International Journal of Pure and Applied Mathematics
Volume 100 No. 3 2015, 391-403
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
doi: http://dx.doi.org/10.12732/ijpam.v100i3.6

ALGEBRAIC DECODING IN TWO-DIMENSIONAL LINEAR CODES

Pramote Jangisarakul[1] [§], Chalie Charoenlarpnopparut[2]

[1,2]School of Information
Computer and Communication Technology
Sirindhorn International Institute of Technology
Thammasat University
Klong Luang, Pathum-Thani 12121, THAILAND

Abstract: The tools of algebraic approach are used to decode two-dimensional linear block code defined as a two-step operation, namely a column-wise encoding matrix and a row-wise encoding matrix. A binomial ideal is essential to initiate the decoding algorithm associated with the Gröbner basis of this ideal and a m-variate division algorithm. To demonstrate the efficiency of the decoding process by using this algorithm can be measured in terms of the correctable percentage with interesting parameters: bit error probability, term ordering, and computational time. By testing erroneous bits in received codewords for several different encoding matrices of 2-D linear block codes, the algorithm can be reliable to correct up error-correcting capability of code. There is a struggle of several variables (indeterminates) in the binomial ideal since almost computational time will be lost finding Gröbner basis of binomial ideal.

AMS Subject Classification: 94B05, 94B35
Key Words: Groebner bases, linear block codes, decoding matrix

1. Introduction

For decades, 1-D linear block codes have been applied to protect digital data

Received: December 18, 2014

[§]Correspondence author

from transmission errors due to the imperfection of the communication channel. When there is the need for transmitting 2-D data such as images and field arrays, designers opt to transform the 2-D data array into a 1-D data vector by scanning row-by-row or column-by-column. As a result, the 2-D correlation among neighboring bits is lost and ignored. In many coding schemes with memory, such as 2-D convolutional codes, the correlation can be employed to gain better error approximations. Although, a 2-D linear block code is a memoryless coding scheme, studying its properties and behavior can lead to a better understanding of 2-D convolutional code designs. The application of Gröbner bases to linear codes was first introduced by Cooper (1993) [1]. The author proposed the use of polynomial expressions for cyclic codes to derive a decoder that makes use of the Gröbner basis approach. The development of the decoder is based on the computation of a set of syndrome equations that are m-variate polynomials. If the number of errors affecting the received codeword does not exceed the error-correcting capability, the roots of syndrome can be obtained. In Borges-Quintana et al, [2, 3], authors introduced a binomial ideal $I_{\mathcal{C}}$ constructed from a binary linear encoder and, the reduced Gröbner basis of this ideal with respect to total degree term ordering was derived. The study can be applied to the decoding problem of binary linear codes. Bounded distance decoding of arbitrary linear codes using Gröbner bases was formulated by Bulygin and Pellikaan in [4]. Solutions of the syndrome equations associated with a certain ideal are essential for this approach. In a recent work, Saleemi and Zimmermann (2010)[5, 6], shown that binary linear codes can be associated with binomial ideals. The Gröbner bases of binomial ideals are studied. In this paper, we present a novel decoding for 2-D linear block codes which are based on Gröbner basis, and also provide a good insight for implementing a code search algorithm. Moreover, the merit of our work lies on the alternative decoding method which can be generalizable to a non-memoryless coding scheme such as 2-D convolutional code based on tail-biting technique.

2. Matrix Description of Encoder

A $k_1 \times k_2$ matrix of input information is encoded and, a $n_1 \times n_2$ matrix of codeword is produced by the encoding matrices. Next, every codeword in $\mathcal{C}$ can be denoted as $V = G_1 \cdot U \cdot G_2$, where U is called *information matrix* of size $k_1 \times k_2$ and $V \in \mathcal{C}$ is called *codeword* of size $n_1 \times n_2$. All entries of U and V are called *bits*. In later discussion, each entry in matrix can be called bit as well. For the case of a binary channel, a bit here means 0 and 1. G_1 and G_2

here are defined as *column-wise encoding matrix* and *row-wise encoding matrix*, respectively. Without loss of generality, both G_1 and G_2 are assumed to be systematic; this means:

$$G_1 = \begin{bmatrix} I_{k_1 \times k_1} \\ P_{1(n_1-k_1)\times k_1} \end{bmatrix} \tag{1}$$

and

$$G_2 = \begin{bmatrix} I_{k_2 \times k_2} & P_{2k_2\times(n_2-k_2)} \end{bmatrix}, \tag{2}$$

where $I_{k_1\times k_1}$, $I_{k_2\times k_2}$ are identity matrices; $P_{1(n_1-k_1)\times k_1}$, $P_{2k_2\times(n_2-k_2)}$ are matrices whose entries are all bits. As a result, the corresponding parity-check matrices can be expressed:

$$H_1 = \begin{bmatrix} P_1^T{}_{k_1\times(n_1-k_1)} \\ I_{(n_1-k_1)\times(n_1-k_1)} \end{bmatrix} \tag{3}$$

and

$$H_2 = \begin{bmatrix} P_2^T{}_{(n_2-k_2)\times k_2} & I_{(n_2-k_2)\times(n_2-k_2)} \end{bmatrix}, \tag{4}$$

where the superscript T denotes the transpose of a matrix. Note that plus and minus for binary field are interchangeable. These parity-check matrices satisfy the constraints:

$$H_1^T G_1 = 0, G_2 H_2^T = 0.$$

Moreover, we define a relationship between the parity-check matrix and the codeword as follows:

$$H_1^T V = H_1^T \cdot G_1 \cdot U \cdot G_2 = 0 \tag{5}$$

$$V H_2^T = G_1 \cdot U \cdot G_2 \cdot H_2^T = 0. \tag{6}$$

Definition 1. The code rate R_c is defined as the ratio between the number of information digits and the number of codeword digits. From the above discussion, it is possible to define the code rate as:

$$R_c = (k_1 \cdot k_2)/(n_1 \cdot n_2). \tag{7}$$

The code rate is one of the key parameters for evaluating the code performances. A high code rate implies that there are few redundant check bits among the codeword bits, but one of disadvantages is that it is difficult to remedy transmission errors.

Definition 2. Let $\mathcal{C}$ be a linear block code over $\mathbb{F}_2$. For all V_1,V_2, $V_3 \in \mathcal{C}$ and for all λ_1,$\lambda_2 \in \mathbb{F}_2$:
(i) V_1+$V_2 \in \mathcal{C}$;
(ii) there is codeword $\mathbf{0} \in \mathcal{C}$;
(iii) $(V_1 + V_2) + V_3 = V_1 + (V_2 + V_3)$;
(iv) $V_1 + V_2 = V_2 + V_1$;
(v) $\lambda_1 V_1 \in \mathcal{C}$.

Proposition 1. *Let both G_1 and G_2 be encoding matrices for $\mathcal{C}$ over $\mathbb{F}_2$. A code defined in the form $V = G_1 \cdot U \cdot G_2$ is a linear block code.*

Proof. (i) and *(v)* If $V_1 = G_1 \cdot U_1 \cdot G_2$ and $V_2 = G_1 \cdot U_2 \cdot G_2$, then $\lambda_1 V_1 + \lambda_2 V_2 = \lambda_1(G_1 \cdot U_1 \cdot G_2) + \lambda_2(G_1 \cdot U_2 \cdot G_2) = G_1(\lambda_1 \cdot U_1 + \lambda_2 \cdot U_2)G_2 = G_1 \cdot U_3 \cdot G_2 \in \mathcal{C}$. *(ii)* If $U_1 = 0$ then $V = G_1 \cdot U_1 \cdot G_2 = \mathbf{0} \in \mathcal{C}$. *(iii)* and *(iv)* Obvious. □

2.1. Minimum Distance

Basic concepts for 1-D linear block codes such as a minimum distance and an error correction algorithm are necessary for developing new aspects of 2-D linear block codes. The following definitions are important results for this work. In the following, we provide concise descriptions of these concepts. The reader is referred to references for further details.

Definition 3. The *Hamming* weight of a codeword is defined as the number of nonzero elements that are in the codeword.

Definition 4. The *minimum Hamming distance* d_{min} of a code is the smallest Hamming distance between distinct codewords. The parameter d_{min} can be used to find the error-correcting capability of a code. If a code can correct up to t errors, where t is the upper bound, then

$$t = \lfloor (d_{min} - 1)/2 \rfloor, \tag{8}$$

where

$$d_{min} = min|V_i - V_j|, \{\forall (V_i, V_j) \in \mathcal{C}, V_i \neq V_j\}. \tag{9}$$

3. Formulation of the Problem

3.1. Binomial Ideal

A binomial ideal as a binary linear code plays a crucial role in the decoding process of 2-D linear block codes, since an application of the binomial ideal of

a linear code can be extended to the decoding process of 1-D error-correcting codes. Therefore, a short summary on the binomial ideal and its applications is given in the following.

A monomial in $\mathbb{F}[x_{11}, \ldots, x_{1n_2}, \ldots, x_{n_1 1}, \ldots, x_{n_1 n_2}]$ defined by X^{α} is a product of the form:

$$X^{\alpha} = x_{11}^{\alpha_{11}} \cdots x_{1n_2}^{\alpha_{1n_2}} \cdots x_{n_1 1}^{\alpha_{n_1 1}} \cdots x_{n_1 n_2}^{\alpha_{n_1 n_2}}, \tag{10}$$

where all components $\alpha_{11}, \ldots, \alpha_{n_1 n_2}$ are binary field.

The situation for a decoding process associated with the binomial ideal of 1-D linear block codes gives in [3, 5, 6]. Then, we apply fundamental points of view for decoding 2-D linear block codes, and refer to the binomial ideal $I_{\mathcal{C}}$:

$$I_{\mathcal{C}} = \langle X^{\overline{V}} - 1, x_{ij}^2 - 1 | \ \forall (i,j) : 1 \leq i \leq n_1, 1 \leq j \leq n_2 \rangle. \tag{11}$$

The matrix $\overline{V}$ does not belong to $\mathcal{C}$. It can be represented as: $\overline{V} = V_m + K$, where $V_m \in \mathcal{C}$ is the codeword that has the maximum weight, and K is a matrix whose all entries are nonzero bits. The matrix K has the same dimensions as the matrix V_m. Then, the number of terms in $(X^{\overline{V}} - 1)$ equals to the number of all possible V_m.

Definition 5. The bit error probability (P_e) of a binary linear code is the probability that an information bit of codeword is erroneously transmitted to the destination.

Because assume the transmission channel has bit error probability given in Definition 5, the transmitted codewords may occur some erroneous bits and can be expressed:

$$\tilde{V} = V + E, \tag{12}$$

where $\tilde{V}$ is called *received codeword* and E is called *error matrix* in transmission channel. In simulation part, we assume erroneous bits e_{ij}, for $1 \leq i \leq n_1$ and $1 \leq j \leq n_2$ in E, are at random as well as the number of them equals t of a code. $\tilde{V}$, V and E in Eq.(12) can be represented by $n_1 \times n_2$ matrices as:

$$V = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n_2} \\ v_{21} & v_{22} & \cdots & v_{2n_2} \\ \vdots & \vdots & \ddots & \vdots \\ v_{n_1 1} & v_{n_1 2} & \cdots & v_{n_1 n_2} \end{bmatrix}, \tag{13}$$

$$E = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1n_2} \\ e_{21} & e_{22} & \cdots & e_{2n_2} \\ \vdots & \vdots & \ddots & \vdots \\ e_{n_1 1} & e_{n_1 2} & \cdots & e_{n_1 n_2} \end{bmatrix}, \tag{14}$$

$$\tilde{V} = \begin{bmatrix} \tilde{v}_{11} & \tilde{v}_{12} & \cdots & \tilde{v}_{1n_2} \\ \tilde{v}_{21} & \tilde{v}_{22} & \cdots & \tilde{v}_{2n_2} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{v}_{n_1 1} & \tilde{v}_{n_1 2} & \cdots & \tilde{v}_{n_1 n_2} \end{bmatrix}. \tag{15}$$

V and $\tilde{V}$ can be mapped into monomial in Eq.(10)

$$X^V = x_{11}^{v_{11}} \cdots x_{1n_2}^{v_{1n_2}} \cdots x_{n_1 1}^{v_{n_1 1}} \cdots x_{n_1 n_2}^{v_{n_1 n_2}} \tag{16}$$

$$X^{\tilde{V}} = x_{11}^{\tilde{v}_{11}} \cdots x_{1n_2}^{\tilde{v}_{1n_2}} \cdots x_{n_1 1}^{\tilde{v}_{n_1 1}} \cdots x_{n_1 n_2}^{\tilde{v}_{n_1 n_2}}. \tag{17}$$

The purpose of the decoding process is to find unknown elements in an *estimated error matrix* defined as $\hat{E}$. Assume that $\hat{E}$ has all unknown elements $\hat{e}_{ij}$, for $1 \leq i \leq n_1$ and $1 \leq j \leq n_2$:

$$\hat{E} = \begin{bmatrix} \hat{e}_{11} & \hat{e}_{12} & \cdots & \hat{e}_{1n_2} \\ \hat{e}_{21} & \hat{e}_{22} & \cdots & \hat{e}_{2n_2} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{e}_{n_1 1} & \hat{e}_{n_1 2} & \cdots & \hat{e}_{n_1 n_2} \end{bmatrix}. \tag{18}$$

3.2. Algorithm for Decoding 2-D Block Codes

Let $\mathcal{G}$ be the Gröbner basis of the binomial ideal $I_{\mathcal{C}}$. Let the received codeword $\tilde{V} \in \mathbb{F}_2^n$ for $n = n_1 n_2$ and be based on the monomial term defined $X^{\tilde{V}}$.

Proposition 2. *If $X^{\tilde{V}}$ reduces to r modulo $\mathcal{G}$ and we denote as $X^{\tilde{V}} \xrightarrow{\mathcal{G}}_+ r$, then the remainder of the m-variate division algorithm identifies E in received codeword $\tilde{V}$.*

Proof. If $\mathcal{G} = \{g_1, g_2, \ldots, g_k\}$ is a Gröbner basis of ideal $I_{\mathcal{C}}$, then for any $V \in \mathcal{C}$, the corresponding X^V can be expressed as a unique linear combination of the basis elements; i.e., there exist unique $\lambda_1, \lambda_2, \ldots, \lambda_k \in \mathbb{F}_2$ such that $X^V = \lambda_1 g_1 + \lambda_2 g_2 +, \ldots, +\lambda_k g_k$. This equation is derived by the m-variate division algorithm $X^V \xrightarrow{\mathcal{G}}_+ 0$. Next, when $\tilde{V} = V + E$ and E is not in $\mathcal{C}$, then we have $X^{\tilde{V}} = \lambda_1 g_1 + \lambda_2 g_2 +, \ldots, +\lambda_k g_k + r$ or $X^{\tilde{V}} \xrightarrow{\mathcal{G}}_+ r$. This implies that the remainder is E. □

If the remainder (r) is zero, the codeword V is equal to the received codeword $\tilde{V}$, and hence there are no errors on $\tilde{V}$. Otherwise, $\tilde{V}$ contains errors. Note that a short summary of the m-variate division algorithm can see in [7].

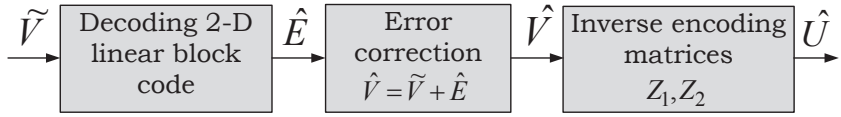

Figure 1: 2-D decoding process with forward error correction

Definition 6. The correctable percentage of a code is defined as the ratio of the number of detected and corrected codewords ($\hat{V} = V$) and the number of received codewords ($\tilde{V}$).

Algorithm 1: Decoding of error-correcting code by means of Gröbner bases.
Step 1: Construct G_1 and G_2 in Eq.(1)-(4).
Step 2: Find the received codeword $\tilde{V}$.
Step 3: Construct $I_{\mathcal{C}}$ with respect to any term ordering.
Step 4: Compute Gröbner basis $\mathcal{G}$ for $I_{\mathcal{C}}$.
Step 5: Use the m-variate division algorithm in order to determine $\hat{E}$ by the reduction process $X^{\tilde{V}} \xrightarrow{\mathcal{G}}_{+} r$.
Step 6: Carry out the remainder of the m-variate division algorithm to represent the errors. For instance, if the remainder is $x_{11}x_{12}x_{23}x_{32}$ and the received codeword $\tilde{V}$ is 3×3 matrix, then the estimated error matrix $\hat{E}$ can be mapped into the matrix:

$$\hat{E} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

As shown in Fig. 1, the $\tilde{V}$ passes into the decoding 2-D linear block code to produce the $\hat{E}$ and then it will be used to correct $\tilde{V}$ with $\hat{V} = \tilde{V} + \hat{E}$, where $\hat{V}$ is defined as *estimated codeword*. If the $\hat{V}$ is correct i.e. $\hat{V} = \tilde{V} = V$, then the error correction is successful, but otherwise $\tilde{V}$ contains errors. In the binary field, addition and subtraction are interchangeable. The *estimated information matrix* denoted by $\hat{U}$ can be expressed:

$$\hat{U} = Z_1 \cdot \hat{V} \cdot Z_2, \tag{19}$$

where Z_1 and Z_2 are a left inverse of G_1 and a right inverse of G_2, respectively, as well as in this work, both matrices have not been demonstrated.

4. Simulation Result

Two examples are selected to demonstrate the efficiency of decoding process by using Algorithm 1. The considerations are focused on some parameters. Singular [8] is chosen to implement several algebraic procedures such as ideals, syzygy modules, m-variate division algorithms, and Gröbner basis. Note that all computational results have been done on Windows operating system with processor speed 2.67 GHz and RAM memory 4.00 GB.

Example 7. This example is to simulate the error-correcting performance of the decoding process by means of Algorithm 1 for five different encoding matrices. To simplify the notation of the different codes, the superscript (i) is introduced which leads to $G_1^{(i)}, G_2^{(i)}$, for $1 \leq i \leq 5$. All encoding matrices are given in Table 1. We do not give details of all entries in each code $G_1^{(i)}, G_2^{(i)}$. Assume that the number of random erroneous bits of E in each received codeword equals the error-correcting capability t of a code. For example, in Table 2, the number of random erroneous bits of E for $G_1^{(1)}, G_2^{(1)}$ has only one. There are 16000 received codewords. The performance of this algorithm can be measured in terms of correctable percentage. The $I_{\mathcal{C}}$ with respect to degree lexicographical ordering. For example, the $I_{\mathcal{C}}$ for $G_1^{(1)}, G_2^{(1)}$ can be expressed as follows:

$$\begin{aligned} I_{\mathcal{C}} &= \langle x_{13}x_{14}x_{22}x_{23}x_{31}x_{33} - 1, x_{13}x_{14}x_{21}x_{23}x_{32}x_{33} - 1, \\ &\quad x_{12}x_{13}x_{23}x_{24}x_{31}x_{33} - 1, x_{12}x_{13}x_{21}x_{23}x_{33}x_{34} - 1, \\ &\quad x_{11}x_{13}x_{23}x_{24}x_{32}x_{33} - 1, x_{11}x_{13}x_{22}x_{23}x_{33}x_{34} - 1, \\ &\quad x_{ij}^2 - 1 | 1 \leq i \leq n_1, 1 \leq j \leq n_2 \rangle, \end{aligned} \tag{20}$$

Table 2 indicates the computation results. The columns represent: the code matrices $G_1^{(i)}$ and $G_2^{(i)}$, the minimum distance, the error-correcting capability, the number of detected and corrected codewords ($\hat{V} = V$), the number of received codewords, and the correctable percentage, respectively. As seen in the table, the correctable percentage of all codes is completely 100 except for $G_1^{(3)}, G_2^{(3)}$ and $G_1^{(5)}, G_2^{(5)}$. Moreover, comparing the same d_{min} with different matrix size of encoding matrices, matrix size of V may affect a performance of corrections.

Table 3 indicates that, for this decoding process, degree lexicographical ordering is the most efficient term ordering, where *lp*, *dp*, and *Dp* are acronyms for lexicographical ordering, degree reverse lexicographical ordering, and degree lexicographical ordering, respectively. The reader is referred to term orderings in [7]. However, for other problems, other term orderings can be more effective.

Table 1: Codes for simulations $G_1^{(i)}, G_2^{(i)}$ in Example 7.

	Codes
$G_1^{(1)}, G_2^{(1)}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$, $\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$
$G_1^{(2)}, G_2^{(2)}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$, $\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}$
$G_1^{(3)}, G_2^{(3)}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$, $\begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$
$G_1^{(4)}, G_2^{(4)}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$, $\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$
$G_1^{(5)}, G_2^{(5)}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$, $\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$

Table 2: Correctable percentage by using Algorithm 1 to decoding of error-correcting codes: $G_1^{(i)}, G_2^{(i)}$

Code	d_{min}	t	$\hat{V} = V$	$\tilde{V}$	correctable percentage
$G_1^{(1)}, G_2^{(1)}$	4	1	16000	16000	100
$G_1^{(2)}, G_2^{(2)}$	6	2	16000	16000	100
$G_1^{(3)}, G_2^{(3)}$	8	3	14992	16000	93.77
$G_1^{(4)}, G_2^{(4)}$	8	3	16000	16000	100
$G_1^{(5)}, G_2^{(5)}$	10	4	15741	16000	98.38

Table 3: Correctable percentage of decoding process by using Algorithm 1 for $G_1^{(i)}, G_2^{(i)}$ with different term orderings.

Code	Term ordering		
	lp	*dp*	*Dp*
$G_1^{(1)}, G_2^{(1)}$	58.32	100	100
$G_1^{(2)}, G_2^{(2)}$	44.62	100	100
$G_1^{(3)}, G_2^{(3)}$	37.61	93.7	93.8
$G_1^{(4)}, G_2^{(4)}$	44.28	100	100
$G_1^{(5)}, G_2^{(5)}$	38.91	98.38	98.53

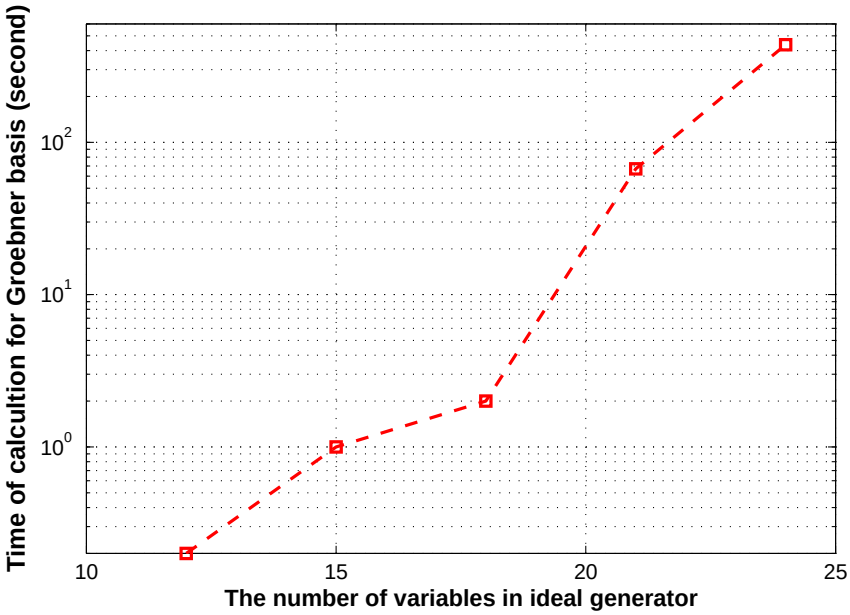

Figure 2: Time of computing Gröbner basis

Fig 2 shows the time required for computing the Gröbner basis of an ideal generator with 12 to 24 variables, and all ideal generators are given in Table **??**. As seen in the graph, the time of computing Gröbner basis is exponential, as it is clearly seen when the number of variables varies from 18 to 24. This is a consequence of the ideal membership problem.

Example 8. This example uses the same encoding matrices $G_1^{(i)}, G_2^{(i)}$ for $1 \leq i \leq 5$, and procedures as in Example 7. The transmission channel has bit error probability P_e. As seen in the Table 4, the decoding process by using $G_1^{(5)}, G_2^{(5)}$ as encoding matrices provides the best correctable percentage. In the Table 5, the time used for decoding process for each code is exponential because of Gröbner basis computation.

Table 4: Correctable percentage of decoding process by using Algorithm 1 for $G_1^{(i)}, G_2^{(i)}$ with bit error probability P_e.

Code	P_e			
	0.05	0.10	0.15	0.20
$G_1^{(1)}, G_2^{(1)}$	95.36	84.06	68.21	53.76
$G_1^{(2)}, G_2^{(2)}$	98.91	92.81	80.58	64.59
$G_1^{(3)}, G_2^{(3)}$	99.23	94.54	83.43	68.46
$G_1^{(4)}, G_2^{(4)}$	99.78	97.60	90.70	78.05
$G_1^{(5)}, G_2^{(5)}$	99.82	98.03	92.12	79.64

Table 5: Time of decoding process in each code.

Code	Average time (s)
$G_1^{(1)}, G_2^{(1)}$	18
$G_1^{(2)}, G_2^{(2)}$	21
$G_1^{(3)}, G_2^{(3)}$	28
$G_1^{(4)}, G_2^{(4)}$	166
$G_1^{(5)}, G_2^{(5)}$	605

5. Conclusion

The main attention for this research lies on the development of novel method for the 2-D decoding process by using an algebraic approach based on Gröbner basis. The basic definitions such as *Hamming* weight, d_{min}, error-correcting capability are still well definitions in 2-D linear block codes. Gröbner basis of this

ideal and the m-variate division algorithm give the final results of estimated error matrix. By testing erroneous bits in received codewords for several different encoding matrices, the algorithm appears to be reliable to correct up error-correcting capability of code. The struggle of several variables(indeterminates) in the binomial ideal is a problem since almost computational time will be lost finding Gröbner basis. Then, fast algorithm for finding this basis of the binomial ideal or the method of reduced variables in this ideal is required. Moreover, it is potential to apply for other codes. Due to several advantages of 2-D codes, comparing with 1-D codes, it is feasible to use 2-D codes for transmission of images and animations in the future [9].

Acknowledgments

The authors thank Prof.Dr. Maria Elena Valcher at the University of Padova for her valuable comments toward this work.

References

[1] A.B. Cooper, Toward a new Method of Decoding Algebraic Codes Using Gröbner Bases, *Trans. 10th Army Conf. Appl. Math. and Comp.*, (1993), 1-11.

[2] M. Borges-Quintana, M.A. Borges-Trenard, and E. Martinez-Moro, On a Gröbner bases structure associated to linear codes, *Journal of Discrete Mathematical Sciences and Cryptography*, **10**, No.2 (2007), 151-191. doi: 10.1080/09720529.2007.10698114.

[3] M. Borges-Quintana, M.A. Borges-Trenard, P. Fitzpatrick and E. Martinez-Moro, Gröbner bases and combinatories for binary codes, *Applicable Algebra in Engineering, Communication and Computing*, **19**, No.5 (2008), 393-411. doi: 10.1007/s00200-008-0080-2.

[4] S. Bulygin, R. Pellikaan, Bounded distance decoding of linear error-correcting codes with Gröbner bases, *Journal of Symbolic Computation*, **44**, (2009), 1626-1643. doi: 10.1016/j.jsc.2007.12.003.

[5] M.Saleemi, K-H. Zimmermann, Linear Codes as Binomial Ideals, *International Journal of Pure and Applied Mathematics*, **61**, No.2 (2010), 147-156.

[6] M.Saleemi, K-H. Zimmermann, Groebner for linear codes, *International Journal of Pure and Applied Mathematics*, **62**, No.4 (2010), 481-491.

[7] W. Adams, P. Loustaunau, *An Introduction to Gröbner Bases*, Graduate Studies in Mathematics, American Mathematical Society, Providence, RI, **3** USA (1994).

[8] G.-M. Greuel, G. Pfister, H. Schonemann, SINGULAR 3.0., *Computer Algebra System for Polynomial Computations*, Centre for Computer Algebra, University of Kaiserlautern (2005), http://www.singular.uni-kl.de.

[9] S. Basu, M.N.S. Swamy, Editorial Preface to Special Issue on Multidimensional Signals and Systems, *IEEE Trans. Circuits Syst. I, Fundamental Theory and Applications*, **49**, No.6 (2002), 709-714. doi: 10.1109/TCSI.2002.1010026.