Codes, matroids and trellises

Peter J. Cameron
School of Mathematical Sciences
Queen Mary and Westfield College
London E1 4NS
U.K.
p.j.cameron@qmw.ac.uk

Abstract

There is a natural bijection between representable matroids and linear codes over a given field, which is not as well known as it ought to be. An early result on this is Greene's theorem asserting that the weight enumerator of the code is a specialisation of the Tutte polynomial of the matroid. This connection also throws light on the minimal trellis for decoding a given code, and gives new results (and easier proofs of old results) on trellis complexity.

1 Introduction

This is a survey paper on the connection between matroids and linear codes, on trellis decoding, and of applications of the code–matroid connection to the problem of determining the smallest trellis for a code equivalent to a given code. Proofs of the assertions appear elsewhere. The paper is based to a large extent on the Ph.D. thesis [8] of Constantinos Papadopoulos, to whom I express my gratitude. Others who have contributed are R. A. Bailey, Carrie Rutherford and Fuad Shareef.

2 Codes

In this section we give brief summaries of the theories of codes and matroids. For more details we refer to standard textbooks such as [6] for codes and [9] for matroids.

Let F be a set called the *alphabet* and n a positive integer. A *word* of length n over F is simply an n-tuple of elements of F; typically we write $a_1 a_2 \cdots a_n$ instead of $(a_1, a_2, \ldots, a_n)$. In the most important case here, F is a field; from now on, this is always assumed to be the case. The code C is *linear* if it is a subspace of F^n. A linear code of length n and dimension k is referred to as an $[n, k]$ code.

Let C be a $[n, k]$ code. A *generator matrix* for C is a $k \times n$ matrix whose rows form a basis for C. The *dual code* $C^\perp$ of C is the set

$$C^\perp = \{x \in F^n : x \cdot c = 0 \text{ for all } c \in C\},$$

where $\cdot$ denotes the standard inner product on F^n.

The *Hamming distance* $d(a, b)$ between words a and b is the number of coordinates where they differ:

$$d(a, b) = |\{i : 1 \leq i \leq n, a_i \neq b_i\}|.$$

The *weight* $\mathrm{wt}(a)$ is the number of non-zero coordinates of a, that is, $\mathrm{wt}(a) = d(a, 0)$, where 0 is the all-zero word. If F is finite, the *weight enumerator* $W_C(x, y)$ of the code C is the homogeneous polynomial

$$W_C(x, y) = \sum_{c \in C} x^{n - \mathrm{wt}(c)} y^{\mathrm{wt}(c)} = \sum_{i=0}^{n} A_i x^{n-i} y^i,$$

where A_i is the number of words of weight i in C.

Two codes C, C' of length n over F are *monomial equivalent* if C' can be obtained from C by permuting the coordinates and multiplying coordinates by non-zero scalars. This is the natural equivalence relation on linear codes, and preserves dimension, weight enumerator, and most significant properties.

The ith *generalised Hamming weight* $d_i(C)$ of a code C, for $1 \leq i \leq k = \dim(C)$, is defined to be the smallest support size of an i-dimensional subcode of C. So $d_1(C)$ is the *minimum weight* of C. The sequence $(d_1(C), \ldots, d_k(C))$ is called the *Hamming weight hierarchy* of C. These numbers are strictly increasing; in fact, the following is true:

Theorem 2.1 *The generalized Hamming weights of a linear code over* $\mathrm{GF}(q)$ *satisfy*

$$d_{i+1} \geq d_i + \left\lceil \frac{d_i(q-1)}{q(q^i-1)} \right\rceil .$$

This result in the case of binary codes is due to Helleseth *et al.* [4].

3 Matroids

Let E be a set. A *matroid* M on E is a pair $(E, \mathcal{I})$, where $\mathcal{I}$ is a non-empty family of subsets of E (called *independent sets*) with the properties

(a) if $I \in \mathcal{I}$ and $J \subseteq I$, then $J \in \mathcal{I}$;

(b) (the *exchange property*) if $I_1, I_2 \in \mathcal{I}$ and $|I_1| < |I_2|$, then there exists $e \in I_2 \setminus I_1$ such that $I_1 \cup \{e\} \in \mathcal{I}$.

Matroids were introduced by Whitney to axiomatise the notion of linear independence in a vector space. Indeed, if E is a family of vectors in a vector space V, and $\mathcal{I}$ is the set of linearly independent subsets of E, then $(E, \mathcal{I})$ is a matroid. More formally, a *representation* of a matroid $(E, \mathcal{I})$ over a field F is a map χ from E to an F-vector space with the property that a subset I of E belongs to $\mathcal{I}$ if and only if $\chi(I)$ is linearly independent. Two representations χ, χ' of M are *equivalent* if there is an invertible linear transformation of V whose composition with χ is χ'.

It follows from the second axiom that all maximal independent sets in a matroid M have the same cardinality k, called the *rank* of M. These maximal independent sets are called the *bases* of M.

It can be shown that the set of all complements of bases of M is the set of bases of another matroid M^* on E, called the *dual* of M.

Matroids can be defined in many equivalent ways. One which will be important to us is by means of the *rank function* ρ, the function from the power set of E to the non-negative integers given by the rule that $\rho(A)$ is the cardinality of any maximal independent subset of A. (Again, the exchange property shows that any two maximal independent subsets of A have the same cardinality.) It is possible to axiomatise the rank functions of matroids, but that will not be necessary here.

One feature of matroids is that they are precisely those structures where the greedy algorithm always succeeds in choosing a base of minimum weight. More formally:

Proposition 3.1 *Suppose that $M = (E, \mathcal{I})$ is a matroid, and that the elements of E are totally ordered. Then*

(a) there is a base $A = \{a_1, \ldots, a_k\}$ with $a_1 < \cdots < a_k$, called the first base*, such that if $X = \{x_1, \ldots, x_k\}$ is any other base with $x_1 < \cdots < x_k$, then $a_i \leq x_i$ for $i = 1, \ldots, k$;*

(b) there is a base $B = \{b_1, \ldots, b_k\}$ with $b_1 < \cdots < b_k$, called the last base*, such that if $X = \{x_1, \ldots, x_k\}$ is any other base with $x_1 < \cdots < x_k$, then $x_i \leq b_i$ for $i = 1, \ldots, k$.*

The first base A is found by the greedy algorithm: $\{a_1\}$ is the smallest independent set of size 1; then a_2 is the smallest element such that $\{a_1, a_2\}$ is independent; and so on. A dual remark applies to the last base.

The first and last bases of the dual matroid M^* are the complements of the last and first bases of M respectively.

4 The code–matroid connection

Let A be a $k \times n$ matrix over a field F, satisfying the condition that the rows of A are linearly independent, so that the row space of A has dimension k.

There are two different structures that can be built from A.

First, the row space of A is an $[n, k]$ code over F, that is, a k-dimensional subspace of F^n. Now row operations on A simply change the basis for the code, leaving the actual code completely unaltered. Column permutations, and multiplications of columns by non-zero scalars, replace the code by a monomial equivalent code.

Second, there is a matroid M on the set $E = \{1, 2, \ldots, n\}$, in which a set I is independent if and only if the family of columns of A whose indices belong to I is linearly independent. (We cannot quite say that the elements of E are the columns and independence is linear independence, since E might have repeated columns.) Moreover, the function χ mapping i to the ith column is a representation of M over F. Now row operations on A don't change M but replace the representation χ by an equivalent representation, while column permutations and scalar multiplications replace M by an isomorphic matroid.

So, if we call two matrices A and A' *CM-equivalent* if A' is obtained from A by a row operation and a monomial transformation of the columns, we see that CM-equivalence classes of matroids correspond bijectively to both monomial equivalence classes of linear codes, and equivalence classes of representations of matroids, under the natural notions of equivalence in each case.

Thus we expect information to transfer back and forth between code and matroid. In later sections, we will implicitly do so, by talking about (for example) the first and last base of a code (meaning the first and last base of the corresponding matroid).

A code C is called *projective* if any two columns of a generator matrix for C are linearly independent (equivalently, if the dual code $C^\perp$ has minimum weight at least 3). The corresponding condition for matroids is that of being a *simple matroid*, that is, all subsets of size at most 2 are independent (so that there are no loops or parallel elements in the matroid). If this holds, then the representation of the matroid can be regarded as being in the projective space $\mathrm{PG}(k-1, F)$ rather than in the vector space F^k. Thus, representations of simple matroids provide another framework for studying point sets in projective spaces. For example, the zeros of a word in the dual code $C^\perp$ are the points of a hyperplane section of the point set, and so the weight enumerator of $C^\perp$ gives the cardinalities of the hyperplane sections.

It is a simple exercise to show the following:

Proposition 4.1 *If the matroid M corresponds to the code C, then the dual matroid M^* corresponds to the dual code $C^\perp$.*

5 Tutte polynomial and weight enumerator

Let M be a matroid on the set E, having rank function ρ. The *Tutte polynomial* of M is most easily defined as follows:

$$T(M; x, y) = \sum_{A \subseteq E} (x-1)^{\rho E - \rho A} (y-1)^{|A| - \rho A}.$$

For example, the Tutte polynomial of the uniform matroid $U_{n,k}$ is

$$T(U_{n,k};x,y) = \sum_{i=0}^{k} \binom{n}{i}(x-1)^i + \sum_{i=k+1}^{n} \binom{n}{i}(y-1)^i.$$

Moreover we have:

Proposition 5.1 *(a) The number of bases of M is equal to $T(M;1,1)$.*

(b) The number of independent sets of M is equal to $T(M;2,1)$.

(c) The number of spanning sets of M is equal to $T(M;1,2)$.

(d) $T(M;2,2) = 2^n$.

The Tutte polynomials of a matroid and its dual are very simply related:

Proposition 5.2

$$T(M^*;x,y) = T(M;y,x).$$

In the remainder of this section we briefly discuss a particular instance where the connection between codes and matroids is useful. The following theorem was proved by Greene [3].

Theorem 5.3 *Let C be a code over a field with q elements, and M the corresponding vector matroid. Then*

$$W_C(x,y) = y^{n-\dim(C)}(x-y)^{\dim(C)}T\left(M; \frac{x+(q-1)y}{x-y}, \frac{x}{y}\right).$$

Note that, if $X = (x+(q-1)y)/(x-y)$ and $Y = x/y$, then

$$(X-1)(Y-1) = q.$$

So the weight enumerator is an evaluation of the Tutte polynomial along a particular hyperbola in the 'Tutte plane'.

The proof will not be given here, but merely indicated. There is a recursive formula for the Tutte polynomial of a matroid in terms of Tutte polynomials of matroids with one fewer element (the so-called "deletion–contraction rule"). Similarly, the weight enumerator of a code can be expressed in terms of the weight enumerators of the codes obtained by shortening and puncturing the code at some position. Then the equality of the two expressions in the theorem follows by induction. (The punctured code corresponds to the deleted matroid, and the shortened code to the contracted matroid.)

From Theorem 5.3 and Proposition 5.2, we can deduce the *MacWilliams relation*, which shows that the weight enumerator of the dual code $C^\perp$ can be calculated from that of C.

Theorem 5.4

$$W_{C^\perp}(x,y) = \frac{1}{|C|} W_C(x+(q-1)y, x-y).$$

Proof Since $C^\perp$ has dimension $n - \dim(C)$ and corresponds to the dual matroid M^*, we have

$$W_{C^\perp}(x,y) = y^{\dim(C)}(x-y)^{n-\dim(C)} T\left(M; \frac{x}{y}, \frac{x+(q-1)y}{x-y}\right).$$

On the other hand, we have

$$\frac{1}{|C|} W_C(x+(q-1)y, x-y) =$$
$$q^{-\dim(C)}(x-y)^{n-\dim(C)}(qy)^{\dim(C)} T\left(M; \frac{qx}{qy}, \frac{x+(q-1)y}{x-y}\right).$$

The two expressions are equal.

Note that this proof is entirely combinatorial, in contrast to the usual proof which involves characters of the additive group of F^n.

6 Trellis decoding

For a small but not quite trivial example, suppose that we are using the binary dual Hamming code of length 7 to send information. The codewords are:

$$\begin{array}{c}0000000\\0011011\\0101101\\0110110\\1001110\\1010101\\1100011\\1111000\end{array}$$

The minimum weight is 4, so we can correct one error and detect two errors.

However, in a practical communication channel, the received word is likely to be an analog signal, sampled at seven time points, so we obtain seven real numbers. Suppose that we receive

$$w = (-0.1, 0.0, 0.2, 0.9, 1.8, 0.9, 1.4) \in \mathbb{R}^7.$$

If we round each value to the nearest of zero and one, we obtain 0001111, which is at distance 2 from the second, third and fifth codewords in the list, so we have a decoding failure.

If we make the (physically realistic) assumptions that the errors at the sampling points are independent identically distributed Gaussian variables, then it can be shown that the most likely codeword to have been transmitted is the one at smallest Euclidean distance from w in $\mathbb{R}^7$, which turns out to be 0101101.

The method of *trellis decoding* provides a way to decode to the nearest Euclidean codeword without first rounding the individual entries of w, and without calculating the Euclidean distance of w to every codeword.

Let F be an alphabet. A *trellis* over F is a directed graph whose vertices come in *layers* $V_0, V_1, \ldots, V_n$, where V_0 contains a single vertex s called the *source*, and V_n contains a single vertex t called the *target*, and whose edges also come in *layers* $E_{i-1,i}$ for $i = 1, \ldots, n$, where an edge in $E_{i-1,i}$ has initial vertex in V_{i-1} and terminal vertex in E_i. Moreover, each edge has a label, which is an element of the alphabet F. Now any path from s to t in F has length n, and the labels of the edges on the path (taken in order) form a word of length n. We say that the trellis is *one-to-one* if different paths realise different words. The set C of all words represented by paths in the trellis is the *code represented by* the trellis. Figure 1 shows a trellis for the dual Hamming code.

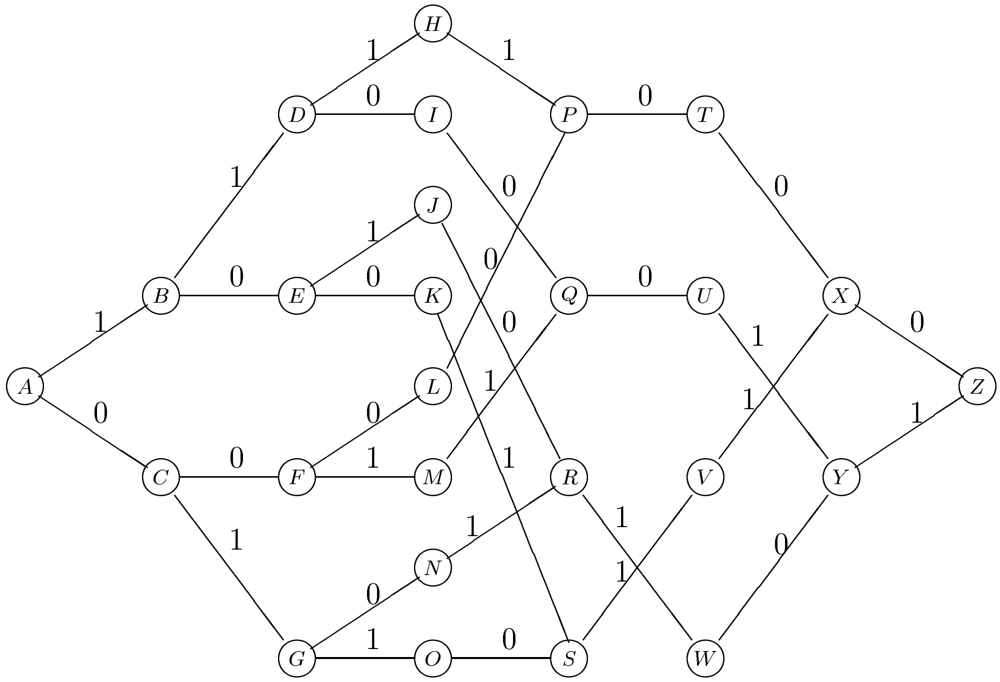

Figure 1: A trellis for the dual Hamming code

Now the trellis is used as follows. Each symbol in the alphabet F is represented by a real number (the signal level corresponding to accurate transmission of that symbol). If $F = \mathrm{GF}(2)$, we may assume without loss of generality that the signal levels for the elements $0, 1 \in F$ are the real numbers $0, 1$. In general, we identify the set F with the corresponding set of real numbers.

Now, when the ith component w_i of the received word w is known, we assign the *length* $l(e) = (w_i - a)^2$ to an edge e with label a in the layer $E_{i-1,i}$. When all components of w have been received, then the sum of the edge lengths in any path from s to t is equal to the squared Euclidean distance from w to the corresponding codeword. So finding the nearest codeword to w is transformed into the problem of finding the shortest path in a directed graph with edge lengths.

Most of the work can be done in real time. Once edge lengths $l(e)$ are assigned to edges $e \in E_{i-1,i}$, we assign to each vertex $v \in E_i$ the number

$$f(v) = \min\{f(v') + l(v', v) : (v', v) \in E_{i-1,i}\}$$

(where the induction begins with $f(s) = 0$). Then $f(v)$ is the length of the shortest path from s to v. Once all components of x are known, then the length of the shortest path from s to t is determined, and the actual path can be found by working backwards. (This is essentially Dijkstra's shortest path algorithm.)

In our example, the numbers work out as shown in Figure 2. We see that the shortest path is

$$A \to C \to G \to N \to R \to W \to Y \to Z,$$

and give the correct message corresponding to this path, which is 0101101.

7 The minimal trellis for a code

Every code is represented by at least one trellis. We can simply take $|C|$ paths from s to t which are edge-disjoint and have only the end vertices in common, and label each path to represent one codeword.

However, the discussion of decoding above shows that a smaller trellis will give more efficient decoding. The algorithm given involves one multiplication for each edge, and k additions and $k-1$ comparisons at each vertex with in-degree k. So the number of arithmetic operations is twice the edge count of the trellis, and the number of comparisons is equal to the *cycle rank* (number of vertices minus number of edges plus one). A third obvious measure of trellis size is the vertex count of the trellis.

Muder's theorem [7] shows that, for a linear code, there is a trellis which is uniformly best in terms of vertex count.

Theorem 7.1 *Let C be a linear code of length n. Then there is a trellis T representing C, with layers $V_0, \ldots, V_n$, such that, if another proper trellis T' for C has layers $V'_0, \ldots, V'_n$, then $|V'_i| \geq |V_i|$ for $i = 0, \ldots, n$. Moreover, if $|V'_i| = |V_i|$ for $i = 0, \ldots, n$, then T' is isomorphic to T. Furthermore, T also minimises the sizes of all the edge layers and the cycle rank.*

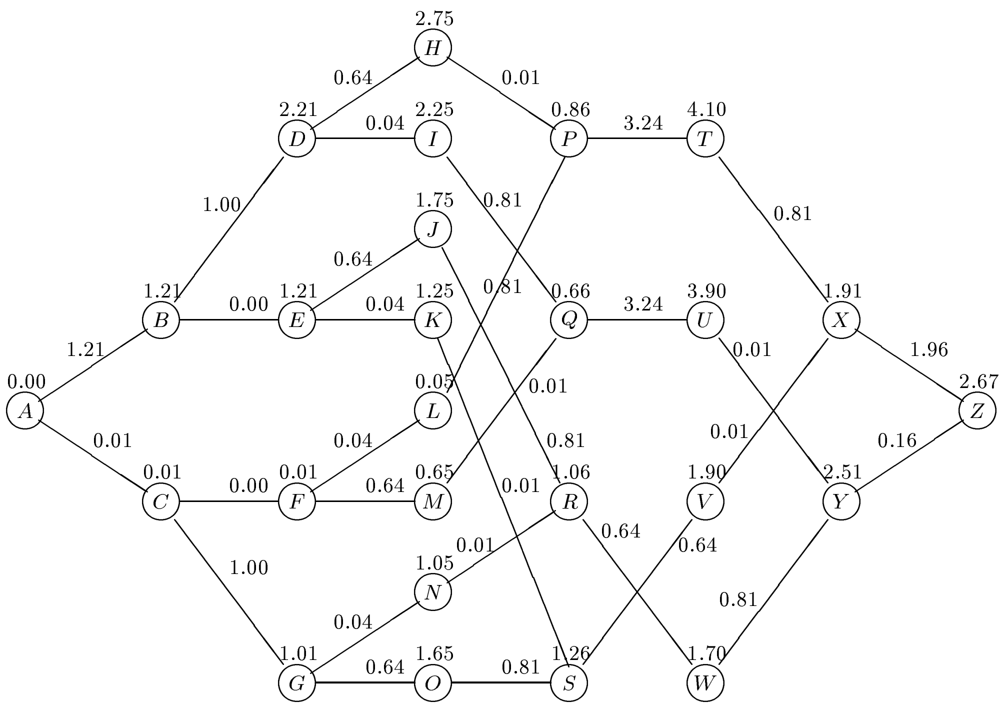

Figure 2: A trellis with edge-lengths and distances

I will not prove this theorem here, but instead describe the simple construction of the Muder trellis given in [1]. This also shows how to calculate the size of the trellis from knowledge of the first and last bases for the matroid associated with the code (see Proposition 3.1).

Let C be a linear code over F, and let $A = \{a_1, \ldots, a_k\}$ be the first base and $B = \{b_1, \ldots, b_k\}$ be the last base, where $a_1 < \cdots < a_k$ and $b_1 < \cdots < b_k$.

For $0 \leq i \leq n$, we define the ith *past subcode* of C to be

$$P_i = \{c \in C : c_j = 0 \text{ for all } j > i\},$$

and the ith *future subcode* to be

$$F_i = \{c \in C : c_j = 0 \text{ for all } j \leq i\}.$$

By convention, $P_n = F_0 = C$. If we take a generator matrix G for C in echelon form, we see that F_i is spanned by the rows of G whose leading 1 occurs to the right of i. Hence

$$\dim(F_i) = |A \cap \{i+1, \ldots, n\}| = k - |A \cap \{1, \ldots, i\}|.$$

Dually,

$$\dim(P_i) = |B \cap \{1, \ldots, i\}|.$$

Also, of course, we have

$$P_i \cap F_i = \{0\},$$

so P_i and F_i generate their direct sum.

Now the following construction produces the minimal trellis for C. The vertices in the ith layer are the elements of the quotient space $V_i = C/(P_i \oplus F_i)$, that is, cosets of $P_i \oplus F_i$ in C. For each codeword c, we put an edge with label c_i from the coset $(P_{i-1} \oplus F_{i-1}) + c \in V_{i-1}$ to the coset $(P_i \oplus F_i) + c \in V_i$. We identify edges with the same label between the same vertices. Note that, for example, two edges with the same initial vertex and the same label also have the same terminal vertex. For suppose that $c - d \in P_{i-1} \oplus F_{i-1}$, say $c - d = p + f$ with $p \in P_{i-1}$ and $f \in F_{i-1}$. Suppose also that $c_i = d_i$. By definition, $p_i = 0$, hence $f_i = (c - d - p)_i = 0$, and $f_i \in F_i$. Thus, $c - d = p + f \in P_i \oplus F_i$, and so the terminal vertices of the corresponding edges are the same.

Similar arguments show:

Proposition 7.2 *The trellis just constructed is isomorphic to the Muder minimal trellis.*

This shows that the first and last bases determine the sizes of the layers in the Muder trellis:

Proposition 7.3 *Let C be a linear code over* $\mathrm{GF}(q)$. *Then the sizes of the vertex and edge layers in the Muder trellis for C are given by*

$$\begin{aligned} |V_i| &= q^{|A \cap \{1,\ldots,i\}| - |B \cap \{1,\ldots,i\}|}, \\ |E_{i-1,i}| &= q^{|A \cap \{1,\ldots,i\}| - |B \cap \{1,\ldots,i-1\}|}, \end{aligned}$$

where A and B are the first and last bases of the corresponding matroid.

For example, if C is the dual Hamming code of the preceding section, then the first base is $\{1, 2, 3\}$ and the last base is $\{4, 6, 7\}$, so we find immediately that the sizes of the vertex layers are $1, 2, 4, 8, 4, 4, 2, 1$ respectively.

8 Column permutations

The various measures of trellis size for a code differ from the more usual coding-theoretic parameters in one significant way: they may differ for equivalent codes (those obtained by column permutations). For example, if we reverse the order of the coordinates of the dual Hamming code, the first base is $\{1, 2, 4\}$ and the last base $\{5, 6, 7\}$; the size of the trellis is unaltered but the individual layers change. Swapping the fourth and fifth columns gives first base $\{1, 2, 3\}$ and last base $\{5, 6, 7\}$, so that the sizes of the layers in the minimal trellis are $1, 2, 4, 8, 8, 4, 2, 1$.

We see that, to construct a small trellis for a code equivalent to C, we want to make the first base come as late as possible, and the last base come as early as possible. This is a difficult problem; it is known to be NP-complete in general (see [5]). However, we can use the geometric structure of some codes to find the smallest trellises.

First a class of codes in which we have no choice. A linear code C of length n and dimension k is said to be *maximum distance separable*, or *MDS*, if the corresponding matroid is the uniform matroid $U_{n,k}$ of rank k on n elements (that is, every set of size k is a base – this is not the usual definition!) A representation of the uniform matroid $U_{n,k}$ over $\mathrm{GF}(q)$ is simply a n-arc in $\mathrm{PG}(k-1, q)$; this instance of the code–matroid connection has been the subject of a lot of research (see [2]).

MDS codes are the worst with respect to trellis size:

Proposition 8.1 *Suppose that $k \leq n/2$.*

(a) The number of vertices in the Muder trellis for an $[n, k]$ code is at most $(n - 2k + 1)q^k + 2(q^k - 1)/(q - 1)$.

(b) The code C has the property that the Muder trellis for any code equivalent to C has $(n - 2k + 1)q^k + 2(q^k - 1)/(q - 1)$ vertices if and only if C is MDS.

Proof The given bound is the number of vertices in the Muder trellis for a code whose first base is $\{1, \ldots, k\}$ and whose last base is $\{n-k+1, \ldots, n\}$. Clearly this is the worst case. Now if C is MDS, then every set of k elements is a base, so the first base is $\{1, \ldots, k\}$ and the last base $\{n - k + 1, \ldots, n\}$ in any ordering. Conversely, suppose that the bound is attained for all codes equivalent to C. Any set of k columns can be brought to the first k positions by some permutation, and so forms a base; so C is MDS.

We conclude with the extended binary Golay code C_{24}, a $[24, 12]$ binary code with minimum weight 8. Since C is self-dual, the complement of any base is a base, and we simply have to arrange the columns so that the first base occurs as late as possible.

A *sextet* is a partition of the 24 coordinates into six *tetrads* or sets of 4, with the property that the union of any two tetrads supports a codeword. Permute the coordinates so that the sets $\{1, 2, 3, 4\}$, $\{5, 6, 7, 8\}$, $\ldots$, $\{21, 22, 23, 24\}$ are the tetrads of a sextet. Then the first base contains $1, 2, 3, 4, 5, 6, 7$ but not 8, and $9, 10, 11$ but not 12. The same argument shows that the last base does not contain 13 or 17, so these are the remaining elements of the first base. Now we can calculate that the number of vertices of the Muder trellis is 2686.

On the other hand, we cannot do better. Since the minimum weight of the code is 8, the future subcode F_i is zero for $i > 16$, and so the last possible position for an element of the first base is 17. Similarly, looking at the smallest possible support of a d-dimensional subcode for $d = 2, 3, \ldots$, we find upper bounds for the positions of the elements of the first base, and we conclude that the best we can do is as given in the preceding paragraph.

This example hints at a connection between the trellis parameters and the Hamming weight hierarchy of a code. Such a connection was worked out by Papadopoulos in his thesis and is given in [1]. It states:

Theorem 8.2 *The first and last base of the* $[n, k]$ *code* C *saatisfy*

$$a_i \leq n - d_{k-i+1}(C) + 1, \qquad b_i \geq d_i(C).$$

References

[1] R. A. Bailey, P. J. Cameron and C. Papadopoulos, Trellis complexity using the first and last bases, in preparation.

[2] A. A. Bruen, J. A. Thas and A. Blokhuis, On M.D.S. codes, arcs in $\mathrm{PG}(n, q)$ with q even, and a solution of three fundamental problems of B. Segre, *Inventiones Mathematicae* **92** (1988), 441–459.

[3] C. Greene, Weight enumeration and the geometry of linear codes, *Studia Appl. Math.* **55** (1976), 119–128.

[4] T. Helleseth, T. Kløve and Ø. Ytrehus, Generalized Hamming weights for linear codes, *IEEE Transactions on Information Theory* **38** (1992), 1133–1140.

[5] K. Jain, I. Măndoiu and V. Vazirani, The "art of trellis decoding" is computationally hard—for large fields, *IEEE Trans. Inform. Theory* **44** (1998), 1211–1214.

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

[7] D. J. Muder, Minimal trellises for block codes, *IEEE Transactions on Information Theory* **IT-34** (1988), 1049–1053.

[8] C. Papadopoulos, *Codes and Trellises*, Ph.D. thesis, University of London, 1999.

[9] D. J. A. Welsh, *Matroid Theory*, Academic Press, London, 1976.