG) = \max\{\gamma_i + \deg u^{(i)} \mid i = 1, \ldots, k\}$.*

(2) *Write $u = \sum_{t\geqslant 0} u_t z^t$, $v = \sum_{t\geqslant 0} v_t z^t$, and $x = \sum_{t\geqslant 0} x_t z^t$ where $x_0 = 0$. Then*

$$v = uG \iff \left\{\begin{matrix} z^{-1}x = xA + uB \\ v = xC + uD \end{matrix}\right\}$$

$$\iff \left\{\begin{matrix} x_{t+1} = x_tA + u_tB \\ v_t = x_tC + u_tD \end{matrix} \quad \text{for all } t \geqslant 0\right\}.$$

We call $x_t \in \mathbb{F}^\gamma$ as in (2) the state of the realization (A, B, C, D) at time t given that the input is u. The space $\mathbb{F}^\gamma$ is called the state space of the encoder G.

The state-space realization $x_{t+1} = x_tA + u_tB$, $v_t = x_tC + u_tD$ with (A, B, C, D) being the controller canonical form has been introduced in [14] and has also been discussed in [4,15]. It is different, however, from the state-space system used in [20,22,21,19]. In those papers the codeword is made up by the combined input and output, while in our case the codeword coincides with the output of the system.

Proof. (1) Suppose first that (A, B, C, D) is the controller canonical form. Then from (2.3) we have $x = (u^{(1)}z, \ldots, u^{(1)}z^{\gamma_1}, \ldots, u^{(k)}z, \ldots, u^{(k)}z^{\gamma_k})$ where for $\gamma_i = 0$ the corresponding block is missing. From this the assertions about x follow. With the data as in Proposition 2.2 one obtains $uB'(zI - A')^{-1} = uB(zI - A)^{-1}T^{-1}$ and therefore the statements are also true for any minimal realization. The second statement in (1) is one of the well-known characterizations of minimal matrices, see [5, p. 495].

(2) We have $v = uG \iff v = u(B(z^{-1}I - A)^{-1}C + D)$ and using the definition for x the first equivalence follows. The other one follows by equating like powers of z. □

Notice that if all row indices γ_i are non-zero, then the matrix B of the controller canonical form has full row rank and $\deg x > \deg u$. If $\gamma_i = 0$ for some i, then

$\ker B \neq 0$ and the inequality $\deg x \leqslant \deg u$ might occur. Via Proposition 2.2 the same is true for any minimal realization. In any case one has $\deg x \leqslant \deg(uG)$.

Let us now consider the controller canonical form (A, B, C, D) of G. Obviously the dynamical equations $x_{t+1} = x_tA + u_tB$, $v_t = x_tC + u_tD$ describe the input-state–output behavior of the canonical shift register realization of the encoder G. The inputs at time t are given by the sequence u_t, the state vectors $x_t \in \mathbb{F}^\gamma$ represent the contents of the memory elements of the register at time t and v_t is the output at that time. Part (1) above tells us in particular $x_0 = 0$, which is the usual assumption that the shift register is empty at the beginning of the encoding process.

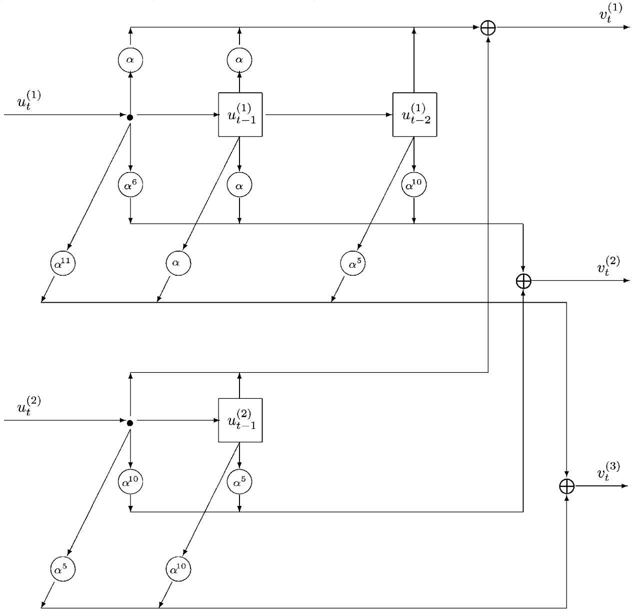
Example 2.4. Let

$$G = \begin{bmatrix} \alpha + \alpha z + z^2 & \alpha^6 + \alpha z + \alpha^{10} z^2 & \alpha^{11} + \alpha z + \alpha^5 z^2 \\ 1 + z & \alpha^{10} + \alpha^5 z & \alpha^5 + \alpha^{10} z \end{bmatrix} \in \mathbb{F}_{16}[z]^{2\times 3}$$

where $\alpha^4 + \alpha + 1 = 0$. Then the multiplication

$$\begin{aligned} uG &= \sum_{t\geqslant 0} u_t z^t G = \sum_{t\geqslant 0} \underbrace{(u_t^{(1)}, u_t^{(2)})}_{u_t} z^t \cdot G \\ &= \sum_{t\geqslant 0} \underbrace{(v_t^{(1)}, v_t^{(2)}, v_t^{(3)})}_{v_t} z^t = \sum_{t\geqslant 0} v_t z^t = v \end{aligned}$$

is realized by the following linear shift register, shown at time t



The controller canonical form is given by

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$C = \begin{pmatrix} \alpha & \alpha & \alpha \\ 1 & \alpha^{10} & \alpha^5 \\ 1 & \alpha^5 & \alpha^{10} \end{pmatrix}, \quad D = \begin{pmatrix} \alpha & \alpha^6 & \alpha^{11} \\ 1 & \alpha^{10} & \alpha^5 \end{pmatrix}.$$

The dynamical equations $x_{t+1} = x_t A + u_t B$, $v_t = x_t C + u_t D$ with the state at time t being $x_t := (u_{t-1}^{(1)}, u_{t-2}^{(1)}, u_{t-1}^{(2)})$ describe exactly the input-state–output behavior of the shift register.

General Assumption 2.5. From now on we will always assume $\mathbb{F} = \mathbb{F}_q$ and that the generator matrix $G \in \mathbb{F}[z]^{k\times n}$ of the code $\mathscr{C} \subseteq \mathbb{F}[z]^n$ is minimal with Forney indices $\gamma_1, \ldots, \gamma_k$ and overall constraint length $\gamma > 0$. Moreover, we fix a minimal realization (A, B, C, D) of G.

Lemma 2.6. *Let $u \in \mathbb{F}[z]^k$ and $v := uG \in \mathscr{C}$. Assume $v_0 \neq 0$ and let $\deg v = N > 0$. Put $x = uB(z^{-1}I - A)^{-1} \in \mathbb{F}[z]^\gamma$. Choose $L \in \{1, \ldots, N\}$. Then the following are equivalent.*

(i) $x_L = 0$,
(ii) $v = \tilde{v} + \hat{v}$ *where* $\tilde{v}, \hat{v} \in \mathscr{C}\backslash\{0\}$ *and* $\deg \tilde{v} < L$, $\hat{v} \in z^L\mathbb{F}[z]^n$.

In the case where (A, B, C, D) is the controller canonical form the lemma simply reflects that if the shift register is back to the zero state (at time $t = L$), then one may regard the information before and after that time instance as two separate information messages and the associated codewords as two separate codewords.

Proof. Recall that $\deg x \leqslant \deg v = N$.

"(i) ⇒ (ii)": Put $\tilde{x} = \sum_{t=0}^{L-1} x_t z^t$ and $\hat{x} = \sum_{t=L+1}^{N} x_t z^t$. If $L = N$, put $\hat{x} = 0$. Then $x = \tilde{x} + \hat{x}$ and $uB = \tilde{x}(z^{-1}I - A) + \hat{x}(z^{-1}I - A)$. Writing $u = \tilde{u} + \hat{u}$ where $\deg \tilde{u} < L$ and $\hat{u} \in z^L\mathbb{F}[z]^k$, we obtain $\deg(\tilde{u}B) < L$ and $\hat{u}B \in z^L\mathbb{F}[z]^\gamma$. Therefore, $\tilde{u}B = \tilde{x}(z^{-1}I - A)$ and $\hat{u}B = \hat{x}(z^{-1}I - A)$. Now put $\tilde{v} := \tilde{u}G = \tilde{u}(B(z^{-1}I - A)^{-1}C + D) = \tilde{x}C + \tilde{u}D$ and $\hat{v} := \hat{u}G = \hat{x}C + \hat{u}D$. Then it is easy to see that (ii) is satisfied.

"(ii) ⇒ (i)": Let $\tilde{v} = \tilde{u}G$ and $\hat{v} = \hat{u}G$, hence $v = (\tilde{u} + \hat{u})G$. Then basicness of G implies $\hat{u} \in z^L\mathbb{F}[z]^k$. Moreover, since G is minimal, we have $\deg \tilde{v} = \max\{\deg \tilde{u}^{(i)} + \gamma_i \mid i = 1, \ldots, k\}$, where $\tilde{u} = (\tilde{u}^{(1)}, \ldots, \tilde{u}^{(k)})$, see Theorem 2.3(1). Thus the assumption $\deg \tilde{v} < L$ implies $\deg \tilde{u}^{(i)} < L - \gamma_i$ for all $i = 1, \ldots, k$. Now, $x = \tilde{u}B(z^{-1}I - A)^{-1} + \hat{u}B(z^{-1}I - A)^{-1}$ and from (2.3) we obtain that $\deg(\tilde{u}B(z^{-1}I - A)^{-1}) < L$ and $\hat{u}B(z^{-1}I - A)^{-1} \in z^{L+1}\mathbb{F}[z]^\gamma$. Thus $x_L = 0$. □

The above gives rise to the distinction of codewords into those which are the sum of two non-overlapping codewords and those which are not. For counting weights it will be advantageous to make an even finer distinction.

Definition 2.7. Let $v \in \mathscr{C}$ such that $v_0 \neq 0$. Let $L \in \mathbb{N}$. The codeword v is called *concatenated at time* $t = L$ if

$$v = \tilde{v} + \hat{v} \text{ where } \tilde{v}, \hat{v} \in \mathscr{C}\backslash\{0\}, \quad \deg \tilde{v} = L - 1, \; \hat{v} \in z^L \mathbb{F}[z]^n.$$

If additionally, $v_L \neq 0$, we call v *tightly concatenated at time* $t = L$. We call a codeword *concatenated* if it is concatenated at some time instance $t = L$. We call it *tightly concatenated* if each of its concatenations is tight. If v is not concatenated, then v is called *atomic*. If v is tightly concatenated or atomic, then v is also called *molecular*.

Parts of this definition can also be found in [16]. Notice that we consider only codewords that start at time $t = 0$, i.e., $v_0 \neq 0$. This is certainly no restriction when it comes to computing the weight. It is obvious that each such codeword is the concatenation of atomic codewords. Moreover, from Lemma 2.6 we know that if v is concatenated at time $t = L$ then the state x_L at time $t = L$ is zero. In this case, the dynamical equations in Theorem 2.3(2) show that v is tightly concatenated at time $t = L$ if and only if $u_L \neq 0$ (since the matrix D has full row rank). Thus for a non-tightly concatenated codeword the shift register is zero and the input is zero for at least one time instance before non-zero input is entering again. For a tightly concatenated codeword the shift register is zero, and there is immediately non-zero input being fed into the system. If v is concatenated at time $t = L$, but not tightly concatenated, then we have $x_L = x_{L+1} = 0$. The converse is not true: it might happen that $x_L = x_{L+1} = 0$ even if v is tightly concatenated at time $t = L$ simply because the matrix B might have a non-trivial kernel. All this is best visualised by using the state diagram.

Definition 2.8

(a) Consider the state space $\mathbb{F}^\gamma$ of the encoder G. We define the *state diagram* of (A, B, C, D) as the labeled directed graph given by the vertex set $\mathbb{F}^\gamma$ and the set of edges

$$\left\{ X \overset{\left(\frac{u}{v}\right)}{\rightarrow} Y | X, Y \in \mathbb{F}^\gamma, u \in \mathbb{F}^k, v \in \mathbb{F}^n : Y = XA + uB, v = XC + uD \right\}.$$

(b) A *path of length* l is a sequence of edges of the form

$$X_{i_0} \overset{\left(\frac{u_0}{v_0}\right)}{\rightarrow} X_{i_1} \overset{\left(\frac{u_1}{v_1}\right)}{\rightarrow} X_{i_2} \overset{\left(\frac{u_2}{v_2}\right)}{\rightarrow} \ldots\ldots \overset{\left(\frac{u_{l-2}}{v_{l-2}}\right)}{\rightarrow} X_{i_{l-1}} \overset{\left(\frac{u_{l-1}}{v_{l-1}}\right)}{\rightarrow} X_{i_l}. \tag{2.4}$$

The path is called a *cycle around* X_{i_0} if $X_{i_0} = X_{i_l}$. The *weight* of the path (2.4) is defined as $\sum_{i=0}^{l-1} \mathrm{wt}(v_i)$.

(c) Let $u, x, v := uG$ be as in Theorem 2.3 and let $\deg v = N$. Then $x_{N+1} = 0$ (since $\deg x \leqslant \deg v$) and we call the path

$$0 = x_0 \overset{\binom{u_0}{v_0}}{\rightarrow} x_1 \overset{\binom{u_1}{v_1}}{\rightarrow} x_2 \overset{\binom{u_2}{v_2}}{\rightarrow} x_3 \cdots\cdots x_N \overset{\binom{u_N}{v_N}}{\rightarrow} x_{N+1} = 0$$

the cycle around zero associated with the codeword $v = uG$.

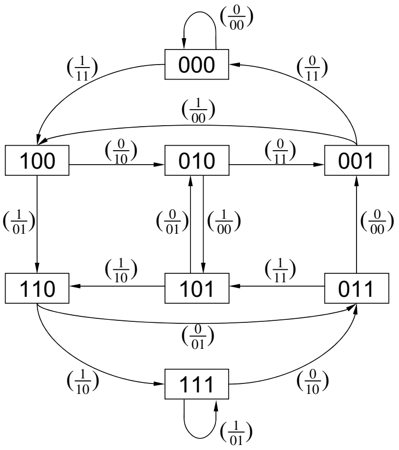
Note that the edges of the state diagram correspond to the transitions in the canonical shift register: $X \overset{\binom{u}{v}}{\rightarrow} Y$ is an edge if for some time instance t the memory vector is given by X, the input is u, and this leads to the next memory vector Y and the output v. Hence there emerge q^k edges at each vertex.

As a consequence, the state diagram contains all information about the encoding process. The cycles around $X_0 = 0$ are in one-one correspondence with the codewords in im G. The message sequence determines the path through the graph and the corresponding v-labels yield the associated codeword. A codeword is atomic if and only if the associated cycle does not pass through the zero state except for starting and end point, see Lemma 2.6. A codeword is molecular if and only if the associated cycle does not contain the edge $0 \overset{\binom{0}{0}}{\rightarrow} 0$, see the discussion right after Definition 2.7. Note also that it is possible to have two different edges between the same vertices, i.e. edges of the form $X \overset{\binom{u}{v}}{\rightarrow} Y$ and $X \overset{\binom{u'}{v'}}{\rightarrow} Y$ where $u \neq u'$. This happens if and only if $\gamma_l = 0$ for at least one l which is equivalent to B having a non-trivial kernel. In this and only this case there are also edges of the form $0 \overset{\binom{u}{v}}{\rightarrow} 0$ in the state diagram such that $u \in \ker B$ is not zero.

Example 2.9. Consider the basic and minimal matrix $G = (1 + z + z^2 + z^3, 1 + z^2 + z^3) \in \mathbb{F}_2[z]^{1\times 2}$. Since $\gamma = 3$ the state diagram has $2^3 = 8$ vertices. The controller canonical form is given by

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

Going through all options for the equations $x_{t+1} = x_t A + u_t B$, $v_t = x_t C + u_t D$ yields the associated state diagram



Notice that the state diagram is not an invariant of the code. It depends on the choice of the minimal generator matrix as well as on the minimal realization. We will discuss this issue in detail in Remark 3.6 and Section 4.

3. The adjacency matrix and the weight distribution

In this section we will illustrate how to compute the weight distribution of a convolutional code in terms of the state diagram. A main tool will be the adjacency matrix associated with the state diagram as it has been defined in [16]. In slightly different forms this matrix appears also in other papers on convolutional codes, see for instance [9, Section 3.10]. Theorem 3.8 has been derived in [16]. First of all we need a reasonable definition for the weight distribution. Of course, it is sufficient to count only the weights of atomic codewords. In order to do so, we need to show that we are dealing with finite numbers if counted appropriately. This will be dealt with in the first lemma. As before General Assumption 2.5 is in force.

Lemma 3.1. *Assume the memory of G, i.e., the maximal Forney index, is given by m.*

(a) *Let $v = uG \in \mathscr{C}$ where $u = \sum_{i=0}^{L} u_i z^i \in \mathbb{F}[z]^k$ with $u_0 \neq 0 \neq u_L$. If $u_{l+1} = u_{l+2} = \cdots = u_{l+m} = 0$ for some $l \in \{0, \ldots, L-m-1\}$, then v is concatenated.*

(b) *For all $\alpha \in \mathbb{N}_0$ we have $\#\{v \in \mathscr{C} | v \text{ atomic}, \mathrm{wt}(v) \leqslant \alpha\} < \infty$.*

Proof. (a) Write $u = \sum_{i=0}^{l} u_i z^i + \sum_{i=l+m+1}^{L} u_i z^i =: \tilde{u} + \hat{u}$. Then $\deg(\tilde{u}G) \leqslant l + m$ and $\hat{u}G \in z^{l+m+1}\mathbb{F}[z]^n$, thus $v = \tilde{u}G + \hat{u}G$ is concatenated at some time $t \leqslant l + m + 1$.

(b) Let $\widehat{G} \in \mathbb{F}[z]^{n\times k}$ be a right inverse of G and let $\widehat{G}$ have maximal row degree $\hat{m}$. Suppose $v = uG \in \mathscr{C}$ is a codeword with at least $m + \hat{m}$ consecutive zero coefficients. Thus, $v = \tilde{v} + \hat{v}$ where $\tilde{v}, \hat{v} \in \mathbb{F}[z]^n\backslash\{0\}$ satisfy $\deg \tilde{v} \leqslant L$ and $\hat{v} \in z^{L+m+\hat{m}+1}\mathbb{F}[z]^n$ for some $L \in \mathbb{N}_0$. Then $u = uG\widehat{G} = \tilde{v}\widehat{G} + \hat{v}\widehat{G}$ and part (a) shows that v is not atomic. All this proves that atomic codewords do not have more than $m + \hat{m} - 1$ consecutive zero coefficients. As a consequence, all atomic codewords of weight at most α have a degree bounded by some $M_\alpha \in \mathbb{N}$ proving the assertion. □

Remark 3.2. It is not hard to prove that if all Forney indices of G are equal to m then part (a) above becomes an if-and-only-if statement. Indeed, let $v = uG \in \mathscr{C}$ be concatenated as $v = \tilde{v} + \hat{v}$ where $\tilde{v}, \hat{v} \in \mathscr{C}\backslash\{0\}$ and $\deg \tilde{v} \leqslant T$ and $\hat{v} \in z^{T+1}\mathbb{F}[z]^n$ for some $T \leqslant \deg u - 1$. We have to show that $u_T = u_{T-1} = \cdots = u_{T-m+1} = 0$. From Lemma 2.6 we know that $x_{T+1} = 0$, thus the dynamical equations in Theorem 2.3(2) yield $0 = x_T A + u_T B = x_{T-1}A^2 + u_{T-1}BA + u_T B$ and finally

$$0 = x_{T-m+1}A^m + (u_{T-m+1}, \ldots, u_T)\begin{pmatrix} BA^{m-1} \\ \vdots \\ BA \\ B \end{pmatrix}. \tag{3.1}$$

Using the controller canonical form it follows directly that if all Forney indices are m, then $A^m = 0$ and the matrix on the very right of (3.1) is a non-singular $\gamma \times \gamma$-matrix (via Proposition 2.2 the same is true then for any minimal realization of G). Hence (3.1) yields the desired result.

The above makes the computation of the distance of a given code to a finite problem, at least theoretically.

Definition 3.3. For all $\alpha, l \in \mathbb{N}$ define $\omega_{l,\alpha} := \#\{v \in \mathscr{C} | v \text{ atomic}, \deg v = l - 1, \text{wt}(v) = \alpha\}$. The power series $\Omega = \Omega(W, L) := \sum_{l=1}^{\infty}\sum_{\alpha=1}^{\infty} \omega_{l,\alpha} W^\alpha L^l \in \mathbb{Q}[[W, L]]$ is called the *weight distribution* of the code $\mathscr{C}$.

Observe that the weight distribution Ω is an invariant of the code and does not depend on a chosen generator matrix. For simplicity we omit the constant term 1 representing the zero codeword. Notice that for block codes we simply have $\Omega = \sum_{\alpha=1}^{n} \omega_{1,\alpha} W^\alpha L$. This is, up to the factor L, the ordinary weight distribution for block codes where the constant term 1, representing the zero codeword, has been omitted. It is clear that the numbers $\omega_{l,\alpha}$ are indeed finite. Moreover, since each codeword of degree $l - 1$ has at most l non-zero coefficients in $\mathbb{F}^n$, we have $\Omega = \sum_{l=1}^{\infty}\sum_{\alpha=1}^{nl} \omega_{l,\alpha} W^\alpha L^l \in \mathbb{Q}[W][[L]]$. We also have for each $\alpha \in \mathbb{N}$ that $\sum_{l=0}^{\infty} \omega_{l,\alpha} L^l$ is a finite sum, due to Lemma 3.1(b). Hence $\Omega = \sum_{\alpha=1}^{\infty}\sum_{l=1}^{\infty} \omega_{l,\alpha} L^l W^\alpha \in \mathbb{Q}[L][[W]]$. Finally observe that $\text{dist}(\mathscr{C}) = \min\{\alpha \in \mathbb{N} | \exists l \in \mathbb{N} : \omega_{l,\alpha} \neq 0\}$ is the degree of the smallest term ocurring in the series expansion with respect to W.

In the sequel we will discuss how one can compute, at least theoretically, the weight distribution of a code using the state diagram. All necessary information is contained in the following matrix Λ. Define

$$\mathscr{F} := \mathbb{F}^{\gamma} \times \mathbb{F}^{\gamma} \quad \text{and} \quad s := q^{\gamma}. \tag{3.2}$$

Definition 3.4. For all $(X, Y) \in \mathscr{F}$ and all $\alpha \in \{0, \ldots, n\}$ put

$$\lambda_{X,Y}^{(\alpha)} := \#\{u \in \mathbb{F}^k \mid Y = XA + uB, \mathrm{wt}(XC + uD) = \alpha\}.$$

The matrix

$$\Lambda := \left(\sum_{\alpha=0}^{n} \lambda_{X,Y}^{(\alpha)} W^{\alpha}\right)_{(X,Y)\in\mathscr{F}} \in \mathbb{Q}[W]^{s\times s}$$

is called the *adjacency matrix* of the realization (A, B, C, D).

Notice that $\lambda_{X,Y}^{(\alpha)}$ is the number of all edges in the state diagram of the form $X \overset{\binom{u}{v}}{\to} Y$ where $\mathrm{wt}(v) = \alpha$. Obviously the weight α is bounded by n. One can regard the adjacency matrix as a generalization of the weight distribution for block codes. Indeed, in that case $G = D$ and $\gamma = 0$. Therefore $\Lambda = \sum_{\alpha=0}^{n} \lambda_{0,0}^{(\alpha)} W^{\alpha}$ where $\lambda_{0,0}^{(\alpha)} = \#\{u \in \mathbb{F}^k | \mathrm{wt}(uD) = \alpha\}$.

Remark 3.5. If we want to display Λ explicitly as a matrix, we need to fix an ordering of the states. If we do so, we will always choose the same ordering for the row index X and for the column index Y and we will pick the zero state as the first index. In this sense, the adjacency matrix of a realization is well-defined up to similarity transformation via a permutation matrix $P \in \mathbb{Q}^{s\times s}$ where $P_{0,0} = 1$.

Just like the state diagram the adjacency matrix depends on the minimal encoder as well as on the minimal realization. We have the following result, which also fits with the previous remark.

Remark 3.6. Suppose (A, B, C, D) and $(\overline{A}, \overline{B}, \overline{C}, \overline{D})$ are two minimal realizations of the minimal generator matrix G. According to Proposition 2.2 there exists $T \in \mathrm{Gl}_{\gamma}(\mathbb{F})$ such that $(\overline{A}, \overline{B}, \overline{C}, \overline{D}) = (TAT^{-1}, BT^{-1}, TC, D)$. Let $\lambda_{X,Y}^{(\alpha)}$ and $\overline{\lambda}_{X,Y}^{(\alpha)}$ be the respective enumerators as defined in 3.4 and Λ, $\overline{\Lambda}$ be the adjacency matrices. Then

$$\begin{aligned}\overline{\lambda}_{X,Y}^{(\alpha)} &= \#\{u \in \mathbb{F}^k | Y = X\overline{A} + u\overline{B}, \mathrm{wt}(X\overline{C} + u\overline{D}) = \alpha\}\\ &= \#\{u \in \mathbb{F}^k | YT = (XT)A + uB, \mathrm{wt}((XT)C + uD) = \alpha\} = \lambda_{XT,YT}^{(\alpha)}\end{aligned}$$

for all $(X, Y) \in \mathscr{F}$ and all $\alpha = 0, \ldots, n$. Thus, $\overline{\Lambda}_{X,Y} = \Lambda_{XT,YT}$ for all $(X, Y) \in \mathscr{F}$. Fixing an ordering of the states as described in the previous remark, we obtain $\overline{\Lambda} = V\Lambda V^{-1}$ where $V \in \mathbb{Q}^{s\times s}$ is a permutation matrix such that $V_{0,0} = 1$. Notice that the permutation on the s indices in $\mathbb{F}^\gamma$ is even a linear mapping. This discussion also shows that the state diagram of the two realizations differ only by a relabeling of the vertices.

Example 3.7

(1) Let $G = \begin{pmatrix} 1 & 1 & 0 \\ 0 & z+1 & z \end{pmatrix} \in \mathbb{F}[z]^{2\times 3}$. Thus $s = 2^\gamma = 2$ and the controller canonical form is $A = (0)$, $B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, $C = \begin{pmatrix} 0 & 1 & 1 \end{pmatrix}$, $D = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix}$. With these data one obtains straightforwardly $\Lambda = \begin{pmatrix} 1+W^2 & 2W \\ 2W^2 & W+W^3 \end{pmatrix}$.

(2) In Example 2.9 we have $s = 2^3 = 8$ and obtain, with an appropriate ordering of the states, directly from the state diagram the matrix

$$\Lambda = \begin{pmatrix} 1 & 0 & 0 & 0 & W^2 & 0 & 0 & 0 \\ W^2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & W^2 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & W^2 & 0 & 0 \\ 0 & 0 & W & 0 & 0 & 0 & W & 0 \\ 0 & 0 & W & 0 & 0 & 0 & W & 0 \\ 0 & 0 & 0 & W & 0 & 0 & 0 & W \\ 0 & 0 & 0 & W & 0 & 0 & 0 & W \end{pmatrix}.$$

Now we can present the result of McEliece about how to compute the weight distribution of a code. We use the notation from (3.2).

Theorem 3.8 (McEliece [16, Theorem 3.1]). *Let Λ be the adjacency matrix of the minimal realiztion (A, B, C, D) and define $\widehat{\Lambda} \in \mathbb{Q}[W]^{s\times s}$ by $\widehat{\Lambda}_{0,0} = \Lambda_{0,0} - 1$ and $\widehat{\Lambda}_{X,Y} = \Lambda_{X,Y}$ for all $(X, Y) \in \mathscr{F}\backslash\{(0, 0)\}$. Then*

$$\Omega = 1 - \Phi^{-1}, \quad \textit{where } \Phi = [(I - L\widehat{\Lambda})^{-1}]_{0,0} \in \mathbb{Q}(W, L).$$

The result is also compliant with the block code case where $\Lambda = \sum_{v\in\mathscr{C}} W^{\mathrm{wt}(v)} \in \mathbb{Q}[W]$ and $\Omega = (\Lambda - 1)L$. The matrix $\widehat{\Lambda}$ can be regarded as the adjacency matrix of the state diagram where the edge $0 \overset{\binom{0}{0}}{\rightarrow} 0$ has been removed. This is necessary for the proof, where one first shows that Φ is the weight distribution of the molecular codewords. As has been discussed after Definition 2.8, the latter can be identified with the cycles around zero in the state diagram without the edge $0 \overset{\binom{0}{0}}{\rightarrow} 0$. Notice that due to Remarks 3.5 and 3.6 the definition of Φ does not depend on the chosen minimal realization. We will present a sketch of the poof. For the details we refer to [16].

Sketch of proof: We call the state diagram where the edge $0 \overset{\binom{0}{0}}{\rightarrow} 0$ has been removed the reduced state diagram. Then one can show straightforwardly that the lth power of $\widehat{\Lambda}$ satisfies $(\widehat{\Lambda}^l)_{X,Y} = \sum_{\alpha=0}^{nl} \hat{\lambda}_{X,Y}^{(l,\alpha)} W^\alpha$ where $\hat{\lambda}_{X,Y}^{(l,\alpha)}$ is the number of paths from X to Y of length l and weight α in the reduced state diagram (this is actually proven in the same way as an analogous case in the first paragraph of the proof of Lemma 5.2 later on). In particular, $\hat{\lambda}_{0,0}^{(l,\alpha)}$ is the number of molecular codewords of length l and weight α. Now

$$\begin{aligned}\Phi &= \Big[(I - L\widehat{\Lambda})^{-1}\Big]_{0,0} = \left[\sum_{l=0}^{\infty}(L\widehat{\Lambda})^l\right]_{0,0}\\ &= 1 + \sum_{l=1}^{\infty}(\widehat{\Lambda}^l)_{0,0}L^l = 1 + \sum_{l=1}^{\infty}\sum_{\alpha=0}^{\infty}\hat{\lambda}_{0,0}^{(l,\alpha)}W^\alpha L^l.\end{aligned}$$

Hence Φ is the weight distribution of the molecular codewords. On the other hand, one can show by induction on $r \in \mathbb{N}$ that $\Omega^r = 1 + \sum_{l=1}^{\infty}\sum_{\alpha=1}^{\infty}\omega_{l,\alpha}^{(r)}W^\alpha L^l$ where $\omega_{l,\alpha}^{(r)}$ is the number of all codewords of weight α and length l that consist of exactly r tightly concatenated atomic codewords. Thus, by definition of the molecular codewords, $\Phi = 1 + \Omega + \Omega^2 + \Omega^3 + \cdots = \frac{1}{1-\Omega}$. □

If one follows the arguments above one has to perform the following steps in order to compute the weight distribution of a given code:

1. Compute Λ.
2. Solve the equation $x(I - L\widehat{\Lambda}) = (1, 0, \ldots, 0)$ for $x \in \mathbb{Q}(W, L)^s$ where we apply Remark 3.5 for writing $\widehat{\Lambda}$ as a matrix.
3. Then $\Phi = x_1$ and $\Omega = 1 - \Phi^{-1}$.

While (1) and (3) are easily done for reasonably sized parameters, step (2) quickly becomes unpractical with growing overall constraint length γ and/or field size q since the size of the adjacency matrix is $q^\gamma \times q^\gamma$. Better algorithms for computing the weight distribution while avoiding the big adjacency matrix are Viterbi's method, see [25] or [9, Section 3.10], or Mason's gain formula as described in [11, Section 10.2]. Further methods can be found in [2,17,18].

Example 3.9. Let G be as in Example 2.9. In this case the adjacency matrix is only 8×8 and we can perform the computation of the weight distribution along the steps (1)–(3) above by using, for instance, Maple. We computed the adjacency matrix Λ already in Example 3.7(2). From that one obtains the weight distribution

$$\begin{aligned}\Omega &= L^4W^6(L + W - LW^2)/(1 - LW - L^2W + L^3W^2 - L^3W^3\\ &\quad - L^4W^2 - L^3W^4 + L^4W^4)\\ &= L^5W^6 + (L^4 + L^6 + L^7)W^7 + (L^6 + L^7 + L^8 + 2L^9)W^8\\ &\quad + (4L^8 + L^9 + 3L^{10} + 3L^{11})W^9 + \mathrm{O}(W^{10}).\end{aligned}$$

Thus, the distance is 6 and the code contains exactly one atomic codeword of weight 6, it has length 5. It contains three atomic codewords (of length 4, 6, and 7, respectively) of weight 7 and 5 atomic codewords of weight 8, two of which have length 9 the other three have length 6, 7, 8, respectively, etc.

At the end of this section we want to briefly mention two distance parameters appearing in a totally different context in the literature that are closely related to our notions. As to our knowledge this relationship, even though quite simple, has not yet been exhibited. In [10, p. 541] the *extended row distances* $\hat{d}_l^r$ are defined. The sequence of these parameters is closely related to the error-correcting performance of the code. In particular, it is desirable that they have a large growth rate, see [10] for further details. The definition shows immediately that

$$\hat{d}_l^r = \min\{\text{wt}(v) \mid v \text{ atomic}, \deg(v) = l\} = \min\{\alpha \in \mathbb{N}_0 \mid \omega_{l+1,\alpha} \neq 0\},$$

where $\omega_{j,\alpha}$ are given in Definition 3.3. Thus these distance parameters can be recovered, at least theoretically, from the weight distribution Ω. There are also other definitions of the extended row distances in the literature. They are slightly different and in general not that closely related to atomic or molecular codewords and some of them even depend on the choice of the (minimal) encoder.

In [7] the active burst distances have been introduced. They form a generalization of the extended row distances and, again, reveal information about the error-correcting capability of the code. They are of particular importance for the investigation of concatenated codes. The lth order *active burst distance* of the code $\mathscr{C} = \operatorname{im} G$ is defined as [7, p. 155]

$$a_l^b := \min\{\text{wt}((uG)_{[0,l]}) \mid x_{l+1} = 0 \text{ and } (x_i, x_{i+1}, u_i) \neq 0 \text{ for all } 0 \leqslant i \leqslant l\},$$

where, as usual, x is the associated state sequence, see also [9, Section 3.2]. Moreover, $(uG)_{[0,l]}$ denotes the codeword truncated after the lth power of z. As we will show now, if G is minimal then

$$a_l^b = \min\{\text{wt}(v) \mid v \text{ molecular}, \deg v = l\} = \min\{\alpha \in \mathbb{N} \mid \hat{\lambda}_{0,0}^{(l+1,\alpha)} \neq 0\}, \qquad (3.3)$$

where $\hat{\lambda}_{0,0}^{(l,\alpha)}$ are given in the proof of Theorem 3.8. The second identity follows directly from that proof. As for the first identity, note first that Lemma 2.6 along with $x_{l+1} = 0$ implies that $v := (uG)_{[0,l]}$ is a codeword. Due to $(x_i, x_{i+1}, u_i) \neq 0$ this codeword is molecular. Finally we have $\deg v = l$, which can be seen as follows. Suppose $\deg v < l$. Since G is minimal we have $\deg u \leqslant \deg v$, see Theorem 2.3(1). Hence $u_l = 0 = v_l$. Using the controller canonical form (A, B, C, D) we obtain $x_{l+1} = 0 = x_l A$ and $v_l = 0 = x_l C$, and (2.2) implies $x_l = 0$. Hence $(x_l, x_{l+1}, u_l) = 0$ which contradicts the choice of v. All this together shows that v is a molecular codeword of degree l. Conversely one can easily see that each such codeword has a state sequence x such that $(x_i, x_{i+1}, u_i) \neq 0$ for all $0 \leqslant i \leqslant l$. This proves the first identity of (3.3). Hence the active burst distances occur in the series Φ as used in Theorem 3.8 for enumerating the molecular codewords. In particular we have that $\min_{l \geqslant 0} a_l^b$ equals the free distance of the code, see also [9, Theorem 3.8].

4. The adjacency matrix as an invariant of the code

In this section we will prove that the adjacency matrix is an invariant of the code, i.e., does not depend on the choice of the minimal generator matrix, provided that one factors out isomorphisms on the state space. As to our knowledge the result of this and the next section are new.

Throughout this section we will use the notation from (3.2). As shown in Remark 3.6 the adjacency matrix depends on the chosen minimal realization. It gives rise to the following notation. On $\mathbb{Q}[W]^{s\times s}$ we define the equivalence relation

$$M \sim M' \Longleftrightarrow \exists T \in \mathrm{Gl}_\gamma(\mathbb{F}) \text{ such that } M'_{X,Y} = M_{XT,YT} \text{ for all } (X,Y) \in \mathscr{F}. \tag{4.1}$$

Moreover, we define $\overline{M}$ to be the equivalence class of $M \in \mathbb{Q}[W]^{s\times s}$.

Now Remark 3.6 tells us that we have a well-defined mapping $G \longmapsto \overline{\Lambda_G}$ from the set of all minimal generator matrices with overall constraint length γ into $\mathbb{Q}[W]^{s\times s}/_\sim$ by simply choosing Λ_G as the adjacency matrix of G with respect to any minimal realization. We will show now that $\overline{\Lambda_G}$ is even an invariant of the code. Indeed, we have

Theorem 4.1. *Let $G, G' \in \mathbb{F}[z]^{k\times n}$ be two minimal generator matrices such that $\mathscr{C} := \operatorname{im} G = \operatorname{im} G'$. Then*

$$\overline{\Lambda_G} = \overline{\Lambda_{G'}}. \tag{4.2}$$

We will write $\overline{\Lambda}(\mathscr{C}) := \overline{\Lambda_G}$ for this invariant and call it the generalized adjacency matrix of the code.

The theorem tells us that the generalized adjacency matrix is a well-defined mapping

$$\begin{array}{rcc} \overline{\Lambda}: \{\mathscr{C} \subseteq \mathbb{F}_q[z]^n \mid \mathscr{C} \text{ code with overall constraint length } \gamma\} & \longrightarrow & \mathbb{Q}[W]^{q^\gamma\times q^\gamma}/_\sim \\ \mathscr{C} & \longmapsto & \overline{\Lambda}(\mathscr{C}). \end{array} \tag{4.3}$$

Proof. Let $\nu_1, \ldots, \nu_k$ and $\mu_1, \ldots, \mu_k$ be the row degrees of G and G', respectively. By assumption we have [3, Theorem 4]

$$G' = UG \text{ for some matrix } U \in \mathrm{Gl}_k(\mathbb{F}[z]), \tag{4.4}$$

where $\mathrm{Gl}_k(\mathbb{F}[z])$ denotes the group of unimodular $k \times k$-matrices over $\mathbb{F}[z]$. Moreover, by uniqueness of the Forney indices, $\{\nu_1, \ldots, \nu_k\} = \{\mu_1, \ldots, \mu_k\}$. Let $\gamma := \sum_{i=1}^k \nu_i = \sum_{i=1}^k \mu_i$ be the overall constraint length of the code.

Since every unimodular matrix $U \in \mathrm{Gl}_k(\mathbb{F}[z])$ is the product of elementary matrices, we may show the result for each type of elementary transformation separately. By virtue of Remark 3.6 we may choose without loss of generality the controller canonical forms (A, B, C, D) and (A', B', C', D') of G and G', respectively. Let Λ

and Λ' be the adjacency matrices of these realizations, respectively. We will show that each type of elementary transformation U results in $\Lambda' \sim \Lambda$.

(1) We show that a permutation of the rows of G results in a transformation of Λ just like in (4.1). Thus let us assume $G' = UG$ where U permutes the ith and jth row of G. Then A' is obtained from A by permuting the ith and jth block row and column, B' is obtained from B by permuting the ith and jth row and the ith and jth block column, C' is obtained from C by permuting the ith and jth block row and finally D' is obtained from D by permuting the ith and jth row. Thus, there exists a permutation matrix $P \in \mathrm{Gl}_\gamma(\mathbb{F})$ such that

$$A' = PAP^{-1}, \quad B' = UBP^{-1}, \quad C' = PC, \quad D' = UD$$

(this is also correct if ν_i or ν_j is zero). Now let $X \overset{\binom{u}{v}}{\rightarrow} Y$ be an edge in the state diagram of (A, B, C, D). Then $Y = XA + uB$ and $v = XC + uD$. From this we obtain $YP^{-1} = (XP^{-1})A' + (uU^{-1})B'$ and $v = (XP^{-1})C' + (uU^{-1})D'$, hence $XP^{-1} \overset{\binom{uU^{-1}}{v}}{\rightarrow} YP^{-1}$ is an edge in the state diagram of (A', B', C', D'). The very definition of the adjacency matrices shows that $\Lambda_{X,Y} = \Lambda'_{XP^{-1},YP^{-1}}$ for all $(X, Y) \in \mathscr{F}$. This in turn implies (4.2).

(2) Next we consider the case where the matrix U in (4.4) multiplies the rows of G with some non-zero constants, say $U = \operatorname{diag}(u_1, \ldots, u_k) \in \mathrm{Gl}_k(\mathbb{F})$. Then

$$A' = A, \quad B' = B, \quad C' = \widehat{U}C, \quad D' = UD,$$

where

$$\widehat{U} = \begin{pmatrix} u_1 I_{\nu_1} & & & \\ & u_2 I_{\nu_2} & & \\ & & \ddots & \\ & & & u_k I_{\nu_k} \end{pmatrix}.$$

Here I_j denotes the $j \times j$-identity matrix. It is easy to see that $\widehat{U}A = A\widehat{U}$ and $UB = B\widehat{U}$, thus $A' = A = \widehat{U}A\widehat{U}^{-1}$ and $B' = B = UB\widehat{U}^{-1}$. Using the same arguments as in case (1) we arrive at (4.2).

(3) Now we consider the case where U adds a *constant* multiple of one row to another. Because of part (1) of this proof we may assume $\nu_1 = \mu_1 \leqslant \cdots \leqslant \nu_k = \mu_k$. Since we did already part (2) of this proof and since G and $G' = UG$ are both minimal we only have to consider the case where the jth row is added to the ith row and $j < i$. Hence $U = I_k + E$ where $E \in \mathbb{F}^{k \times k}$ has a 1 at position (i, j) and 0 elsewhere. Let us first assume $\nu_j > 0$. Put

$$\widehat{U} = \begin{pmatrix} I_{\nu_1} & & & & & & \\ & \ddots & & & & & \\ & & I_{\nu_j} & & & & \\ & & & \ddots & & & \\ & & M & & I_{\nu_i} & & \\ & & & & & \ddots & \\ & & & & & & I_{\nu_k} \end{pmatrix} \in \mathrm{Gl}_\gamma(\mathbb{F}), \tag{4.5}$$

where

$$M = \begin{pmatrix} I_{\nu_j} \\ 0 \end{pmatrix} \in \mathbb{F}^{\nu_i \times \nu_j}.$$

Then we have

$$A' = A, \quad B' = B, \quad C' = \widehat{U}C, \quad D' = UD.$$

Furthermore, it is easy to see that $MA_j = A_i M$ and $B_i M = B_j$ where A_i, B_i are the diagonal blocks of the matrices A, B as given in Definition 2.1. From this we obtain $\widehat{U}A = A\widehat{U}$ and $UB = B\widehat{U}$. Now we can use the same arguments as in case (2) to finish the proof. If $\nu_j = 0$ we have $A' = A$, $B' = B$, $C' = C$, and $D' = UD$. Using $\widehat{U} := I_\gamma$ and the fact that the jth row of B is zero, we have again $UB = B\widehat{U}$ and we can argue as before.

(4) Finally we have to consider the case where the matrix U adds a non-constant multiple of one row of G to another. Without restriction we may assume that z^l times the jth row is added to the ith row. Since G' is supposed to be minimal again, we have $l \leqslant \nu_i - \nu_j$. Hence $U = I_k + E$ where $E \in \mathbb{F}^{k\times k}$ has the entry z^l at position (i, j) and 0 elsewhere. Let again first $\nu_j > 0$. Consider $\widehat{U}$ as in (4.5) but where M now is of the form

$$M = \begin{pmatrix} 0_{l\times\nu_j} \\ I_{\nu_j} \\ 0 \end{pmatrix} \in \mathbb{F}^{\nu_i \times \nu_j}.$$

Then we obtain

$$A' = A, \quad B' = B, \quad C' = \widehat{U}C + \widehat{E}D, \quad D' = D,$$

where $\widehat{E} \in \mathbb{F}^{\gamma\times k}$ has a 1 at position (r, j) with $r = \sum_{\tau=1}^{i-1} \nu_\tau + l$ and 0 elsewhere. Furthermore, $A_i M = MA_j + N$ where $N \in \mathbb{F}^{\nu_i\times\nu_j}$ has a 1 at position $(l, 1)$ and 0 elsewhere. Thus $A\widehat{U} = \widehat{U}A + \widehat{N}$ where $\widehat{N} \in \mathbb{F}^{\gamma\times\gamma}$ satisfies $\widehat{N}_{r,t} = 1$ with r as above and $t = \sum_{\tau=1}^{j-1} \nu_\tau + 1$ and all other entries are zero. Moreover, one has $\widehat{E}B = \hat{N}$, since $\nu_j > 0$, as well as $B\widehat{U} = B$ since $l > 0$. Suppose now that $X \overset{\left(\begin{smallmatrix} u \\ v \end{smallmatrix}\right)}{\to} Y$ is an edge in the state diagram of (A, B, C, D). Then $Y = XA + uB$ and $v = XC + uD$. One computes

$$Y\widehat{U}^{-1} = (X\widehat{U}^{-1})A' + (u - X\widehat{U}^{-1}\widehat{E})B'$$

and

$$v = (X\widehat{U}^{-1})C' + (u - X\widehat{U}^{-1}\widehat{E})D'.$$

Thus, putting $\tilde{u} = u - X\widehat{U}^{-1}\widehat{E}$, we obtain that $X\widehat{U}^{-1} \overset{\left(\begin{smallmatrix} \tilde{u} \\ v \end{smallmatrix}\right)}{\to} Y\widehat{U}^{-1}$ is an edge of the state diagram of (A', B', C', D'). Again this implies (4.2). In the case $\nu_j = 0$ we have $A' = A$, $B' = B$, $C' = C + \widehat{E}D$, and $D' = D$ where $\widehat{E}$ is as before. Since the jth row of B is zero, one has $\widehat{E}B = 0$ and thus the equations $Y = XA + uB$, $v = XC + uD$

are equivalent to the equations $Y = XA' + (u - X\widehat{E})B'$, $v = XC' + (u - X\widehat{E})D'$ and we can argue as above. This completes the proof. □

Notice that the proof also shows how the controller canonical form changes under unimodular transformations of the minimal generator matrix. However, we will not need that result explicitly.

Remark 4.2. It is worth noting that for codes with overall constraint length $\gamma = 1$ all associated adjacency matrices are identical. Indeed, assume we have two adjacency matrices $\Lambda, \Lambda' \in \mathbb{Q}[W]^{q\times q}$ associated with the code $\mathscr{C}$. By Theorem 4.1 there exists an element $a \in \mathbb{F}\backslash\{0\}$ such that $\Lambda'_{X,Y} = \Lambda_{aX,aY}$ for all $X, Y \in \mathbb{F}$. If (A, B, C, D) is any minimal realization of a minimal encoder of $\mathscr{C}$, then obviously $[Y = XA + uB \Longleftrightarrow aY = aXA + auB]$ and $\mathrm{wt}(XC + uD) = \mathrm{wt}(aXC + auD)$ for all $X, Y \in \mathbb{F}$. Therefore, Definition 3.4 shows immediately that $\Lambda_{aX,aY} = \Lambda_{X,Y}$ and thus $\Lambda' = \Lambda$.

Next we will briefly turn to monomially equivalent codes. Defining monomial equivalence just like for block codes it is straightforward to show that it preserves the generalized adjacency matrices, see Theorem 4.4 below. In the next section we will see that for certain classes of codes even the converse of that theorem is true.

Definition 4.3. Two matrices $G, G' \in \mathbb{F}[z]^{k\times n}$ are called *monomially equivalent* if $G' = GPR$ for some permutation matrix $P \in \mathrm{Gl}_n(\mathbb{F})$ and a non-singular diagonal matrix $R \in \mathrm{Gl}_n(\mathbb{F})$. Thus, G and G' are monomially equivalent if and only if they differ by a permutation and a rescaling of the columns. We call two codes $\mathscr{C}, \mathscr{C}' \subseteq \mathbb{F}[z]^n$ *monomially equivalent* and write $\mathscr{C} \sim \mathscr{C}'$ if $\mathscr{C} = \mathrm{im}\, G$ and $\mathscr{C}' = \mathrm{im}\, G'$ for some monomially equivalent generator matrices.

It is clear that monomially equivalent codes have the same distance and the same weight distribution. As we show next they even have the same generalized adjacency matrix.

Theorem 4.4. *Let $\mathscr{C}, \mathscr{C}' \subseteq \mathbb{F}[z]^n$ be two convolutional codes. Then*

$$\mathscr{C} \sim \mathscr{C}' \Longrightarrow \overline{\Lambda}(\mathscr{C}) = \overline{\Lambda}(\mathscr{C}').$$

Proof. Let $G, G' \in \mathbb{F}[z]^{k\times n}$ be generator matrices of $\mathscr{C}$ and $\mathscr{C}'$, respectively, such that $G' = GPR$ for some permutation matrix P and a non-singular constant diagonal matrix R. Then the controller canonical forms satisfy $A' = A$, $B' = B$, $C' = CPR$, $D' = DPR$. Thus, if $X \overset{(\frac{u}{v})}{\rightarrow} Y$ is an edge in the state diagram associated with (A, B, C, D) then $X \overset{(\frac{u}{vPR})}{\rightarrow} Y$ is an edge in the state diagram of (A', B', C', D'). Since $\mathrm{wt}(v) = \mathrm{wt}(vPR)$ we obtain the desired result. □

5. The adjacency matrix as a complete invariant for one-dimensional binary codes

In this section we will derive some results about the generalized adjacency matrix. The guiding question is as to what properties do codes share if they have the same generalized adjacency matrix. We will first show that such codes always have the same Forney indices. Secondly, we will show that binary one-dimensional codes with the same generalized adjacency matrix are monomially equivalent. It is not known to us whether this is also true for one-dimensional codes over arbitrary fields. However, it is certainly not true for general *higher-dimensional* codes as we know from (binary) block code theory, see [8, Example 1.6.1]. As a consequence, we obtain that if two binary one-dimensional codes share the same generalized adjacency matrix, then so do their duals. This indicates the existence of a MacWilliams duality theorem for the adjacency matrices of convolutional codes, and, indeed, such a theorem has been proven for codes with overall constraint length one in the paper [1], see (5.3) at the end of this section. The general case has to remain open for future research.

We begin with showing that two codes sharing the same generalized adjacency matrix have the same Forney indices. Observe that two such codes certainly have the same overall constraint length since that determines the size of the adjacency matrix.

Theorem 5.1. *Let $\mathscr{C}, \mathscr{C}' \subseteq \mathbb{F}[z]^n$ be two convolutional codes with the same overall constraint length and such that $\overline{\Lambda}(\mathscr{C}) = \overline{\Lambda}(\mathscr{C}')$. Then $\mathscr{C}$ and $\mathscr{C}'$ have the same dimension and, up to ordering, the same Forney indices.*

For the proof we will need the following lemma.

Lemma 5.2. *Let the data be as in General Assumption 2.5 and Λ be the associated adjacency matrix. Define*

$$\rho_r = \operatorname{rank} \begin{pmatrix} B \\ BA \\ \vdots \\ BA^r \end{pmatrix}$$

for $r \in \mathbb{N}_0$. Then for any $r \geqslant 1$ we have $\#\{Y \in \mathbb{F}^\gamma \mid (\Lambda^r)_{0,Y} \neq 0\} = q^{\rho_{r-1}}$.

Notice that by virtue of Proposition 2.2 the rank ρ_r is independent from the choice of the minimal realization.

Proof. For $r \in \mathbb{N}$ let $(\Lambda^r)_{X,Y} = \sum_{\alpha=0}^{rn} \lambda_{X,Y}^{(r,\alpha)} W^\alpha$. We first show by induction on r that $\lambda_{X,Y}^{(r,\alpha)}$ is the number of the paths from X to Y of length exactly r and weight α. The case $r = 1$ is clear. Moreover,

$$(\Lambda^{r+1})_{X,Y} = \sum_{Z\in\mathbb{F}^\gamma} \Lambda_{X,Z}(\Lambda^r)_{Z,Y} = \sum_{Z\in\mathbb{F}^\gamma}\sum_{\alpha=0}^{n}\lambda_{X,Z}^{(1,\alpha)}W^\alpha \sum_{\alpha=0}^{rn}\lambda_{Z,Y}^{(r,\alpha)}W^\alpha$$

$$= \sum_{\alpha=0}^{(r+1)n}\sum_{Z\in\mathbb{F}^\gamma}\sum_{\beta=0}^{\alpha}\lambda_{X,Z}^{(1,\beta)}\lambda_{Z,Y}^{(r,\alpha-\beta)}W^\alpha.$$

By induction hypothesis $\lambda_{Z,Y}^{(r,\alpha-\beta)}$ is the number of paths from Z to Y of length r and weight $\alpha-\beta$ and likewise $\lambda_{X,Z}^{(1,\beta)}$ is the number of edges from X to Z and weight β. Since all paths from X to Y go through exactly one state Z after one step, we obtain the desired result about $\lambda_{X,Y}^{(r,\alpha)}$.

Now our assertion is equivalent to saying that there are exactly $q^{\rho_{r-1}}$ states that can be reached from the zero state by a path of length exactly r. This is obviously true for $r=1$ since the existence of an edge $0 \xrightarrow{\binom{u}{v}} Y$ is equivalent to the existence of u such that $Y=uB$ and there are $q^{\text{rk}\,B}=q^{\rho_0}$ different states Y possible. In general, the existence of a path

$$0 \xrightarrow{\binom{u_1}{v_1}} X_{j_1} \xrightarrow{\binom{u_2}{v_2}} \cdots \xrightarrow{\binom{u_r}{v_r}} X_{j_r}$$

is equivalent to the existence of $u_1,\dots,u_r$ such that

$$X_{j_r} = (u_r, u_{r-1},\dots,u_1)\begin{pmatrix} B\\ BA\\ \vdots\\ BA^{r-1}\end{pmatrix}$$

showing that this allows for $q^{\rho_{r-1}}$ different states X_{j_r}.

Now it is not hard to prove the theorem above.

Proof of Theorem 5.1. Let $G\in\mathbb{F}[z]^{k\times n}$ and $G'\in\mathbb{F}[z]^{k'\times n}$ be minimal generator matrices of $\mathscr{C}$ and $\mathscr{C}'$ and (A,B,C,D) and (A',B',C',D') the controller canonical forms, respectively. By assumption there exists a matrix $T\in\text{Gl}_\gamma(\mathbb{F})$ such that

$$\Lambda'_{X,Y} = \Lambda_{XT,YT} \quad \text{for all } (X,Y)\in\mathscr{F}. \tag{5.1}$$

From this we deduce that $k=k'$. Indeed, since there are q^k edges emerging from the zero state we have $\sum_{Y\in\mathbb{F}^\gamma}\Lambda_{0,Y}=\sum_{\alpha=0}^{n}a_\alpha W^\alpha$ where $\sum_{\alpha=0}^{n}a_\alpha=q^k$. On the other hand, the above yields $\sum_{Y\in\mathbb{F}^\gamma}\Lambda_{0,Y}=\sum_{Y\in\mathbb{F}^\gamma}\Lambda'_{0,Y}$ and using the same argument this yields $k=k'$. As for the Forney indices we proceed as follows. By induction on r one easily derives from (5.1) that $((\Lambda')^r)_{X,Y}=(\Lambda^r)_{XT,YT}$ for all $(X,Y)\in\mathscr{F}$. Thus, Lemma 5.2 implies $\rho_r=\rho'_r$ for all $r\in\mathbb{N}_0$ where

$$\rho_r = \operatorname{rank}\begin{pmatrix} B \\ BA \\ \vdots \\ BA^r \end{pmatrix}, \quad \rho_r' = \operatorname{rank}\begin{pmatrix} B' \\ B'A' \\ \vdots \\ B'A'^r \end{pmatrix}.$$

Now let $\gamma_1, \ldots, \gamma_k$ and $\gamma_1', \ldots, \gamma_k'$ be the Forney indices of G and G', respectively. By definition of A and B we have

$$\rho_0 = \operatorname{rank} B = \#\{i \mid \gamma_i > 0\} \quad \text{and} \quad \operatorname{rank} BA^r = \#\{i \mid \gamma_i > r\} \text{ for all } r \in \mathbb{N}.$$

Moreover, due to the specific form of the matrices,

$$\rho_r = \operatorname{rank}\begin{pmatrix} B \\ BA \\ \vdots \\ BA^{r-1} \end{pmatrix} + \operatorname{rank} BA^r.$$

Therefore,

$$\rho_r - \rho_{r-1} = \#\{i \mid \gamma_i > r\} \quad \text{for } r \in \mathbb{N}.$$

Analogous identities hold true for the other code. Using now $\rho_r = \rho_r'$ for all $r \in \mathbb{N}_0$ it follows

$$\#\{i \mid \gamma_i > r\} = \#\{i \mid \gamma_i' > r\} \quad \text{for all } r \in \mathbb{N}_0.$$

This shows that the Forney indices coincide up to ordering. □

Now we come to the main result of this section. The proof will make use of the Equivalence Theorem of MacWilliams about weight preserving transformations for block codes. Moreover, a technical lemma for isomorphisms on $\mathbb{F}_2^\gamma$ will be proven in the Appendix 6. It is open whether the result below can be generalized to arbitrary fields. One should, however, bear in mind that the theorem is not true for higher-dimensional codes since it even fails for binary block codes, see [8, Example 1.6.1].

Theorem 5.3. *Let $\mathscr{C}, \mathscr{C}' \subseteq \mathbb{F}_2[z]^n$ be two binary one-dimensional codes such that $\overline{\Lambda}(\mathscr{C}) = \overline{\Lambda}(\mathscr{C}')$. Then $\mathscr{C}$ and $\mathscr{C}'$ are monomially equivalent.*

Proof. Let $\mathbb{F} = \mathbb{F}_2$ and $G, G' \in \mathbb{F}[z]^{1\times n}$ be minimal generator matrices of $\mathscr{C}$ and $\mathscr{C}'$, respectively. By assumption $\mathscr{C}$ and $\mathscr{C}'$ have the same overall constraint length, say γ. Without loss of generality we may assume $\gamma \geqslant 1$. Let Λ and Λ' be the adjacency matrices of the controller canonical forms of G and G', respectively. By assumption there exists $T \in \mathrm{Gl}_\gamma(\mathbb{F})$ such that (5.1) holds true. We will first show that $T = I_\gamma$. It is quite interesting to show that this is even true if we give up the additivity of the mapping T. More precisely, we will simply assume that there exists a bijection π on $\mathbb{F}^\gamma$ such that $\pi(0) = 0$ and

$$\Lambda'_{X,Y} = \Lambda_{\pi(X),\pi(Y)} \quad \text{for all } (X, Y) \in \mathscr{F}$$

and will show that π is the identity. This is certainly true if $\gamma = 1$ (recall $\mathbb{F} = \mathbb{F}_2$), thus we may assume $\gamma \geqslant 2$. The controller canonical forms of G and G' are given by (A, B, C, D) and (A, B, C', D') where

$$A = \begin{pmatrix} & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \\ & & & & \end{pmatrix} \in \mathbb{F}^{\gamma\times\gamma}, \quad B = (1, 0, \ldots, 0) \in \mathbb{F}^{\gamma}$$

and C, D, C', D' are defined as in Definition 2.1. In the rest of the proof we will use for $X = (x_1, \ldots, x_\gamma) \in \mathbb{F}^\gamma$ and $1 \leqslant a \leqslant b \leqslant \gamma$ the notation $X_{[a,b]} := (x_a, \ldots, x_b)$. For any pair $(X, Y) \in \mathscr{F}$ we have

$$\begin{aligned} \Lambda_{X,Y} \neq 0 &\Longleftrightarrow \exists\, u \in \mathbb{F} : Y = XA + uB \\ &\Longleftrightarrow Y = (u, X_{[1,\gamma-1]}) \text{ for some } u \in \mathbb{F}. \end{aligned}$$

In particular, u is uniquely determined by Y. On the other hand all this is equivalent to $\Lambda'_{\pi(X),\pi(Y)} \neq 0$, hence to the existence of some $u' \in \mathbb{F}$ such that $\pi(Y) = (u', \pi(X)_{[1,\gamma-1]})$. All this gives us

$$Y_{[2,\gamma]} = X_{[1,\gamma-1]} \Longrightarrow \pi(Y)_{[2,\gamma]} = \pi(X)_{[1,\gamma-1]} \quad \text{for all } (X, Y) \in \mathscr{F}.$$

In Lemma A.1 in Appendix we show that this implies that π is the identity. As a consequence we have $\Lambda' = \Lambda$ for all $\gamma \geqslant 1$. As for the rest of the proof notice that since $\ker B = \{0\}$, there is at most one edge $X \overset{\binom{u}{v}}{\to} Y$ for any given pair $(X, Y) \in \mathscr{F}$. Thus the entries of Λ and Λ' are monomials in W. As a consequence, we have for any $(X, Y) \in \mathscr{F}$

$$\Lambda_{X,Y} = W^\alpha \Longleftrightarrow \operatorname{wt}(XC + Y_1 D) = \alpha.$$

With the corresponding equivalence for $\Lambda'_{X,Y} = W^\alpha$ we finally arrive at

$$\operatorname{wt}\left((X, u)\begin{pmatrix} C \\ D \end{pmatrix}\right) = \operatorname{wt}\left((X, u)\begin{pmatrix} C' \\ D' \end{pmatrix}\right) \quad \text{for all } X \in \mathbb{F}^\gamma, u \in \mathbb{F}.$$

But then Lemma 5.4 below yields $\begin{pmatrix} C' \\ D' \end{pmatrix} = \begin{pmatrix} C \\ D \end{pmatrix} P$ for some permutation matrix $P \in \mathrm{Gl}_n(\mathbb{F})$. Hence $G' = GP$ meaning that the two matrices are monomially equivalent. □

It remains to prove the following.

Lemma 5.4. *Let $\mathbb{F}$ be any finite field and let $M, M' \in \mathbb{F}^{k\times n}$ be such that*

$$\operatorname{wt}(uM) = \operatorname{wt}(uM') \quad \textit{for all } u \in \mathbb{F}^k. \tag{5.2}$$

Then M and M' are monomially equivalent.

Proof.[1] Let us first assume that M has rank k. Then the assumption (5.2) implies that M' has rank k, too. Defining the block codes $\mathscr{B} = \operatorname{im} M$ and $\mathscr{B}' = \operatorname{im} M'$ we obtain a well-defined weight-preserving bijective linear transformation

$$\mathscr{B} \longrightarrow \mathscr{B}', \quad uM \longmapsto uM'.$$

By virtue of the Equivalence Theorem of MacWilliams, see for instance [8, Theorem 7.9.4] the two block codes are monomially equivalent. Thus there exist a permutation matrix P and a non-singular diagonal matrix R in $\mathrm{Gl}_n(\mathbb{F})$ such that $M' = MPR$.

Let now rank $M = r < k$ and assume without loss of generality that $M = \begin{pmatrix} M_1 \\ 0 \end{pmatrix}$ where $M_1 \in \mathbb{F}^{r\times n}$ has full row rank. Then $(0, u_2)M = 0$ for all $u_2 \in \mathbb{F}^{k-r}$ and (5.2) implies that $M' = \begin{pmatrix} M_1' \\ 0 \end{pmatrix}$ for some $M_1' \in \mathbb{F}^{r\times n}$ with full row rank. Now we have $\mathrm{wt}(u_1M_1) = \mathrm{wt}(u_1M_1')$ for all $u_1 \in \mathbb{F}^r$ and by the first part of this proof M_1 and M_1' are monomially equivalent. But then the same is true for M and M'. □

We close the section with briefly discussing the question whether there might exist a MacWilliams duality theorem for convolutional codes. For block codes this famous theorem states that the weight distribution of the dual code is fully determined by the weight distribution of the original code and a transformation formula is given, see, e.g., [12, Theorem 3.5.3]. For convolutional codes one might think of two possible generalizations of this result, either to the weight distribution Ω or to the adjacency matrix Λ. As we will describe next, both cases have already been touched upon in the literature. In [23] it has been shown that there does not exist a MacWilliams duality theorem for the weight distribution Ω of convolutional codes. Precisely, the following example has been presented. Consider $G_1 = [1, z, 1+z]$, $G_2 = [z, z, 1+z] \in \mathbb{F}_2^{1\times 3}$. Then one shows that the weight distributions of the two codes $\mathscr{C}_1 := \operatorname{im} G_1$ and $\mathscr{C}_2 = \operatorname{im} G_2$ coincide. Indeed, they are both given by $\Omega = \frac{L^2W^4}{1-LW^2}$. The dual codes are given by

$$\mathscr{C}_1^\perp = \operatorname{im}\begin{pmatrix} 1 & 1 & 1 \\ z & 1 & 0 \end{pmatrix}, \quad \mathscr{C}_2^\perp = \operatorname{im}\begin{pmatrix} 1 & 1 & 0 \\ 1+z & 0 & z \end{pmatrix},$$

and it turns out that they have different weight distributions

$$\Omega_{\mathscr{C}_1^\perp} = \frac{L^2W^2 + LW^3 + 2L^2W^3 - L^2W^5}{1 - LW - LW^2},$$

$$\Omega_{\mathscr{C}_2^\perp} = \frac{LW^2 + 3L^2W^3 - L^2W^5}{1 - LW - LW^3}.$$

As a consequence there cannot exist a MacWilliams transformation mapping the weight distribution of a given code onto the weight distribution of the dual without

[1] One can prove this result straightforwardly for the field $\mathbb{F}_2$. However, I wish to thank Gert Schneider for pointing out the connection to MacWilliams' Equivalence Theorem to me.

using any further information. The example even shows more. Since multiplication by z is weight-preserving, the mapping $uG \longmapsto uG'$ yields an isometry between the codes $\mathscr{C}$ and $\mathscr{C}'$. But obviously, the codes are not monomially equivalent, showing that there is no MacWilliams Equivalence Theorem for convolutional codes in this form (one would have to allow at least rescaling by powers of z in monomial equivalence). Let us now discuss the adjacency matrices of these codes. Since the two codes are not monomially equivalent we know from Theorem 5.3 that the generalized adjacency matrices of the two codes are not identical. Indeed, one computes

$$\Lambda_1 = \begin{pmatrix} 1 & W^2 \\ W^2 & W^2 \end{pmatrix}, \quad \Lambda_2 = \begin{pmatrix} 1 & W \\ W^3 & W^2 \end{pmatrix}.$$

Of course, the generalized adjacency matrices of the dual codes are different since the weight distributions are. They are given by

$$\Lambda_1^{\perp} = \begin{pmatrix} 1+W^3 & W+W^2 \\ W+W^2 & W+W^2 \end{pmatrix}, \quad \Lambda_2^{\perp} = \begin{pmatrix} 1+W^2 & 2W \\ 2W^2 & W+W^3 \end{pmatrix}.$$

At this point the question arises whether there exists a MacWilliams duality theorem for the adjacency matrices of convolutional codes. In the paper [1, Theorem 4] such a transformation has been established for codes with overall constraint length one. It is derived in a notation totally different from ours and results in essence in a separate formula for each entry of the adjacency matrix. Recall also from Remark 4.2 that for codes with overall constraint length one the adjacency matrix is even uniquely determined by the code. In a forthcoming paper [6] we will show that the formulas from [1] can be rewritten in compact form as

$$\Lambda^{\perp} = q^{-k}(1+(q-1)W)^n (H^{-1}\Lambda^{\mathsf{T}} H)|_{\frac{1-W}{1+(q-1)W}}, \tag{5.3}$$

where $M|_a$ denotes substitution of a for W in every entry of the matrix M and where the transformation matrix $H \in \mathbb{C}^{q\times q}$ is defined as $H_{X,Y} = \chi(XY)$ for all $X, Y \in \mathbb{F}_q$ with a non-trivial character χ on $\mathbb{F}_q$. This matrix also appears in the duality theorem for the complete weight enumerator for block codes, see [13, p. 144]. For $q = 2$ we have $H = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ and the formula can straightforwardly be verified for the two codes and their duals given above.

We strongly believe that this transformation can be generalized to codes with bigger overall constraint length. At least in the one-dimensional binary case with arbitrary overall constraint length we can establish the following support for this conjecture.

Corollary 5.5. *Let $\mathscr{C}, \mathscr{C}' \subseteq \mathbb{F}_2[z]^n$ be two binary one-dimensional codes. Then*

$$\overline{\Lambda}(\mathscr{C}) = \overline{\Lambda}(\mathscr{C}') \Longrightarrow \overline{\Lambda}(\mathscr{C}^{\perp}) = \overline{\Lambda}(\mathscr{C}'^{\perp}).$$

Proof. By virtue of Theorem 5.3 the assumption implies that $\mathscr{C}$ and $\mathscr{C}'$ are monomially equivalent. It is trivial to see that then also $\mathscr{C}^\perp$ and $\mathscr{C}'^\perp$ are monomially equivalent and thus Theorem 4.4 yields $\overline{\Lambda}(\mathscr{C}^\perp) = \overline{\Lambda}(\mathscr{C}'^\perp)$. □

Unfortunately, the corollary does not reveal a formula transforming $\overline{\Lambda}(\mathscr{C})$ into $\overline{\Lambda}(\mathscr{C}^\perp)$.

We close the section with the following.

Conjecture 5.6. *Let* $\mathscr{C}, \mathscr{C}' \subseteq \mathbb{F}[z]^n$ *be codes such that* $\overline{\Lambda}(\mathscr{C}) = \overline{\Lambda}(\mathscr{C}')$. *Then* $\overline{\Lambda}(\mathscr{C}^\perp) = \overline{\Lambda}(\mathscr{C}'^\perp)$.

6. Open problems

With this paper we want to initiate an investigation of the weight distribution and weight preserving maps for convolutional codes. The central object of our approach is the adjacency matrix of the associated state diagram. In Theorem 5.3 we showed that for one-dimensional binary codes this matrix uniquely determines the code up to monomial equivalence. One immediately wonders whether this is true for one-dimensional codes over arbitrary fields as well. From block code theory it is known, however, that such a result cannot be expected for *higher-dimensional* codes. It would be helpful to see some examples with positive overall constraint length. Another open problem is the issue of MacWilliams Equivalence for convolutional codes. This expression refers to the study of isometries (weight-preserving isomorphisms) between codes. As far as we know it is an open question whether isometries between convolutional codes can be described explicitly. Finally, of course there remains Conjecture 5.6. While for codes with overall constraint length one a transformation between the adjacency matrices of a code and its dual has been derived in [1], and can be written as in (5.3), the general case has to remain open for future research.

Appendix A

In the following lemma we use again the notation $X_{[a,b]} := (X_a, X_{a+1}, \ldots, X_b)$ for $X = (X_1, \ldots, X_\gamma) \in \mathbb{F}^\gamma$ and all $1 \leqslant a \leqslant b \leqslant \gamma$. For $X_{[a,a]}$ we write, of course, simply X_a.

Lemma A.1. *Let* $\mathbb{F} = \mathbb{F}_2$ *and* $\gamma \geqslant 2$. *Furthermore, let* $\pi : \mathbb{F}^\gamma \longrightarrow \mathbb{F}^\gamma$ *be a bijective map with* $\pi(0) = 0$ *and satisfying*

$$\pi(u, X_{[1,\gamma-1]})_{[2,\gamma]} = \pi(X)_{[1,\gamma-1]} \quad \textit{for all } X \in \mathbb{F}^\gamma \textit{ and all } u \in \mathbb{F}. \tag{A.1}$$

Then π *is the identity map.*

Proof. Denote by $e_1, \ldots, e_\gamma$ the standard basis vectors on $\mathbb{F}^\gamma$.

(1) Using $X = 0$ and $u = 1$ we obtain $\pi(e_1)_{[2,\gamma]} = \pi(0)_{[1,\gamma-1]} = (0, \ldots, 0)$, thus $\pi(e_1) = (a, 0, \ldots, 0)$. Bijectivity of π implies $a = 1$, thus $\pi(e_1) = e_1$.

(2) Using $X = e_\gamma$ and $u = 0$ we obtain

$$\pi(u, X_{[1,\gamma-1]})_{[2,\gamma]} = \pi(0)_{[2,\gamma]} = 0 = \pi(e_\gamma)_{[1,\gamma-1]},$$

thus $\pi(e_\gamma) = (0, \ldots, 0, a)$ and again bijectivity of π implies $\pi(e_\gamma) = e_\gamma$.

(3) Now we proceed by induction. Assume that there is some $r \geqslant 1$ such that $\pi(X) = X$ for all $X \in \mathbb{F}^\gamma$ satisfying $\operatorname{wt}(X) \leqslant r$ and $X_1 = 1$. By (1) this is true for $r = 1$. Then we have to show
(i) $\pi(\widetilde{X}) = \widetilde{X}$ for all $\widetilde{X}$ such that $\operatorname{wt}(\widetilde{X}) \leqslant r$,
(ii) $\pi(\widetilde{X}) = \widetilde{X}$ for all $\widetilde{X}$ such that $\operatorname{wt}(\widetilde{X}) \leqslant r + 1$ and $\widetilde{X}_1 = 1$.

Ad (i): Pick $X \in \mathbb{F}^\gamma$ such that $\operatorname{wt}(X) \leqslant r$ and $X_1 = 1$. Put $X^{(1)} = (0, X_{[1,\gamma-1]})$. Then $\pi(X^{(1)})_{[2,\gamma]} = \pi(X)_{[1,\gamma-1]} = X_{[1,\gamma-1]}$, thus $\pi(X^{(1)}) = (a_1, X_{[1,\gamma-1]})$. Put now $X^{(i)} = (0, \ldots, 0, X_{[1,\gamma-i]}) \in \mathbb{F}^\gamma$. We proceed by induction on i. Thus by hypothesis we may assume

$$\pi(X^{(i)}) = (a_i, \ldots, a_1, X_{[1,\gamma-i]}). \tag{A.2}$$

Then $X^{(i+1)} = (0, X^{(i)}_{[1,\gamma-1]})$ and thus

$$\pi(X^{(i+1)})_{[2,\gamma]} = \pi(X^{(i)})_{[1,\gamma-1]} = (a_i, \ldots, a_1, X_{[1,\gamma-i-1]}).$$

Therefore $\pi(X^{(i+1)}) = (a_{i+1}, \ldots, a_1, X_{[1,\gamma-i-1]})$. Hence (A.2) holds true for all $i = 1, \ldots, \gamma - 1$. Now $X^{(\gamma-1)} = e_\gamma$. Hence by (2) of this proof $e_\gamma = \pi(X^{(\gamma-1)}) = (a_{\gamma-1}, \ldots, a_1, X_1)$. This implies $a_1 = \cdots = a_{\gamma-1} = 0$ and hence $\pi(X^{(i)}) = X^{(i)}$ for all $i = 1, \ldots, \gamma - 1$. Since each $\widetilde{X} \in \mathbb{F}^\gamma$ such that $\operatorname{wt}(\widetilde{X}) \leqslant r$ is of the form $X^{(i)}$ for a suitable X satisfying $\operatorname{wt}(X) \leqslant r$ and $X_1 = 1$, this proves (i).

Ad (ii): Let $\widetilde{X} \in \mathbb{F}^\gamma$ such that $\widetilde{X}_1 = 1$ and $\operatorname{wt}(\widetilde{X}) \leqslant r + 1$. Then $\widetilde{X} = (1, X_{[1,\gamma-1]})$ for some $X \in \mathbb{F}^\gamma$ such that $\operatorname{wt}(X) = \operatorname{wt}(X_{[1,\gamma-1]}) \leqslant r$. By part (i) we know that $\pi(X) = X$ as well as

$$\pi(0, X_{[1,\gamma-1]}) = (0, X_{[1,\gamma-1]}). \tag{A.3}$$

Now (A.1) yields $\pi(\widetilde{X})_{[2,\gamma]} = \pi(X)_{[1,\gamma-1]} = X_{[1,\gamma-1]}$. Hence $\pi(\widetilde{X}) = (a, X_{[1,\gamma-1]})$ and bijectivity of π along with (A.3) yields $a = 1$. Thus $\pi(\widetilde{X}) = \widetilde{X}$. □

Acknowledgments

I would like to thank the anonymous reviewers as well as Gert Schneider for their helpful comments.

References

[1] K.A.S. Abdel-Ghaffar, On unit constrained-length convolutional codes, IEEE Trans. Inform. Theory IT-38 (1992) 200–206.

[2] P. Fitzpatrick, G.H. Norton, Linear recurring sequences and the path weight enumerator of a convolutional code, Electron. Lett. 27 (1991) 98–99.

[3] G.D. Forney Jr., Convolutional codes I: Algebraic structure, IEEE Trans. Inform. Theory IT-16 (1970) 720–738. (see also corrections in IEEE Trans. Inform. Theory 17 (1971) 360).

[4] G.D. Forney Jr., Structural analysis of convolutional codes via dual codes, IEEE Trans. Inform. Theory IT-19 (1973) 512–518.

[5] G.D. Forney Jr., Minimal bases of rational vector spaces, with applications to multivariable linear systems, SIAM J. Contr. 13 (1975) 493–520.

[6] H. Gluesing-Luerssen, G. Schneider, On the MacWilliams duality for convolutional codes, in preparation.

[7] S. Höst, R. Johannesson, V.V. Zyablov, Woven convolutional codes I: Encoder properties, IEEE Trans. Inform. Theory IT-48 (2002) 149–161.

[8] W.C. Huffman, V. Pless, Fundamentals of Error-Correcting Codes, Cambridge University Press, Cambridge, 2003.

[9] R. Johannesson, K.S. Zigangirov, Fundamentals of Convolutional Coding, IEEE Press, New York, 1999.

[10] J. Justesen, E. Paaske, M. Ballan, Quasi-cyclic unit memory convolutional codes, IEEE Trans. Inform. Theory IT-36 (1990) 540–547.

[11] S. Lin, D.J. Costello Jr., Error Control Coding: Fundamentals and Applications, Prentice Hall, 1983.

[12] J.H.v. Lint, Introduction to Coding Theory, third ed., Springer, 1999.

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

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

[15] R.J. McEliece, The algebraic theory of convolutional codes, in: V. Pless, W. Huffman (Eds.), Handbook of Coding Theory, vol. 1, Elsevier, Amsterdam, 1998, pp. 1065–1138.

[16] R.J. McEliece, How to compute weight enumerators for convolutional codes, in: M. Darnell, B. Honory (Eds.), Communications and Coding (P.G. Farrell 60th birthday celebration), Wiley, New York, 1998, pp. 121–141.

[17] M. Motani, C. Heegard, Computing weight distributions of convolutional codes via shift register synthesis, Applied Algebra, Algorithms and Error-Correcting Codes; 13th Intern. Symp. AAECC-13 (Honolulu/USA), Lecture Notes in Computer Science LN 1719, Springer, 1999, pp. 314–323.

[18] I. Onyszchuk, Finding the complete path and weight enumerators of convolutional codes, JPL TDA Progress Report 42-100, 1990.

[19] J. Rosenthal, Connections between linear systems and convolutional codes, in: B. Marcus, J. Rosenthal (Eds.), Codes, Systems, and Graphical Models, Springer, Berlin, 2001, pp. 39–66.

[20] J. Rosenthal, J.M. Schumacher, E.V. York, On behaviors and convolutional codes, IEEE Trans. Inform. Theory IT-42 (1996) 1881–1891.

[21] J. Rosenthal, R. Smarandache, Maximum distance separable convolutional codes, Appl. Algebra Eng. Comm. Comput. 10 (1999) 15–32.

[22] J. Rosenthal, E.V. York, BCH convolutional codes, IEEE Trans. Inform. Theory IT-45 (1999) 1833–1844.

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

[24] E.D. Sontag, Mathematical Control Theory; Deterministic Finite Dimensional Systems, Springer, 1990.

[25] A.J. Viterbi, Convolutional codes and their performance in communication systems, IEEE Trans. Commun. Technol. COM-19 (1971) 751–772.