Journal of Computer Science

Original Research Paper

Efficient Decoding Algorithm for Binary Quadratic Residue Codes Using Reduced Permutation Sets

[1]Hamza Boualame, [1]Mostafa Belkasmi and [2]Idriss Chana

[1]Department of Web and Mobile Engineering, ENSIAS College of Engineering, Mohammed V University, in Rabat, Morocco
[2]Department of Computer Science, Ecole Supérieure de Technologie Moulay-Ismail University, Meknes, Morocco

Article history
Received: 19-12-2022
Revised: 21-02-2023
Accepted: 03-03-2023

Corresponding Author:
Hamza Boualame
Department of Web and Mobile Engineering, ENSIAS College of Engineering, Mohammed V University, in Rabat, Morocco
Email: boualame.hamza@gmail.com

Abstract: The Quadratic Residue (QR) codes have a rich mathematic structure. Unfortunately, their Algebraic Decoding (AD) is not generalizable for all QR codes. In this study, an efficient hard decoding algorithm is proposed to generalize the decoding of the binary systematic Quadratic Residue (QR) codes. The proposed decoder corrects t erroneous bits or less, in the received word, based on a reduced set of permutations derived from the large automorphism group of QR codes. This set of permutations is applied to the received word to move the error positions and trap all of them in redundancy. Then, to evaluate the proposed method, we applied it to many binary QR codes of moderate code length starting with 17 until 113 with reducible and irreducible generator polynomials. The proposed decoder was validated by inserting all possible error patterns, that have t or less erroneous positions, as input of the proposed decoder and the output is always a correct codeword. The complexity study, in terms of the number of operations used, reveals that the light permutation decoding LPD algorithm significantly decreases decoding complexity without performance loss. So, it is qualified to be a good competitor to decode QR codes with lower lengths but is the best for QR codes with higher lengths.

Keywords: Automorphism Group, Permutation Decoding, Quadratic Residue Codes, Syndrome Decoding

Introduction

The digitization of several sectors of activity is increasing every day and the need for more secure communication systems is also increasing. These later are manifested generally in the storage and transmitting data, with the greatest reliability, from one user to another that are mainly distant. This transmission is carried out through a noisy physical communication channel (Shannon, 1948). The basic model of digital communication is presented in Fig. 1 and is known as the Shannon paradigm. This fundamental model improves the reliability and efficiency of communication systems under varying conditions. It provides the ability to understand and analyze the behavior of communication systems in the presence of various types of impairments, such as noise, interference, and fading. It helps the researchers to determine the best encoding and decoding techniques and the most efficient error correction algorithms to implement.

In more cases the received message certainly includes errors. The processing of data at the reception is limited to restore the integrality of the transmitted information. This concept is based on the detection and correction of errors by using channel coding techniques. The integration of this module at the transmitter and the receiver improves the quality of the transmission and guarantees the reliability of the communication system. It enables the introduction of controlled redundancy bits into the binary message which is then exploited at the receiver to remove the errors introduced by the channel and to find the most likely transmitted message related to the received one. This process is mainly done by using algebraic structures of the Errors Correcting Codes (ECC). In communication systems, linear block codes are widely used and cyclic codes are the most important and attractive class of block codes.

In 1958, Prange introduces a new powerful subclass of codes that possess a half code rate and a good minimum distance (Prange, 1957). This subclass of codes is called the Quadratic Residue QR codes which belong to the family of cyclic codes and has rich mathematic structure. They have the best error performances among binary codes and offer the information a high resistance against channel perturbations. They can be an excellent candidate to be adopted for a very noisy channel (MacWilliams and

Sloane, 1977) and the next generations of mobile communication systems. For this reason, the researchers consider these codes that are the best codes known in the theory of ECC due to their algebraic structure. So, QR codes with lengths less than 100 may have greater application potential for short packet transmission with low latency (Dong *et al*., 2022). Why not extend the areas of the use of the longer QR codes? practically, the (24,12,8) extended QR code has been used in imaging systems for space exploration (Wicker, 1994) and high frequency radio systems (Honary *et al*., 1994).

The decoding of QR codes seems very hard. Because the algorithms mainly used to decode other subclasses of cyclic codes, namely BCH and Reed Solomon codes, could not decode all the QR codes. This decoding hardness is manifested in the way that the algebraic structure of QR codes presents some restrictions to adopting a universal decoder. It means that the decoding system followed, to determine the erroneous positions for a code length, must, necessarily, be modified or at least adapted for each QR code.

On the other hand, real works began in the eighties (Elia, 1987) and continue today. They have produced several techniques for decoding QR codes which can be classified into three broad categories: Algebraic decoding techniques, quasi algebraic decoding techniques, and non-algebraic methods that consider the QR codes as linear block codes. The former has some points of similarities in the decoding procedure which are based on a set of syndromes basically known to find the error locator polynomial $L(z)$. This polynomial be able to establish if we can solve a nonlinear equation system (i.e., the Newton identities related with the syndrome components of the code) by using Sylvester's resultant (Reed *et al*., 1990; 1992) or Gröbner basis (Chen *et al*., 1994). Then, to find the erroneous positions a substitution process of $L(z)$ is carried out by using the roots of the generator polynomial of the code. This is usually done by using the Chien search (Chien *et al*., 1969). For the QR codes, some of the syndrome components are missing due to the algebraic properties and the required eliminations using successive substitutions will be difficult to obtain. So, when the number of errors occurs in the transmitted codeword is quite high, the set of unknown syndromes increases as well as the number of nonlinear equations. Hence, the decoding procedure definitely is not trivial as we do not have enough syndrome components.

However, the authors (He *et al*., 2001) arrived, 2001, to determine the value of the unknown syndromes. They proposed a new technic that serves to make a correspondence between the unknown syndromes and the known ones which define the best $(v + 1)$ $(v + 1)$ matrix $S(I, J)$ with strict conditions. Thereafter, this method makes the decoding procedure somehow less complex and has still opened the door for the researchers to decode the next lengths of QR codes. The benefit of using this method consists of the possibility to apply the efficient Inverse Free Berlekamp Massey (IFBM) algorithm to decode the QR codes (Chang *et al*., 2003; Wang *et al*., 2013). The IFBM method is an iterative approach that requires $2t$ consecutive known syndromes as the input for determining $L(z)$. It has been considered a useful decoding procedure for different binary QR codes. On top of that, there is another commonly used technique to decode a cyclic code but is seldom used to decode the binary QR code. This technique is known as the Euclidean algorithm (Shih *et al*., 2008; 2009). It is based on the division of the syndrome polynomial $S(x)$ and x^n-1. It allows us to determine $L(z)$ by computing the gcd $(S(x); x^n\text{-}1)$, but also all the n consecutive syndromes must be established.

Through the above, it is relatively apparent and straightforward to decode the binary QR code with the required consecutive syndromes and apply the abovementioned algebraic decoding algorithms. Despite that, these decoding algorithms are very complex and need highly complicated computations by utilizing an enormous number of operations over a Galois field. They present a difficult hardware implementation.

Since the Lookup Table Decoding (LTD) is used as a quasi-algebraic decoding technique. In 2009 the authors (Chen *et al*., 2009) propose a new decoding scheme for the binary systematic QR (47,24,11) code based on a lookup table decoding approach. Then, a Reduced Size Lookup Table (RSLT) was proposed (Lin *et al*., 2010). They made some improvements to the decoding parameter by reducing the number of syndromes stored and consequently the size of the decoding table.

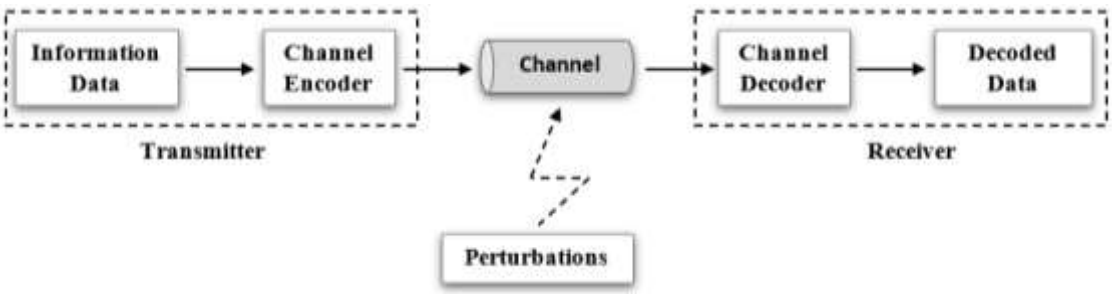

Fig. 1: Simplified model of a digital communication system

After that, new decoding algorithms (Lin *et al*., 2012; Lee *et al*., 2013; Li *et al*., 2018; Huang *et al*., 2018; Gholami and Roostaie, 2021) based on these two previous works have been proposed to improve the decoding. These methods retain the employment of the mathematical properties of this family of codes to determine the erroneous positions. They have been proposed due to the limitation of the AD algorithms. The LTD and their improved versions are slightly less complex and practically provide another way to decode the QR code. It achieves the same goal by looking up a pre-calculated table. These algorithms need to store the lookup tables for all error patterns which requires a real storage capacity in the DSP system and a tangible CPU time when the code length is quite high.

In 1962, Prange introduced the permutation decoding PD algorithm based on the principle of error trapping decoding scheme (Prange, 1962). This category of algorithms is dedicated to decoding systematic cyclic codes. After that, MacWilliams developed (Macwilliams, 1964) a serial decoder for cyclic code. She is interested in finding a large group of automorphisms of the systematic cyclic codes that operate in a particular manner to correct the highest number of errors that do not exceed the error correction capability of code. The PD applies a set of code preserving permutations to the received codeword in order to move, towards the redundancy, of errors appearing in the received word. The algorithm based on this approach is less expensive and simpler to implement than other decoding methods. It is best suited to codes that are invariant under a large group of automorphisms (MacWilliams and Sloane, 1977) and is most useful for alphabets with high error correcting capabilities.

Prange and MacWilliams introduced an (*S, V*) permutation group, in which (*S*) is a group of cyclic shifts and (*V*) is a sequence of squares. After that, two general research areas have been getting started:

- First, to study the application domain of this technique and to verify analytically the capability of decoding several codes. Based on the algebraic properties of the code, they investigate (Benyamin-Seeyar *et al*., 1986; Jia *et al*., 1992) an exact lower bound on the code length. They show that the size of the permutation group is proportional to the code dimension or to the error correction capability of the code. Then, they extend the study of the capability of the large (*S, V*) permutation decodable cyclic codes (Jia *et al*., 1994; Jia and Le-Ngoc, 1995). They use a maximum number of squaring permutations, called steps, required to move all errors. They have established equations that represent the exact relations between the parameters of the code *n*, *k*, *t* and the number of the steps
- Second, to optimize and minimize the number of permutations used in the decoding. This process was done algebraically (Wolfmann, 1983) or by genetic algorithms (Nouh *et al*., 2013)

In 2010 the authors (Key *et al*., 2010) decode Reed Muller codes by using permutation decoding. They applied the decoding algorithm to the first and second order Reed Muller codes. They show that they have a large automorphism group containing the translation group, making them good candidates for permutation decoding. In 2017 the authors (Pace and Sonnino, 2017) construct linear codes that present large automorphism groups and they are suitable for permutation decoding. Propose a new permutation decoding method for RM codes (Kamenev *et al*., 2019). After that, the permutation decoding is applied to polar codes. The authors (Pillet *et al*., 2021) deal with polar code automorphisms that kept the code invariant under permutation. They propose a low latency Automorphism Ensemble (AE) decoding and they prove that polar codes under AE decoding are more efficient than classical polar codes decoders. In 2022 the authors (Bioglio *et al*., 2023) propose some improvements by introducing the notion of redundant automorphism permutations. They list all the permutations that can give a different codeword candidate under successive cancellation-based AE decoding to further reduce the automorphism set size.

For QR codes, the situation is somewhat easier. The decoding of this family of codes is even more efficient because they present a large group of automorphism. In this study, a universal hard decoder based on a subset of permutations, derived from the automorphism group of QR codes, is the subject of this study's interest. The light permutation decoding LPD algorithm corrects *t* erroneous bits or less in the received word. It applies a reduced set of permutations to move the error positions and trap all of them in redundancy. That means that there is at least one permutation that can perform this. We have applied two kinds of decoding algorithms based on two permutation sets. The first is the (*S, V*) permutation set and the second is the (*S, V, T*) permutation set which the permutation (*T*) is the modular multiplicative inverse of the position of the symbols.

As we mentioned before, the algebraic decoding algorithms are very complex and need highly complicated computations by utilizing an enormous number of operations over a Galois field. So, the proposed decoder performs without the necessity for both the unknown syndrome computation and the error locator polynomial on the one hand, and on the other hand, it avoids constructing and stocking a sizeable pre-calculated table that needs real storage capacity like the LTD algorithms and DS algorithm. However, the LPD uses two essential operations: The application of the permutation and the calculation of syndrome weights.

Then, to evaluate the proposed method, we applied it to many binary QR codes of moderate code length starting

with 17 until 113 with reducible and irreducible generator polynomials. We have tried $\binom{n}{i}$ error patterns in which $0 < i \leq t$ is the input of the proposed decoder and the saucerful rate is 100%. In addition, we compare the computational complexity, in the worst case, of the LPD algorithm with the best existing decoding techniques, namely, the cyclic weight decoding algorithms, the difference of syndromes decoding algorithm, and the modified reduced lookup table decoding algorithm. The complexity study reveals that the proposed decoder significantly decreases decoding complexity without performance loss. So, it is qualified to be a good competitor to decode QR codes with lower lengths but is the best for QR codes with higher lengths.

Preliminary and Background of the QR Codes

Cyclic Codes

Definition 1 Let $GF(2)$ denote a Galois finite field of order 2. A binary linear code over $GF(2)$ of length n is called cyclic code if every cyclic shift, of coordinate $i \rightarrow (i + 1) \bmod n$, of a codeword in C is also a codeword in C. It means that if the components of an n-tuple $(c_0, c_1,..., c_{n-1}) \in C$ another n-tuple $(c_{n-1}, c_0, c_1,..., c_{n-2}) \in C$.

When studying binary cyclic codes over $GF(2)$, we will most often represent the codewords in polynomial form. There is a bijective correspondence between the n-tuple vectors $c = (c_0, c_1,..., c_{n-1})$ in $GF(2)$ and the polynomial $c(x) = c_0 + c_1x+c_2x^2+...+ c_{n-1}x^{n-1}$ in $GF(2)[x]$.

Definition 2 Let a nonzero binary cyclic code C be ideal in a polynomial ring $R_n = GF(2)[x]/(x^n -1)$ over a Galois finite field. Then, there exists a unique monic polynomial $g(x)$ of minimal degree such divides x^n-1 in $GF(2)[x]$. Then, $g(x)$ is called the generator polynomial of cyclic code C.

Therefore, the binary cyclic code $C(n, k, d)$ over $GF(2)$ where the positive integers n and k denote, respectively, the code length and the code dimension. The positive integer d represents the minimum distance. Code $_C$ consists of 2^k multiples of $g(x)$ of degree n-k. In other words, the 2^k information words are extended by $(n$-$k)$ redundant parity bits that are algebraically attached. Let $b = (b_0, b_1,..., b_{k-1})$ be k-dimensional binary information bits corresponding to the information polynomial $b(x) = b_0 + b_1x + b_2x^2 +...+b_{k-1}x^{k-1}$. Then, the code $c(x)$ over $GF(2)[x]$ corresponding to $b(x)$ is obtained by the encoding operation:

$$c(x) = b(x)g(x) \tag{1}$$

Such that, $c(x) = \sum_{i=0}^{n-1} c_i x^i \in C$, where, $ci \in \{0,1\}$ and $b(x) = \sum_{i=0}^{k-1} b_i x^i$ the information polynomial, where $b_i \in \{0,1\}$ and $g(x) = \sum_{i=0}^{n-k-1} g_i x^i$, where $g_i \in \{0,1\}$.

The Cyclic Codes in Systematic Form

In general, to facilitate the decoding of cyclic codes, we use a systematic encoding procedure. So, if $g(x)$ divide $b(x)x^{n-k}$, then we get the following equality:

$$b(x)x^{n-k} = q(x)g(x) + d(x) \tag{2}$$

$$b(x)x^{n-k} + d(x) = q(x)g(x) \tag{3}$$

Then, by (1) the code polynomial $c(x)$ is equal to:

$$c(x) = b(x)x^{n-k} + d(x) \tag{4}$$

where the associated vector $d = (d_0, d_1,..., d_{k-1})$ of $d(x)$ is the remainder obtained from the division operation. Then, by multiplying both sides of (1) by $x^k \bmod (x^n -1)$, i.e.:

$$d(x)x^k + b(x) = \left(q(x)x^k\right)g(x) \tag{5}$$

By definition 1, the term $(d(x)x^k + b(x))$, which is a multiple of $g(x)$, can be given by:

$$c(x) = d(x)x^k + b(x) \tag{6}$$

where the $b(x)$ is in the lower k-bit of $c(x)$:

$$\begin{aligned} c &= \left(c_0, c_1, ..., c_{n-1}\right) = \left(c_b, c_d\right) \\ &= \left(b_0, b_1, ..., b_{k-1}, d_0, d_1, ..., d_{n-k-1}\right) \end{aligned} \tag{7}$$

The information symbols are embedded in the obtained codeword. Then, the polynomial $c(x)$ is a systematic code.

The Quadratic Residue Codes

The $C(n, k, d)$ binary quadratic residue codes over $GF(2)[x]$, with a code rate $R \geq \frac{1}{2}$, are a subclass of the cyclic codes of odd prime length $n = 8u \pm 1$ where u is some integer, $k = \frac{n+1}{2}$ is the set of information symbols and d represents the minimal distance. Let's define the error correcting capability of C to be denoted by $t = \lfloor (d-1)/2 \rfloor$, where $\lfloor x \rfloor$ indicate the first integer less than or equal to x.

Then, to construct a C, we must specify the smallest positive integer m which is required to comply with the following condition: $n|2^m -1$. For a given length, we will define the set Q_n to be as being a series of non-zero squares modulo n:

$$Q_n = \left\{ i \mid i \equiv j^2 \left(mod\ \ n\right) for\ 1 \leq j \leq n-1 \right\} \tag{8}$$

The set Q_n contains $\frac{n-1}{2}$ elements and is called the defining set of cover a finite field $GF(2^m)$ of order 2^m.

Let $p(x)$ be the primitive polynomial of degree m which accepts α as the root. This α is the generator of the field $GF(2^m)$, knowing that each element in the field can be expressed as a function of α. Finally, to construct the generator polynomial $g(x)$ of the QR codes, we will not need all the elements of the field $GF(2^m)$, but the construction requires a set T_n extracted from $GF(2^m)$.

Indeed, we can define the set T_n for a given QR code as $T_n = \{\beta^i \in GF(2^m) | i \in Q_n\}$ which is called the variety of g and is the set of all roots of $g(x)$ over $GF(2^m)$ i.e., $g(\beta) = 0$, then $(x-\beta)|g(x)$. Where, $\beta = \alpha^{(2m-1)/n}$ is the primitive n^{th} root of unity in $GF(2^m)$ and the generator polynomial given as:

$$g(x) = \prod_{i \in Q_n} (x - \beta^i) \tag{9}$$

For a code length up to 113, Table 1 lists all the necessary parameters that we need to construct the generator polynomial $g(x)$ for a given QR (n, k, d). Namely, the Galois finite field $GF(2^m)$, the primitive polynomial $p(x)$, and the value of the first element in T_n.

The Light Permutation Decoding Algorithm for the QR Codes

Let's assume that, over $GF(2)[x]$, a codeword $c(x) = \sum_{i=0}^{n-1} c_i x^i$ is transmitted via a noisy channel. This transfer of information is carried out with error and has at least a direct effect on the polarity of one symbol. Then, let's define, respectively, the received word and the error by $r(x) = \sum_{i=0}^{n-1} r_i x^i$ and $e(x) = \sum_{i=0}^{n-1} e_i x^i$ where $r(x)$, $e(x) \in GF(2)[x]$ and r_i, $e_i \in \{0,1\}$. Thus, $r(x)$ is mathematically represented by the following identity:

$$r(x) = c(x) + e(x) \tag{10}$$

which the addition is done in $GF(2)$. This later has a nonzero term in erroneous positions, bearing in mind that it represents the deformation undergone by the channel. So, if we suppose that this deformation produces v errors in $r(x)$. Then, $e(x)$ will include, therefore, v non-zero terms as:

$$e(x) = x^{p_1} + x^{p_2} + ... + x^{p_v} \tag{11}$$

where, $p_1, p_2, ...p_v \in \mathbb{Z}n$ indicates the erroneous positions for $0 \leq p_1 < p_2 < ... < p_v < n$.

Obviously, we can algebraically decode the received word $r(x)$ once we determine $e(x)$, which amounts to finding the error positions $\{p_1, p_2, ..., p_v\}$ for $v \leq t$. Let's divide $r(x)$ by $g(x)$, we obtain:

$$r(x) = q(x)g(x) + s(x) \tag{12}$$

The remainder $s(x)$ is zero as long as there are no errors in $r(x)$. It means that $r(x)$ is a factor of $g(x)$ and the received word is exactly a codeword. Then, if $deg(s(x)) \neq 0$, $s(x)$ is a polynomial of degree $\leq n-k-1$ and is the syndrome of the received word $r(x)$.

By (1), $c(x)$ is a multiple of $g(x)$. Then, combining (10) and (12), we have the following relationship between the error pattern and the syndrome:

$$e(x) = (b(x) + q(x))g(x) + s(x) \tag{13}$$

then:

$$e(x) = q(x)g(x) + s(x) \tag{14}$$

The syndrome $s(x)$ is also the remainder after we divide $e(x)$ by $g(x)$.

The principal mission of a decoder is manifested by two great acts, in particular, to detect and correct the errors produced by the channel during the transfer of information. This must be done without forgetting the complexity, reliability, and efficiency which are the ultimate objectives of such a decoding scheme. The ancient decoders proposed to decode the QR code have unfortunately been unsuccessful to find a universal decoding scheme. It means that the technique used to correct the erroneous positions for a code, must necessarily, be modified or at least adapted for each QR code.

Table 1: The needed parameters to construct the generator polynomial $g(x)$ of QR codes

n	k	d	Galois finite field $GF(2^m)$	Primitive polynomial $p(x)$	Primitive n^{th} root β
17	9	5	$GF(2^8)$	$x^8+x^5+x^3+x^2+1$	α^{15}
23	12	7	$GF(2^{11})$	$x^{11}+x^2+1$	α^{89}
31	16	7	$GF(2^5)$	x^5+x^3+1	α^{1}
41	21	9	$GF(2^{20})$	$x^{20}+x^3+1$	α^{25575}
47	24	11	$GF(2^{23})$	$x^{23}+x^5+1$	α^{178481}
71	36	11	$GF(2^{35})$	$x^{35}+x^2+1$	$\alpha^{483939977}$
73	37	13	$GF(2^9)$	x^9+x^4+1	α^{7}
79	40	15	$GF(2^{39})$	$x^{39}+x^4+1$	$\alpha^{6958934353}$
97	49	15	$GF(2^{48})$	$x^{48}+x^8+x^6+x^5+x^4+x^3+x^2+x+1$	$\alpha^{2901803883615}$
113	57	15	$GF(2^{28})$	$x^{28}+x^3+1$	$\alpha^{2375535}$

In this study, we propose an efficient decoding technique for the QR codes by using a subset of permutations derived from the automorphism group of QR codes. We show and prove that this method can rise to the challenge of being an efficient model to decode QR codes. The proposal decoding algorithm retains the employment of the mathematical properties of the QR code to determine the erroneous positions. It performs without the necessity for both the unknown syndrome computation and the error locator polynomial $L(z)$. It avoids constructing and stocking a sizeable pre-calculated table that needs real storage capacity like the LTD decoding method.

The Automorphism Group of the QR Codes

Usually, we are interested in the positive characteristics of QR codes. It makes them at the heart of the concerns of several authors. The cyclic character of this family of codes gives a flexible behavior against errors. It provides the ability to pass from one vector to another that is the same. From a mathematics point of view, two binary linear codes are the same if they are isomorphic as vector spaces with the same length and over the same field. This concept led us to introduce the term "code equivalence".

Definition 3 two codewords $c_1 \subseteq GF(2)$ and $c_2 \subseteq GF(2)$ are said to be equivalent if c_1 can be obtained from c_2 by permuting the coordinate places of c_1 and multiplying each coordinate position by a non-zero field element.

With generator matrix, G in systematic form, an automorphism group of code C is an isomorphism from C to C. Then, the permutation of the position of the symbols in a code $c \subseteq GF(2)$ forms the automorphism group of C. Let $Aut(C) = \{\pi \in S_n \mid \pi(C) = C\}$ be an automorphism group. It is the set of permutations that map $_C$ to itself without changing their weight distribution and translocating errors from one location to another. A typical permutation π of the position of the symbols is a bijective function between i into $\pi(i)$ means that the vector $c = (c_1, c_2,..., c_n)$ goes into $\pi(c) = (\pi(c_1), \pi(c_2)..., \pi(c_n))$.

Definition 4 a group P of permutations of code C is transitive if, for a given n distinct symbols $c_1, c_2,...,c_n$ and another n distinct symbols $c'_1, c'_2,..., c'_n$, there is a $\pi \in G$ such that $\pi(c_1) = c'_1, \pi(c_2) = c'_2 ..., \pi(c_n) = c'_n$.

If ρ is another permutation of the position of the symbols, the product of the two permutations π and ρ means that we apply π first then ρ. Thus, the following identity is correct $\rho(\pi(c)) = \pi(\rho(c))$.

For binary cyclic codes, Prange proposed a method for producing a series of distinct data sets that may be applied to any cyclic code (Prange, 1962). This method is based on the observation that all cyclic codes are invariant under the symbol position permutations. By definition, a group of permutations (S, V) includes all the cyclic permutations (cyclic shift) and all their powers denoted respectively by (S) and (V). Hence, MacWilliams announced, by the following theorem, the automorphism group of the QR codes (MacWilliams and Sloane, 1977).

Theorem 1 the automorphism group of QR codes is generated by three permutations:

$$S : i \mapsto i+1 \tag{15}$$

$$V : i \mapsto \rho^2 i \tag{16}$$

$$T : i \mapsto -\frac{1}{i} \tag{17}$$

where, i represents a symbol position of the code and ρ is an integer prime to n.

In fact, the automorphism group of QR codes consists of the $\frac{1}{2}n(n^2-1)$ permutations. For detailed proof, see (MacWilliams and Sloane, 1977).

If we want to apply, on the codeword, the permutation (S) for the finite number $\omega < n$ and the permutation (V) for the finite number $\mu < m-1$, in which m is the smallest positive integer who is required to comply with the following condition: $n|2^m-1$. Then (15) and (16) become:

$$S^{\omega} : i \mapsto (i+\omega) mod\ n \tag{18}$$

$$V^{\mu} : i \mapsto \left(\left(\rho^2\right)^{\mu} i\right) mod\ n \tag{19}$$

Example

Suppose that:

$$f(x) = 1 + x + x^2 + x^4 + x^6 \tag{20}$$

is a cyclic codeword of length 7 over $GF(2)$. Let $\omega = 4$, $\mu = 2$, and only for more simplicity do we take $\rho^2 = 2$. Then, by applying S^4 to $f(x)$ we obtain:

$$f_S(x) = x + x^3 + x^4 + x^5 + x^6 \tag{21}$$

x^6 and by applying V^3 to $f(x)$ we obtain:

$$f_V(x) = 1 + x + x^2 + x^4 + x^5 \tag{22}$$

and by applying T to $f_S(x)$ we obtain:

$$f_T(x) = x + x^2 + x^3 + x^5 + x^6 \tag{23}$$

Hence the polynomial $f_S(x)$, $f_V(x)$, and $f_T(x)$ are also a codeword and every power of the permutation V leaves the zero-position unchanged.

The Decoding of the Binary QR Codes by Using Reduced Permutation Sets

Now, the large enough permutation automorphism group of code is established. We can officially discuss the decoding techniques that will exploit the aforementioned properties. In his research, MacWilliams principally introduces, in (MacWilliams and Sloane, 1977), a category of decoding algorithms for short cyclic codes, with a very small number of errors, known as "permutation decoding" and it is completely explained in (MacWilliams and Sloane, 1977). They assumed that the PD algorithm is best suited to codes that are invariant under a large group of permutations and they conjectured that the PD algorithm fit for cyclic codes. For this study, the concept of PD-sets was extended to correct a larger number of errors by using reduced permutation sets.

For a determined $b(x)$ of a given code, this method employs a particular set of permutations called light PD-set derived from the automorphism group of QR codes. Its makes use of both the group (S) of cyclic shift and group (V) of a sequence of squares. Since these two subsets of permutations transform $c \in C$ into an equivalent $c' \in C$ with the same length and the same properties. It means that the syndrome of the obtained codeword is zero and $_{c}^{'}$ always is a factor of $g(x)$. Then, the error position $\{p_1, p_2,..., p_v\}$, for $v \leq t$, associated with this erroneous version of the transmitted code word will, however, be moved into $\{p'_1, p'_2,..., p'_v\}$. So, the idea behind this method is to apply the particular elements of the PD-set to the received word until all the errors are included in redundancy.

Definition 5 if C is a t-error correcting code, then a PD-set for C is a subset of the automorphism group of C which is such that every error pattern of t coordinate positions is moved by at least one permutation into the redundancy.

Finding adequate PD-sets for a code is not trivial. If such a set can be determined, then the LPD algorithm can be used to move the errors of which the code is capable into the check positions. In the case of systematic QR codes, any selection of successive positions represents a data set. Let C (n, k, $2t$+1) be a QR code and $c \in C$, with:

$$c(x)=\sum_{i=0}^{n-1} c_i x^i \tag{24}$$

where, $ci \in \{0,1\}$. Clearly, the permutation group given by (15) and (16) can be applied separately to $c(x)$ as follows:

$$c_S(x)=S^{\omega}\left[c(x)\right]=\left(\sum_{i=0}^{n-1} c_i x^{i+\omega}\right) mod\left(x^{n-1}\right) \tag{25}$$

and:

$$c_V(x)=V^{\mu}\left[c(x)\right]=\left(\sum_{i=0}^{n-1} c_i x^{i\times}\left(\rho^2\right)^{\mu}\right) mod\left(x^n-1\right) \tag{26}$$

Since every binary systematic QR code is preserved by the (S) PD-set and the (V) PD-set the new polynomials $c_S(x)$ and $c_V(x)$ are simply included in C with the coordinate position permuted (definition 1). Let's now, apply a couple of permutations at the same time to $c(x)$. We obtain the following equations, respectively, for the (S, V) PD-set and (S, V, T) PD-set:

$$c_{S,V}(x)=V^{\mu}S^{\omega}\left[c(x)\right]=\left(\sum_{i=0}^{n-1} c_i x^{(i+\omega)\times\left(\rho^2\right)^{\mu}}\right) mod\left(x^n-1\right) \tag{27}$$

and:

$$c_{S,V,T}(x)=V^{\mu}TS^{\omega}\left[c(x)\right]=\left(\sum_{i=0}^{n-1} c_i x^{\omega_i'\times\left(\rho^2\right)^{\mu}}\right) mod\left(x^n-1\right) \tag{28}$$

where, $\omega_i'=-\frac{1}{i+\omega}$ and is the modular multiplicative inverse of n. Then, if n is odd and ρ is the primitive root of n, the code is invariant under the (V) permutation group and $V^{\mu}S^{\omega}[c(x)]$ $V^{\mu}TS^{\omega}[c(x)]$ is in C.

Let's assume that at most $v \leq t$ errors occur in the transmitted codeword. So, the error vector will include, therefore, v non-zero terms, i.e., $e(x) = xp_1 + xp_2 + ... + xp_v$. If $|p_i - p_j| \geqslant k$ such that i and j are two distinct positions. Then, the (S) PD-set can alone move the error into the check positions. If the maximum number of consecutive zero terms in $e(x)$, however, is less than k, then the use only of the shift permutation set will be very restrained. It will be valid only for a single error correction. The same problem arises when we apply only the (V) PD set because every power of the permutation (V) leaves the zero-position unchanged.

On the decoder side and based on (12) and (14), we have shown that the syndrome determined by the process of dividing $r(x)$ by $g(x)$ is also the same after dividing $e(x)$ by $g(x)$. Then, it can be shown that the syndrome of the permuted received words $V^{\mu}S^{\omega}[r(x)]$ is relatively the same as that obtained from the divisions of ($V^{\mu}S^{\omega}[e(x)]$ modulo (x^n-1)):

$$S_{\mu\omega}(x)=\left(\left(V^{\mu}S^{\omega}\left[e(x)\right] mod\left(x^n-1\right)\right)\right) mod\ g(x) \tag{29}$$

is the syndrome of the permuted $e(x)$.

Description of Algorithm

Let's $C(n, k, 2t + 1)$ be a binary systematic QR code, and H of size $(n\text{-}k)\times n$ is the systematic parity check matrix given by:

$$H=\left[A^T \mid I_{n-k}\right] \tag{30}$$

where, A^T denotes the transpose of A and in k is the identity matrix of size n-k. Let's assume that at most t errors occur in the transmitted codeword. Then, the syndrome can be determined by the following equation:

$$s = rH^T \tag{31}$$

Based on the abovementioned theoretical definitions, equations, and theorem 1, the procedure of the proposed decoding algorithm performs as follows. Once we have found, by Theorem 1, a PD set for the given QR code. Suppose that the Hamming weight of the error vector satisfies that $wt(e) \leq t$ and has the syndrome $wt(s) > t$. We consider the first algorithm which presents the decoding protocol of the systematic QR codes by operating the product of two permutations. So, the idea behind this is to apply first the (S) PD-set, for $\omega < n$, to the received word. Then we apply the (V) PD-set for $\mu < m-1$.

The (S, V) PD algorithm of the QR codes is chiefly based on two major nested loops. The initial loop, in line 4, requires n circular permutations by shifting the received word. The second loop, in line 6, is a power permutation of the shifted received word and is require m-2 permutations. The following theorem illustrates a break condition of the decoding process.

Theorem 2 supposes an error vector $e = e_0e_1\cdots e_{n-1}$ of weight t occurs, where $2t+1 \leqslant d$. Let y be the received vector, with syndrome $s = Hy^T$. If the syndrome weight $wt(s) \leqslant t$, then the information symbols $y_r \cdots y_{n-1}$ are correct and $s = (e_0\cdots e_{r-1})^T$ gives the errors. If $wt(s) > t$, then at least one information symbol is incorrect. For detailed proof, MacWilliams and Sloane (1977).

Based on theorem 2 the decoder can decide correctly in the first k coordinate positions corresponding to the information symbols and correct the errors in n-k coordinate positions corresponding to the parity check symbols. So, we compute the syndromes $s_{\mu\omega} = r^{(\mu\omega)}H^T$, until ω and μ are found such that $wt(s_{\mu\omega}) \leq t$. The non-zero symbols in the syndrome $s_{\mu\omega}$ correspond to the erroneous position in the parity check symbols of the permuted received word $r_p^{(\mu\omega)}$. Then we can construct a decided codeword $D^{(\mu\omega)} \in C$. Finally, we decode by inverting the permutation $S^{-\omega}(T^{-1}(V^{-\mu}(D^{(\mu\omega)})))$. The flowchart of the ($S$, V) PD algorithm is illustrated in Fig. 2.

Algorithm 1: The (S, V) PD of the QR codes

1 Input: Received Codeword r, n, m, t, H^T
2 Output: Corrected Codeword D
3 ω ← 0
4 **while** $\omega < n$ **do**
5 $\mu \leftarrow 0$
6 **while** $\mu < m - 1$ **do**
7 $r^{(\mu\omega)} \leftarrow V^{\mu}(S^{\omega}(r))$
8 $s_{\mu\omega} \leftarrow r^{(\mu\omega)} \times H^T$
9 **if** $wt(s_{\mu\omega}) \leq t$ **then**
10 $D_p^{(\mu\omega)} \leftarrow r_p^{(\mu\omega)} \wedge s_{\mu\omega}$
11 $D_m^{(\mu\omega)} \leftarrow r_m^{(\mu\omega)}$
12 D ← $S^{-\omega}$ ($V^{-\mu}(D^{(\mu\omega)})$)
13 Break
14 **end**
15 $\mu \leftarrow \mu + 1$
16 **end**
17 ω ← ω + 1
18 **end**

Algorithm 2: The (S, V, T) PD of the QR codes

1 Input: Received codeword r, n, m, t, H^T
2 Output: Corrected Codeword D
3 ω ← 0
4 **while** $\omega < n$ **do**
5 $r'^{(\omega)} \leftarrow T((S^{\omega}(r)))$
6 $\mu \leftarrow 0$
7 **while** $\mu < m - 1$ **do**
8 $r'^{(\mu\omega)} \leftarrow V^{\mu}(r'^{(\omega)})$
9 $S'_{\mu\omega} \leftarrow r'^{(\mu\omega)} \times H^T$
10 **if** $wt\left(S'_{\mu\omega}\right) \leq t$ **then**
11 $D_p^{(\mu\omega)} \leftarrow r_p'^{(\mu\omega)} \wedge s'_{\mu\omega}$
12 $D_m^{(\mu\omega)} \leftarrow r_m'^{(\mu\omega)}$
13 $D \leftarrow S^{-\omega}(T^{-1}(V^{-\mu}(D^{(\mu\omega)})))$ and Break
14 **end**
15 $\mu \leftarrow \mu + 1$
16 **end**
17 ω ← ω + 1
18 **end**

In order to aid efficient and effective understanding of the decoding process. Fig. 3 represents an example of the decoding process of the QR (23, 12, 3) code using the (S, V) PD-set. The nodes colored in green represent the correct symbols and the others colored in red, blue, and purple are considered erroneous. Each row represents the result of a symbol permutation of the previous row and their syndrome weight is displayed to the right. Then, for $\omega = 1$ and $\mu = 3$, the decoder was able to move the three errors toward redundancy.

Let's consider the second algorithm which presents the decoding protocol of the QR codes by using the product of three permutations (S, V, T). In addition to the permutations (S, V) applied in the first algorithm, it uses the permutation (T) which is based on the modular multiplicative inverse of the position of the symbol. So, if the (S, V) PD failed to move out the errors of the information set to redundancy, we apply, in line 5 of algorithm 2, this permutation only once for each shifted received word and the rest looks like the protocol shown in the decoding scheme of the first algorithm.

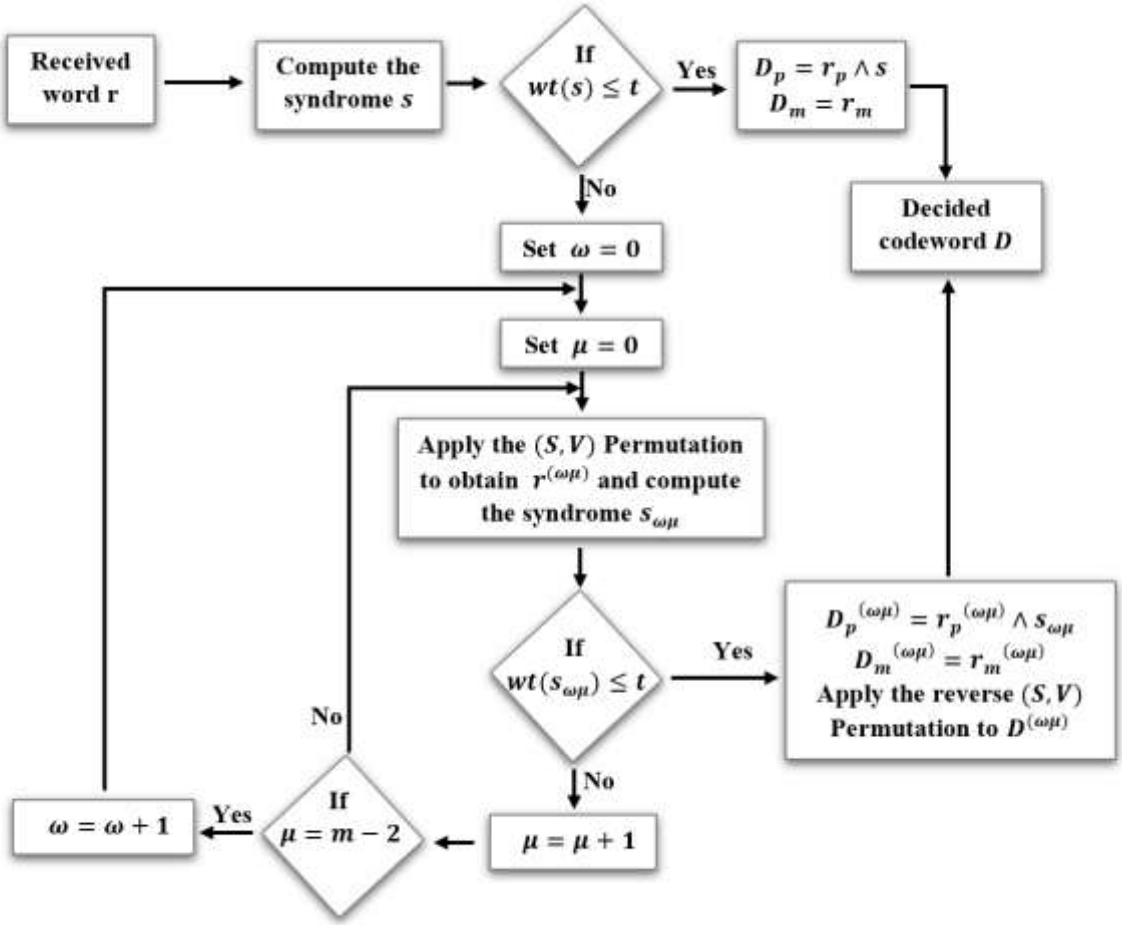

Fig. 2: Flowchart of the (*S*, *V*) PD algorithm for decoding a QR code

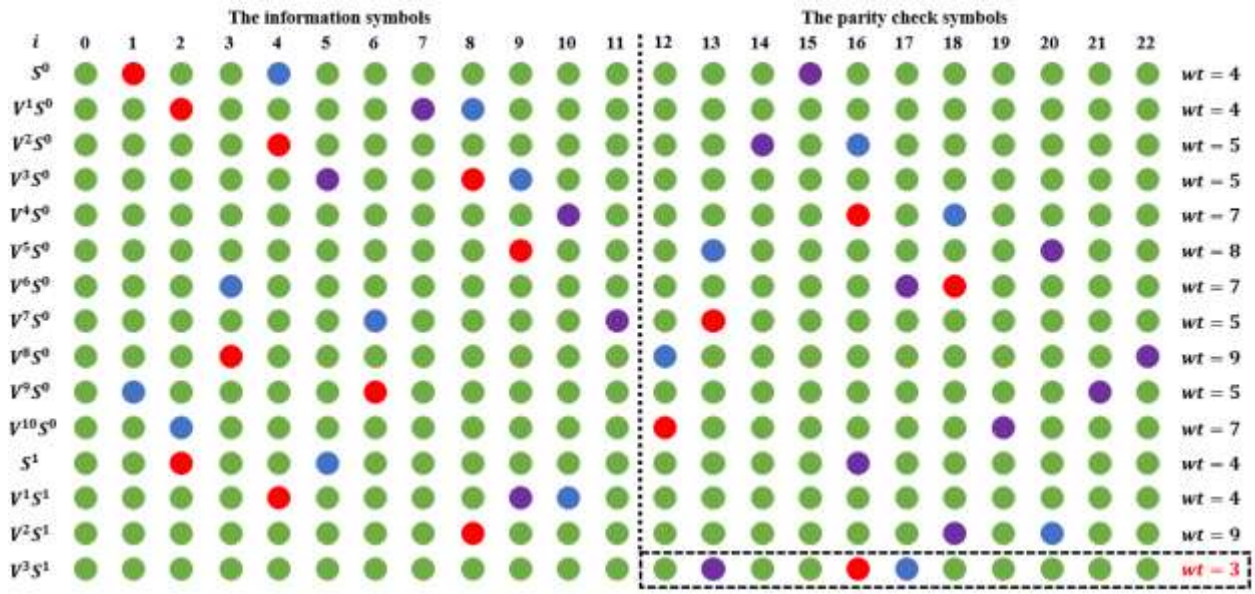

Fig. 3: Example of the decoding procedure of the QR (23, 12, 3) by using the PD algorithm, such that $m = 11$, the correction capacity is $t = 3$ errors, and $e(x) = x + x^4 + x^{15}$

Results and Discussion

This section is dedicated to the performance evaluation of the proposed decoding algorithm. We have made a habit of evaluating decoders by measuring the Bit Error Rate (BER) for each Signal Noise Ration's (SNR) value by carrying out the Monte Carlo method, using a given modulation and passing the codeword over a fixed channel. But in this case, we were mainly interested in the output of the decoder according to a conditional input. In other words, we need to evaluate the efficiency of our decoder through the success and failure rate of a set of erroneous inputs such that $wt(e) < t$. Then, Table 2 illustrates the simulation parameters. It presents the value of the integer ρ^2, then n_{opt} represent the number of cyclic shifts (S) and m is the permutation order (V). Normally, the decoder must use n cyclic shifts, but by simulation, we have ended up with a number, called n_{opt} less than the one theoretically proposed. In the last column of the table, we present the necessary number of error combinations to be tested for each code as a function of its capacity t.

So, we want to underline that this algorithm is written in *C* language. It is used to test, separately, all codes included in Table 2 and to see the behavior of the decoder for each code. Therefore, it is necessary to check all the error combinations according to each code length to validate the decoding algorithm. Let's take, as an example, the QR (23,12,7) code which we need to check $\sum_{i=1}^{3}\binom{23}{i} = 2047$ error pattern containing at most three errors to validate the decoding.

When the decoder tests all possible error combinations. The results of the evaluation of the decoding algorithms (S, V) and (S, V, T) are presented respectively in Tables 3-4. Let's take Table 3, it can be divided into two parts. The first contains the QR codes which are completely decoded without any problem. We refer to the codes of length n that are equal to 17, 23, 31, 41, 47, and 71. In the second part, the failure of the (S, V) PD-set is manifested only when $wt(e) = t$. We notice that the number of uncorrectable combinations remains very small in comparison with the number of tested combinations. We speak of very small proportions.

To face this limitation, we have widened the permutation set from the (S, V) permutation set to the (S, V, T) permutation set only for the codes with length n equal to 73, 79, 97, and 113 when $wt(e) = t$. We obtained good results since the proposed decoder manages to correct all shown in Table 4.

Complexity Analysis

We would like to point out that the success rate of our decoder is 100% which guarantees decoding efficiency. This preliminary conclusion brings us to achieve the main objective of this study which manifests itself in the proposal of a method that generalizes the decoding of binary QR codes. In fact, this result has repeatedly been made plain that our proposed method is practically powerful and capable to decode a set of QR codes in the same way.

Before considering this conclusion, it is necessary to mention that proving the generalization of decoding on a set of QR codes is not sufficient. We must prove the robustness of this method in terms of complexity. It is not easy to directly compare the permutation decoding method to previously known AD algorithms in terms of decoding complexity since there is no universal algebraic for this category that can always decode every QR code.

The basic concept of permutation decoding consists of the determination of a set of code preserving permutations. For such code, the original PD algorithm proposed by MacWilliams exploits all the automorphism group permutations. According to theorem 1, this set of permutations contains $\frac{1}{2}n\left(n^2-1\right)$ permutations. Indeed, the LPD is based on a reduced set of permutations to decode the QR. Then, Fig. 4 presents the cardinal number of the permutation set comparison between the light PD and the original PD proposed. The LPD significantly decreases the number of permutations without performance loss.

On the other hand, we compare the computational complexity of the LPD algorithm with the best existing decoding techniques. Initially, this comparison study will be based on the decoding of the systematic QR code (47,24,11). Then, it can therefore be directly generalized to other QR codes. We provide the computational complexity analysis when correcting 1 to t errors, for the Cyclic Weight (CW) decoding algorithms in (Lin *et al*., 2012), the Difference of Syndromes (DS) decoding algorithm in (Li *et al*., 2018), and the Modified Reduced Lookup Table Decoding (MRLTD) algorithm (Gholami and Roostaie, 2021). These three algorithms are based on storing error patterns and their corresponding syndromes in a table such that each one uses a different size. They are based on the same decoding principle of which they examine the realization of all possible cases based on the distribution of t non-zero positions on the error vector. These errors can appear on t distinct positions among the n positions of the received word. Then, they can distinguish three possible cases: First, the information part contains all the erroneous positions. Second, all the t errors appear in the parity check section. Third, each section contains a number of errors strictly less than t, such that the sum of the errors of each section equals t.

Table 2: The simulation parameters

n	k	t	n_{opt}	m	ρ^2	$\sum_{i=1}^{t}\binom{n}{i}$
17	9	2	3	8	2	153
23	12	3	8	11	2	2047
31	16	3	27	5	9	4991
41	21	4	32	20	2	112791
47	24	5	34	23	2	1729647
71	36	5	28	35	2	14051255
73	37	6	73	9	25	186404113
79	40	7	79	39	2	3200838655
97	49	7	97	48	2	13902476209
113	57	7	113	28	9	41293903801

Table 3: Decoding results by using the (*S*, *V*) permutation decoding set

n	*k*	*t*	*wt*(*e*) = 1 (%)	*wt*(*e*) = 2 (%)	*wt*(*e*) = 3 (%)	*wt*(*e*) = 4 (%)	*wt*(*e*) = 5 (%)	*wt*(*e*) = 6 (%)	*wt*(*e*) = 7 (%)
17	9	2	100	100	-	-	-	-	-
23	12	3	100	100	100	-	-	-	-
31	16	3	100	100	100	-	-	-	-
41	21	4	100	100	100	100	-	-	-
47	24	5	100	100	100	100	100	-	-
71	36	5	100	100	100	100	100	-	-
73	37	6	100	100	100	100	100	99	-
79	40	7	100	100	100	100	100	100	98,8
97	49	7	100	100	100	100	100	100	94,61
113	57	7	100	100	100	100	100	100	96,5

Table 4: Decoding results by using the (*S*, *V*, *T*) permutation decoding set

n	*k*	*t*	*wt*(*e*) = 5 (%)	*wt*(*e*) = 6 (%)	*wt*(*e*) = 7 (%)
73	37	6	100	100	-
79	40	7	100	100	100
97	49	7	100	100	100
113	57	7	100	100	100

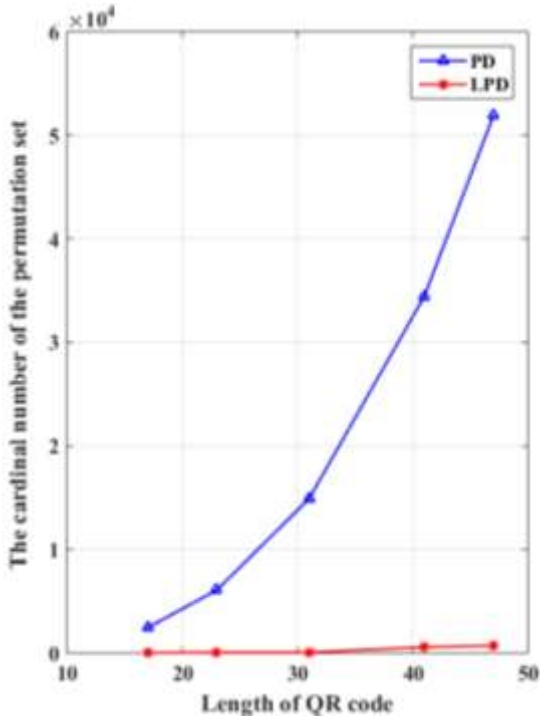

Fig. 4: The comparison between the light PD and the original PD proposed by MacWilliams in terms of the cardinal number of the permutation set

In order to investigate an appropriate comparison between these algorithms. We consider the worst case, for $t = 5$, with the highest computational complexity. Let $GF(+)$, $GF(<<)$, and $R(+)$ denote, respectively, the addition, the shift operation in a finite field, and the real addition. Table 5 displays in detail the amount of calculation performed by the CW, DS, MRLTD, and the LPD algorithm. In the worst case, the CW decoding process is applied twice in order to decode the received message.

Consequently, it performs 13344 $GF(+)$, 106950 $R(+)$, 46 $GF(<<)$, 9296 times search and its memory requirements amount to 20,43 Kbytes. Then, the DS algorithm stores 24 syndromes in the look up table corresponding to the single error patterns. The MRLTD algorithm needs 300 syndromes with their corresponding error patterns. Hence, the memory requirement for storing is quite 2, 6 Kbytes. We noticed that DS and MRLTD decoders follow a different decoding scheme but they require the same amount of calculation. They require 1800 $GF(+)$, 20769 $R(+)$ and 900 times search.

Similarly, let's look at Algorithm 1 is chiefly based on two major nested loops. The initial loop, in line 4, requires not circular permutations and is in charge of the cyclic shift of the received word. The second loop is a power permutation of the shifted received word and is require m - 2 permutations. So, for a given ω and μ the LPD algorithm permutes the received word by (27). Then, computes a syndrome s and its weight $wt(s)$. It means that require 24 $GF(+)$, 23 $R(+)$ and 1 $GF(<<)$ for second iteration. It requires $n_{opt} \times (m\text{-}2)$ iterations in the worst case. Then, the amount of calculation of the LPD algorithm is 17136 $GF(+)$, 16422 $R(+)$, and 714 $GF(<<)$.

Now we want to study the general case. It means that we will extend the complexity study for DS and LPD to decode systematic QR codes of greater length.

The choice to compare only with DS is not arbitrary, but because it is better than CW and MRLTD as Table 5 shows. Let's analyze Fig. 5, we can extract two important remarks:

- From length $n = 41$ the number of operations $R(+)$ used by DS is higher than the one proposed (graph (a)). In fact, the same thing happens for longer quadratic code lengths, where the number of $R(+)$ operations used by DS is 10^8 higher than the one LPD algorithm (graph (c))
- For code lengths less than 100, the DS decoder requires a lower number of GF(+) operations than the LPD algorithm with a small difference (graph (a) and (b)). But for code lengths greater than 100, the DS decoder requires a very large number of $GF(+)$ operations compared to the LPD algorithm

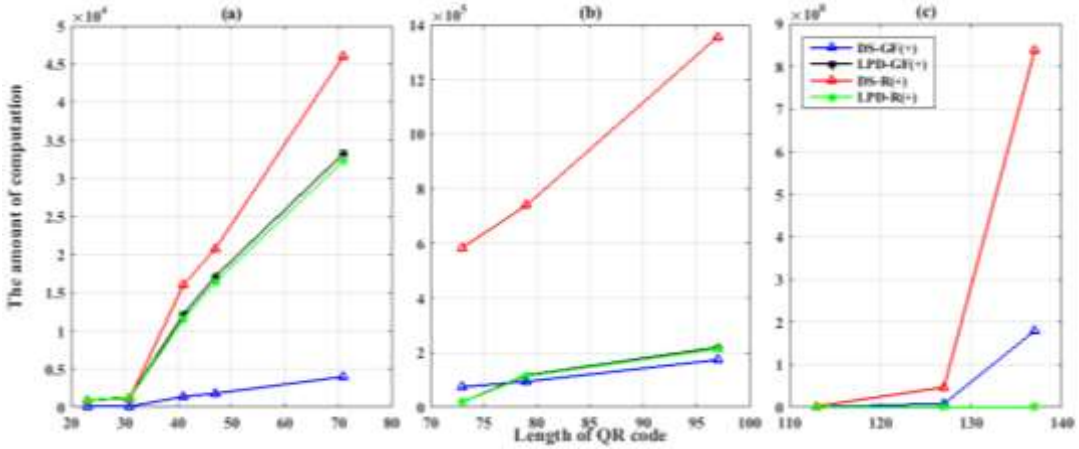

Fig. 5: The computational complexity comparison between the DS algorithm and the LPD algorithm for higher systematic QR code length

Table 5: The computational amount of each step in the CW algorithm, the DS algorithm, the MRLTD algorithm, and the LPD algorithm when we're used to decode QR (47, 24, 11)

Decoder	Iteration number	Algorithm steps description	GF(+)	R(+)	GF(<<)	Search Time	Memory storage (Kbytes)
CW	2	Compute syndrome s and $w(s)$	24	23	-	-	-
		Search whether s is in the CLT	-	-	-	2 324	-
		$S_{dj} = s + S_j$ and					
		Whether $w(S_{dj})$ + w(e_j)	6 648	53 452	23	2 324	-
		Total	13 344	106 950	46	9 296	20.43
DS	3	Compute syndrome s and $w(s)$	24	23	-	-	-
		$S_{di} = s + S_i$ and					
		whether $w(S_{di}) \leq t$ -1	24	552	-	24	-
		$s_{dij} = s + s_i + s_j$ and					
		whether $w(s_{dij}) \leq t$ -2	552	6 348	-	276	-
		Cyclically shift the received word	-	-	24	-	-
		Total	1 800	20 769	48	900	0.07
MRLTD	1	Compute syndrome s and $w(s)$	24	23			
		$s^{(i)} = s + s_i$ and	576	6 900		300	
		whether $w(s^{(i)}) \leq t$ -1	24	23	23	-	-
		Compute syndrome s' and $w(s')$					
		$s'(i) = s' + s_i$ and	576	6 900		300	
		whether $w(s'^{(i)}) \leq t$ -1					
		Flip the first bit and	24	23	-	-	-
		compute syndrome s'					
		$s'(i) = s' + s_i$ and	576	6 900	-	300	-
		whether $w(s'^{(i)}) \leq t$ -1					
		Total	1 800	20 769	23	900	2.6
LPD	714	Compute syndrome s and $w(s)$	24	23	1	-	-
		Total	17 136	16 422	714	-	-

Through these two remarks, we can say that the LPD is a good competitor to decoding QR codes of lower length but is the best for QR codes of higher length. In addition to the above mentioned, there are other factors such as storage capacity and table lookup process which must be considered. Hence, these two factors increase exponentially with the increase in the length of the code. The light permutation decoding algorithm avoids constructing and stocking a sizeable pre-calculated table that needs real storage capacity.

Conclusion

In this study, we propose an efficient hard decoding algorithm for binary QR codes. This decoding algorithm corrects t erroneous bits or less, in the received word, based on a reduced set of permutations derived from the large automorphism group of QR codes. This set of permutations is applied to the received word to move the error positions and trap all of them in redundancy. We showed that the binary QR codes are suitable for light permutation decoding and we have confirmed MacWilliams's conjecture. Then, the proposed method was assessed by applying it to the aforementioned binary QR codes with reducible and irreducible generator polynomials. The proposed decoder output behavior was examined by computing the success rate of the decoder for all the QR codes tested. We have inserted n_i error patterns with $0 < i \leq t$ as input and the decoder success rate is 100%. Then we could

go so far as to affirm that the proposed decoder is a *t*-bounded decoder of the aforementioned QR codes. Furthermore, comparing this new decoding scheme with the previously known decoding algorithms, the LPD algorithm has a similar error correction performance to that of the QR hard decoding algorithms in the theory. So, it can be utilized to decode any binary QR code in the theory. In addition, we compare the computational complexity, in the worst case, of the LPD algorithm with the best existing decoding techniques. The results are attractive in practice because the LPD algorithm significantly decreases the number of operations in the decoding process and avoids constructing and stocking a sizeable pre-calculated table. It is more feasible for hardware implementation.

The main limitation of the present study is that the proposed method is valid only for binary QR codes below 200 and not for non-binary QR codes. So, in the near future, the authors of this study will devote their efforts to decoding QR codes beyond 200 efficiently and to make them more useful in practical systems as a short error correcting code.

Acknowledgment

Thank you to the publisher for their support in the publication of this research article. We are grateful for the resources and platform provided by the publisher, which have enabled us to share our findings with a wider audience. We appreciate the efforts of the editorial team in reviewing and editing our work, and we are thankful for the opportunity to contribute to the field of research through this publication.

Funding Information

The authors have not received any financial support or funding to report.

Author's Contributions

Hamza Boualame: Problem identification, conceptualization, data set identification, written original drafted, methodology, development and algorithmic programming, results evaluation and interpretation. Written reviewed, edited and validation.

Mostafa Belkasmi: Problem identification, conceptualization, methodology development, algorithmic programming, results evaluation and interpretation. Written, reviewed, edited, and validated.

Idriss Chana: Written, reviewed, edited and validated.

Ethics

This article is original and contains unpublished material. There are no ethical issues involved in this manuscript.

References

Benyamin-Seeyar, A., Shiva, S., & Bhargava, V. (1986). Capability of the error-trapping technique in decoding cyclic codes. *IEEE Transactions on Information Theory*, *32*(2), 166-180. https://ieeexplore.ieee.org/abstract/document/1057170

Bioglio, V., Land, I., & Pillet, C. (2023). Group properties of polar codes for automorphism ensemble decoding. *IEEE Transactions on Information Theory*. https://ieeexplore.ieee.org/abstract/document/10025774

Chang, Y., Truong, T. K., Reed, I. S., Cheng, H. Y., & Lee, C. D. (2003). Algebraic decoding of (71, 36, 11), (79, 40, 15) and (97, 49, 15) quadratic residue codes. *IEEE Transactions on Communications*, *51*(9), 1463-1473. https://ieeexplore.ieee.org/abstract/document/1231644

Chen, X., Reed, I. S., Helleseth, T., & Truong, T. K. (1994). Use of Grobner bases to decode binary cyclic codes up to the true minimum distance. *IEEE Transactions on Information Theory*, *40*(5), 1654-1661. https://ieeexplore.ieee.org/abstract/document/333885

Chen, Y. H., Truong, T. K., Huang, C. H., & Chien, C. H. (2009). A lookup table decoding of systematic (47, 24, 11) quadratic residue code. *Information Sciences*, *179*(14), 2470-2477. https://doi.org/10.1016/j.ins.2009.03.011

Chien, R., Cunningham, B., & Oldham, I. (1969). Hybrid methods for finding roots of a polynomial-With application to BCH decoding (Corresp.). *IEEE Transactions on Information Theory*, *15*(2), 329-335. https://ieeexplore.ieee.org/abstract/document/1054283

Dong, J., Li, Y., Liu, R., Guo, T., & Lau, F. C. (2022). Efficient Decoder for Turbo Product Codes Based on Quadratic Residue Codes. *Electronics*, *11*(21), 3598. https://doi.org/10.3390/electronics11213598

Elia, M. (1987). Algebraic decoding of the (23, 12, 7) Golay code (Corresp.). *IEEE Transactions on Information Theory*, *33*(1), 150-151. https://ieeexplore.ieee.org/abstract/document/1057262

Gholami, M., & Roostaie, Z. (2021). On the Decoding of [47, 24, 11] and [48, 24, 12] Quadratic Residue Codes by Some New Fast Algorithms. *Iranian Journal of Science and Technology, Transactions A: Science*, *45*, 683-694. https://doi.org/10.1007/s40995-021-01066-8

He, R., Reed, I. S., Truong, T. K., & Chen, X. (2001). Decoding the (47, 24, 11) quadratic residue code. *IEEE Transactions on Information Theory*, *47*(3), 1181-1186. https://ieeexplore.ieee.org/abstract/document/915677

Honary, B., Hunt, B., & Maundrell, M. (1994). Improving automatic link establishment through a new soft decision trellis decoder for the (24, 12) Golay code. https://digital-library.theiet.org/content/conferences/10.1049/cp_19940489

Huang, J., Zhou, T., Chang, H. C., & Xie, D. (2018). An optimized cyclic weight algorithm of (47, 24, 11) QR code and hardware implementation. *IEEE Access*, *6*, 36995-37002. https://ieeexplore.ieee.org/abstract/document/8392684

Jia, M., & Le-Ngoc, T. (1995, November). Performance of multiple-step (T, U) permutation decoding of cyclic codes. In *Proceedings of GLOBECOM 1995 Mini* (pp. 1-5). IEEE. https://ieeexplore.ieee.org/abstract/document/502921

Jia, M., Benyamin-Seeyar, A., & Le-Ngoc, T. (1992). Exact lower bounds on the codelength of three-step permutation-decodable cyclic codes. *IEEE Transactions on Information Theory*, *38*(6), 1812-1817. https://ieeexplore.ieee.org/abstract/document/165457

Jia, M., Benyamin-Seeyar, A., & Le-Ngoc, T. (1994). On the capability of (T, U) permutation decoding method. *IEEE Transactions on Communications*, *42*(234), 192-195. https://ieeexplore.ieee.org/abstract/document/577006

Kamenev, M., Kameneva, Y., Kurmaev, O., & Maevskiy, A. (2019, July). A new permutation decoding method for Reed-Muller codes. In *2019 IEEE International Symposium on Information Theory (ISIT)* (pp. 26-30). IEEE. https://ieeexplore.ieee.org/abstract/document/8849320

Key, J. D., McDonough, T. P., & Mavron, V. C. (2010). Reed–Muller codes and permutation decoding. *Discrete Mathematics*, *310*(22), 3114-3119. https://doi.org/10.1016/j.disc.2009.06.001

Lee, H. P., Chang, C. H., & Chu, S. I. (2013). High-speed decoding of the binary Golay code. *Journal of Applied Research and Technology*, *11*(3), 331-337.

Li, Y., Duan, Y., Chang, H. C., Liu, H., & Truong, T. K. (2018). Using the difference of syndromes to decode quadratic residue codes. *IEEE Transactions on Information Theory*, *64*(7), 5179-5190. https://ieeexplore.ieee.org/abstract/document/8350085

Lin, T. C., Lee, H. P., Chang, H. C., & Truong, T. K. (2012). A cyclic weight algorithm of decoding the (47, 24, 11) quadratic residue code. *Information Sciences*, *197*, 215-222. https://doi.org/10.1016/j.ins.2012.02.020

Lin, T. C., Lee, H. P., Chang, H. C., Chu, S. I., & Truong, T. K. (2010). High speed decoding of the binary (47, 24, 11) quadratic residue code. *Information Sciences*, *180*(20), 4060-4068. https://doi.org/10.1016/j.ins.2010.06.022

MacWilliams, F. J., & Sloane, N. J. A. (1977). *The theory of error-correcting codes* (Vol. *16*). Elsevier. ISBN-10: 9780444850102.

Macwilliams, J. (1964). Permutation decoding of systematic codes. *The Bell System Technical Journal*, *43*(1), 485-505. https://ieeexplore.ieee.org/abstract/document/6770917

Nouh, S., Chana, I., & Belkasmi, M. (2013). Decoding of block codes by using genetic algorithms and permutations set. *International Journal of Communication Networks and Information Security (IJCNIS)*, *5*(3), 201-209.

Pace, N., & Sonnino, A. (2017). On linear codes admitting large automorphism groups. *Designs, Codes and Cryptography*, *83*, 115-143. https://doi.org/10.1007/s10623-016-0207-6

Pillet, C., Bioglio, V., & Land, I. (2021, October). Polar codes for automorphism ensemble decoding. In *2021 IEEE Information Theory Workshop (ITW)* (pp. 1-6). IEEE. https://ieeexplore.ieee.org/abstract/document/9611504

Prange, E. (1957). *Cyclic error-correcting codes in two symbols*. Air force Cambridge research center.

Prange, E. (1962). The use of information sets in decoding cyclic codes. *IRE Transactions on Information Theory*, *8*(5), 5-9. https://ieeexplore.ieee.org/abstract/document/1057777/

Reed, I. S., Truong, T. K., Chen, X., & Yin, X. (1992). The algebraic decoding of the (41, 21, 9) quadratic residue code. *IEEE Transactions on Information Theory*, *38*(3), 974-986. https://ieeexplore.ieee.org/abstract/document/135639

Reed, I. S., Yin, X., & Truong, T. K. (1990). Algebraic decoding of the (32, 16, 8) quadratic residue code. *IEEE Transactions on Information Theory*, *36*(4), 876-880. https://ieeexplore.ieee.org/abstract/document/53750/

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, *27*(3), 379-423. https://ieeexplore.ieee.org/abstract/document/6773024/

Shih, P. Y., Su, W. K., Lin, T. C., & Truong, T. K. (2008, May). On decoding of quadratic residue codes with irreducible generator polynomials. In *2008 International Conference on Communications, Circuits and Systems* (pp. 35-37). IEEE. https://ieeexplore.ieee.org/abstract/document/4657721/

Shih, P. Y., Su, W. K., Lin, T. C., & Truong, T. K. (2009, February). Modified decoding of binary Quadratic Residue codes by using Euclidean algorithm. In *2009 11th International Conference on Advanced Communication Technology* (Vol. *3*, pp. 1628-1630). IEEE. https://ieeexplore.ieee.org/abstract/document/4809384/

Wang, L., Li, Y., Truong, T. K., & Lin, T. C. (2013). On decoding of the (89, 45, 17) quadratic residue code. *IEEE Transactions on Communications*, *61*(3), 832-841. https://ieeexplore.ieee.org/abstract/document/4809384/

Wicker, S. B. (1994). *Error control systems for digital communication and storage*. Prentice-Hall, Inc.

Wolfmann, J. (1983). A permutation decoding of the (24, 12, 8) Golay code (Corresp.). *IEEE Transactions on Information Theory*, *29*(5), 748-750. https://ieeexplore.ieee.org/abstract/document/1056726/