Journal of Informatics and Mathematical Sciences
Vol. 10, Nos. 1 & 2, pp. 1–11, 2018
ISSN 0975-5748 (online); 0974-875X (print)
Published by RGN Publications
http://dx.doi.org/10.26713/jims.v10i1-2.677

http://www.rgnpublications.com

Research Article

On Isometric Error-Correcting Codes over Finite Fields of Prime Order

T.I. Katsaounis

Department of Mathematics, The Ohio State University, Mansfield OH 44906, USA
katsaounis.1@osu.edu

Abstract. Error-correcting codes obtained from each other via a Hamming distance preserving transformation, called isometry, are equivalent. In this paper, we consider three different isometries which yield permutationally equivalent linear codes, monomially equivalent linear codes, and equivalent nonlinear codes, respectively. For each case, we derive some new necessary and sufficient conditions for equivalence using the concept of weight vector of a code or distance matrix of a code. Results hold for error-correcting codes over a finite field of prime order.

Keywords. Error-correcting code; Isometry; Permutational equivalence; Linear equivalence; Equivalent codes

MSC. 68R05; 94B05; 11T71

Received: September 3, 2017 **Accepted:** November 23, 2017

1. Introduction

Equivalence of error-correcting codes under Hamming distance preserving transformations, called isometries, is an important problem studied in combinatorics. For some of the earlier results about code isometries, see, for example, [5] and [11]; for more recent references see, for example, [1] and [8]. The usual definitions of isometric error-correcting codes require existence of appropriately defined isometries that transform a code to another; however they do not provide a method for finding such isometries. In this paper, some new necessary and sufficient conditions for equivalence are presented which give a way to identify a proper isometry that transforms a code to another, if it exists. Error-correcting codes are used in communication systems when the objective is to transmit and recover signals (codewords) as accurately as possible (see, for example, [14]). This paper focuses on error-correcting codes over a finite field of prime order. The number of transmitted bits (coordinates in codewords) that are corrupted (errors) which can be corrected (or detected) by a code is determined by its *minimum* Hamming distance d (the Hamming distance between two codewords is the number of coordinates where two codewords differ). If $d \geq 2e+1$, then up to e errors can be corrected (and up to $d-1$ errors can be detected). Error-correcting codes obtained from each other via a Hamming distance preserving transformation have the same error-correction (detection) capability and, as such, they are considered equivalent.

This paper deals with permutational and monomial equivalence of linear error correcting codes, and equivalence of general error correcting codes (over a field of prime order). There are several known results about permutational and monomial equivalence of linear error-correcting codes. A comprehensive review of such results is beyond the scope of this paper. The primary goal of this paper is investigation of necessary and sufficient conditions for equivalence of general error correcting codes. First, some related results about permutational and monomial equivalence of linear codes are presented in order to demonstrate the succession of key concepts that lead to the results about general codes. Permutational equivalence is applicable when the order in which coordinates (bits) are transmitted is irrelevant; for example, when a channel is “memoryless” (i.e. the probability of an error is conditionally independent of previous transmissions and errors). Obviously, in such cases, a permutation of coordinates does not affect the quality of a code. As can be shown, a permutation of coordinates is an isometry that preserves linearity. Specifically, linear codes obtained from each other via a permutation of coordinates are called *permutationally isometric* or *permutationally equivalent* (see, for example, [1] and [8] for details). Monomial equivalence becomes applicable when reordering of the symbols in coordinates is irrelevant also; for example, when a channel is “symmetric” (i.e. the probability that a transmitted symbol is corrupted is the same for any of the symbols in a coordinate). In such cases, a reordering of the symbols in any of the coordinates does not affect the quality of a code. In general, any codes obtained from each other via a permutation of coordinates in codewords combined with a permutation of symbols in one or more coordinates (an isometry) are called *equivalent*. In the case of linear codes though, we need to consider permutations of the non-zero symbols in F_q in order to preserve linearity. In particular, linear codes obtained from each other via a permutation of coordinates in codewords combined with a permutation of non-zero symbols in one or more coordinates are called *monomially equivalent* (or *linearly equivalent*); see, for example, [1], [3], [8] and [10].

Section 2 states some basic definitions and preliminary results used throughout this paper (for more details see, for example, [7] and [12]). Section 3 gives necessary and sufficient conditions for permutational equivalence of linear error-correcting codes, using the invariance property of weight vector of a code (Lemma 3.1 and Theorem 3.1). Section 4 presents necessary and sufficient conditions for monomial equivalence of linear error-correcting codes, based on the concept of distance matrix of a code (Theorems 4.1 and 4.2). Section 5 provides necessary and sufficient conditions for equivalence of nonlinear error-correcting codes by extending the results of Section 4 (Theorems 5.1 and 5.2). Finally, Section 6 gives a discussion of the results and the conclusions.

2. Preliminaries

Let $F_q = \{0,1,\cdots,q-1\}$ be the finite field of order q, where q is a prime and consider the finite vector space of all n-tuples $u = (u_1,u_2,\cdots u_n)$ over F_q denoted by V^n. A (n,m,d) code (with length n, size m codewords, and minimum distance d) is a nonempty subset C of V^n. An n-tuple u in V^n is called codeword and the values u_i, $i = 1,\cdots,n \in F_q$ are called coordinates. A code is called binary if $q = 2$, ternary if $q = 3$, quarternary if $q = 4$, and q-ary if $q > 4$. The finite vector space V^n equipped with the Hamming distance metric is called the Hamming metric space. Two codes over the Hamming metric space are isometric if there exists a bijective mapping $f : C \in (V^n,d) \to C' \in (V^n,d)$ which preserves the Hamming distances between codewords, i.e. $d(u,v) = d(f(u),f(v))$, for all $u,v \in (V^n)$ (see, for example, [8, p. 36]). If u,v,w are codewords in the Hamming metric space, $d(u,v)$ is the Hamming distance between codewords u,v, and $\alpha \in F_q - \{0\}$, then $d(u,v) = d(u+w,v+w)$ and $d(u,v) = d(\alpha u,\alpha v)$ hold. As seen from these fundamental properties, Hamming distance is a natural distance metric when dealing with isometric codes. Note that, the Hamming distance of a codeword from the zero codeword is the Hamming weight of the codeword. We use the term distance (weight) to refer to Hamming distance (weight).

A linear code is a set of vectors in V^n over F_q that forms a k-dimensional subspace of V^n-closed under addition and scalar multiplication modulo q (in F_q). A nonlinear code is a set of vectors in V^n over F_q that is not a subspace of V^n. In this paper, (without loss of generality) an (n,m,d) code C is represented by an $m \times n$ matrix $C = (c_{ij})$, with elements in F_q such that its m rows are the codewords. Note that, a reordering of codewords in C is not important (it does not affect the quality of the code); nevertheless, it corresponds to a row permutation in C, and can be expressed via a row permutation matrix. In general, a code is characterized by its distance distribution, consisting of the frequencies of distances between the codewords. A linear code though is usually characterized by its weight distribution, consisting of the frequencies of weights of the codewords (which coincides with its distance distribution). However, equality of distance distributions is not sufficient for equivalence of two general codes. Similarly, equality of weight distributions is not sufficient for (permutational or monomial) equivalence of linear codes (see, for example, [8, p. 36]). In our setting, the concepts of the Hamming distance matrix of a (linear or nonlinear) code and the weight vector of a (linear) code are useful in establishing sufficiency. The weight vector $\mathbf{w}(C)$ of a (n,m,d) linear code C is defined as the $(m \times 1)$ vector with elements the weights of the codewords (where, without loss of generality, the order of the weights is the same as the order of the codewords in matrix C). Also, the Hamming distance matrix $\mathbf{H}(C)$ of a (n,m,d) code C is defined as the $(m \times m)$ matrix with elements $\mathbf{H}[i,j] = d(u^i,u^j)$, where $d(u^i,u^j)$ is the distance between codewords u^i and u^j, $i,j \in \{1,\cdots,m\}$ (where, without loss of generality, the order of the distances between pairs of codewords follows the order of the codewords in matrix C; e.g. $d(u^1, u^2)$ is the distance between the first and second codewords (rows) in C etc).

A linear $C(n,m,d)$ code can be represented via a $(k \times n)$ *generator* matrix G (where $k \le m$), whose rows form a set of k linearly independent codewords. A generator matrix (essentially a basis of the corresponding subspace) is not unique, but as is known, the set of all $(k \times n)$ generator matrices can be obtained via left multiplication from an arbitrary generator matrix by all invertible $k \times k$-matrices $A \in GL_k(q)$, where $GL_k(q)$ is the group of all regular $k \times k$ matrices (square matrices with inverse) over F_q. A related code is the dual code $C^\perp$, defined as the $n-k$-dimensional linear code $C^\perp = \{h \in F_q^n \mid h \cdot u = 0$, for all $u \in C\}$, where $h \cdot u$ denotes the usual dot product of two vectors. Note that the dual of C is a linear code even when C is a nonlinear code, and, in general, $C \subseteq (C^\perp)^\perp$. If C is a linear code then $C = (C^\perp)^\perp$. As a result, another useful representation of a linear code is via a $(n-k \times n)$ generator matrix of the dual (called parity check matrix). Finally, the *projection* of the $m \times n$ matrix C (which corresponds to code $C(n,m,d)$) is defined to be the $m \times t$ submatrix of C consisting of columns $p_1, \cdots, p_t$ of C, denoted here by $C[p_1, \cdots, p_t]$.

3. Permutational Equivalence of Linear Codes

Recall that, permutationally equivalent linear codes are obtained from each other via a permutation of coordinates. Permutational equivalence is a natural notion of equivalence for linear codes, since a permutation of coordinates does not affect linearity. Note that, generator matrices that yield permutationally equivalent linear codes are not necessarily permutationally equivalent. As an example, the (3×6) generator matrices $G = [110000, 001100, 000011]$ and $G' = [111111, 011011, 001001]$ (over F_2) generate permutationally equivalent linear codes, but they are not permutationally equivalent. In the following, consider two (n,m,d) linear codes represented by $(m \times n)$ matrices C and C', respectively. We have the following definition.

Definition 3.1. Two linear codes C and C' with m codewords of length n are permutationally equivalent if there exists a $(m \times m)$ permutation matrix $\mathbf{R}$ and a $(n \times n)$ permutation matrix $\mathbf{P}$ such that $C' = \mathbf{R}\, C\, \mathbf{P}$.

A permutation of p $(\le n)$ coordinates (represented by $p \times p$ permutation matrix $\mathbf{P}$) is a weight preserving isometry, since it fixes the element 0 in F_q (an isometry is weight preserving if and only if it fixes 0 $(\in F_q)$; see, for example, [8, p. 69]). As a result, permutationally equivalent linear codes have the same weight distribution (a necessary condition for permutational equivalence). However, this is not a sufficient condition for permutational equivalence, as seen from the fact that linear codes with the same weight distribution might not be permutationally equivalent (see, for example, [8, p. 36]). In addition, permutationally equivalent linear codes have the same weight vector (also a necessary condition). Obtaining a sufficient condition for permutational equivalence using the invariance property of weights is not trivial. In Lemma 3.1, a necessary and sufficient condition for permutational equivalence is derived by considering weight vectors of projections.

Lemma 3.1. *Two linear codes C and C' with m codewords of length n are permutationally equivalent if and only if there exists a $(m \times m)$ permutation matrix $\mathbf{R}$ such that for $t = 1, \cdots, n$ the following holds:*

$$v\mathbf{w}(C'[s(t)]) = \mathbf{R}\,\mathbf{w}(C[t]), \tag{3.1}$$

for any $(m \times 1)$ projection $C[t]$ of C and for a permutation $(s(1), \cdots, s(n))$ of $(1, \cdots, n)$; $\mathbf{w}(C[p])$ is the weight vector of the projection consisting of coordinate p of C.

Proof. *Necessity*: If codes C and C' are permutationally equivalent, then (by Definition 3.1) there exists a permutation matrix $\mathbf{R}$ which maps C to C'. The same permutation matrix also maps the weight vector of C to the weight vector of C'. Therefore,

$$\mathbf{w}(C') = \mathbf{R}\,\mathbf{w}(C) \tag{3.2}$$

must hold. The weight vector of a code C can obtained as the sum of the weight vectors of projections $C[t]$, $t = 1, \cdots, n)$ of C, that is,

$$\mathbf{w}(C) = \sum_{t=1}^{n} \mathbf{w}(C[t]). \tag{3.3}$$

By equations (3.2) and (3.3) the following holds:

$$\mathbf{w}(C') = \mathbf{R}\left(\sum_{t=1}^{n} \mathbf{w}(C[t])\right) = \sum_{t=1}^{n} (\mathbf{R}\,\mathbf{w}(C[t])) = \sum_{t=1}^{n} \mathbf{w}(C'[s(t)]), \tag{3.4}$$

for some permutation of coordinates $(s(1), \cdots, s(n))$ of $(1, \cdots, n)$. Thus, Condition (3.1) holds.

Sufficiency: Suppose that Condition (3.1) holds. By taking the sum over all $t = 1, \cdots, n$, we have:

$$\sum_{t=1}^{n} \mathbf{w}(C'[s(t)]) = \sum_{t=1}^{n} (\mathbf{R}\,\mathbf{w}(C[t])) = \mathbf{R}\left(\sum_{i=1}^{n} \mathbf{w}(C[t])\right). \tag{3.5}$$

By equation (3.3) and the left and right hand parts of equation (3.5) we obtain:

$$\mathbf{w}(C') = \mathbf{R}\,\mathbf{w}(C).$$

Thus, permutation matrix $\mathbf{R}$ maps C to C', up to a permutation of coordinates $((s(1), \cdots, s(n))$ of $(1, \cdots, n))$, which can be found by inspection. Thus, the two codes are permutationally equivalent. □

Theorem 3.1 below is a generalization of Lemma 3.1, and provides an alternative necessary and sufficient condition for permutational equivalence based on projections with (fixed) length t, where $1 < t < n - 1$.

Theorem 3.1. *Two linear codes C and C' are permutationally equivalent if and only if there exists a permutation matrix $\mathbf{R}$ such that, for a given t $(1 \le t \le n - 1)$ the following holds:*

$$\mathbf{w}(C'[s(j_1), \cdots, s(j_t)]) = \mathbf{R}\,\mathbf{w}(C[j_1, \cdots, j_t]), \tag{3.6}$$

for any $m \times t$ projection $C[j_1, \cdots, j_t]$ of C, and for a permutation $(s(1), \cdots s(n))$ of coordinates $(1 \cdots n)$, where $\mathbf{w}(C[p_1, \cdots, p_t])$ is the weight vector of $m \times t$ projection $C[p_1, \cdots, p_t]$ of C (based on coordinates $\{p_1, \cdots, p_t\}$ from $\{1, \cdots, n\}$).

Proof. *Necessity*: Follows from equation (3.3) and the necessary part of Lemma 3.1.

Sufficiency (by contradiction): Assume that Condition (3.6) holds but the codes are not permutationally equivalent. Since the codes are not permutationally equivalent by the necessary part of Lemma 3.1 there must exist (at least one) $m \times 1$ projection of C' for which Condition (3.1) does not hold. Consider an $m \times t$ projection of C' which includes such $m \times 1$ projection(s). By applying equation (3.3) it can be shown that Condition (3.6) cannot hold for such $m \times t$ projection. But this contradicts our assumption. Therefore, Condition (3.1) must hold for all $m \times 1$ projections, and the result follows from the sufficiency part of Lemma 3.1. □

4. Monomial Equivalence of Linear Codes

Another natural notion of equivalence for linear codes is monomial equivalence. The term "monomial" comes from the fact that a permutation of coordinates combined with permutations of the non-zero symbols in one or more coordinates (see Section 1) is a *monomial transformation*. A useful aspect of a monomial transformation is that it can be expressed via a *monomial matrix*, that is, a matrix which has exactly one non-zero element of F_q in each row and column. Note that, a monomial transformation is invertible, and composites and inverses of monomial transformations are also monomial transformations (see, for example, [15]). In this case, two linear codes (say C and C'), are monomially equivalent if and only if their generator matrices (say G and G', respectively) are also monomially equivalent (see, for example, [2] and [11]). In fact, the definition and theorems below apply in the same way to generator matrices (G and G') without any modifications. We have the following definition.

Definition 4.1. Two linear codes C and C$'$ are monomially equivalent if there exists a permutation $\mathbf{R}$, a diagonal matrix $\mathbf{L}$ with diagonal elements non-zero elements of F_q and a permutation $\mathbf{P}$, such that $C' = \mathbf{R}\, C\, \mathbf{A}$, where $\mathbf{A} = \mathbf{L}\,\mathbf{P}$ is a monomial matrix.

Since a monomial transformation is a distance preserving transformation (see, for example, [8]), monomially equivalent linear codes have the same weight (distance) distribution (a necessary condition for monomial equivalence). For the same reason, monomially equivalent linear codes have the same weight vector (also a necessary condition). In this case though, we cannot establish sufficiency by considering weight vectors of projections as before. Nevertheless, we can obtain a necessary and sufficient condition for monomial equivalence by considering distance matrices of projections, as shown in Theorem 4.1.

Theorem 4.1. *Two linear codes C and C' are monomially equivalent if and only if there exists a permutation $\mathbf{R}$ and a permutation $(s(1),\cdots,s(n))$ of $(1,\cdots,n)$ such that, for each $t = 1,\cdots,n$, the following holds:*

$$\mathbf{H}(C'[s_t]) = \mathbf{R}\,(\mathbf{H}(C[t])\,\mathbf{R}^T, \tag{4.1}$$

for any projection $C[t]$ of C, where $\mathbf{H}(C[p])$ is the distance matrix of the projection consisting of coordinate p of C, and $\mathbf{R}^T$ denotes the transpose of $\mathbf{R}$.

Proof. Necessity: Assume that C and C' are monomially equivalent. By definition, there exists a permutation $\mathbf{R}$ that transforms C to C', which also satisfies:

$$\mathbf{H}(C') = \mathbf{R}\, \mathbf{H}(C)\, \mathbf{R}^T. \tag{4.2}$$

Also, the distance matrix of a code C can obtained as the sum of the distance matrices of its projections $C[t]$, $t = 1, \cdots, n$, as follows:

$$\mathbf{H}(C) = \sum_{t=1}^{n} \mathbf{H}(C[t]). \tag{4.3}$$

By equations (4.2) and (4.3), we have:

$$\mathbf{H}(C') = \mathbf{R} \left(\sum_{t=1}^{n} \mathbf{H}(C[t]) \right) \mathbf{R}^T = \sum_{t=1}^{n} \left(\mathbf{R}\, \mathbf{H}(C[t])\, \mathbf{R}^T \right) = \sum_{t=1}^{n} \mathbf{H}(C'[s(t)]). \tag{4.4}$$

From the last two terms in equation (4.4), Condition (4.1) holds for a permutation $(s(1), \cdots, s(n))$ of $(1, \cdots, n)$.

Sufficiency: Suppose that Condition (4.1) holds. By taking the sum over all $t = 1, \cdots, n$, we obtain the following:

$$\sum_{t=1}^{n} \mathbf{H}(C'[s(t)]) = \sum_{t=1}^{n} \left(\mathbf{R}\, \mathbf{H}(C[t])\, \mathbf{R}^T \right) = \mathbf{R} \left(\sum_{i=1}^{n} \mathbf{H}(C[t]) \right) \mathbf{R}^T. \tag{4.5}$$

By equation (4.3) and the left-hand and right-hand parts of equation (4.5), we can write:

$$\mathbf{H}(C') = \mathbf{R}\, \mathbf{H}(C)\, \mathbf{R}^T.$$

Therefore, $\mathbf{R}$ along with permutation $(s(1), \cdots, s(n))$ of $(1, \cdots, n)$ transforms C to C', up to a permutation of non-zero symbols in one or more coordinates. The last permutation exist since the weights of codewords remain the same under Condition (4.1) and can be found by inspection. □

Theorem 4.2 below is a generalization of Theorem 4.1, and provides an alternative necessary and sufficient condition for monomial equivalence of two linear codes based on distance matrices of projections with (fixed) length t $(1 < t < n - 1)$.

Theorem 4.2. *Two linear codes C and C' are monomially equivalent if and only if there exists a permutation $\mathbf{R}$ and a permutation $(s(1), \cdots, s(n))$ of $(1 \cdots, n)$ such that, for a given (fixed) t, $1 \le t \le n - 1$, the following condition holds:*

$$\mathbf{H}(C'[s(j_1), \cdots, s(j_t)]) = \mathbf{R}\, \mathbf{H}(C[j_1, \cdots, j_t])\, \mathbf{R}^T, \tag{4.6}$$

for any projection $C[j_1, \cdots, j_t]$ of C, where $\mathbf{H}(C[p_1, \cdots, p_t])$ is the distance matrix of projection $C[p_1, \cdots, p_t]$ of C based on coordinates $\{p_1, \cdots, p_t\}$ from $\{1, \cdots, n\}$.

Proof. Necessity: Follows from equation (4.3) and the necessary part of Theorem 4.1.

Sufficiency (by contradiction): Assume that Condition (4.6) holds but the codes are not monomially equivalent. Since the codes are not monomially equivalent by the necessary part of Theorem 4.1, there exists (at least one) projection with length one of C', for which Condition (4.1) does not hold. Consider a projection of C' with length t which includes such projection(s). Using equation (4.3), it can be shown that Condition (4.6) does not hold for such projection with length t. But this contradicts our assumption. Therefore, Condition (4.1) must hold for any projection with length one. Thus, result follows from the sufficiency part of Theorem 4.1. □

5. Equivalence of Nonlinear Codes

As mentioned in Section 1, a permutation of coordinates combined with arbitrary permutations of symbols in one or more coordinates can affect linearity. Also, there is no general "monomial theorem" for nonlinear codes (see, for example, [4], [5] and [9]). However, a way to deal with permutations of symbols (in one or more coordinates) is by viewing such permutations as *translations*. In general, a translation is a geometric transformation that moves every element in a vector space by the same amount in a given direction (given by a *translation vector*). In our setting, a permutation of the m symbols $x_{j_1},\cdots,x_{j_m}$ in the x_jth coordinate (of C) can be written as multiplication of a translation matrix $\mathbf{T}(a)$ by $(x_{j_1},\cdots,x_{j_m},1)$, where

$$\mathbf{T}(a)=\begin{bmatrix}1 & 0 & \cdots & 0 & a_1\\ 0 & 1 & \cdots & 0 & a_2\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & \cdots & 1 & a_m\\ 0 & 0 & \cdots & 0 & 1\end{bmatrix},$$

and $(x_1,\cdots,x_m,1)$ are the *homogeneous coordinates* of vector $x=(x_1,\cdots,x_m)$; $a=(a_1,\cdots,a_m,1)$ is a translation vector with $a_1,\cdots,a_m\in F_q$. Note that, the inverse of a translation matrix is obtained by reversing the direction of a translation vector, and the product of two translation matrices is given by adding the corresponding translation vectors. Note also that, $\mathbf{T}(\mathbf{0})=\mathbf{I}$ gives the identity permutation (where $\mathbf{0}$ denotes the $(m+1)\times 1$ zero vector and $\mathbf{I}$ the $(m+1)\times(m+1)$ identity matrix). We have the following definition.

Definition 5.1. Two nonlinear codes C and C' are equivalent, if there exists a permutation $\mathbf{R}$, a permutation $\mathbf{P}$ and matrices $\mathbf{T}(a_j)$, $j=1,\cdots,n$, such that $\mathbf{T}(a_j)$ is a translation matrix, which yield the (homogeneous coordinates of) jth coordinate (say $\mathbf{x}_j$, $j=1,\cdots,n$) of C' when multiplied by the (homogeneous coordinates of) jth coordinate of $\mathbf{R}\,C\,\mathbf{P}$.

Since a permutation of coordinates combined with permutations of symbols in one or more coordinates preserves the distances between codewords (see also, for example, [8]), equivalent nonlinear codes have the same distance distribution — a necessary but not a sufficient condition for equivalence of nonlinear codes. For example, the ternary codes $C=\{000,011,022\}$ and $C'=\{000,022,220\}$ have the same distance distribution, however they are not equivalent. For the same reason, equivalent nonlinear codes have the same distance matrix (also a necessary but not a sufficient condition for equivalence). In this case, by extending the results of Theorem 4.1, a necessary and sufficient condition for equivalence of nonlinear codes can be obtained, as in Theorem 5.1 below.

Theorem 5.1. *Two nonlinear codes C and C' are equivalent if and only if there exists a permutation $\mathbf{R}$ and a permutation $(s(1),\cdots,s(n))$ of $(1,\cdots,n)$ such that, for $t=1,\cdots,n$*

the following holds

$$\mathbf{H}(C'[s_t]) = \mathbf{R}\,\mathbf{H}(C[t])\,\mathbf{R}^T, \tag{5.1}$$

for any projection $C[t]$ *of* C*, where* $\mathbf{H}(C[p])$ *is the distance matrix of projection consisting of coordinate* p *of* C.

Proof. Necessity: Equivalent codes have the same distances between all pairs of codewords, and thus, the same distance matrix up to permutation of rows. Result follows from the necessary part of Theorem 4.1.

Sufficiency: By the sufficiency part of Theorem 4.1, there exists a permutation $\mathbf{R}$ and a permutation $\mathbf{P}$ that transform C to C'. Thus, the two codes are equivalent up to a permutation of symbols (equivalently elements $c_1, \cdots, c_m$ in a translation matrix $\mathbf{T}$) in one or more columns, which can be found by inspecting the columns of codes C' and $\mathbf{R}\,C\,\mathbf{P}$. □

Next, in Theorem 5.2, Condition 4.6 in Theorem 4.2 is extended to obtain a more general necessary and sufficient condition for equivalence of two nonlinear codes.

Theorem 5.2. *Two codes* C *and* C' *are equivalent if and only if there exists a permutation* $\mathbf{R}$ *and a permutation* $(s(1), \cdots, s(n))$ *of* $(1, \cdots, n)$ *such that, for a given* t*,* $1 \le t \le n-1$*, the following condition holds:*

$$\mathbf{H}(C'[s(j_1), \cdots, s(j_t)]) = \mathbf{R}\,\mathbf{H}(C[j_1, \cdots, j_t])\,\mathbf{R}^T, \tag{5.2}$$

for any projection $C[j_1, \cdots, j_t]$ *of* C*, where* $\mathbf{H}(C[p_1, \cdots, p_t])$ *is the distance matrix of the projection based on coordinates* $\{p_1, \cdots, p_t\}$ *from* $\{1, \cdots, n\}$.

Proof. Necessity: Follows from the necessary part of Theorem 4.2.

Sufficiency: By the sufficiency part of Theorem 4.2, there exists a row and a column permutation that transform C to C'. Thus the two codes are equivalent up to a permutation of symbols (equivalently elements $c_1, \cdots, c_m$ in a transformation matrix $\mathbf{T}$) in one or more columns, which can be found by inspecting the columns of codes C' and $\mathbf{R}\,C\,\mathbf{P}$. □

6. Discussion and Conclusions

The necessary and sufficient conditions for (linear or nonlinear) error-correcting codes presented in this paper provide some new methods for detecting isometric error-correcting codes. In particular, Theorem 3.1 provides a fast method for detecting permutational equivalence of two linear codes, by considering projections with a small length. Similarly, Theorem 4.2 and Theorem 5.2 provide a fast method for detecting momomial equivalence of linear codes or equivalence of nonlinear codes, respectively. As an example, non permutational equivalence of Hadamard (linear) codes can be detected from projections with length equal to three. This can be readily seen from the fact that Hadamard codes (which are obtained from Hadamard matrices ([6]) are orthogonal arrays of strength two (see [14]). Therefore, by the definition of an orthogonal array, there are projections onto dimension three which are not identical. Thus, an efficient way to detect non permutationally equivalent Hadamard codes is using $t = 3$ in Theorem 3.1. An algorithm that implements the various conditions discussed in this paper is available from the author by request.

In view of practical considerations, when the dual of a linear (n,m,d) code is a smaller code $(n-k<k)$, permutational equivalence of linear codes can be detected faster by applying Lemma 3.1 or Theorem 3.1 (with no modifications) to their dual codes. Note that, if a linear code C is permutationally equivalent to C', then its dual code is permutationally equivalent to the dual of C', $C^{'\perp}$ (this can be verified using $C' = \mathbf{R}\, C\, \mathbf{P}$, uniqueness of an orthogonal subspace, and the fact that a permutation matrix is an orthogonal matrix). Note also that, if a linear code C is self-dual so is C', if the two codes are permutationally equivalent. Also, when the rank of a linear code (k) is much smaller than the number of its codewords (m), monomial equivalence can be detected faster using generator matrices (recall that Theorems 4.1 and 4.2 apply in the same way to the generator matrices G, G' of C and C', respectively). Furthermore, if two linear codes C and C' are monomially equivalent, then their dual codes are also monomial equivalent (this can be shown using equation $C' = \mathbf{R}\, C\, \mathbf{P}$, uniqueness of an orthogonal subspace, and the fact that a permutation matrix is an orthogonal matrix). Thus, when the duals of two linear codes are smaller, monomial equivalence can be detected faster by applying Theorem 4.1 or 4.2 (with no modifications) to their duals or their parity check matrices.

In the case of linear codes over a finite field F_q^r, with q^r a power of a prime number, the more general notion of semilinear equivalence arises from isometries that map a subspace to a subspace, and include automorphisms of a code (see, for example, [8], [9] and [13]). In our case, where q is a prime, semilinear equivalence coincides with monomial equivalence (notice that F_q has only the trivial automorphism). Note that, for binary linear codes permutational, monomial and semilinear equivalence coincide (see also, for example [7, p. 20] and [1, p. 30]). All our results about linear error-correcting codes over F_q can be extended to linear error-correcting codes over F_q^r (future research).

Acknowledgment

The author is immensely grateful to the referees, and to two anonymous reviewers of an earlier version of this manuscript, whose valuable comments helped improve the presentation of this paper.

Competing Interests

The author declares that she has no competing interests.

Authors' Contributions

The author wrote, read and approved the final manuscript.

References

[1] A. Betten, M. Braun, H. Fripertinger, A. Kerber, A. Kohnert and A. Wasserman, Error-correcting linear codes-classification by isometry and applications, *Algorithms and Computation in Mathematics* (18), Springer, New York (2006).

[2] K. Bogart, D. Goldberg and J. Gordon, An elementary proof of the MacWilliams theorem on equivalence of codes, *Information and Control* **37** (1) (1978), 19 – 22.

[3] P.G. Bonneau, Poids et equivalence des codes linaires, *Informatique Theorique et Applications* **21** (3) (1987), 331 – 339.

[4] C.J. Colbourn and J.H. Dinitz (eds.), *The CRC Handbook of Combinatorial Designs*. CRC Press Series on Discrete Mathematics and its Applications, Chapman and Hall/CRC Press (2006).

[5] I. Constantinescu and W. Heise, On the concept of code-isomorphy, *Journal of Geometry* **57** (1996), 63 – 69.

[6] J. Hadamard, Resolution d'une question relative aux determinants. *Bulletin des Sciences Mathematiques* **17** (1893), 240 – 246.

[7] W.G. Huffman and V. Pless, *Fundamentals of Error-correcting Codes*, Cambridge University Press, Cambridge, UK (2003).

[8] P. Kaski and P.R.J. Ostergard, Classification Algorithms for Codes and Designs, *Algorithms and Computations in Mathematics* (15), Springer, New York (2006).

[9] C. Lam, Finding error-correcting codes using computers, in *NATO Advanced Study Institute on Information Security and Related Combinatorics: Information Security, Coding Theory and Related Combinatorics*, D. Crnkovic' and V. Tonchev (eds.), IOS Press, Amsterdam, 278–284 (2011).

[10] Z. Liu and Z. Sun, On the equivalence of linear codes, *Applicable Algebra in Engineering, Communication and Computing* **22** (2) (2011), 137 – 145.

[11] F.J. MacWilliams, *Combinatorial Problems of Elementary Abellian Groups*, Ph.D Dissertation, Radcliffe College, Cambridge, Mass. (1962).

[12] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, North Holland Mathematical Library, New York (1977).

[13] J. Moori, Finite groups, designs and codes, in *NATO Advanced Study Institute on Information Security and Related Combinatorics: Information Security, Coding Theory and Related Combinatorics*, D. Crnković and V. Tonchev (eds.), IOS Press, Amsterdam, 202 – 230 (2011).

[14] W.W. Peterson, *Error-Correcting Codes*, MIT Press, Cambridge, Mass. (1961).

[15] H.N. Ward aand J.A. Wood, Characters and the equivalence of codes, *Journal of Combinatorial Theory, Series A* **73** (2) (1996), 348 – 352.