

The simulation results for GS list-decoding of RS codes

ZhiAn Zheng DangHai PHAM Tomohisa Wada

Department of information Engineering, University of the Ryukyus, Okininawa, 903-0129 Japan

E-mail: zheng@lsi.ie.u-ryukyu.ac.jp , phdang@lsi.ie.u-ryukyu.ac.jp , wada@ie.u-ryukyu.ac.jp

Abstract Reed-Solomon codes are block-based error correcting codes with a wide range of applications in digital communications and storage. The basis discussion on error correction capacity of Guruswami-Sudan(GS) decoding for RS codes is depicted nevertheless it appears in some other paper. The interpolation and factorization procedure is also presented using kotter's algorithm and Roth-Ruckenstein algorithm respectively. Finally, we give the simulation result of error-correcting performance for different code rate.

Keyword list-decoding, hasse derivative, finite field, interpolation, factorization

I. Introduction

Reed-Solomon(RS) codes are nonbinary linear block codes over $GF(q)$ of length n and dimension k with minimum distance $d_{\min} = n - k + 1$, denoted as $RS(n, k, d_{\min})$. Since the nonbinary nature and maximum minimum distance property of codeword, the RS codes are widely used in digital communications and storage system.

In most of these existing systems, RS codes are decoded via an algebraic hard decision decoding (HDD) algorithm using BM algorithm [1] which can correct errors up to only half of the minimum distance $t_0 = \lfloor (d_{\min} - 1) / 2 \rfloor$, e.g. a fraction $(1 - R) / 2$ of n errors (where $R = k / n$ for $RS(n, k)$ codes). In 1999, GS(Guruswami-Sudan) list-decoding algorithm[2] was arising which can correct errors up to a fraction $1 - \sqrt{R}$ of n errors which is larger than or equal to error correcting capacity of BM algorithm.

This paper approaches the GS list-decoding algorithm from the point of view of a researcher who wants to know the detailed structure of the decoding procedures and the potential of the GS list-decoding.

In section II, we show a brief introduction of some concepts that are necessary to understand the algorithm. Section III is a brief review of the GS list-decoding. Section IV gives detailed procedures of the GS list-decoding. In section V, we discuss the potential of the algorithm. Section VI is a simulation result. Conclusions are offered in Section VII.

II. Some relevant concepts

Before the detailed introduction of the GS list-decoding, this section shows some relevant concepts which are necessary to understand the algorithm.

2.1 Feng Monomial ordering

Definition1, Feng Monomial ordering:

The $(1, \nu)$ revlex ordering of the bivariate monomials is defined as follows

$$x^i y^j < x^{i'} y^{j'} \quad (1)$$

if either $i + \nu j < i' + \nu j'$ or $i + \nu j = i' + \nu j'$ and $j < j'$.

With the above definition, we can arrange the monomials in order, then one bivariate polynomial can be expressed as summation of ordered monomials ,

$$Q(x, y) = \sum_{a=0}^{w_x} \sum_{b=0}^{w_y} q_{a,b} x^a y^b = \sum_{i=0}^{C'} q_i \Phi_i(x, y) \quad (2)$$

Here $1 = \Phi_0 < \Phi_1(x, y) < \Phi_2(x, y) < \dots$

2.2 Weighted degree

Definition2, Weighted degree:

The (u, ν) weighted degree of a monomial $x^i y^j$ is defined as $iu + \nu j$. Then the (u, ν) -weighted degree of a bivariate polynomial $Q(x, y)$ shown in formula (2) is

$$\deg^{(u, \nu)} Q(x, y) = \max \{ au + \nu b : a < w_x, b < w_y \} \quad (3)$$

2.3 Definition of $N(\nu, l)$

Lemma1 gives the number of monomials of a bivariate polynomial for given maximum weighted $(1, \nu)$ degree.

Lemma1: Let $N(v,l)$ be the number of monomials of a bivariate polynomial with weighted $(1, v)$ degree less than or equal to l , Then

$$N(v,l) = \left(\left\lfloor \frac{l}{v} \right\rfloor + 1 \right) (l + 1 - \frac{v}{2} \left\lfloor \frac{l}{v} \right\rfloor) \quad (4)$$

2.4 Definition of $B(w_b, v)$

Let $B(w_b, v)$ be the number of monomials of a bivariate polynomial for the last monomial of y^{w_b} with respect to $(1, v)$ revlex order.

Lemma2 (see detailed proof in [6]):

$B(w_b, v) = |\{(i, j) : i + vj \leq w_b v\}| - 1$, then

$$B(w_b, v) = \frac{vw_b^2}{2} + \frac{(v+2)w_b}{2} \quad (5)$$

III. GS list-decoding

For $RS(n, k)$ codes, k symbols message is represented by the coefficients of a degree $k-1$ message polynomial $f(x)$, and a length n codeword is formed by evaluating $f(x)$ at the nonzero elements of $GF(q)$

$$c = (f(1), f(\alpha), \dots, f(\alpha^{q-2})) \quad (6)$$

For some fixed ordering of the n field elements $\{\alpha^i\}$. This evaluation map view of RS codes leads to the GS list-decoding algorithm.

The GS list-decoding algorithm is shown in Fig.1.

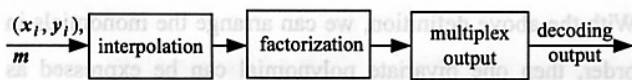


Fig.1. GS list-decoding structure

- 1) Interpolation: Find a bivariate interpolation polynomial $Q(x, y)$ of minimal $(1, k-1)$ weighted degree that pass through the point (x_i, y_i) with prefixed m times. Here $i=1, 2, \dots, n$, $x_i = 1, \alpha, \dots, \alpha^i, \dots, \alpha^{q-2}$ and y_i is received data.
- 2) Factorization: Given the interpolation polynomial $Q(x, y)$, identify all the factors of $Q(x, y)$ of the form $y - f(x)$ with $\deg f(x) < k$. Produce a list of the codewords that correspond to these factors.
- 3) Multiplex output: Choose the most likely output in the codewords list from factorization step.

IV. Procedures of the GS list-decoding

As shown in section II, the GS list-decoding consists of interpolation, factorization and Multiplex output 3 parts. In this section, we describe detailed structure of these three steps

3.1 Kotter's solution for Interpolation

Guruswami and Sudan were the first to use the idea of Hasse derivative to formulate the problem of list-decoding as bivariate polynomial interpolation, which enable the algorithm efficient.

Consider the bivariate polynomial with coefficient chosen from a finite field:

$$Q(x, y) = \sum_{a=0}^{w_x} \sum_{b=0}^{w_y} q_{a,b} x^a y^b \quad (7)$$

The (r, s) th Hasse derivative of a bivariate polynomial $Q(x, y)$ is defined for integers $r, s \geq 0$ as:

$$Q^{[r,s]}(x_i, y_i) = \sum_{b=s}^{w_y} \sum_{a=r}^{w_x} \binom{a}{r} \binom{b}{s} x_i^{a-r} y_i^{b-s} q_{a,b} \quad (8)$$

To perform interpolation, we will need to ensure that bivariate polynomial and their derivatives pass through certain points. We say that a bivariate polynomial $Q(x, y)$ passes through a point (x_i, y_i) with multiplicity m if $Q^{[r,s]}(x_i, y_i) = 0$ for all integers $r, s \geq 0$ that $r + s < m$.

The interpolation step actually consists of solving the linear system:

$$Q^{[r,s]}(x_i, y_i) = 0 \quad r + s < m \quad (9)$$

for all triples (x_i, y_i) . The cost of the interpolation C shown as formula (10) is the number of linear equations or constraints that need to be satisfied for the interpolation.

$$C = \frac{1}{2} m(m+1)n \quad (10)$$

Additionally, Kotter[3] used the ideas of linear-functional, modules, and cumulative kernels to construct $Q(x, y)$ that

likely output.

V. Error correction potential of the GS list-decoding

In order to discuss the error correction potential of the GS list decoding, let's recall the following two important theorems.

Theorem1 (The interpolation theorem) Let

$$Q(x, y) = \sum_{i=0}^C a_i \Phi_i(x, y), \quad (14)$$

Where the monomials are ordered according to an arbitrary monomial order. Then a nonzero $Q(x, y)$

Polynomial exists that interpolates the points $(x_i, y_i), i=1,2,\dots,n$ with multiplicity m at each point if

$$C = n \binom{m+1}{2} \quad (15)$$

Theorem2.(The factorization theorem) Let $Q(x, y)$ be an interpolating polynomial of $(1, v)$ -weighted degree $\leq l$ such that $Q^{[r,s]}(x_i, y_i) = 0$ for $i=1,2,\dots,n$ and for all $r+s < m$. (That is, each (x_i, y_i) is interpolated up to order m .) Let $p(x)$ be a polynomial of degree at most v such that $y_i = p(x_i)$ for at least K_m values of i in $\{1,2,\dots,n\}$. If $mK_m > l$, then $(y - p(x)) | Q(x, y)$.

Let us now establish a connection between the correction distance t_m , the multiplicity m , and the maximum $(1, v)$ -weighted degree of $Q(x, y)$. The point of the interpolation theorem is that the number of variables in the interpolating polynomial must exceed the number of constraints, which is $C = n \binom{m+1}{2}$. Recall from Lemma 1 that the number of monomials of weighted $(1, v)$ -degree l is $N(v, l)$. So by the interpolation theorem, we must have

$$n \binom{m+1}{2} < N(v, l) \quad (16)$$

By the factorization theorem, we must also have

$$mK_m \geq l+1 \quad (17)$$

Since $N(v, l)$ is increasing its second argument,

Then we have

$$N(v, mK_m - 1) > n \binom{m+1}{2} \quad (18)$$

For $m \geq 1$ we will define K_m to be the smallest value for which (11) is true:

$$K_m = \min\{K : N(v, mK - 1) > n \binom{m+1}{2}\} \quad (19)$$

Then the error correction distance (potential) t_m is,

$$t_m = n - K_m \quad (20)$$

VI. Simulation results

In this section, the simulation results are presented. Described as section IV, the error correcting potential of GS list-decoding is given by formula (19) and (20), based on it the error correcting capacity (potential) are simulated shown as Fig2.

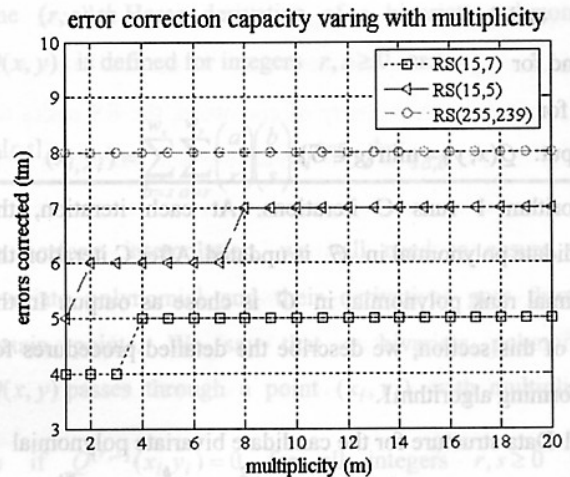
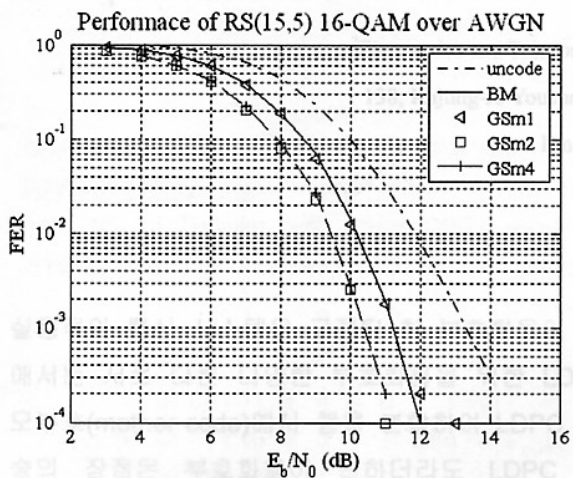


Fig2 Error correction capacity of GS list-decoding

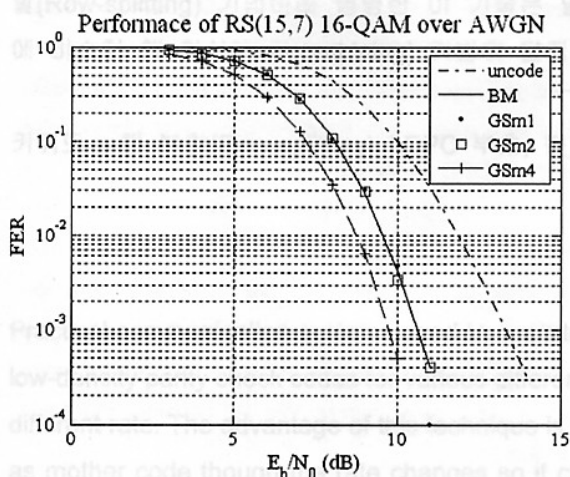
Recalling formula (4)(19)(20), t_m is a non-decreasing function of multiplicity m . Additionally, by our simulation experience, there is a multiplicity m_0 beyond which $t_{m_0} = t_{m_0+1} = \dots$ no further increases are possible. We denote this as t_∞ . That is increasing the multiplicity m can increase the error correction capacity, up till the point t_∞ is reached. Fig2 shows GS list-decoding capacity for three different rate RS codes. Based on it we know that GS list-decoding is efficient for low rate codes but it is no any improvement for high rate codes.

Figure3 shows the error correcting performance for RS(15,5)

and RS(15,7) respectively. The following notation will be used in the legend. "uncode" refers to the error correcting performance of no using frame error code(FEC) systems, "BM" refers to conventional bounded distance decoding scheme, and "GSm1","GSm2","GSm4" refer to GS list-decoding scheme with multiplicity 1,2,4 respectively. Based on fig3, we see that comparison to BM decoding, about 0.8dB coding gain is achieved for GS decoding of RS(15,5) with multiplicity 2, and also about 0.8dB coding gain is achieved for RS(15,7) with multiplicity 4. In addition, by algorithm 1, we must notice that the larger m , larger computational complexity.



(a) Error correcting performance for RS(15,5)



(b) Error correcting performance for RS(15,7)

Fig3. Error correcting performance of GS list-decoding

VII. Conclusions

In this paper, we showed the simulation of GS list-decoding. Some detailed procedure of interpolation step of GS list-decoding is described. As an important discussion for the error correcting potential of GS list-decoding is also emphasized. Based on the simulation results, GS list-decoding can achieve considerable coding gain comparison with conventional BM decoding for low rate codes, but it is no any improvement for high rate codes.

REFERENCES

- [1] E.R.Berlekamp, *Algebraic Coding Theory*. New York: McGraw-Hill, 1968.
- [2] V.Guruswami and M.Sudan, "Improved decoding of Reed-Solomon and algebraic geometric codes," *IEEE Trans.Inform.Theory*, vol.45, pp.1757-1767, Sept.1999
- [3] R.Koetter, "Fast generalized minimum-distance decoding of algebraic-geometry and Reed-Solomon codes," *IEEE Trans.Inform.Theory*, vol.42, pp.721-736, May 1996
- [4] R.M.Roth and G.Ruckenstein, "Efficient decoding of Reed-Solomon codes Beyond Half the Minimum Distance," *IEEE Trans.Info.Theory*, vol.46,no.1, pp.246-256,Jan.2000
- [5] Xinmiao Zhang and Keshab K.Parhi, "Fast factorization architecture in soft-decision Reed-Solomon Decoding," *IEEE Trans. on VLSI systems*, Vol.13,NO.4,2005
- [6] R.J.McEliece, "The Guruswami-Sudan decoding algorithm for Reed-Solomon codes," *IPN Progress Reports*, <http://www.systems.Caltech.edu/EE/Faculty/rjm/>, April 2003