An Algebraic Method for Decoding q-ary Codes via Submodules of $\mathbb{Z}^n$

Malihe Aliasgari and Mohammad-Reza Sadeghi

***Abstract*—In this paper, by using a relation between binomial ideal and submodules of $\mathbb{Z}^n$ in [3], a submodule associated with the integer programming (IP) problem is defined. By computing the reduced Gröbner basis (RGB) of the submodule, the decoding problem of non-binary q-ary codes is considered as an integer program problem. Decoding complexity is investigated and the effective factors in complexity are also determined. Furthermore, an example of the decoding method for a 3-ary code is provided.**

***Index Terms*—Gröbner basis, group code, $\mathbb{Z}$-module, integer programming.**

I. Introduction

THE integer programming (IP) in non-negative integers is NP-complete. Algorithms with average complexity even theoretically bad worst case can be useful for a special class of NP-complete problems. Recently, various algebraic IP solvers based on the theory of Gröbner basis (GB) have been proposed [5], [10]. The key idea is to translate an IP problem into an ideal associated with the constraints of the problem. The GB is equivalent to the test set of the IP problem. The test set of the IP problem can be directly computed by using algebraic software such as CoCoA or Maple. There are two methods for translating an IP problem into a certain ideal:

- Translating by adding extra variables;

  This method was the first connection which established by Conti and Traverso [5].
- Translating without adding extra variables;

  This method is a geometric interpretation of Conti-Traverso algorithm where introduced by Thomas [10].

The main difficulty of these solvers is the size of the GB generated by both methods which is quite large.
An extension of Conti-Traverso procedure to solve IP with modulo arithmetic conditions is proposed by Ikegami and Kaji [7]. As a consequence, the class of binary codes have been decoded as an IP problem. In order to carry out the Conti-Traverso method, the FGLM-based trick [4] has been used to compute a reduced Gröbner basis (RGB) in [9].

The maximum likelihood decoding (MLD) is one of the fundamental topics in the theory of error correcting codes. Many efforts have been devoted to find an efficient algorithm for the MLD of block codes. The MLD based on an IP problem for binary linear codes was introduced by Feldman et al. [6]. Also, MLD using GB is reduced into a class of IP [7]. The hard-decision MLD on the binary symmetric channel consists of finding an error vector with the smallest Hamming weight, such that the sum of the received word with the error vector is a codeword. A linear programming objective for $q > 2$ has been solved [7]. Since the Hamming metric is inefficient for decoding non-binary codes, the modular IP method in [7] does not allow one to perform decoding non-binary codes.

The study of binomial ideal derived from arbitrary codes or lattices is currently of great interest in coding theory, for example in the framework of decoding methods for binary codes [4] or algorithms for decoding arbitrary lattices [8] (integer and non-integer) using GB [1]. The notion of GB for submodules of $\mathbb{Z}^n$ was introduced in [3], where it has been attempted to avoid using Buchberger algorithm in the computation of GB of pure binomial ideals.

In this work, we propose a new method for solving IP problem via GB. Our method implements Conti-Traverso method on $\mathbb{Z}^n$-module and based on an IP formulation of the MLD, a method for decoding q-ary codes is presented.

This work is organized as follow. In Section II we give a brief sketch to some basic concepts needed in this work. In Section III we introduce the idea of associating a $\mathbb{Z}^{m+n}$-module to IP problem and then by using the division algorithm, non-binary q-ary codes are decoded. Section IV is devoted to discuss about complexity. Finally, we draw a conclusion in Section V.

Notations: Throughout the paper, small boldface letters and capital letters are used to represent vectors and matrices. For a vector $\mathbf{x}$, we use x_j to represent the j-th component of $\mathbf{x}$. For a matrix X, the matrix X^t denote the transpose of X.

II. General Setting

In this section, we briefly introduce some definitions and results from group code, IP problem, G-norm and GB which are necessary for the subsequent text.

A. Group code

Let G_i be a finite alphabet of symbols. A block code $\mathcal{C}$ is any subset of a sequence space $G = G_1 \times \cdots \times G_n$. Elements of $\mathcal{C}$ are called codewords. If $\mathcal{C} = \{\mathbf{c}_1, \ldots, \mathbf{c}_M\}$ has M codewords then $\mathcal{C}$ is an (n, M) code. An Abelian group block code $\mathcal{C}$ is a subgroup of a sequence space G where each G_i is an Abelian group. Since any finite Abelian group can be expressed as a direct product of finite cyclic groups, without loss of generality, we can assume that G_i is cyclic. If $|G_i| = g_i$ for $i = 1, \ldots, n$ then G_i is isomorphic to $\mathbb{Z}_{g_i}$, the integer group module g_i. So we suppose that $G = \mathbb{Z}_{g_1} \times \mathbb{Z}_{g_2} \times \cdots \times \mathbb{Z}_{g_n}$ under addition. If all code symbols are given from the same alphabet $\mathcal{A} = \{\alpha_1, \ldots, \alpha_q\}$, then $\mathcal{C}$ is called a q-ary code over $\mathcal{A}$. Also, when each $G_i = \{0, 1\}$ the code $\mathcal{C}$ is a binary code. More details are given in [2].

Manuscript received December 17, 2013. The associate editor coordinating the review of this letter and approving it for publication was E. Paolini.

The authors are with the Department of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran (e-mail: {ariyadokht, msadeghi}@aut.ac.ir).

Digital Object Identifier 10.1109/LCOMM.2014.030914.132773

Here, we consider $\mathcal{C}$ as a subgroup of q-ary code $G = \mathbb{Z}_q^n$, where $\mathbb{Z}_q = \{0, 1, \ldots, q-1\}$. The minimal subset of $\{\mathbf{c}_1, \ldots, \mathbf{c}_M\}$ which generates $\mathcal{C}$ is said a generator set for $\mathcal{C}$. The dual code $\mathcal{C}^\perp$ of $\mathcal{C}$ is defined as follow

$$\mathcal{C}^\perp = \{\mathbf{c} \in G : \langle \mathbf{a}, \mathbf{c} \rangle = 0 \pmod q, \text{ for all } \mathbf{a} \in \mathcal{C}\},$$

where $\langle \mathbf{a}, \mathbf{c} \rangle$ is denoted the inner product of vectors $\mathbf{a}$ and $\mathbf{c}$. A matrix $H = \left[\mathbf{c}_1^* \ldots \mathbf{c}_m^*\right]^t$ whose rows form a spanning set of $\mathcal{C}^\perp$ is called a parity check matrix of $\mathcal{C}$. It follows that $\mathcal{C}$ is the null space of H.

B. Decoding as an integer programming problem

The problem of finding a non-negative integer vector which minimizes or maximizes a target function with respect to some linear constraint equations with integer coefficients, is called an IP problem. Let $H \in \mathbb{Z}^{m\times n}$, $\mathbf{s}^t \in \mathbb{Z}^m$ and $\mathbf{w} \in \mathbb{R}^n$. The IP problem finds an integer vector $\mathbf{e} \in \mathbb{Z}_{\geq 0}^n$ such that $\langle \mathbf{w}, \mathbf{e} \rangle$ is minimized subject to $H\mathbf{e}^t = \mathbf{s}$ and it is denoted by $\mathrm{IP}_{H,\mathbf{w}}(\mathbf{s})$. The matrix form of the problem is:

$$\mathrm{IP}_{H,\mathbf{w}}(\mathbf{s}) = \begin{cases} \text{minimize } \langle \mathbf{w}, \mathbf{e} \rangle \\ \text{subject to} \begin{cases} H\mathbf{e}^t = \mathbf{s} \\ \mathbf{e} \in \mathbb{Z}_{\geq 0}^n. \end{cases} \end{cases}$$

Also, the modular IP for an integer $q \geq 2$ is denoted by $\mathrm{IP}_{H,\mathbf{w},q}(\mathbf{s})$ which minimizes $\langle \mathbf{w}, \mathbf{e} \rangle$ subject to $H\mathbf{e}^t \equiv \mathbf{s} \pmod q$. In this case $H \in \mathbb{Z}_q^{m\times n}$ and $\mathbf{s} \in \mathbb{Z}_q^m$. A vector $\mathbf{e} \in \mathbb{Z}_q^n$ which satisfies $H\mathbf{e}^t \equiv \mathbf{s} \pmod q$ is called optimal if $\mathbf{e}$ minimizes the $\langle \mathbf{w}, \mathbf{e} \rangle$.

The Conti-Traverso algorithm solves the IP problem by translating it into a problem dealing with polynomials. Then they introduced a GB to solve $\mathrm{IP}_{H,\mathbf{w}}$. Ikegami et al. adapted the idea of the Conti-Traverso algorithm to solve the modular IP. The MLD can be considered as an IP problem.

Let $\mathcal{C}$ be a code with parity check matrix H and let $\mathbf{c} = (c_1, \ldots, c_n) \in \mathcal{C}$ be the transmitted codeword. We consider a baseband model in that it suppresses all of the modulation and demodulation functions. The corruption of a codeword by channel noise is modeled as an additive process. Let $\mathbf{r} \in G$ be the received word output by the demodulator and entering the decoder. Based on syndrome decoding, syndromes of received codewords are computed by matrix H, i.e. $\mathbf{s} = H\mathbf{r}^t$. MLD consists of finding the most likely codeword $\mathbf{c}$ given the observation $\mathbf{r}$. Since $H\mathbf{c}^t = 0$ for a codeword $\mathbf{c}$, a null syndrome indicates that a received word is a codeword. For a non-zero syndrome vector, the presence of error is detected. In hard-decision MLD for a binary code $\mathcal{C}$ is finding an error vector $\mathbf{e} = (e_1, \ldots, e_n) \in \mathbb{Z}_2^n$ with the smallest Hamming weight among all the vectors in $\mathbb{Z}_2^n$ such that the sum of the received word with the error vector is a codeword. Since the Hamming weight $\mathbf{e}$ is equal to $\sum_{i=1}^n e_i$, MLD is equivalent to solving $\mathrm{IP}_{H,\mathbf{1},2}(H\mathbf{r}^t)$ where $\mathbf{1} = (1, \ldots, 1)$.

C. Gröbner bases of $\mathbb{Z}^n$-submodules

If $\boldsymbol{a} = (a_1, \ldots, a_n)$ and $\boldsymbol{b} = (b_1, \ldots, b_n)$ are in $\mathbb{Z}_{\geq 0}^n$, then the degree lexicographic order is defined by $\boldsymbol{a} \prec_{\text{lex}} \boldsymbol{b}$, if either $\sum_{i=0}^n a_i < \sum_{i=0}^n b_i$ or $\sum_{i=0}^n a_i = \sum_{i=0}^n b_i$ and the left-most nonzero coordinate of $\boldsymbol{b}-\boldsymbol{a}$ is positive. Also, the degree reverse lexicographic order is defined by $\boldsymbol{a} \prec_{\text{rlex}} \boldsymbol{b}$ if either $\sum_{i=0}^n a_i < \sum_{i=0}^n b_i$ or $\sum_{i=0}^n a_i = \sum_{i=0}^n b_i$ and the right-most nonzero coordinate of $\boldsymbol{b}-\boldsymbol{a}$ is positive. We say that $\boldsymbol{b}$ divides $\boldsymbol{a}$ and write $\boldsymbol{b}|\boldsymbol{a}$ if $b_i \leq a_i$ for all i.

For $\boldsymbol{a} \in \mathbb{Z}^n$, we use the decomposition $\boldsymbol{a} = \boldsymbol{a}^+ - \boldsymbol{a}^-$ where $(\boldsymbol{a}^+)_i = \max\{a_i, 0\}$ and $\boldsymbol{a}^- = (-\boldsymbol{a})^+ \geq 0$. Define $\mathrm{supp}(\boldsymbol{a}) := \{i : a_i \neq 0\} \subseteq \{1, \ldots, n\}$. It should be noted that $\boldsymbol{a}^+$ and $\boldsymbol{a}^-$ have disjoint supports and each part of $\boldsymbol{a}^+$ and $\boldsymbol{a}^-$ are uniquely determined by $\boldsymbol{a}$. Boffi and Logar have introduced the theory of GB on $\mathbb{Z}^n$-modules by extending the order to $\mathbb{Z}^n$ [3].

We consider $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{Z}^n$ and $\mathcal{F} = \{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_r\} \subset \mathbb{Z}^n$ such that $\boldsymbol{f}_i^+$ is greater than $\boldsymbol{f}_i^-$ with respect to $\prec_{\text{lex}}$ or $\prec_{\text{rlex}}$ order. In [3], $\boldsymbol{a}$ reduces to $\boldsymbol{b}$ module $\mathcal{F}$ in one step if there exists $i = 1, \ldots, r$ such that either $\boldsymbol{f}_i^+$ divides $\boldsymbol{a}^+$ and $\boldsymbol{b} = \boldsymbol{a} - \boldsymbol{f}_i$ or $\boldsymbol{f}_i^+$ divides $\boldsymbol{a}^-$ and $\boldsymbol{b} = \boldsymbol{a} + \boldsymbol{f}_i$. Also, $\boldsymbol{a}$ reduces to $\boldsymbol{b}$ module $\mathcal{F}$ if there exists $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \in \mathbb{Z}^n$ such that $\boldsymbol{a}$ reduces to $\boldsymbol{b}_1$ in one step module $\mathcal{F}$, $\boldsymbol{b}_1$ reduces to $\boldsymbol{b}_2$ in one step module $\mathcal{F}, \ldots,$ $\boldsymbol{b}_{k-1}$ reduces to $\boldsymbol{b}_k$ in one step module $\mathcal{F}$ and $\boldsymbol{b}_k = \boldsymbol{b}$. If none of $\boldsymbol{f}_i^+$ divides $\boldsymbol{b}^+$ and $\boldsymbol{b}^-$ for all $i = 1, \ldots, r$, then $\boldsymbol{b}$ is called a reduction of $\boldsymbol{a}$ module $\mathcal{F}$ and it is denoted by $\mathrm{Red}(\boldsymbol{a}, \mathcal{F})$.

Definition 1. *Let $\mathcal{M}$ be a $\mathbb{Z}^n$-submodule. A set $\mathcal{G} = \{\boldsymbol{g}_1, \ldots, \boldsymbol{g}_s\}$ is a GB of $\mathcal{M}$ with respect to the given order if for every $\boldsymbol{a} \in \mathcal{M}$ there exists $\boldsymbol{g}_i$ such that $\boldsymbol{g}_i^+ | \boldsymbol{a}^+$. Also, $\mathcal{G}$ is called a reduced one if $\boldsymbol{g}_i^+ | \boldsymbol{g}_j^+$ then $i = j$ and none of the $\boldsymbol{g}_i^+$ divides $\boldsymbol{g}_j^-$ for all $i, j = 1, \ldots, s$.*

Every GB of a $\mathbb{Z}^n$-module $\mathcal{M}$ forms a set of generators.

D. G-norm

Decoding q-ary codes consists of finding the codeword with the smallest distance to the received word $\mathbf{r}$. If $q = 2$ then solving the modular IP problem $IP_{H,\mathbf{1},2}(\mathbf{s})$ where H is the parity check matrix of the binary linear code and $\mathbf{s} = \mathbf{r}H^t \pmod 2$, gives the complete decoding of the received word $\mathbf{r}$. Since Hamming distance can not be stated as a linear programming problem for a non-binary q-ary codes we resort on [1] in which an appropriate norm, called the G-norm, has been introduced. The G-norm is equivalent to the assumed order on $\mathbb{Z}^n$-module. If G is a vector space over $\mathbb{Z}_2$ then the G-norm in G is defined as follows.

Definition 2. *Let $\boldsymbol{c} = (c_1, \ldots, c_n)$ be a q-ary codeword in G. The G-norm of $\boldsymbol{c}$ which is denoted by $\|\boldsymbol{c}\|_G$ is defined as*

$$\|\boldsymbol{c}\|_G = c_1 + c_2 + \cdots + c_n,$$

where the operations are performed in $\mathbb{R}$.

It should be noted that the G-norm is equivalent to both degree lexicographic and degree reverse lexicographic orders [1]. From now on we work with a degree reverse lexicographic order as a primary order and use the notation $\prec$ instead of $\prec_{rlex}$.

Lemma 3. *Let $\boldsymbol{a}, \boldsymbol{c} \in G$ and $\boldsymbol{c} \prec \boldsymbol{a}$. If $\boldsymbol{a}$ is the closest codeword to $\boldsymbol{c}$ with respect to $\prec$, then $\boldsymbol{a}$ is the closest codeword to $\boldsymbol{c}$ with respect to the G-norm $\|.\|_G$.*

Proof: The proof is given in [1]. ∎

III. $\mathbb{Z}$-Module Associated to the Integer Programming Problem

In [5] GB is used for solving $\mathrm{IP}_{H,\mathbf{w}}(\mathbf{s})$. The extension of this algorithm to solve modular IP was presented in [7]. Also, a hard-decision MLD for a binary code via IP and GB was presented. We use the Conti et al. algorithm for decoding q-ary codes. Since there is a correspondence between pure saturated binomial ideals of $K[x_1,\dots,x_n]$ and $\mathbb{Z}^n$-modules, we consider the constraints of the IP problem $\mathrm{IP}_{H,\mathbf{w},q}(\mathbf{s})$ as a submodule of $\mathbb{Z}^{m+n}$ instead of a binomial ideal. In this section we show how to decode non-binary q-ary codes with G-norm.

Consider $\mathcal{C}\subseteq\mathbb{Z}_q^n$ as a non-binary q-ary code and H be a parity-check matrix for the code. Let $\mathbf{s}=H\mathbf{r}^t \pmod q$. Since G-norm for every $\mathbf{e}\in G$ is equal to $\|\mathbf{e}\|_G=\sum_{i=1}^n e_i$, we assume that $\mathbf{w}=\mathbf{1}$ so that the matrix form of the decoding group code becomes:

$$\mathrm{IP}_{H,\mathbf{1},q}(\mathbf{s})=\begin{cases}\text{minimize } \|\mathbf{e}\|_G\\ \text{subject to}\begin{cases}H\mathbf{e}^t\equiv\mathbf{s}\pmod q\\ \mathbf{e}\in\mathbb{Z}_q^n.\end{cases}\end{cases}$$

Let $H=\begin{bmatrix}\mathbf{h}_1\cdots\mathbf{h}_n\end{bmatrix}$ be a full rank $m\times n$ basis matrix for $\mathcal{C}^\perp$ with column vectors $\mathbf{h}_i\in\mathbb{Z}_q^m$ for $i=1,\dots,n$. We consider $H'=\begin{bmatrix}H|\mathrm{diag}_m(q)\end{bmatrix}^t$, where $\mathrm{diag}_m(q)=qI_m$ and I_m is an $m\times m$ identity matrix. Now, we define the square $(m+n)\times(m+n)$ matrix

$$H^\star=\begin{bmatrix}H^t & -I_n\\ \mathrm{diag}_m(q) & 0_{m\times n}\end{bmatrix},\tag{1}$$

where $0_{m\times n}\in\mathbb{Z}^{m\times n}$ is the all zero matrix. We define two maps $\iota_{mn}:\mathbb{Z}^m\to\mathbb{Z}^{m+n}$ where $\iota_{mn}(h_1,\dots,h_m)=(h_1,\dots,h_m,0,\dots,0)$ and $\pi_{mn}:\mathbb{Z}^{m+n}\to\mathbb{Z}^n$ which $\pi_{mn}(h_1,\dots,h_{m+n})=(h_{m+1},\dots,h_{m+n})$. The map ι_{mn} extends an m-tuple vector to an $m+n$ vector by adding n zero components at the end of the vector and π_{mn} maps an $(m+n)$-tuple vector to its last n components.

Let $\mathcal{M}$ be a $\mathbb{Z}^{m+n}$-submodule associated to the row vectors of $H^\star$. By using the method in [3] the GB of submodule $\mathcal{M}$ which avoids the Buchberger algorithm is computed. Let $\mathcal{G}$ denote the RGB associated to $\mathcal{M}$. We consider $\mathcal{G}$ as the RGB of the constraint of $\mathrm{IP}_{H,\mathbf{1},q}$ problem. To present decoding method for non-binary q-ary codes via IP, first we state the following Lemma.

Lemma 4. *Let $H\in\mathbb{Z}_q^{m\times n}$ and $\boldsymbol{s}\in\mathbb{Z}_q^m$. Consider $\mathcal{G}$ as the RGB of $\mathbb{Z}^{m+n}$-module associated to the row vectors of $H^\star$. If $\boldsymbol{s}$ is a linear combination of column vectors of H then $\mathrm{Red}(\iota_{mn}(\boldsymbol{s}),\mathcal{G})\in 0^m\times\mathbb{Z}^n$.*

Proof: Let $\iota_{mn}(\mathbf{s})_i$ be the i-th component of the vector $\iota_{mn}(\mathbf{s})$. Since $\mathbf{s}$ is a vector in a column space of the matrix H, we have

$$\iota_{mn}(\mathbf{s})_i=\begin{cases}h_{i1}x_1+\cdots+h_{in}x_n & i=1,\dots,m,\\ 0 & i=m+1,\dots,m+n,\end{cases}$$

where $x_i\in\mathbb{R}$. We replace zero components of $\iota_{mn}(\mathbf{s})_i$ with x_i-x_i, $i=m+1,\dots,m+n$. In other words, $\iota_{mn}(\mathbf{s})$ decompose to sum of the following vectors

$$\begin{gathered}(h_{11}x_1+\cdots+h_{1n}x_n,\dots,h_{m1}x_1+\cdots+h_{mn}x_n,\bar{x}_1,\dots,\bar{x}_n)\\ +(0,\dots,0,x_1,\dots,x_n)\\ =x_1\mathbf{h}_1^\star+\cdots+x_n\mathbf{h}_n^\star+(0,\dots,0,x_1,\dots,x_n)\end{gathered}$$

where $-x_i$ is denoted by $\bar{x}_i$ and $\mathbf{h}_i^\star$ is the i-th row vector of matrix $H^\star$. Since $x_1\mathbf{h}_1^\star+\cdots+x_\mathbf{n}\mathbf{h}_n^\star\in\mathcal{G}$, thus the reduction of $\iota_{mn}(\mathbf{s})$ with respect to $\mathcal{G}$ is a vector in $0^m\times\mathbb{Z}^n$. ∎

Now by using the above Lemma we state our main theorem.

Theorem 5. *Using the previous notation, if $\boldsymbol{s}\in\mathbb{Z}_q^m$ is in a column space of H then $\pi_{mn}(\mathrm{Red}(\iota_{mn}(\boldsymbol{s}),\mathcal{G}))$ is the optimal solution of $IP_{H,\boldsymbol{1},q}(\boldsymbol{s})$.*

Proof: Let $\mathbf{h}_i^\star,\mathbf{h}_i'$ be the i-th row vector of matrices $H^\star,H'$, respectively. For $i=1,\dots,n$, $\mathbf{h}_i^\star$ is obtained from $\mathbf{h}_i'$ where its $(m+i)$-th component in $\mathbf{h}_i^\star$ is -1. Also in computing the reduction of $\iota_{mn}(\mathbf{s})$ it is checked whether or not $\boldsymbol{g}_i^+|\iota_{mn}(\mathbf{s})^+$ for $\boldsymbol{g}_i\in\mathcal{G}$. Equivalently, $\iota_{mn}(\mathbf{s})$ is subtracted from vectors of $\mathcal{M}$ because $\mathcal{G}$ is a generator for $\mathcal{M}$. The n last coordinates of a vector $\iota_{mn}(\mathbf{s})$ are zero. Because of the existence of $-I_n$ matrix in $H^\star$, in each subtraction, the $(m+i)$-th component one unit increases for $i=1,\dots,n$. Furthermore, $\mathbf{s}$ is in a column space of H thus by Lemma 4, $\mathrm{Red}(\iota_{mn}(\mathbf{s}),\mathcal{G})\in 0^m\times\mathbb{Z}^n$. Let $\mathbf{e}=\pi_{mn}(\mathrm{Red}(\iota_{mn}(\mathbf{s}),\mathcal{G}))$ where satisfies in equation $H\mathbf{e}^t\equiv\mathbf{s}\pmod q$. Meanwhile, the reduction is performed by considering degree lexicographic or degree reverse lexicographic order. By Lemma 3, these orders are compatible with G-norm. The uniqueness of $\mathrm{Red}(\iota_{mn}(\mathbf{s}),\mathcal{G})$ results having the least G-norm among all n-tuple vectors. So $\mathbf{e}$ has the least G-norm and it is the optimal solution. ∎

Let $\mathcal{C}$ be a q-ary code with parity-check matrix H. Also assume that $\mathcal{G}$ is a RGB associated with the constraint of $\mathrm{IP}_{H,\mathbf{1},q}(H\mathbf{r}^t)$ where $\mathbf{r}$ is a received word. Since $H\mathbf{r}^t$ is a vector in column space of H, by Theorem 5 hard-decision decoding of $\mathbf{r}$ with respect to the G-norm is the codeword $\mathbf{c}=\mathbf{r}-\mathbf{e}$ where $\mathbf{e}=\pi_{mn}(\mathrm{Red}(\iota_{mn}(H\mathbf{r}^t),\mathcal{G}))$.

We want to find a codeword $\mathbf{c}^*$ which satisfies in $\min_{\mathbf{c}^*\in C}\{\|r-c^*\|_G,\|c^*-r\|_G\}$. We observe that $\mathbf{c}$ is the closest codeword to $\mathbf{r}$ with respect to the G-norm. Once again solve $\mathrm{IP}_{H,\mathbf{1},q}(\mathbf{s}')$, where $\mathbf{s}'=H\mathbf{r}_0^t \pmod q$ and $\mathbf{r}_0=-\mathbf{r} \pmod q$. In fact, the vector $\mathbf{e}'$ which is the reduction of the vector $\mathbf{s}'$ by $\mathcal{G}$ is the optimal solution of $\mathrm{IP}_{H,\mathbf{1},q}(\mathbf{s}')$. Let $\mathbf{c}'=\mathbf{r}_0-\mathbf{e}'$, so $\mathbf{c}'$ is the closest codeword to $\mathbf{r}_0$ with respect to the G-norm. Put $\mathbf{c}_0=-\mathbf{c}' \pmod q$, thus $\|\mathbf{c}_0-\mathbf{r}\|$ is minimized. Finally, we choose either $\mathbf{c}$ or $\mathbf{c}_0$ whichever minimizes $\{\|\mathbf{r}-\mathbf{c}\|_G,\|\mathbf{c}_0-\mathbf{r}\|_G\}$. If $\|\mathbf{e}\|=\|\mathbf{e}'\|$ then we choose $\mathbf{c}$ or $\mathbf{c}_0$ randomly. This codeword is the nearest one to the received vector $\mathbf{r}$ which its error vector has the smallest G-norm.

Example 6. *Let $\mathcal{C}$ be a 3-ary code in $\mathbb{Z}_3^5$, whose parity-check matrix is $H=\begin{bmatrix}1&1&2&0&0\\0&1&0&1&1\end{bmatrix}$. Let $\boldsymbol{r}=(2,2,2,0,0)$ be a received word and $\boldsymbol{e}=(e_1,\dots,e_5)$ be an error vector. Solving the following IP problem is equivalent to decoding $\boldsymbol{r}$ to the nearest codeword of 3-ary code $\mathcal{C}$,*

$$IP_{H,\boldsymbol{1},3}(\boldsymbol{s})=\begin{cases}\textit{minimize } \|\boldsymbol{e}\|_G\\ \textit{subject to}\begin{cases}e_1+e_2+2e_3=2(\mathrm{mod}\ 3)\\ e_2+e_4+e_5=2(\mathrm{mod}\ 3)\\ e_i\in\mathbb{Z}_3, i=1,\dots,5\end{cases}\end{cases}$$

where $\boldsymbol{s}\equiv\boldsymbol{r}H^t(\mathrm{mod}\ 3)=(2,2)$. Based on our method, we consider the $\mathbb{Z}^{12}$-submodule associated with the row vectors

of matrix $H^\star$

$$H^\star = \begin{pmatrix} 1 & 0 & 3 & 0 & 0 & 0 & 0 & \bar{1} & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 3 & 0 & 0 & 0 & 0 & \bar{1} & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & \bar{1} & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & \bar{1} & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & \bar{1} \\ 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

$\mathcal{G}$ *the* RGB *associated with the submodule is*

$$\left\{ \begin{matrix} 000000300000, & 000003000000, & 000030000000, & 000300000000 \\ 003000000000, & 010000000001\bar{1}, & 000000000003, & 000000000011\bar{1} \\ 1000000\bar{1}0000, & 000000020\bar{1}00, & 000000010100, & 0000000\bar{1}0200 \\ 000000011\bar{1}001, & 000000001110\bar{1}, & 00000001\bar{1}0\bar{1}\bar{1}, & 000000002\bar{1}02 \\ 00000000\bar{2}\bar{1}01, & 0000000\bar{1}1002\bar{1}, & 000000003000 & \end{matrix} \right\}.$$

The reduction of $\iota_{75}(\boldsymbol{s})$ *with respect to* $\mathcal{G}$ *is computed and* $\pi_{75}(Red(\iota_{75}(\boldsymbol{s}), \mathcal{G})) = (02000)$. *Thus* $\boldsymbol{e} = (02000)$ *and* $\boldsymbol{c} = \boldsymbol{r} - \boldsymbol{e} = (20200)$. *Since* $\|\boldsymbol{e}\|_G = 2 \neq 1$, *we compute the reduction of* $\iota_{75}(\boldsymbol{s}')$ *for* $\boldsymbol{s}' = \boldsymbol{r}_0 H^t$ *and* $\boldsymbol{r}_0 = -\boldsymbol{r}(\mathrm{mod}\ 3) = (11100)$. *Therefore* $\pi_{75}(\mathrm{Red}(\iota_{75}(\boldsymbol{s}'), \mathcal{G})) = (01000), \boldsymbol{e}' = (01000)$ *and* $\boldsymbol{c}' = (10100)$. *Since* $\|\boldsymbol{e}'\|_G = 1$ *is less than* $\|\boldsymbol{e}\|_G$, $\boldsymbol{c}_0 = -\boldsymbol{c} = (20200)$ *is the desired codeword. In this case,* $\boldsymbol{c}$ *is equal to* $\boldsymbol{c}_0$.

IV. Discussion on Complexity

By and large, for decoding q-ary codes via G-norm we solve the IP problem twice and as a consequence we compute the reduction by $\mathcal{G}$ twice. Thus, the main part of the decoding complexity is due to the division algorithm. The reduction in submodules of $\mathbb{Z}^n$ is equivalent to subtraction of vectors.

We analyze the complexity by counting the number of required arithmetic operations. Hereafter, we set $\iota_{mn}(\mathbf{h}'_r) = \upsilon_r$. To obtain the complexity we prove the following Lemma.

Lemma 7. *Let* $\mathcal{G} = \{\boldsymbol{g}_1, \ldots, \boldsymbol{g}_t\}$ *be a* RGB *of* $\mathbb{Z}^{m+n}$*-module* $\mathcal{M}$ *associated with* q*-ary code* $\mathcal{C} \subseteq \mathbb{Z}_q^n$. *Then* $-q \leq g_{lj} \leq q$ *for* $j = 1, \ldots, m+n$ *and* $l = 1, \ldots, t$.

Proof: Let $K[x_1, \ldots, x_{m+n}]$ denote the polynomial ring over field K. Consider $X^{\mathbf{b}} = x_1^{b_1} \ldots x_{m+n}^{b_{m+n}}$ where $\mathbf{b} \in \mathbb{Z}_{\geq 0}^{m+n}$. Given any point $\mathbf{c} = (c_1, \ldots, c_{m+n}) \in \mathbb{Z}_{\geq 0}^{m+n}$ we use the notation $X^{\mathbf{c}} = X^{\mathbf{c}^+} - X^{\mathbf{c}^-}$. The following binomial ideal

$$I = (X^{\upsilon_1} - x_{m+1}, \ldots, X^{\upsilon_n} - x_{m+n}, x_1^q - 1, \ldots, x_m^q - 1)$$

is equivalent to $\mathcal{M}$. Let $\mathcal{G}_I$ be a GB of I. Exponents of binomials in $\mathcal{G}_I$ are equivalent to vectors in $\mathcal{G}$. Basically, the accuracy of our claim depends on existence of $x_k^q - 1$, $1 \leq k \leq m$. The Buchberger criterion is used to obtain $\mathcal{G}_I$. In the first step the S-polynomials of I for $1 \leq k, k' \leq m$ and $1 \leq r, s \leq n$, are as follows

$$S(X^{\upsilon_r} - x_{m+r}, X^{\upsilon_s} - x_{m+s}) = \frac{x_{m+s}X^{\upsilon_r} - x_{m+r}X^{\upsilon_s}}{\gcd(X^{\upsilon_r}, X^{\upsilon_s})},$$
$$S(X^{\upsilon_r} - x_{m+r}, x_k^q - 1) = \frac{X^{\upsilon_r} - x_k^q x_{m+r}}{x_k^{\upsilon_{rk}}},$$

$$S(x_k^q - 1, x_{k'}^q - 1) = x_k^q - x_{k'}^q.$$

The equivalent vectors to these S-polynomials have components greater than $-q-1$ and less than $q+1$. The remainders of these polynomials with respect to I are computed and this process continues. Thus the application of the Buchberger criterion to the powers of $\mathcal{G}_I$ yields the statement. ∎

In each stage of reduction it is checked whether or not ${\boldsymbol{g}_l}^+$ divides $\iota_{mn}(\mathbf{s})$, $l = 1, \ldots, t$. We assume that ${\boldsymbol{g}_1}^+ | \iota_{mn}(\mathbf{s})$ and $\mathbf{b}_1 = \mathrm{Red}(\iota_{mn}(\mathbf{s}), \boldsymbol{g}_1)$. Let γ be the greatest positive component of vector $\boldsymbol{g}_1$. Since $\mathbf{s} \equiv \mathbf{r}H^t (\mathrm{mod}\ q)$, we consider all the components of $\mathbf{s}$ to be at most $q-1$. In the first stage of the reduction we have at most $[\frac{q-1}{\gamma}]$ subtractions of $(m+n)$-tuple vectors. By Lemma 7, $1 \leq \gamma \leq q$ so an upper bound on the reduction operation takes place when $\gamma = 1$. Thus at most $(m+n)q$ subtractions in the first stage are done. In the next step $\boldsymbol{g}_1^+ \nmid \mathbf{b}_1$. During the subtraction, the components of $\mathbf{b}_1$ grow up to $[\frac{q-1}{\gamma}]q + q - 1$ or at most $(q-1)(q+1)$. We choose $\boldsymbol{g}_1, \ldots, \boldsymbol{g}_t$, where during the division algorithm none of the vectors $\boldsymbol{g}_1^+, \ldots, \boldsymbol{g}_{l-1}^+$ divides $\mathbf{b}_l$. This procedure continues in the same manner. In the l-th stage at most $(m+n)(q-1)(q+1)^{l-1}$ subtractions are done and $0 \leq b_{lk} \leq (q-1)(q+1)^l$. Hence the complexity of reduction is bounded above by

$$\sum_{l=1}^{t} (m+n)(q-1)(q+1)^{l-1} \leq (m+n)(q+1)^t,$$

where t, n, m and q are the number of elements in GB, the length and the dimension of q-ary code, respectively.

V. Conclusion

In a nutshell, the main result of this paper is decoding q-ary codes as an IP problem. In the previous works, binary codes had been decoded via binomial ideals associated with the IP problem. In this paper, we use G-norm to decode non-binary q-ary codes and consider an IP problem of decoding as a submodules of $\mathbb{Z}^n$. The number of elements in GB, length, dimension and character of q-ary code play a key role in the decoding complexity.

Acknowledgment

The authors would like to thank Dr. Enrico Paolini and the reviewers for their comments on improving the paper.

References

[1] M. Aliasgari, M.-R. Sadeghi, and D. Panario, "Gröbner bases for lattices and an algebraic decoding algorithm," *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1222–1230, 2013.

[2] A. H. Banihashemi and F. R. Kschischang, "Tanner graphs for block codes and lattices: construction and complexity," *IEEE Trans. Inf. Theory*, vol. 47, pp. 822–834, 2001.

[3] G. Boffi and A. Logar, "Computing Gröbner bases of pure binomial ideals via submodules of $\mathbb{Z}^n$," *J. Symbolic Computation*, vol. 47, pp. 1297–1308, 2012.

[4] M. Borges-Quintana, M. A. Borges-Trenard, P. Fitzpatrick, and E. Martínez-Moro, "Gröbner bases and combinatorics for binary codes," *AAECC*, vol.19, pp. 393–411, 2008.

[5] P. Conti and C. Traverso, "Buchberger algorithm and integer programming," *Applied algebra, algebraic algorithms and error-correcting codes*, Lecture Notes in Comput. Sci. vol. 539, pp. 130–139, 1991.

[6] J. Feldman, M. J. Wainwright, and D. R. Karger, "Using linear programming to decode binary linear codes," *IEEE Trans. Inf. Theory*, vol. 51, pp. 954–972, 2005.

[7] D. Ikegami and Y. Kaji, "Maximum likelihood decoding for linear block codes using Gröbner bases," *IEICE Trans. Fund. Electron. Commun. Comput. Sci. E86-A*, vol. 3, pp. 643–651, 2003.

[8] M. R. Sadeghi, A. H. Banihashemi, and D. Panario, "Low density parity check lattices: construction and decoding analysis," *IEEE Trans. Inf. Theory*, vol. 52, pp. 4481–4495, 2006.

[9] I. Márquez-Corbella and E. Martínez-Moro, "Algebraic structure of the minimal support codewords set of some linear codes," *Advances in Mathematics of Commun.*, vol. 5, pp. 233–244, 2011.

[10] R. R. Thomas, "A geometric Buchberger algorithm for integer programming," *Mathematics of Operations Research*, vol. 20, pp. 864–884, 1995.