Trellis Complexity Bounds for Decoding Linear Block Codes

A. B. Kiely, S. Dolinar, and L. Ekroot
Communications Systems Research Section

R. J. McEliece
California Institute of Technology
and
Communications Systems Research Section

W. Lin
California Institute of Technology

We consider the problem of finding a trellis for a linear block code that minimizes one or more measures of trellis complexity. The domain of optimization may be different permutations of the same code or different codes with the same parameters. Constraints on trellises, including relationships between the minimal trellis of a code and that of the dual code, are used to derive bounds on complexity. We define a partial ordering on trellises: If a trellis is optimum with respect to this partial ordering, it has the desirable property that it simultaneously minimizes all of the complexity measures examined. We examine properties of such optimal trellises and give examples of optimal permutations of codes, most notably the (48,24,12) quadratic residue code.

I. Introduction

A minimal trellis is a labeled graph that can be used as a template for encoding or decoding. In [6], we examined properties of trellises for fixed permutations of a code. A code's minimal trellis is unique as long as the ordering of the code's symbols is fixed. However, different permutations of the symbols yield different minimal trellises. An optimum minimal trellis for the code is one that minimizes a suitable measure of trellis complexity over all possible permutations of the code. There are no known efficient algorithms for constructing optimum minimal trellises.

We expand the results of [6] to examine the problem of finding a permutation that minimizes one or more trellis complexity measures. We extend these results to the problem of finding a minimal complexity trellis over all codes with the same parameters. We identify certain sufficient conditions for a code or a permutation to simultaneously minimize all of the complexity measures.

In Section II, we discuss dimension/length profiles of a code [3,11], which are equivalent to Wei's generalized Hamming weights [12]. The dimension/length profiles are used to derive some straightforward complexity bounds. We summarize some properties of these profiles, including duality relationships.

We define a partial ordering on minimal trellises in Section III. If the minimal trellises for two codes are comparable in terms of this partial ordering, then each of the complexity measures for one trellis is bounded by the same measure evaluated for the other trellis. This partial ordering can sometimes be used to identify the permutation of a code with the least (or most) complex minimal trellis, or the code with the lowest (or highest) complexity trellis of all codes with the same parameters. The extremal codes determined by this partial ordering turn out to meet the complexity bounds described in Section II. We illustrate certain properties and give examples of such permutations and codes.

II. Trellis Complexity Bounds

The minimal trellis results of [6] assume a fixed coordinate ordering for the code. However, the trellis structure and, hence, trellis complexity are different for different permutations of the code coordinates. Massey refers to the procedure of reordering the code symbols to reduce the trellis complexity as "the *art* of trellis decoding" [9, p. 9].

In this section, we identify code parameters that affect the possible trellis complexity, describe upper and lower bounds based on these parameters, and illustrate properties of certain codes that have low complexity trellises. Our results apply to a gamut of possible complexity measures introduced in [6]: the maximum vertex (state) and edge dimensions $(s_{\max}, e_{\max})$, the total vertex and edge spans (ν, ε), and the total numbers of vertices, edges, and mergers (V, E, M). In this article, all theorems are presented without proof; proofs are supplied in a separate article.[1]

First, some notation: Let $\mathcal{S}_n$ denote the set of all permutations of $\{1, 2, \cdots, n\}$, and for any $\pi \in \mathcal{S}_n$, let $\mathcal{C}\pi$ denote the code $\mathcal{C}$ with coordinates reordered according to π. Because the code and dual code provide symmetric constraints on the code's minimal trellis, the complexity bounds are developed by considering the characteristics of both the code and its dual. We refer to an (n, k, d) code over $GF(q)$ with dual distance $d^{\perp}$ as an $(n, k, d, d^{\perp})$ code.

A. Bounds Relating One Complexity Measure to Another

The following lemma arises from the definitions of $s_{\max}$ and $e_{\max}$ and from the fact that the vertex and edge dimensions, v_i and e_i, change by no more than one unit from one index to the next.

Lemma 1. The vertex dimensions and edge dimensions are upper bounded by

$$v_i \le \min\{i, n-i, s_{\max}\}, \qquad 0 \le i \le n$$

$$e_i \le \min\{i, n+1-i, e_{\max}\}, \qquad 1 \le i \le n$$

Summing the inequalities in Lemma 1 leads to the following bounding relationships among the complexity measures.

Theorem 1. The total complexity measures ν, ε, V, E are upper bounded in terms of the maximum complexity measures $s_{\max}$, $e_{\max}$ by

$$\nu \le s_{\max}(n - s_{\max}) \tag{1}$$

[1] A. B. Kiely, S. Dolinar, R. J. McEliece, L. Ekroot, and W. Lin, "Trellis Decoding Complexity of Linear Block Codes," submitted to *IEEE Trans. Inform. Theory.*

$$\varepsilon \leq e_{\max}(n + 1 - e_{\max}) \tag{2}$$

$$V \leq \left[n + \frac{q+1}{q-1} - 2s_{\max} \right] q^{s_{\max}} - \frac{2}{q-1} \tag{3}$$

$$E \leq \left[n + \frac{2q}{q-1} - 2e_{\max} \right] q^{e_{\max}} - \frac{2q}{q-1} \tag{4}$$

Since the average edge dimension over all stages is ε/n and the average vertex dimension over the last n depths is ν/n, loose lower bounds on V and E can be obtained from Jensen's inequality.

Theorem 2. The total complexity measures V, E are lower bounded in terms of the total span length complexity measures ν, ε by

$$V \geq 1 + nq^{\nu/n}$$

$$E \geq nq^{\varepsilon/n}$$

There are also tighter lower bounds on V and E in terms of ν and ε.

Theorem 3. Given a total span length ν, or equivalently ε, let $\Delta\varepsilon = \varepsilon - e^-(n + 1 - e^-)$ and $\Delta\nu = \nu - s^-(n - s^-)$, where $e^- \leq (n+1)/2$ and $s^- \leq n/2$ are the largest integers such that $\Delta\varepsilon \geq 0$ and $\Delta\nu \geq 0$. Then

$$V \geq \left[n + \frac{q+1}{q-1} - 2s^- \right] q^{s^-} - \frac{2}{q-1} + (q-1)q^{s^-}\Delta\nu$$

$$E \geq \left[n + \frac{2q}{q-1} - 2e^- \right] q^{e^-} - \frac{2q}{q-1} + (q-1)q^{e^-}\Delta\varepsilon$$

This theorem follows from the observation that, for a given ν or ε, a vertex or edge dimension profile such as the one in Fig. 1 minimizes V or E. Notice the similarity of these lower bounds in terms of s^- and e^- with the corresponding upper bounds, Eqs. (3) and (4), in terms of $s_{\max}$ and $e_{\max}$.

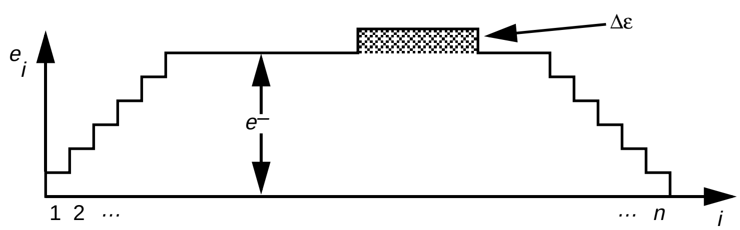

Fig. 1. An edge dimension profile that minimizes *E* subject to a constraint on total edge span ε.

B. Complexity Lower Bounds Based on MSGM Span Length

Every row of a generator matrix for an $(n, k, d, d^{\perp})$ code must have edge-span length $\varepsilon_i \geq d$ and vertex-span length $\nu_i \geq d - 1$. Applying this simple bound to both the code and the dual code and using the fact that $\nu^{\perp} = \nu = \varepsilon - k$ leads to the following lower bounds on the span length complexity measures ν and ε.

Theorem 4. The total lengths ν and ε of the vertex spans and edge spans for any $(n, k, d, d^{\perp})$ code are lower bounded by

$$\nu \geq \max\left\{k(d-1), (n-k)(d^{\perp}-1)\right\}$$

$$\varepsilon \geq k + \max\left\{k(d-1), (n-k)(d^{\perp}-1)\right\}$$

Applying the Singleton bound to the inequalities in this theorem gives the weaker bounds $\nu \geq (d-1)(d^{\perp}-1)$ and $\varepsilon \geq k + (d-1)(d^{\perp}-1)$.

We say that a code meeting the bounds in Theorem 4 with equality is a minimal span code. An example is the $(n, 1, n, 2)$ repetition code. To construct a nondegenerate $(n, k, d, 2)$ binary minimal span code for any $d > 2$ and $n \geq d + (k-1)\lceil d/2 \rceil$, let the first row of the minimal span generator matrix (MSGM) be

$$\underbrace{111\cdots1}_{d}\underbrace{000\cdots0}_{n-d}$$

and form each successive row by cyclically shifting the previous row at least $\lceil d/2 \rceil$ positions but not more than d positions to the right, such that the total of all the shifts is $n - d$. The dual of a minimal span code is also a minimal span code. These codes are not usually good in terms of distance, though they have very low complexity trellises.

The span length bounds in Theorem 4, combined with the bounds of Eqs. (1) and (2), lead to lower bounds on the complexity measures $s_{\max}$, $e_{\max}$ for any $(n, k, d, d^{\perp})$ code:

$$s_{\max}(n - s_{\max}) \geq \max\left\{k(d-1), (n-k)(d^{\perp}-1)\right\}$$

$$e_{\max}(n + 1 - e_{\max}) \geq k + \max\left\{k(d-1), (n-k)(d^{\perp}-1)\right\}$$

A slightly weaker version of this bound on $s_{\max}$ has been proved for both linear and nonlinear codes [8]. This bound implies, for instance, that the average edge dimension $e_{\max}$ can never be lower than the asymptotic coding gain kd/n. We can also obtain bounds on V and E for any $(n, k, d, d^{\perp})$ code by substituting the right-hand sides of the bounds in Theorem 4 for ν and ε in Theorems 2 and 3.

C. Dimension/Length Profiles

We can see from the definitions of the complexity measures in [6] that a permutation of $\mathcal{C}$ that makes f_i and p_i large (small) wherever possible will produce a low (high) complexity trellis. It is useful, therefore, to find bounds on these quantities.

The support of a vector x is the set of nonzero positions in x. The support of a set of vectors is the union of the individual supports.

Definition 1. For a given code $\mathcal{C}$ and any $0 \leq i \leq n$, let $K_i(\mathcal{C})$ be the maximum dimension of a linear subcode of $\mathcal{C}$ having support whose size is no greater than i. The set $\{K_i(\mathcal{C}), i = 0, \cdots, n\}$ is called the dimension/length profile (DLP) [3,11].

The DLP and similar concepts have been used recently by many other researchers to bound trellis complexity. Extensive bibliographies are given in [3] and Kiely et al.[2] Since the past and future subcodes $\mathcal{P}_i$ and $\mathcal{F}_i$ are subcodes of $\mathcal{C}$ with support size no larger than i and $n-i$, respectively, the past and future subcode dimensions are bounded by the DLP

$$p_i \leq \max_{\pi \in \mathcal{S}_n} p_i(\mathcal{C}\pi) = K_i(\mathcal{C}) \tag{5}$$

$$f_i \leq K_{n-i}(\mathcal{C}) \tag{6}$$

These bounds, which also appeared in [4, Eq. (1.4)], are tight in the following sense: For any i, there exists a permuted version of $\mathcal{C}$ that meets the bound of Eq. (5) and one that meets Eq. (6), though it may not be possible to meet both simultaneously. The DLP of a code can be used to lower bound the trellis complexity for any permutation of that code, as we shall see in Section II.E.

Since each $K_i(\mathcal{C})$ is associated with a linear subcode of $\mathcal{C}$, we can use bounds on the best possible linear codes (i.e., codes with the largest possible minimum distance) to upper bound the DLP:

Theorem 5. For an $(n, k, d, d^\perp)$ code $\mathcal{C}$ and any $0 \leq i \leq n$,

$$p_i \leq K_i(\mathcal{C}) \leq \overline{K}_i(n, k, d, d^\perp)$$

$$f_i \leq K_{n-i}(\mathcal{C}) \leq \overline{K}_{n-i}(n, k, d, d^\perp)$$

where

$$\overline{K}_i(n, k, d, d^\perp) \triangleq \min[k_{\max}(i, d), k - n + i + k_{\max}(n - i, d^\perp)]$$

and $k_{\max}(m, d)$ is the largest possible dimension for any q-ary linear block code of length m and minimum distance d. The set $\left\{\overline{K}_i(n, k, d, d^\perp), i = 0, \ldots, n\right\}$ is called the upper dimension/length profile (UDLP) for the code parameters $(n, k, d, d^\perp)$.

Bounds based on the UDLP may be loose, as it may not be possible for a single $(n, k, d, d^\perp)$ code and its dual to both have a series of subcodes, all with the maximum code dimensions. However, these bounds are important practically, because much data about the best possible codes have been tabulated [1] and, in many cases, the UDLP bounds can be achieved with equality.

Since for any (n, k) code $\mathcal{C}$, p_i and f_i both reach maximum values of k ($f_0 = k$ and $p_n = k$) and can fall from these values at a maximum rate of one unit per trellis stage, p_i and f_i are lower bounded as follows:

$$K_i(\mathcal{C}) \geq p_i \geq \underline{K}_i(n, k) \triangleq \max(0, k - n + i) \tag{7}$$

$$K_{n-i}(\mathcal{C}) \geq f_i \geq \underline{K}_{n-i}(n, k) = \max(0, k - i) \tag{8}$$

The set $\{\underline{K}_i(\mathcal{C}), i = 0, 1, \cdots, n\}$ is called the lower dimension/length profile (LDLP) for the code parameters (n, k). The LDLP stays at 0 until the last possible depth before it can rise linearly at the rate of one dimension per depth to reach its final value of k at depth n. The LDLP can be used to upper bound the complexity of a minimal trellis for an arbitrary (n, k) code.

[2] Ibid.

D. Properties of Dimension/Length Profiles

The DLPs possess many of the same properties as the past and future subcode dimensions that they bound. For example, the monotonicity and unit increment properties of $\{p_i\}$ also hold for $K_i(\mathcal{C})$, $\overline{K}_i(\mathcal{C})$, and $\underline{K}_i(\mathcal{C})$: The increments $\overline{K}_{i+1}(n,k,d,d^{\perp})-\overline{K}_i(n,k,d,d^{\perp})$, $K_{i+1}(\mathcal{C})-K_i(\mathcal{C})$, and $\underline{K}_{i+1}(n,k)-\underline{K}_i(n,k)$ must equal 0 or 1 for all i. Similarly, duality properties can be easily extended.

There is a convenient relationship between the DLP of a code and that of its dual, stated in [4, Eq. (1.12)] and [3, Theorem 3], which is equivalent to the duality relationship for generalized Hamming weights [12, Theorem 3]. Similar relationships hold for the upper and lower dimension/length profiles:

Lemma 2. For all $0 \le i \le n$, the DLP, UDLP, and LDLP satisfy the following duality relationships:

$$\begin{aligned} K_i(\mathcal{C}^{\perp}) &= i - k + K_{n-i}(\mathcal{C}) \\ \overline{K}_i(n, n-k, d^{\perp}, d) &= i - k + \overline{K}_{n-i}(n,k,d,d^{\perp}) \\ \underline{K}_i(n, n-k) &= i - k + \underline{K}_{n-i}(n,k) \end{aligned}$$

E. Complexity Bounds From Dimension/Length Profiles

The DLP bounds, Eqs. (5) and (6), combined with the complexity definitions lead to simple bounds on trellis complexity that are useful when the DLP of a given code is known. These bounds can be tightened slightly by using the additional fact that the vertex and edge dimensions must be nonnegative everywhere.

Theorem 6. The complexity measures for the minimal trellis $\mathcal{T}(\mathcal{C}\pi)$ corresponding to any permutation π of a given (n, k) code $\mathcal{C}$ are lower bounded by

$$s_{\max}(\mathcal{C}\pi) \ge \max_{i\in[0,n]} (k - K_i(\mathcal{C}) - K_{n-i}(\mathcal{C})) \tag{9}$$

$$e_{\max}(\mathcal{C}\pi) \ge \max_{i\in[1,n]} (k - K_{i-1}(\mathcal{C}) - K_{n-i}(\mathcal{C})) \tag{10}$$

$$\varepsilon(\mathcal{C}\pi) \ge \sum_{i=0}^{n} \max\{0, k - K_i(\mathcal{C}) - K_{n-i}(\mathcal{C})\} \tag{11}$$

$$V(\mathcal{C}\pi) \ge \sum_{i=0}^{n} q^{\max\{0, k - K_i(\mathcal{C}) - K_{n-i}(\mathcal{C})\}} \tag{12}$$

$$E(\mathcal{C}\pi) \ge \sum_{i=1}^{n} q^{\max\{0, k - K_{i-1}(\mathcal{C}) - K_{n-i}(\mathcal{C})\}} \tag{13}$$

$$M(\mathcal{C}\pi) \ge \frac{1}{q} \sum_{i=1}^{n} [K_i(\mathcal{C}) - K_{i-1}(\mathcal{C})] q^{\max\{0, k - K_{i-1}(\mathcal{C}) - K_{n-i}(\mathcal{C})\}} \tag{14}$$

The DLP bound, Eq. (9), on state complexity has been derived in [3,11].[3] Some of the bounds in Theorem 6 can be improved slightly when $\mathcal{C}$ is nondegenerate, because this condition implies that $e_i \geq 1$.

The UDLP bound (Theorem 5) leads to similar lower bounds on trellis complexity that apply to all codes with given code parameters.

Theorem 7. The complexity measures for the minimal trellis $\mathcal{T}(\mathcal{C})$ representing any $(n, k, d, d^{\perp})$ code $\mathcal{C}$ are lower bounded by

$$s_{\max}(\mathcal{C}) \geq \max_{i \in [0,n]} \left[k - \overline{K}_i(n,k,d,d^{\perp}) - \overline{K}_{n-i}(n,k,d,d^{\perp})\right] \tag{15}$$

$$e_{\max}(\mathcal{C}) \geq \max_{i \in [1,n]} \left[k - \overline{K}_{i-1}(n,k,d,d^{\perp}) - \overline{K}_{n-i}(n,k,d,d^{\perp})\right] \tag{16}$$

$$\varepsilon(\mathcal{C}) \geq \sum_{i=1}^{n} \max\left\{0, k - \overline{K}_{i-1}(n,k,d,d^{\perp}) - \overline{K}_{n-i}(n,k,d,d^{\perp})\right\} \tag{17}$$

$$V(\mathcal{C}) \geq \sum_{i=0}^{n} q^{\max\left\{0, k - \overline{K}_i(n,k,d,d^{\perp}) - \overline{K}_{n-i}(n,k,d,d^{\perp})\right\}} \tag{18}$$

$$E(\mathcal{C}) \geq \sum_{i=1}^{n} q^{\max\left\{0, k - \overline{K}_{i-1}(n,k,d,d^{\perp}) - \overline{K}_{n-i}(n,k,d,d^{\perp})\right\}} \tag{19}$$

$$M(\mathcal{C}) \geq \frac{1}{q} \sum_{i=1}^{n} \left[\overline{K}_i(n,k,d,d^{\perp}) - \overline{K}_{i-1}(n,k,d,d^{\perp})\right] q^{\max\left\{0, k - \overline{K}_{i-1}(n,k,d,d^{\perp}) - \overline{K}_{n-i}(n,k,d,d^{\perp})\right\}} \tag{20}$$

Finally, the LDLP bounds, Eqs. (7) and (8), lead immediately to simple explicit upper bounds on the various complexity measures that apply to all codes with a given length and dimension.

Theorem 8. The complexity measures for the minimal trellis $\mathcal{T}(\mathcal{C})$ corresponding to any (n, k) code $\mathcal{C}$ are upper bounded by

$$s_{max}(\mathcal{C}) \leq \min(k, n-k) \tag{21}$$

$$e_{max}(\mathcal{C}) \leq \min(k, n-k+1) \tag{22}$$

$$\varepsilon(\mathcal{C}) \leq k(n-k+1) \tag{23}$$

$$V(\mathcal{C}) \leq \left[n + \frac{q+1}{q-1} - 2\min(k, n-k)\right] q^{\min(k,n-k)} - \frac{2}{q-1} \tag{24}$$

[3] A. Lafourcade and A. Vardy, “Lower Bounds on Trellis Complexity of Block Codes,” submitted to *IEEE Trans. Inform. Theory.*

$$E(\mathcal{C}) \leq \left[n + \frac{2q}{q-1} - 2\min(k, n-k+1)\right] q^{\min(k,n-k+1)} - \frac{2q}{q-1} \tag{25}$$

$$M(\mathcal{C}) \leq \left[\frac{1}{q-1} + \max(0, 2k-n)\right] q^{\min(k,n-k)} - \frac{1}{q-1} \tag{26}$$

The inequality, Eq. (21), is the well-known Wolf bound [13]. Note that Eqs. (2) through (4) are tighter than Eqs. (23) through (25), except when Eqs. (21) and (22) are met with equality, in which case the bounds are the same.

III. Best and Worst Trellises

A. Uniform Comparability

In general, to determine which of two minimal trellises is less complex, we must first choose the relevant complexity measure. However, in some cases, one trellis may be simpler than another at every stage and depth with respect to all of the complexity measures simultaneously.

Definition 2. For two (n, k) codes $\mathcal{C}_1, \mathcal{C}_2$ having minimal trellises $\mathcal{T}(\mathcal{C}_1)$ and $\mathcal{T}(\mathcal{C}_2)$, we say that $\mathcal{T}(\mathcal{C}_1) \preceq \mathcal{T}(\mathcal{C}_2)$ if $p_i(\mathcal{C}_1) \geq p_i(\mathcal{C}_2)$ and $f_i(\mathcal{C}_1) \geq f_i(\mathcal{C}_2)$ for all i. If either $\mathcal{T}(\mathcal{C}_1) \preceq \mathcal{T}(\mathcal{C}_2)$ or $\mathcal{T}(\mathcal{C}_2) \preceq \mathcal{T}(\mathcal{C}_1)$, then the two trellises are uniformly comparable.

The binary relation $\preceq$ defines a partial ordering on any set of codes with the same length and dimension. If $\mathcal{T}(\mathcal{C}_1) \preceq \mathcal{T}(\mathcal{C}_2)$ and $\mathcal{T}(\mathcal{C}_2) \preceq \mathcal{T}(\mathcal{C}_1)$, then the two minimal trellises have equivalent complexity, though they may not have the same structure.

Note that if $\mathcal{T}(\mathcal{C}_1) \preceq \mathcal{T}(\mathcal{C}_2)$, then at every depth and stage, $\mathcal{T}(\mathcal{C}_1)$ has no more vertices or edges than $\mathcal{T}(\mathcal{C}_2)$, but the converse is not necessarily true. We define comparability in terms of past and future dimensions rather than edge and vertex dimensions because this gives a closer connection to the dimension/length profiles.

Theorem 9. If $\mathcal{T}(\mathcal{C}_1) \preceq \mathcal{T}(\mathcal{C}_2)$, then all of the following trellis complexity measures for $\mathcal{C}_1$ are upper bounded by those for $\mathcal{C}_2$:

(1) Maximum state complexity: $s_{\max}(\mathcal{C}_1) \leq s_{\max}(\mathcal{C}_2)$

(2) Total span lengths: $\varepsilon(\mathcal{C}_1) \leq \varepsilon(\mathcal{C}_2)$, $\nu(\mathcal{C}_1) \leq \nu(\mathcal{C}_2)$

(3) Total vertices: $V(\mathcal{C}_1) \leq V(\mathcal{C}_2)$

(4) Total edges: $E(\mathcal{C}_1) \leq E(\mathcal{C}_2)$

(5) Total number of path mergers: $M(\mathcal{C}_1) \leq M(\mathcal{C}_2)$

If two minimal trellises are not uniformly comparable, then the choice of the less complex trellis may depend on which of the complexity measures is used as the criterion.

Uniform comparability is a very strong property that is not guaranteed to exist between any two trellises. Our motivation for defining it and studying its consequences lies in the correspondingly strong results obtained for the problem of finding a minimal trellis in the first place, i.e., finding the least complex trellis that represents a fixed permutation of a fixed code. As shown by McEliece,[4] the minimal trellis is uniformly less complex at every stage and depth than any other trellis that represents the code.

[4] R. J. McEliece, "On The BCJR Trellis for Linear Block Codes," submitted to *IEEE Trans. Inform. Theory.*

We define four categories of best and worst minimal trellises based on uniform comparability:

Definition 3. For a fixed code $\mathcal{C}$, a permutation π^* and the corresponding minimal trellis $\mathcal{T}(\mathcal{C}\pi^*)$ are

(1) Uniformly efficient if $\mathcal{T}(\mathcal{C}\pi^*) \preceq \mathcal{T}(\mathcal{C}\pi)$ for all $\pi \in \mathcal{S}_n$

(2) Uniformly inefficient if $\mathcal{T}(\mathcal{C}\pi) \preceq \mathcal{T}(\mathcal{C}\pi^*)$ for all $\pi \in \mathcal{S}_n$

Definition 4. An $(n, k, d, d^\perp)$ code $\mathcal{C}^*$ and its corresponding minimal trellis $\mathcal{T}(\mathcal{C}^*)$ is

(2) Uniformly concise if $\mathcal{T}(\mathcal{C}^*) \preceq \mathcal{T}(\mathcal{C})$ for all $(n, k, d, d^\perp)$ codes $\mathcal{C}$

(2) Uniformly full if $\mathcal{T}(\mathcal{C}) \preceq \mathcal{T}(\mathcal{C}^*)$ for all (n, k) codes $\mathcal{C}$

If a minimal trellis is uniformly efficient or uniformly concise, we can drop the qualifier "minimal" and refer to it simply as a uniformly efficient trellis or a uniformly concise trellis, respectively. As shown later in Theorem 17, the two worst-case categories, uniformly inefficient and uniformly full, turn out to be equivalent.

The inclusion of $d^\perp$ in the above definition elucidates symmetries that are hidden by consideration of only n, k, and d. First, it preserves duality relationships, as we shall see below in Theorem 10. Second, from a practical point of view, d and $d^\perp$ have symmetric impact on the potential trellis complexity. There also appears to be a deep connection between d and $d^\perp$ for good codes: Often when d is large, $d^\perp$ must also be large, e.g., the extended Hamming codes and maximum distance separable (MDS) codes.

A direct consequence of [6, Theorem 1] is that uniform comparability of codes and their duals are equivalent:

Theorem 10. $\mathcal{T}(\mathcal{C}_1) \preceq \mathcal{T}(\mathcal{C}_2)$ if and only if $\mathcal{T}(\mathcal{C}_1^\perp) \preceq \mathcal{T}(\mathcal{C}_2^\perp)$. Consequently,

(1) A permutation π^* is uniformly efficient for $\mathcal{C}$ if and only if π^* is uniformly efficient for $\mathcal{C}^\perp$.

(2) A permutation π^* is uniformly inefficient for $\mathcal{C}$ if and only if π^* is uniformly inefficient for $\mathcal{C}^\perp$ [4, Theorem 1].

(3) $\mathcal{C}^*$ is uniformly concise if and only if $\mathcal{C}^{*\perp}$ is uniformly concise.

(4) $\mathcal{C}^*$ is uniformly full if and only if $\mathcal{C}^{*\perp}$ is uniformly full.

In the next sections, we show that the trellis complexity bounds derived in Section II.E are met exactly for the four categories of extremal minimal trellises.

B. Best Permutations

The following theorem shows that uniformly efficient trellises are those that achieve the DLP bounds in Eqs. (5), (6), and Theorem 6 with equality.

Theorem 11. A permutation π^* is uniformly efficient for a nondegenerate code $\mathcal{C}$ if and only if $\mathcal{C}\pi^*$ meets the DLP bounds, Eqs. (5) and (6), with equality, i.e.,

$$p_i(\mathcal{C}\pi^*) = K_i(\mathcal{C}) \text{ and } f_i(\mathcal{C}\pi^*) = K_{n-i}(\mathcal{C}) \text{ for all } i.$$

This guarantees that $\mathcal{C}\pi^*$ meets all of the lower bounds on complexity, Eqs. (9) through (14), with equality. Conversely, if $\mathcal{C}\pi^*$ meets any one of the lower bounds, Eqs. (11) through (13), with equality, then π^* is a uniformly efficient permutation for $\mathcal{C}$.

Theorem 11 shows that uniformly efficient permutations, which are defined in terms of trellis comparability, turn out to be the same as "efficient" [3] or "strictly optimum" [4] orderings, which were defined in terms of the DLP bounds. Note that a code may not have a permutation that meets these conditions.

A uniformly efficient permutation, if it exists, is not unique: If π^* is uniformly efficient for $\mathcal{C}$, then so is the reverse of π^*, and in fact the number of uniformly efficient permutations must be at least as large as the automorphism group of the code. There may also be different permutations that are uniformly efficient and produce distinct MSGMs for the code.

Even though uniform efficiency is a very strong property to require of a trellis, there are many codes that have uniformly efficient permutations. For example, the standard permutation of any Reed–Muller code is uniformly efficient [4, Theorem 2]. Additional examples of uniformly efficient codes are given in Section III.C, which lists trellises that are both uniformly efficient and uniformly concise.

We now give some theoretical results that impose necessary conditions on uniformly efficient permutations.

Theorem 12. Suppose $\mathcal{C}$ is a code that has some uniformly efficient permutation π^*. Then for any i, j such that $i + j \leq n$,

$$K_{i+j}(\mathcal{C}) \geq K_i(\mathcal{C}) + K_j(\mathcal{C})$$

Theorem 13. If π^* is a uniformly efficient permutation for an $(n, k, d, d^\perp)$ code $\mathcal{C}$, then $\mathcal{C}\pi^*$ contains codewords of the form $X^d 0^{n-d}$, $0^{n-d} X^d$, and $\mathcal{C}^\perp \pi^*$ contains codewords of the form $X^{d^\perp} 0^{n-d^\perp}$, $0^{n-d^\perp} X^{d^\perp}$, where 0^j denotes j consecutive zeros, and X^j denotes some sequence of j nonzero symbols from $GF(q)$.

Corollary 1. If $\mathcal{C}$ is a binary $(n, k, d, d^\perp)$ code that has some uniformly efficient permutation π^*, then $\min(d, d^\perp)$ must be even.

By Corollary 1, the (23,12,7,8) Golay code has no uniformly efficient permutation; neither does the $(2^m - 1, 2^m - m - 1, 3, 2^{m-1})$ Hamming code for any $m \geq 3$. Consequently, no nontrivial perfect binary linear code has a uniformly efficient permutation.

Although many codes lack uniformly efficient permutations, there may be some permutation that simultaneously minimizes all of the trellis complexity measures. For example, the (7,4) Hamming code is sufficiently small that we can verify by exhaustive search that there are permutations that are optimal with respect to all of the complexity measures despite not being uniformly efficient.

For self-dual codes, [6, Theorem 3] tells us that there is always a single permutation that simultaneously minimizes E, V, and M. We suspect that not every code has a permutation that simultaneously minimizes all of the complexity measures, though we do not yet know of an example that confirms this conjecture.

C. Best Codes

Uniformly concise codes are optimum in a rather strong sense. Not only do they have an efficient permutation, but they also minimize all of the trellis complexity measures compared to all codes with the same parameters. The following theorem shows that codes that achieve the bounds in Theorems 5 and 7 with equality are uniformly concise.

Theorem 14. An $(n, k, d, d^\perp)$ code $\mathcal{C}^*$ is uniformly concise if the dimensions of its past and future subcodes meet the bounds in Theorem 5 with equality, i.e.,

$$p_i(\mathcal{C}^*) = \overline{K}_i(n, k, d, d^\perp) \text{ and } f_i(\mathcal{C}^*) = \overline{K}_{n-i}(n, k, d, d^\perp) \text{ for all } i$$

In this case, $\mathcal{C}^*$ meets all of the lower bounds on complexity, Eqs. (15) through (20), with equality. Conversely, if $\mathcal{T}(\mathcal{C}^*)$ meets any of the bounds of Eqs. (17) through (19) with equality, then $\mathcal{C}^*$ is uniformly concise.

Table 1 lists known uniformly concise binary codes. In each case, the complexity values listed are the lowest possible for any code with the same parameters. From Theorem 10, the dual of each code is also uniformly concise. Generator matrices for many of these codes are given in Kiely et al.[5] All of the rate 1/2 codes in the table are either self-dual or have duals that are permuted versions of the original code.

Theorem 15. All $(2^m, m+1, 2^{m-1}, 4)$ first-order Reed–Muller codes and their duals, the $(2^m, 2^m - m - 1, 4, 2^{m-1})$ extended Hamming codes, are uniformly concise.

There are also examples of code parameters $(n, k, d, d^\perp)$ for which no uniformly concise trellis can exist. The $\mathcal{R}(r,m)$ Reed–Muller codes when $(m = 6, r = 2, 3)$, $(m = 7, r = 2, 3, 4)$ are codes that do not meet the UDLP bounds. This is established by comparing the UDLP bounds to the known optimal permutations for the Reed–Muller codes.

Results such as the examples above and Theorems 12 and 13 illustrate that in many instances the UDLP bounds on complexity are not tight. An area of further research is to produce tighter bounds on trellis complexity based on the code parameters $(n, k, d, d^\perp)$.

D. Worst Minimal Trellises

The following theorems show that uniformly inefficient and uniformly full minimal trellises are the same as the trellises that achieve the LDLP bounds with equality.

Theorem 16. An (n, k) code $\mathcal{C}$ is uniformly full if and only if the dimensions of the past and future subcodes of $\mathcal{C}$ meet the bounds of Eqs. (7) and (8) with equality, i.e.,

$$p_i(\mathcal{C}) = \max(0, k - n + i) \text{ and } f_i(\mathcal{C}) = \max(0, k - i) \text{ for all } i$$

In this case, $\mathcal{C}$ meets all of the upper bounds on complexity, Eqs. (21) through (26), with equality. Conversely, if $\mathcal{C}$ meets any one of the upper bounds, Eqs. (23) through (25), with equality, then $\mathcal{C}$ is uniformly full.

Theorem 17. A minimal trellis $\mathcal{T}(\mathcal{C}\pi^*)$ is uniformly full if and only if π^* is a uniformly inefficient permutation of $\mathcal{C}$.

Many codes have uniformly inefficient trellises in their standard permutations. For example, the minimal trellises for all cyclic, extended cyclic, and shortened cyclic codes are uniformly inefficient [5,7]. However, not every code has a uniformly inefficient permutation.

Additional examples of codes with uniformly inefficient trellises are given in the following two theorems.

Theorem 18. A self-dual code always has a uniformly inefficient permutation.

Theorem 19. If and only if a code is maximum distance separable (MDS), every permutation π is uniformly inefficient and the corresponding trellis complexity measures equal the upper bounds in Eqs. (21) through (26).

---

[5] Kiely et al., op cit.

Table 1. Some known uniformly concise binary codes.[a]

Code (parameters)	E	V	M	$s_{\max}$	$e_{\max}$	ε
Minimal span codes[b]						
$(d+(k-1)\lceil \frac{d}{2} \rceil, k, d, 2)$	$2kd$	$2+2k(d-1)$	$2k-1$	$\min\{2, d-2\}$	2	kd
Dual	$4k(d-2)+4$	$2+2k(d-1)$	$2k(d-3)+3$	2	2	$k(d-2)+n$
Reed–Muller[c] $\mathcal{R}(1,m)$ $(2^m, 1+m, 2^{m-1}, 4)$	$(2^{2m+1}+2^4)/3$ -4	$(2^{2m+1}+2^4)/3$ $-3(2^{m-1})-2$	$3(2^{m-1})-1$	m	m	$(m-1)2^m$ $+2$
Extended Hamming Dual	$(2^{2m+2}+2^5)/3$ $-4-3(2^{m+1})$	$(2^{2m+1}+2^4)/3$ $-3(2^{m-1})-2$	$(2^{2m+1}+2^4)/3$ $-9(2^{m-1})-1$	m	$m+1$	$m(2^m-2)$
Extended Golay $\mathcal{G}_{24}$ $(24,12,8,8)$ self-dual	3580	2686	895	9	9	136
Reed–Muller $\mathcal{R}(2,6)$ $(32,16,8,8)$ self-dual	6396	4798	1599	9	9	202
Quadratic residue $(48,24,12,12)$ self-dual	860156	645118	115039	16	16	502
$(10,5,4,4)$[d] Formally self-dual	60	46	15	3	3	24
$(12,6,4,4)$[d] Formally self-dual	76	58	19	3	3	30
$(16,4,8,2)$[d]	88	78	11	3	3	36
Dual	132	78	55	3	4	44
$(20,6,8,4)$[d]	236	206	31	4	4	66
Dual	348	206	143	4	5	74
$(24,7,8,4)$[d]	300	262	39	4	4	82
Dual	444	262	183	4	5	92
$(24,8,8,4)$[d]	364	302	63	5	5	86
Dual	476	302	175	5	5	94
$(40,7,16,4)$[d]	940	878	63	5	5	170
Dual	1628	878	751	5	6	196
$\mathcal{R}(1,3) \oplus \mathcal{R}(1,3)$ $(16,8,4,4)$ self-dual	88	67	22	3	3	36
$\mathcal{G}_{24} \oplus \mathcal{G}_{24}$ $(48,24,8,8)$ self-dual	7160	5371	1790	9	9	272

[a] Codes are grouped with their duals, which are also uniformly concise.
[b] $d > 2$, $k \leq 3$.
[c] Complexity expressions for first-order Reed–Muller and extended Hamming codes are valid for $m \geq 3$, except $e_{max} = 3$ when $m = 3$.
[d] See Kiely et al., op cit.

This theorem follows from the fact that a code is MDS if and only if every subset of k columns of its generator matrix is linearly independent. A peculiar consequence of Theorem 19 is that every permutation of an MDS code is also uniformly efficient, as noted by Forney [3]. This observation emphasizes that uniform efficiency is only a relative measure of trellis complexity.

IV. Conclusion

In this article, we extended the analysis of [6] to consider permutations of a code that minimize the complexity of a trellis representation that can be used for encoding or decoding. The analysis for a fixed code generalizes naturally to similar results for codes allowed to vary over a domain of optimization.

We identified two useful domains, the set of permutations of a given code and the set of all codes with given code parameters. Within each domain, we defined uniformly best and worst minimal trellises that are guaranteed to simultaneously minimize or maximize all of the complexity measures. We showed that it is easy to generalize the bounds on maximum state complexity derived by other authors from the dimension/length profile of a code to similar bounds on all the complexity measures over each optimization domain. Furthermore, if a minimal trellis attains the bounds for some of the complexity measures, it must necessarily be uniformly extremal, but this is not true for the simpler measures of maximum state or edge dimension considered by other authors. This lends further credence to the argument that a measure of total complexity (such as the total number of edges) is more useful than a measure of maximum complexity [10].[6]

Unlike the case of a fixed permutation of a given code, uniformly best and worst minimal trellises are not guaranteed to exist within the larger domains of optimization. However, we demonstrated the usefulness of the concepts by presenting several examples of uniformly best trellises, most notably the optimum permutation of the (48,24) quadratic residue code [2], heretofore unknown. Conversely, by deriving some necessary existence conditions, we also identified some cases for which uniformly extremal minimal trellises cannot exist.

We showed that the useful relationships between the trellis complexity of a code and that of its dual developed in [6] extend naturally to optimizations over larger code domains. This approach yields many of the same results obtained by other authors for dimension/length profiles or generalized Hamming weights, but it emphasizes that all the duality results stem from fundamental minimal trellis relationships valid for a fixed permutation of a code. In fact, we have argued that the symmetry of the constraints imposed by the code and its dual on trellis complexity is so fundamental that the minimum distance of the dual code should be included as one of the intrinsic code parameters that limits achievable complexity.

Acknowledgment

A portion of W. Lin's contribution was sponsored by AFOSR grant no. F4960-94-1-005. Part of this work was presented at the 32nd Annual Allerton Conference on Communication, Control, and Computing in Allerton, Illinois, in October 1994.

References

[1] A. E. Brouwer and T. Verhoeff, "An Updated Table of Minimum-Distance Bounds for Binary Linear Codes," *IEEE Trans. Inform. Theory*, vol. 39, pp. 662–677, 1993.

[2] S. Dolinar, L. Ekroot, A. Kiely, W. Lin, and R. J. McEliece, "The Permutation Trellis Complexity of Linear Block Codes," *Proc. 32nd Annual Allerton Conference on Communication, Control, and Computing*, Allerton, Illinois, October 1994.

[6] McEliece, op cit.

[3] G. D. Forney, "Dimension/Length Profiles and Trellis Complexity of Linear Block Codes," *IEEE Trans. Inform. Theory*, vol. 40, no. 6., pp. 1741–1752, November 1994.

[4] T. Kasami, T. Takata, T. Fujiwara, and S. Lin "On the Optimum Bit Orders With Respect to the State Complexity of Trellis Diagrams for Binary Linear Codes," *IEEE Trans. Inform. Theory*, vol. 39, pp. 242–245, 1993.

[5] T. Kasami, T. Takata, T. Fujiwara, and S. Lin, "On Complexity of Trellis Structure of Linear Block Codes," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1057–1064, 1993.

[6] A. B. Kiely, S. Dolinar, R. J. McEliece, L. Ekroot, and W. Lin, "Minimal Trellises for Linear Block Codes and Their Duals," *The Telecommunications and Data Acquisition Progress Report 42-121, January–March 1995*, Jet Propulsion Laboratory, Pasadena, California, pp. 148–158, May 15, 1995.

[7] F. R. Kschischang and V. Sorokine, "On the Trellis Structure of Block Codes," *Proc. 1994 IEEE International Symposium on Information Theory*, Trondheim, Norway, p. 337, June 27–July 1, 1994.

[8] A. Lafourcade and A. Vardy, "Asymptotically Good Codes Have Infinite Trellis Complexity," *IEEE Trans. Inform. Theory*, vol. 41, no. 2, pp. 555–559, March 1995.

[9] J. L. Massey, "Foundations and Methods of Channel Coding," in *Proc. of the Int. Conf. on Info. Theory and Systems*, vol. 65, NTG-Fachberichte, September 1978.

[10] R. J. McEliece, "The Viterbi Decoding Complexity of Linear Block Codes," *Proc. 1994 IEEE International Symposium on Information Theory*, Trondheim, Norway, p. 341, June 27–July 1, 1994.

[11] A. Vardy and Y. Be'ery, "Maximum-Likelihood Soft Decision Decoding of BCH Codes," *IEEE Trans. Inform. Theory*, vol. 40, pp. 546–554, 1994.

[12] V. K. Wei, "Generalized Hamming Weights for Linear Codes," *IEEE Trans. Inform. Theory*, vol. 37, pp. 1412–1418, 1991.

[13] J. K. Wolf, "Efficient Maximum Likelihood Decoding of Linear Block Codes Using a Trellis," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 76–80, 1978.