Comparing decoding methods for quaternary linear codes ⋆

R. D. Barrolleta, [1] J. Pujol, [2] and M. Villanueva [3]

Departament d'Enginyeria de la Informació i de les Comunicacions
Universitat Autònoma de Barcelona
Cerdanyola del Vallès, Spain

Abstract

Permutation decoding is a technique which involves finding a subset S, called PD-set, of the permutation automorphism group of a code C. Constructions of small PD-sets for partial decoding for two families of $\mathbb{Z}_4$-linear codes (Hadamard and Kerdock) are given. Moreover, different decoding methods for $\mathbb{Z}_4$-linear codes are compared by showing their performance applied to these two families.

Keywords: $\mathbb{Z}_4$-linear codes, PD-sets, decoding, Hadamard codes, Kerdock codes.

1 Introduction

Any nonempty subset C of $\mathbb{Z}_2^n$ is a binary code and a subgroup of $\mathbb{Z}_2^n$ is a *binary linear code*. Equivalently, a subgroup $\mathcal{C}$ of $\mathbb{Z}_4^n$ is a *quaternary linear code*. Quaternary linear codes can be seen as binary codes under the Gray map

⋆ Research partially supported by the Spanish MINECO under Grant TIN2013-40524-P, and by the Catalan AGAUR under Grant 2014SGR-691.

[1] Email: rolanddavid.barrolleta@uab.cat
[2] Email: jaume.pujol@uab.cat
[3] Email: merce.villanueva@uab.cat

$\Phi : \mathbb{Z}_4^n \to \mathbb{Z}_2^{2n}$ defined as $\Phi((y_1, \ldots, y_n)) = (\phi(y_1), \ldots, \phi(y_n))$, where $\phi(0) = (0,0)$, $\phi(1) = (0,1)$, $\phi(2) = (1,1)$, $\phi(3) = (1,0)$, for all $y = (y_1, \ldots, y_n) \in \mathbb{Z}_4^n$. If $\mathcal{C}$ is a quaternary linear code, the binary code $C = \Phi(\mathcal{C})$ is a $\mathbb{Z}_4$-*linear code*. Since $\mathcal{C}$ is a subgroup of $\mathbb{Z}_4^n$, it is isomorphic to $\mathbb{Z}_2^\gamma \times \mathbb{Z}_4^\delta$ and we say that $\mathcal{C}$ (or equivalently the $\mathbb{Z}_4$-linear code $C = \Phi(\mathcal{C})$) is of type $2^\gamma 4^\delta$ [7].

The *Hamming weight* $\mathrm{wt}_H(v)$ of $v \in \mathbb{Z}_2^n$ is the number of nonzero coordinates in v. The *Hamming distance* $d_H(u,v)$ between $u, v \in \mathbb{Z}_2^n$ is the number of coordinates in which u and v differ, that is, $d_H(u,v) = \mathrm{wt}_H(u+v)$. The *Lee weight* $\mathrm{wt}_L(u)$ of $u \in \mathbb{Z}_4^n$ is the addition of the weights of its coordinates, whereas the *Lee distance* $d_L(u,v)$ between two words $u, v \in \mathbb{Z}_4^n$ is $d_L(u,v) = \mathrm{wt}_L(u-v)$. The elements of $\mathbb{Z}_4$ have the following Lee weights: $\mathrm{wt}_L(0) = 0$, $\mathrm{wt}_L(1) = \mathrm{wt}_L(3) = 1$ and $\mathrm{wt}_L(2) = 2$. The minimum (Lee) distance of $\mathcal{C}$ is $d(\mathcal{C}) = \min\{d_L(u,v) : u, v \in \mathcal{C}, u \neq v\}$, which coincides with the minimum (Hamming) distance of $C = \Phi(\mathcal{C})$, $d(C) = \min\{d_H(u,v) : u, v \in C, u \neq v\}$.

Magma is a software package designed to solve computationally hard problems. It supports the basic facilities for linear codes over integer residue rings and Galois rings [6], including additional functionality for quaternary linear codes. A new package that expands the current functionality for these codes, including functions to decode, has been developed by the authors [2]. This package and a manual with the description of all implemented functions can be downloaded from `http://ccsg.uab.cat`.

In this paper, we give PD-sets, which are necessary to apply the permutation decoding, for two well known families of $\mathbb{Z}_4$-linear codes (Hadamard and Kerdock) having a high error correcting capability. Then, we compare the permutation decoding with other decoding methods. We will show their performance by using the functions that we have implemented in Magma.

2 Decoding Methods

Let $\mathcal{C}$ be a quaternary linear code of length n, type $2^\gamma 4^\delta$, and minimum distance d. This section describes four different algorithms for decoding vectors from the ambient space over $\mathbb{Z}_4$ or the binary space under the Gray map.

Syndrome Decoding: This is a method for linear codes [10], which can be applied to quaternary linear codes. The syndrome of $u \in \mathbb{Z}_4^n$ with respect to a parity check matrix $\mathcal{H}$ of $\mathcal{C}$ is $s = \mathcal{H}u$. Note that $\mathcal{C}$ consists of all vectors whose syndrome is equal to the zero vector. Every vector in $\{0,2\}^\gamma \times \mathbb{Z}_4^{n-\gamma-\delta}$ is a syndrome and there is a one-to-one correspondence between cosets of $\mathcal{C}$ and its syndromes. Let $\mathcal{C}_s$ be the coset of $\mathcal{C}$ consisting of all vectors in $\mathbb{Z}_4^n$ having syndrome s. This method consists of computing a table pairing each possible

syndrome s with an error vector of minimum weight e_s in $\mathcal{C}_s$. After receiving $u \in \mathbb{Z}_4^n$, compute its syndrome $s = \mathcal{H}u$, and u is decoded as $c = u - e_s$.

Lifted Decoding: This is a general method for linear codes over Galois rings [1], which consists of lifting decoding algorithms for two binary linear codes C_0 and C_1, known as the residue and torsion codes of $\mathcal{C}$. Let t_0 and t_1 be the error-correcting capability of C_0 and C_1, respectively. Assume the received vector $u = c + e$, where $c \in \mathcal{C}$ and $e \in \mathbb{Z}_4^n$ is the error vector. Then, all error vectors e such that $\tau_1 + \tau_3 \leq t_0$ and $\tau_2 + \tau_3 \leq t_1$, where τ_i is the number of occurrences of i in e, can be corrected.

Coset Decoding: This method can be applied to any code over a finite field (not necessarily linear) [11]. It is based on representing $C = \Phi(\mathcal{C})$ as the union of cosets of a linear subcode K. Specifically, $C = \bigcup_{i=0}^{t}(K + c_i)$, where $K = \{x \in C : x + C = C\}$, $c_0, c_1, \ldots, c_t$ are coset representatives of C with respect to K and c_0 is the zero codeword. If C is linear, then $C = K$. After receiving a vector $u \in \mathbb{Z}_4^n$, consider the linear codes $K_0 = K \cup (K + \Phi(u))$, $K_1 = K \cup (K + c_1 + \Phi(u))$, ..., $K_t = K \cup (K + c_t + \Phi(u))$. If $d(\cup_{i=0}^{t} K_i) < d(K)$, u is decoded as the codeword c such that $\Phi(c) = \Phi(u) + e$, where e is a word of minimum weight of $\cup_{i=0}^{t} K_i$. This method is based on computing the minimum weight of $t + 1$ linear codes K_i, $i \in \{0, \ldots, t\}$. Although it is known that the problem of computing the minimum weight for linear codes is NP-hard, the Brouwer-Zimmermann algorithm can be used [12].

Permutation Decoding: An information set I for $C = \Phi(\mathcal{C})$ is a set of $\gamma + 2\delta$ coordinate positions such that $|C_I| = 2^{\gamma+2\delta}$, where $C_I = \{v_I : v \in C\}$ and v_I is the vector v restricted to I. The permutation automorphism group $\mathrm{PAut}(C)$ of C is the group generated by all permutations that preserve the set of codewords. Permutation decoding was first introduced for linear codes [10] and it can be adapted to work with $\mathbb{Z}_4$-linear codes [5]. Let C be systematic t-error-correcting code with information set I. The method consists on moving all errors in a received vector u out of I by using an automorphism of C. It strongly depends on the existence of some special subsets of $\mathrm{PAut}(C)$, called PD-sets. Specifically, a subset $S \subseteq \mathrm{PAut}(C)$ is said to be an s-PD-set C if every s-set of coordinate positions is moved out of I by at least one element of S, where $1 \leq s \leq t$. When $s = t$, S is said to be a PD-set.

3 PD-Sets and Performance Analysis

A *binary Hadamard code* of length n has $2n$ codewords and minimum distance $n/2$ [10]. The quaternary linear codes such that, under the Gray map, give a binary Hadamard code are called *quaternary linear Hadamard codes*. These

codes have been studied and classified in [9].

The permutation automorphism group $\mathrm{PAut}(\mathcal{H}_{\gamma,\delta})$ of a quaternary linear Hadamard code $\mathcal{H}_{\gamma,\delta}$ of length $\beta = 2^{m-1}$ and type $2^\gamma 4^\delta$, $\gamma = m - 2\delta + 1$, can be regarded as a certain subset of $\mathrm{GL}(\delta+\gamma, \mathbb{Z}_4)$ [3]. Let $\mathcal{M} \in \mathrm{PAut}(\mathcal{H}_{\gamma,\delta})$ and m_i be the ith row of $\mathcal{M}$. Define $\mathcal{M}^*$ as the matrix where the first row is m_1 and the ith row is $m_1 + m_i$ for $i \in \{2, \ldots, \delta\}$ and $m_1 + 2m_i$ for $i \in \{\delta+1, \ldots, \delta+\gamma\}$. The question of whether a subset of $\mathrm{PAut}(\mathcal{H}_{\gamma,\delta})$ leads to a s-PD-set of size $s+1$ for $H_{\gamma,\delta} = \Phi(\mathcal{H}_{\gamma,\delta})$ is addressed by searching a set $\mathcal{P} = \{\mathcal{M}_i \ : \ 0 \le i \le s\} \subseteq \mathrm{PAut}(\mathcal{H}_{\gamma,\delta})$ such that no two matrices $(\mathcal{M}_i^{-1})^*$ and $(\mathcal{M}_j^{-1})^*$ for $i \neq j$ have a row in common [3]. In this paper, we show the existence of $\mathcal{P}$ for any $\mathcal{H}_{0,\delta}$ by giving its explicit construction. We follow a similar technique to the one described for simplex codes in [8]. An s-PD-set for $\mathcal{H}_{\gamma,\delta}$, $\gamma \neq 0$, is provided by applying a recursive construction over $\mathcal{P}$ [3].

Let $\mathcal{R} = \mathrm{GR}(4^{\delta-1})$ be the Galois extension of dimension $\delta - 1$ over $\mathbb{Z}_4$. Let $f(x) \in \mathbb{Z}_2[x]$ be a primitive polynomial of degree $\delta - 1$. Let $\ell = 2^{\delta-1} - 1$. There is a unique primitive basic irreducible polynomial $h(x)$ dividing $x^\ell - 1$ in $\mathbb{Z}_4[x]$. Let α be a root of $h(x)$. It is well known that any $r \in \mathcal{R}$ can be written uniquely as $r = a + 2b$, where $a, b \in \{0, 1, \alpha, \ldots, \alpha^{\ell-1}\}$. We take $\mathcal{R}$ as the following ordered set $\mathcal{R} = \{r_1, \ldots, r_{4^{\delta-1}}\} = \{0 + 2\cdot 0, \ldots, \alpha^{\ell-1} + 2\cdot 0, \ldots, 0 + 2\cdot\alpha^{\ell-1}, \ldots, \alpha^{\ell-1} + 2\cdot\alpha^{\ell-1}\}$. For all $i \in \{0, \ldots, f_{0,\delta} = \lfloor (2^{2\delta-2} - \delta)\delta^{-1} \rfloor\}$, we consider the $\delta \times \delta$ quaternary matrices

$$\mathcal{N}_i^* = \left(\begin{array}{c|c} 1 & r_{\delta i+1} \\ \vdots & \vdots \\ 1 & r_{\delta(i+1)} \end{array}\right). \tag{1}$$

Theorem 3.1 *The set $\mathcal{P} = \{\mathcal{M}_i \ : \ 0 \le i \le f_{0,\delta}\}$, where $\mathcal{M}_i = \mathcal{N}_i^{-1}$, satisfies that no two matrices $(\mathcal{M}_i^{-1})^*$ and $(\mathcal{M}_j^{-1})^*$ for $i \neq j$ have a row in common.*

This section aims to provide a comparison study of the performance of the four decoding method described for the different $\mathbb{Z}_4$-linear Hadamard codes $H_{0,\delta}$. We use the $f_{0,\delta}$-PD-set $\mathcal{P}$ given by Theorem 3.1 when permutation decoding is executed. The time spent generating $\mathcal{P}$ is included in all tests. Note that $f_{0,\delta}$ is the greater s for which we can find s-PD-sets of size s+1 for $H_{0,\delta}$.

Although creating the syndrome table is a one-time task, which is carried out before decoding the received vectors, sometimes it can be difficult to create and store it. Moreover, if it contains many elements, it can be hard to find the corresponding error vector from a given syndrome. Thus, syndrome and lifted decoding are not suitable for decoding $\mathbb{Z}_4$-linear Hadamard codes $H_{\gamma,\delta}$ with $\delta \ge 4$. Note that to correct up to s errors, the total number of syndromes is $2(\sum_{i=0}^{s} \binom{\beta}{i})$ for lifted decoding and $\sum_{i=0}^{s} \binom{2\beta}{i}$ for syndrome decoding.

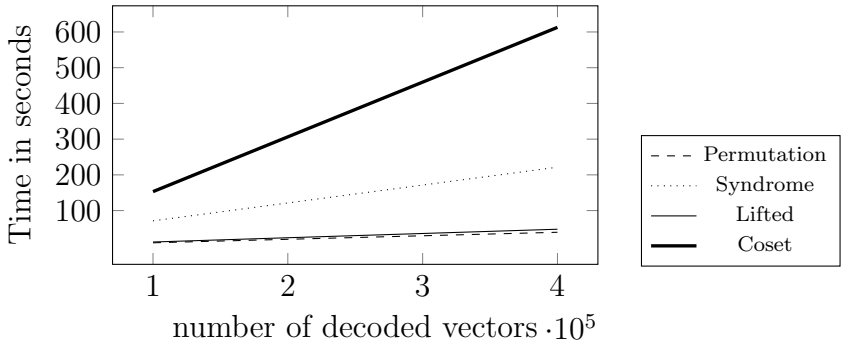

Fig. 1. Time for decoding using the $\mathbb{Z}_4$-linear Hadamard code $H_{0,3}$.

Example 3.2 Let $H_{0,3}$ be the (nonlinear) $\mathbb{Z}_4$-linear Hadamard code of length 32 and type $2^0 4^3$. Figure 1 displays the time in seconds to decode random received vectors with at most 4 errors by using permutation, syndrome, lifted, and coset decoding. Despite lifted decoding has a similar performance (in terms of time) than permutation decoding, it is the unique method that cannot correct all received vectors since $t_0 = t_1 = 3$. We observe a negligible amount of time generating the 4-PD-set obtained by Thereom 3.1, fact that facilitates permutation decoding to be the best method (both in terms of time and correctly decoded received vectors) when trying to correct up to 4 errors.

Example 3.3 Let $H_{0,\delta}$ be the $\mathbb{Z}_4$-linear Hadamard code of length $2^{2(\delta-1)}$ and type $2^0 4^\delta$. The following table shows the time in seconds to decode 100,000 random received vectors with at most $f_{0,\delta}$ errors by using permutation and coset decoding for $\delta \in \{3, \ldots, 7\}$. Permutation decoding has better performance than coset decoding for each $H_{0,\delta}$.

δ	3	4	5	6	7
Permutation decoding	9.85	19.13	64.92	243.97	1079.62
Coset decoding	153.08	26.53	101.75	628.43	5879.25

Example 3.4 Let $\mathcal{K}(m)$ be the quaternary Kerdock code of length 2^m and type 4^{m+1} [7]. Suppose that $2^m - 1$ is not a prime number. We have implemented a construction to obtain s-PD-sets of size $s+1$ for $K(m) = \Phi(\mathcal{K}(m))$, where $s+1$ is the greatest divisor of $2^m - 1$ such that $s + 1 \leq 2^m/(m+1)$ [4]. We obtain similar results to the Hadamard codes studied in this section when applying the different decoding methods. Lifted and syndrome decoding become impracticable for $m \geq 6$. According to the implementation in Magma, permutation decoding is the method with the best behaviour.

Tests have been performed in MAGMA version V2.21-4, running on a server with an Intel Xeon processor (clock speed 3.30GHz).

References

[1] Babu, N.S., and K.H. Zimmermann, *Decoding of linear codes over Galois rings*, IEEE Trans. Inform. Theory. **47**(2001), 1599–1603.

[2] Barrolleta, R., J. Pernas, J. Pujol, and M. Villanueva, "Codes over $\mathbb{Z}_4$. A MAGMA package," Autonomous University of Barcelona, 2015. http://ccsg/uab.cat.

[3] Barrolleta, R., and M. Villanueva, *Partial permutation decoding for binary linear and $\mathbb{Z}_4$-linear Hadamard codes*, (2015), arXiv:1512.01839.

[4] Barrolleta, R., and M. Villanueva, PD-*sets for $\mathbb{Z}_4$-linear codes: Hadamard and Kerdock codes*, in Proceedings of the IEEE International Symposium on Information Theory, 2016.

[5] Bernal, J. J.,, J. Borges, C. Fernández-Córboda, and M. Villanueva, *Permutation decoding of $\mathbb{Z}_2\mathbb{Z}_4$-linear codes*, Des. Codes and Cryptogr. **76**(2015), 269–277.

[6] Cannon, J. J., and W. Bosma (Eds.), "Handbook of MAGMA Functions," Edition 2.13, 4350 pages, 2006. http://magma.maths.usyd.edu.au/magma/.

[7] Hammons, A.R., P.V. Kumar, A.R. Calderbank, N.J.A. Sloane, and P. Solé, *The $\mathbb{Z}_4$-linearity of kerdock, preparata, goethals and related codes,* IEEE Trans. Inform. Theory **40**(1994), 301–319.

[8] Fish, W., J. D. Key, and E. Mwambene, *Partial permutation decoding for simplex codes,* Adv. Math. Commun. **6**(2012), 505–516.

[9] Krotov, D. S., *$\mathbb{Z}_4$-linear Hadamard and extended perfect codes,* Electron. Note Discr. Math. **6**(2001), 107–112.

[10] MacWilliams, F. I., and N. J. Sloane, "The Theory of Error-Correcting Codes," North-Holland, New York, 1977.

[11] Villanueva, M., F. Zeng, and J. Pujol, *Efficient representation of binary nonlinear codes: constructions and minimum distance computation*, Des. Codes and Cryptogr. **76**(2015), 3–21.

[12] Zimmerman, K.-H. "Integral Hecke Modules, Integral Generalized Reed-Muller Codes, and Linear Codes," Tech. Rep. 3-96, Technische Universität Hamburg-Harburg, 1996.