U6CAU Proceedings Volume 1, Number 1, 58 - 66, September, 2019 **Maiden Edition**
on Harnessing African Potentials for Sustainable Development, Calabar, Nigeria **ISSN 1596-1273**

IMPROVED COMPLEXITY IN DECODING REED-SOLOMON CODES

Amadou Keita
Mathematics Department, University of The Gambia, The Gambia
Corresponding Author: akeita@utg.edu.gm

ABSTRACT

The Guruswami-Sudan (GS) algorithm is a standard algorithm that decodes beyond the classical decoding bound for Reed-Solomon codes. An analysis of some complexity improvement techniques namely polynomial reconstruction and basis transformation has been done to enhance the decoding capabilities of the algorithm and comparison of the techniques.

Key words: Reed-Solomon, decoding, basis transformation, polynomial reconstruction, complexity.

INTRODUCTION

Reed-Solomon codes were invented in 1960 and within a decade an efficient algorithm was discovered to decode them. They are efficiently encoded and decoded. Moreover, they are widely used because of their favourable properties including high error correction capability, burst-error (the type of error in which two or more symbols in a string of $\mathbb{F}_2^n$ change from 0 to 1 or vise-versa and the length of the burst error is the number of string symbols from the first corrupted to the last corrupted symbol) correction capability, and erasure-recovery capability. For Reed-Solomon codes with block length n and dimension k, the Johnson Theorem states that for a Hamming ball of radius $n-\sqrt{nk}$ there can be at most $O(n^2)$ codewords. The first efficient polynomial time algorithm that agrees with the theorem was discovered by Sudan (Sudan, 1997), which was improved by GS (Guruswami and Sudan, 1999). The GS paper (Guruswami and Sudan, 1999) considered the following problem:

Theorem 1: *Given n distinct elements $(x_1, x_2, \cdots, x_n)$ in $\mathbb{F}_q$ and another n elements $(y_1, y_2, \cdots, y_n)$ in $\mathbb{F}_q$, find polynomials $f(x)$ of degree at most $k-1$ that agree at at least t points.*

The algorithm basically takes as inputs a received word $\beta = (\beta_1, \cdots, \beta_n)$ (this is basically some n points $(\alpha_1, \beta_1), \cdots, (\alpha_n, \beta_n)$ where $(\alpha_1, \cdots, \alpha_n)$ are distinct field elements), the number of agreements t and the dimension k of the Reed-Solomon code. The algorithm constructs a bivariate polynomial which is then factored with some conditions. These factors, when they satisfy (as equations) at least t of the n points, are listed. The main idea behind the GS algorithm, reason why it works, is the introduction of more constraints on $Q(x,y)$. Fundamentally, in Sudan (Sudan, 1997), the $(1,k-1)$-weighted degree of $Q(x,y)$ being below a fixed upper bound is a very useful constraint. If the $(1,k-1)-$ weighted degree of $Q(x,y)$ is ∂ then $Q(x,f(x))$, where $\deg f(x) \le k-1$, has total degree at most ∂. It is easy to see that if $f(x)$ is a desired codeword, then $Q(x,f(x))$ has more than ∂ zeros which implies $Q(x,f(x)) \equiv 0$. The factorization step easily follows since $Q\big(x,f(x)\big) \equiv 0 \; iff \; y-f(x) | Q(x,y)$. This idea was adopted in the GS algorithm with some improvement on the choice of the multiplicity, m as compared to the case in Sudan (Sudan, 1997). The introduction of m tightens the decoding radius.

This paper presents some preliminaries, discusses some complexity reduction techniques and compares two of these techniques.

The GS Algorithm

The main idea behind the GS algorithm, as explained, works partly as a result of the following result.

Theorem 2 *If a polynomial of degree at most $k-1$ has at least k roots, then this polynomial is the zero polynomial.*

Proof. Take $R(x) \in \mathbb{F}[x]$ with $\deg R(x) \le$

$k-1$ and assume that $R(x)$ has at least k roots. Suppose there exists an $\tilde{x}$ such that $R(\tilde{x})=0$. Then $(x-\tilde{x})|R(x)$. If $R(x)$ has $(k-1)+1=k$ distinct roots $\tilde{x}_1,\tilde{x}_2,\cdots,\tilde{x}_k$, then $(x-\tilde{x}_1)(x-\tilde{x}_2)\cdots(x-\tilde{x}_k)|R(x)$. This results in a degree k polynomial that forms a factor of a degree at most $k-1$ polynomial. Therefore, $R(x)$ must be a zero polynomial. Q.E.D.

Theorem 2 implies that $d=n-k+1$ is a lower bound for the minimum distance of a Reed-Solomon code.

i. Univariate Polynomial Representation: The GS algorithm executes the decoding process of Reed-Solomon codes in two steps - interpolation and factorization. The interpolation step constructs a bivariate polynomial. It is easier to work with a univariate polynomial, instead, so a description of a general way to put down a univariate representation for any bivariate polynomial is given. Consider a bivariate polynomial, $Q(x,y)$, and define the set of all its monomials as $M_{x,y}:=\{x^iy^j: i,j\geq 0\}$. Denote $\mathbb{Z}_{\geq 0}$ as the set of nonnegative integers and construct the map

$$\varphi: M_{x,y} \longrightarrow \mathbb{Z}^2_{\geq 0}$$
$$\varphi(x^iy^j) \longmapsto (i,j),$$

which is a bijection of sets. Definitions, recorded as Outcomes 1, 2, 3 and 4, are concepts defined in the work of McEliece (McEliece, 2003).

Outcome 1 (Monomial ordering) *A monomial order is a total order '<' on the set of monomials $M_{x,y}$ such that*

- if $\lambda_1\leq\delta_1$, $\lambda_2\leq\delta_2$, then $(\lambda_1,\lambda_2)\leq(\delta_1,\delta_2)$;
- if $\lambda\leq\delta$ and $\lambda,\delta,\gamma\in\mathbb{Z}_{\geq 0}$, then $\lambda+\gamma\leq\delta+\gamma$.

Outcome 2 (Weighted degree) *For u,v nonnegative integers, the (u,v)-weighted degree of a monomial x^iy^j is $deg_{(u,v)}x^iy^j=ui+vj$.*

Outcome 3 ((u,v)-lexicographic order) *For two monomials $x^{i_1}y^{j_1}<x^{i_2}y^{j_2}$, implies that $ui_1+vj_1<ui_2+vj_2$, or $ui_1+vj_1=ui_2+vj_2$ and $i_1<i_2$.*

Outcome 4 ((u,v)-reverse lexicographic order) *For two monomials $x^{i_1}y^{j_1}<x^{i_2}y^{j_2}$, implies that $ui_1+vj_1>ui_2+vj_2$, or $ui_1+vj_1=ui_2+vj_2$ and $i_1>i_2$.*

Then for the set $M_{x,y}$ in a lexicographic or reverse lexicographic order, there is essentially a univariate representation $\varphi_0(x,y)<\varphi_1(x,y)<\varphi_2(x,y)<\cdots<\varphi_l(x,y)$ for some finite number of $\varphi_i(x,y)\neq 0$ in the variable y. That is to say, the bivariate polynomial can be written as

$$Q(x,y)=\sum_{i,j\geq 0}q_{i,j}x^iy^j=\sum_{j=0}^{l}c_j\varphi_j(x,y) \quad (1)$$

where c_j is an element of the field. This is important because it reduces the problem at the factorisation step to a root finding problem. Given an interpolation polynomial (any polynomial) Q, a well-known technique of finding its singularities is studying its partial derivatives. Hasse derivatives shall be used to study the singularities of Q (points where Q intersects itself) simply because partial derivatives of polynomials over fields of small characteristic are not well behaved.

Outcome 5 ((r,s)-Hasse derivative (Cassuto, Bruck and McEliece, 2013)**)** *The (r,s) Hasse derivative of a polynomial $Q(x,y)$, denoted $D_{r,s}Q(x,y)$, for any integer pair $r\geq 0$, $s\geq 0$ is*

$$D_{r,s}Q(x,y)=\sum_{i,j}\binom{i}{r}\binom{j}{s}q_{i,j}x^{i-r}y^{j-s}$$

where $q_{i,j}$ is the coefficient of x^iy^j in $Q(x,y)$.

In the GS algorithm, studying singularities by exploring partial derivatives was avoided for a more suitable technique. The technique was to shift a coordinate system to a chosen point (x_i,y_i) as a new origin, and then insist that the non-zero coefficients be of high degree. The shifting for $Q(x,y)$, where $\alpha,\beta\in\mathbb{F}_q$ is $Q_{\alpha,\beta}(x,y):=Q(x+\alpha,y+\beta)$.

This technique is essentially applying Hasse derivatives to the polynomial $Q(x,y)$ since $D_{r,s}Q(\alpha,\beta)=\text{coefficient}_{x^ry^s}Q(x+\alpha,y+\beta)$.

The following Theorem and Outcome were given in the work of McEliece (McEliece, 2003), where an elegant prove using binomial expansion was given for the theorem.

Theorem 3: *Let*
$Q(x,y) = \sum_{i,j} q_{i,j} x^i y^j \in \mathbb{F}_q[x,y]$.
For any $(\alpha,\beta) \in \mathbb{F}_q^2$, *the shift*
$Q(x+\alpha, y+\beta) = \sum_{r,s} D_{r,s} Q(\alpha,\beta) x^r y^s$.

Outcome 6: *Then*
$Q(x,y) = \sum_{r,s} D_{r,s} Q(\alpha,\beta)(x-\alpha)^r (y-\beta)^s$.

A second look at the factorization step of the GS algorithm suggests that using Outcome 6, the anticipated list of polynomials is just
$\mathcal{L} := \{f(x) \in \mathbb{F}_q[x] : (y - f(x)) | Q(x,y)\}$.

Complexity reduction

Outcome 7: (Hamming distance) *Let* $x = (x_1, x_2, \cdots, x_n)$ *and* $y = (y_1, y_2, \cdots, y_n)$ *be two strings in* $\mathbb{F}_q^n$. *The number of inequal symbols* $d(x,y) = |\{i \le n | x_i \ne y_i\}|$ *is the Hamming (minimum) distance between strings* x *and* y.

Outcome 8: (Reed-Solomon code) *Let* $(x_1, \cdots, x_n)$ *be* n *distinct elements in* $\mathbb{F}_q$. *The map* $\Psi: \mathbb{F}_q^k \longrightarrow \mathbb{F}_q^n$ with
$(f_0, f_1, \cdots, f_{k-1}) \longmapsto (f(x_1), f(x_2), \cdots, f(x_n))$,
where $(f_0, f_1, \cdots, f_{k-1})$ is a k-dimensional vector such that
$f(x) = f_0 + f_1 x + \cdots + f_{k-1} x^{k-1} \in \mathbb{F}_q[x]$
is some encoding process. The image space of this linear map forms a Reed-Solomon code.

i. Error-rate as a function of Message rate: Let $\mathcal{C}$ be an $[n,k,d]_q$ error-correcting code (i.e. $\mathcal{C}$ has codeword length n, dimension k and minimum distance d over a q -ary alphabet $\mathbb{F}_q$). This means the map $\mathcal{C}: \mathbb{F}_q^k \longrightarrow \mathbb{F}_q^n$ such that if $c_1, c_2 \in \mathcal{C}$ then c_1 and c_2 differ in at least d coordinates. For a Reed-Solomon code, $d \ge n - k + 1$ with $k < n \le q$. This construction is used to study the error-rate dependence on the message rate in decoding the Reed-Solomon code $\mathcal{C}$. Let e be the number of distinct coordinates between some two vectors in $\mathbb{F}_q^n$ where one is transmitted (a codeword) and another, a message is received. The number of alterations that can be corrected - the error in the transmission - is well-known to be $2e < d$. The error-rate is defined as $\varepsilon := \frac{e}{n}$, while the message rate is $\kappa := \frac{k}{n}$. The error-rate dependence on the message rate is studied in two cases. These are: (i) when both the error-rate and the message rate are fixed, and (ii) when the error-rate is not fixed. Basically, both rates are fixed when the number of errors is within the unique-decoding bound. A unique decoding bound μ is a bound for an error correcting code such that for a ball of radius μ centered at a received message, there can be only one codeword within the ball. A powerful algorithm by Peterson (Peterson, 1960) constructed that for a decoding radius $e < \frac{n-k}{2}$, decoding can be done without any complications. Thus the error-rate is $\varepsilon = \frac{1-\kappa}{2}$. It might happen that a list of codewords are found within a certain ball of radius $e \ge \frac{n-k}{2}$. Hence, unique-decodability might not be achieved here. A polynomial-time algorithm with the capability of decoding up to $e = \lfloor n - \sqrt{nk} \rfloor$ radius is acquired, thus
$\varepsilon = 1 - \sqrt{\kappa} = \frac{1-\kappa}{1+\sqrt{\kappa}} \ge \frac{1-\kappa}{2}$.

ii. Polynomial Reconstruction: For $\mathcal{C}$, an $[n,k,d]_q$ Reed-Solomon code, Theorem 1 is just
$\Gamma := \{f(x) \in \mathbb{F}_q[x] | \deg f(x) \le k-1,$
$f(x_i) = y_i$ for at least t points$\}$.

Finding all polynomials that agree at t points is equivalent to listing all codewords within a Hamming ball of radius $n - t$. Let $k < \lambda n$ for $0 < \lambda < 1$. Considering the GS problem, define a polynomial

$$f_j(x_i) := \frac{y_i - y_j}{x_i - x_j}, i \ne j. \quad (2)$$

Denote $\mathcal{C}$ an $[n,k,d]_q$ Reed-Solomon code having codewords agreeing at at least t points as an (n,k,t) instance of the GS problem. Let $j = 1$ in Equation (2), then

$$f_1(x_i) := \frac{y_i - y_1}{x_i - x_1}, i = 2, \cdots, n$$

with $n-1$ points $\left(\frac{y_2-y_1}{x_2-x_1}, \frac{y_3-y_1}{x_3-x_1}, \cdots, \frac{y_n-y_1}{x_n-x_1}\right)$. If $f(x_1)=y_1$ and $\deg f(x) \leq k-1$, then $\deg f_1(x) \leq k-2$ since the n-distinct input points $(x_1, x_2, \cdots, x_n)$ is reduced to an $(n-1)$-distinct input points $(x_2, \cdots, x_n)$. At

Theorem 4: *There exists a polynomial $f(x)$ such that $degf(x) \leq k-1$ and $f(x_i)=y_i$ for at least t points and $f(x_1)=y_1$ iff there is a polynomial $f_1(x)$ of degree at most $k-2$ such that*

$$f_1(x_i) := \frac{y_i-y_1}{x_i-x_1}, i = 2, \cdots, n$$

for at least $t-1$ points.

Proof. By Euclidean algorithm, let

$$\frac{f(x)}{x-x_1} = f_1(x) + \frac{f(x_1)}{x-x_1}.$$

This implies that $f(x) = f_1(x)(x-x_1)+y_1$. It is straight forward to see that indeed $\deg f_1(x) \leq k-2$.

Conversely, $f_1(x)(x-x_1)$ is a polynomial so define $f(x) := f_1(x)(x-x_1)+y_1$. Observe that $f(x_1)=y_1$ and $\deg f(x) \leq k-1$. Q.E.D.

Define

$$\Gamma_j := \Bigg\{ f_j(x) \in \mathbb{F}_q[x] | \deg f_j(x) \leq k-2, \\ f_j(x_i) = \frac{y_i-y_j}{x_i-x_j}, \\ i \neq j \text{ for at least } t-1 \text{ points} \Bigg\},$$

such that

$\Gamma = \cup_{j=1}^{n} \{f_j(x)(x-x_j)+y_j | f_j(x) \in \Gamma_j\}$.

This is because for all $f \in \Gamma$, $f(x_i)=y_i$ for at least t points and can be recovered from Γ_j by the reduction process explained above.

Theorem 5: *Let $k < \alpha n$, $0 < \alpha < 1$ and $\sqrt{nk}-c < t \leq \sqrt{nk}-c+1$ for $c > 0$. Let $\Gamma = \{f(x) \in \mathbb{F}_q[x] | degf(x) \leq k-1,$ $f(x_i) = y_i for\ at\ least\ t\ points\}$.*

Then the

1. *number of polynomials,*

$$|\Gamma| \leq O\left(n^{c\frac{2\sqrt{\alpha}}{(1-\sqrt{\alpha})^2}+c+2}\right);$$

2. *polynomials in Γ can be listed in*

this stage, it has reduced to an $(n-1, k-1, t-1)$ instance of the problem. Theorems 4 and 5 were results from (Muralidhara and Sen, 2009).

$$O\left(n^{\frac{2\sqrt{\alpha}}{(1-\sqrt{\alpha})^2}+c+10}\right) \textit{ time.}$$

The above result guarantees that Reed-Solomon codes can be successfully decoded even beyond the GS radius provided polynomial reconstruction is introduced. After a certain number of reductions, the GS algorithm is applied on the reduced points and a simple combinatorics to acquire the desired complexities, which are polynomial.

iii. Alternative Basis Transformation

(a) Syndrome-Based Decoding: The basic idea of error correction is to see to it that when a codeword is transmitted through a channel, which could influence limited alterations on the transmitted symbols, the receiver should successfully decode the transmitted message. The idea of encoding messages to be transmitted over a channel is basically to improve the decoding capabilities since alterations might occur during transmission.

Outcome 9: (Parity-Check Matrix) *This is a matrix whose rows form a basis for the dual space $\mathcal{C}^{\perp} \leq \mathbb{F}_q^n$. It is a $k \times n$ matrix.*

Outcome 10: (Syndrome (Berlekamp, 2015)**)** *The syndrome $\boldsymbol{s}$ of any received vector r is defined by the equation $s = \mathcal{H}r$, where $\mathcal{H}$ is the parity-check matrix.*

Let e be the number of errors that occur on a transmitted codeword $x \in C$ such that $y \in \mathbb{F}_q^n$ is received. Denote vector E as the error vector where the coordinates of E hold information about the locations of alterations. Thus, $E = (E_1, E_2, \cdots, E_n)$ with $E_i \in \mathbb{F}_q$ and

$$E_i = \begin{cases} 1 & \text{if there is an error at coordinate } i, \\ 0 & otherwise. \end{cases}$$

Let $\mathcal{J}$ be the set of error locations in a received vector y (i.e. all $E_i = 1$). In the book by Ron Roth (Roth, 2006), for syndrome coefficients $S_0, S_1, \cdots, S_{n-k-1}$ and E the error vector, the syndrome polynomial $S(x)$ is defined as

$$S(x) := \sum_{i=0}^{n-k-1} S_i x^i \equiv \sum_{j=1}^{n} \frac{E_j v_j}{1-\delta_j x} \operatorname{mod} x^{n-k}$$

where $\delta_j, v_j, j = 1, \cdots, n$ are some multipliers. In syndrome decoding, the error-locator polynomial is defined as

$$\Lambda(x) := \prod_{j\in\mathcal{J}} (1-\delta_j x)$$

and the error-evaluator polynomial is defined as

$$\Omega(x) := \sum_{j\in\mathcal{J}} E_j v_j \prod_{i\in\mathcal{J}\setminus\{j\}} (1-\delta_i x) \quad .$$

Observe that $\Lambda(0) \neq 0$ so the (well-defined) fraction.

The basic idea of syndrome-based decoding of Reed-Solomon codes to reformulate Theorem 1 in terms of modules over a univariate polynomial ring (in other words, Key Equations) will be used.

(b) Modules over $\mathbb{F}_q[x]$: Let $\alpha = (\alpha_1, \cdots, \alpha_n)$ be some transmitted codeword and $\beta = (\beta_1, \cdots, \beta_n)$ be the received word. The GS algorithm has order of multiplicity parameter m (which is 1 in Sudan (Sudan, 1997)) for the n points $(\alpha_1, \beta_1), (\alpha_2, \beta_2), \cdots, (\alpha_n, \beta_n)$. Let l be the length of the list $\mathcal{L}$ and τ be the maximum decoding radius. The bivariate polynomial

$$Q(x,y) = \sum_{t=0}^{l} Q_t(x) y^t \in \mathbb{F}_q[x,y]$$

is the desired polynomial for the interpolation step which satisfies some conditions including the $(1, k-1)$-weighted degree must be as small as possible. This means, for any monomial $x^i y^j$ in $Q(x,y)$, $i + (k-1)j < m(n-\tau)$. By our construction of $Q(x,y)$, this weighted degree condition has that $\deg Q_t(x) < m(n-\tau) - (k-1)t$, for $t = 0,1,\cdots,l$. Hence, define

$N_t := m(n-\tau) - (k-1)t, \text{for} t = 0,1,\cdots,l$.

All one needs to know about the multiplicity parameter is that for any monomial $x^i y^j$ in $Q(x,y)$, the coefficient q_{ij} of $x^i y^j$ is zero whenever $i + j < m$.

Let $R(x)$ be the Lagrange interpolation polynomial such that $R(\alpha_i) = \beta_i$ and define some polynomial $G(x) := \prod_{i=1}^{n} (x - \alpha_i)$.

From Outcome 6,

$$Q(x,y) = \sum_{r,s} D_{r,s} Q(\alpha,\beta)(x-\alpha)^r (y-\beta)^s \qquad =$$

$$\frac{\Omega(x)}{\Lambda(x)} = \frac{\sum_{j\in\mathcal{J}} E_j v_j}{\sum_{j\in\mathcal{J}} (1-\delta_j x)} \equiv \sum_{j\in\mathcal{J}} \frac{E_j v_j}{1-\delta_j x} \operatorname{mod} x^{n-k}$$

suggests a Key Equation that relates the error-locator and the error-evaluator polynomials as

$$\frac{\Omega(x)}{\Lambda(x)} \equiv S(x) \operatorname{mod} x^{n-k}. \qquad (3)$$

Here, the decoder basically computes $S(x)$ from the received word y, solves the Key Equation (i.e. Equation (3)) for the error-locator polynomial $\Lambda(x)$ to determine its roots, and finally compute the error-evaluator $\Omega(x)$ to determine the error values.

$$= \sum_{r,s} D_{r,s} Q_t(\alpha) \beta^t (x-\alpha)^r (y-\beta)^s.$$

Therefore, having $r = 0$, $0 \leq t \leq l$, and $0 \leq s < m$ gives that

$$Q(x,y) = \sum_{s=0}^{l} D_{0,s} Q_t(\alpha) \beta^t (y-\beta)^s,$$

where $Q(x,y)$ has a multiplicity at least m at (α_i, β_i) if and only if $(x-\alpha_i)^{m-s}$ divides $D_{0,s} Q_t(\alpha) \beta^t$ for all s, t with $0 \leq s < m$ and $t = 0,1,\cdots,l$. Below, a result which is fundamental in the reformulation process is proved.

Theorem 6: *A bivariate polynomial* $Q(x,y) \in \mathbb{F}_q[x,y]$ *has multiplicity at least* m *at every* (α_i, β_i) *if and only if* $G(x)^{m-s} | D_{0,s} Q(x, R(x))$ *for all* s *with* $0 \leq s < m$.

It shall be proved that the multiplicity condition implies each factor $(x-\alpha_i)^{m-s}$ divides $D_{0,s} Q(x, R(x))$ and by the uniqueness in the factors for all i, the product $G(x)^{m-s}$ divides it. For the converse, remainder theorem and Hasse derivatives will be helpful.

Proof. Let $(\alpha_i, \beta_i) \in \mathbb{F}_q^2$, $R(x) \in \mathbb{F}_q[x]$ such that $R(\alpha_i) = \beta_i$. Suppose $Q(x,y)$ has multiplicity at least m at (α_i, β_i). Equation (1) gives a univariate representation such that

$D_{0,s} Q(x, R(x)) = \sum_{i \geq m-s} D_{0,s} c_i \varphi_i(x, R(x))$.

Shift to the origin - assume $(\alpha_i, \beta_i) = (0,0)$. This means $R(0) = 0$, so $x | R(x)$. Consider $s < m$ with each polynomial $D_{0,s} \varphi_i(x,y)$ with no terms of degree less than $m - s$. Since each

$D_{0,x^{m-s}|D_{0,s} c_i \varphi_i(x,R(x))} \text{ for all } s \text{ with } 0 \leq s < m. {}_{s} \varphi_i(x,y)$

have only terms with degrees greater than or equal to $m-s$, and $x|R(x)$, then shifting back, it follows that

$$(x-\alpha_i)^{m-s}|D_{0,s}c_i\varphi_i\big(x,R(x)\big) \text{ for all } s,i$$
$$\text{with } 0 \leq s < m \text{ and } 1 \leq i \leq n.$$

Since $(x-\alpha_i)$ are distinct for all i, then the product $G(x)^{m-s}$ divides $D_{0,s}Q(x,R(x))$ too. Conversely, suppose

$$G(x)^{m-s}|D_{0,s}Q\big(x,R(x)\big) \text{ for all } s$$
$$\text{with } 0 \leq s < m$$

. Then each factor (shifted to the origin, i.e. $\alpha_i = 0$) $x^{m-s}|D_{0,s}c_i\varphi_i(x,R(x))$. This means that for some polynomial $U_s(x)$,

$$D_{0,s}Q\big(x,R(x)\big) = x^{m-s}U_s(x)$$
$$\text{with } 0 \leq s < m.$$

By Outcome 6, it follows that

$$\begin{aligned} Q(x,y) &= \textstyle\sum_s D_{0,s}Q(x,R(x))(y-R(x))^s \\ &= \textstyle\sum_{s<m} D_{0,s}Q(x,R(x))(y-R(x))^s + \\ &\quad \textstyle\sum_{s\geq m} D_{0,s}Q(x,R(x))(y-R(x))^s \\ &= \textstyle\sum_{s<m} x^{m-s}U_{(m-s)}(x)(y-R(x))^s + \\ &\quad \textstyle\sum_{s\geq m} D_{0,s}Q(x,R(x))(y-R(x))^s. \end{aligned}$$

Observe that a common factor $(y-R(x))^s$ has only terms of degree at least s since $x|R(x)$. Therefore, every term in $Q(x,y)$ has degree greater than m. Q.E.D.

This proves that the multiplicity and weighted degree conditions of the GS interpolation step are satisfied by $Q(x,y)$ if and only if there exists a polynomial $B_s(x) \in \mathbb{F}_q[x]$ such that

$$D_{0,s}Q(x,R(x)) = B_s(x)G(x)^{m-s}$$
$$\text{with} 0 \leq s < m \quad (4)$$

and

$$\deg B_s(x) < l(n-k) - m\tau + s,$$
$$\text{with } 0 \leq s < m.$$

In the following, a reformulation of the GS interpolation conditions in terms of modules over a univariate polynomial ring, $\mathbb{F}_q[x]$, is given.

Outcome 11: *A bivariate* $Q(x,y) \in \mathbb{F}_q[x,y]$ *is an interpolation polynomial if and only if it is parametrizable as* $\sum_{a=0}^{l} Q_a(x)y^a$.

Proof. Let $Q(x,y) \in \mathbb{F}_q[x,y]$ be an interpolation polynomial. It follows from the proof of Theorem 6 that

$$\begin{aligned} Q(x,y) = \textstyle\sum_{a=0}^{l} \Big[&\textstyle\sum_{i=0}^{m-1} B_i(x)G(x)^{m-i}(-R(x))^{i-a}\binom{i}{a} + \\ &\textstyle\sum_{i=m}^{l} B_i(x)(-R(x))^{i-a}\binom{i}{a}\Big] y^a \end{aligned}$$
$$=: \textstyle\sum_{a=0}^{l} Q_a(x)y^a \in \mathbb{F}_q[x]$$

where polynomials $B_i(x)$ as variables are considered to be variables.

Conversely, suppose

$$Q(x,y) = \textstyle\sum_{a=0}^{l} Q_a(x)y^a$$

and that

$\deg Q_a(x) < m(n-\tau) - a(k-1)$ for all a with $0 \leq a \leq l$. The parametrization becomes a valid interpolation polynomial. Q.E.D.

Block-Hankel Matrix: Equation (4), with the definition of Hasse derivatives, can be recorded as

$$\textstyle\sum_{t=s}^{l} \binom{t}{s} Q_t(x)(R(x))^{t-s} = B_s(x)G(x)^{m-s},$$

for all s with $0 \leq s < m$.

Let the reciprocal polynomials be the following

$$\begin{aligned} \bar{R}(x) &= x^{n-1}R(x^{-1}), \\ \bar{G}(x) &= x^n G(x^{-1}) = \textstyle\prod_{i=1}^{n}(1-\delta_i x), \\ \bar{B}_s(x) &= x^{l(n-k)-m\tau-s-1}B(x^{-1}), \\ \Lambda_t(x) &= x^{N_t-1}Q_t(x^{-1}), \end{aligned}$$

which, when inserted in Equation (5) yield

$$\textstyle\sum_{t=s}^{l} \binom{t}{s} \Lambda_t(x)x^{(l-t)(n-k)}\bar{R}(x)^{t-s} =$$
$$\bar{B}_s(x)\bar{G}(x)^{m-s}, \text{ for all } s \text{ with } 0 \leq s < m.$$

Since $\bar{G}(0) \neq 0$, the following is well-defined

$$T^{(s,t)}(x) := \frac{\bar{R}(x)^{t-s}}{\bar{G}(x)^{m-s}}.$$

Thus, considering the $\binom{m+1}{2}n$ system of homogeneous linear equations, it follows that

$$\textstyle\sum_{t=s}^{l} \binom{t}{s} \Lambda_t(x)x^{(l-t)(n-k)}T^{(s,t)}(x) \equiv$$
$$\bar{B}_s(x)\text{mod}x^{m(n-\tau)+l(n-k)-s(n-1)}, 0 \leq s < m.$$

Outcome 12: (Guruswami-Sudan Syndrome (Zeh, Gentner, & Augot, 2011)**)** *The syndrome polynomials*

$S^{(0,0)}(x), S^{(0,1)}(x), \cdots, S^{(0,l)}(x), S^{(1,1)}(x), \cdots, S^{(m-1,l)}(x)$ with

$$S^{(s,t)}(x) = \textstyle\sum_{i=0}^{(m-s)n+N_t-1} S_i^{(s,t)}x^i$$

are given by

$$S_i^{(s,t)} = T_{i+(s+1+t(n-1)-mn)}^{(s,t)}, \text{for} t = s, \cdots, l.$$

By Outcome 12, the homogeneous system can be modified, considering weighted degree conditions, as

$$\sum_{t=s}^{l} \Lambda_t(x) S^{(s,t)}(x) \equiv \bar{B}_s(x) \bmod x^{m(n-\tau)+l(n-k)-s(n-1)}, \quad (6)$$

where deg $\bar{B}_s(x) < l(n-k) - m\tau + s$, for all s with $0 \le s < m$. With the high degree terms in Equation (6) considered, there are

$$\sum_{s=0}^{m-1} (m-s)n = \binom{m+1}{2}$$

homogeneous equations and they can be parametrized as

$$\sum_{t=s}^{l} \sum_{i=0}^{N_t-1} Q_t^i S_{j+i}^{(s,t)} = 0, 0 \le j < (m-s)n, 0 \le s < m, \quad (7)$$

where Q_t^i is a univariate polynomial (a coefficient).

Outcome 13: (Hankel matrix (Zeh et al., 2011)**)** *An* $m \times n$ *matrix* S *is a Hankel matrix if* $S_{i,j} = S_{i-1,j+1}$ $\forall i, j$ *with* $1 \le i \le m-1$, $0 \le j < n-1$.

The linear system in Equation (7) gives a Block-Hankel matrix (each sub-matrix $S^{(s,t)}$ is a Hankel matrix).

(d) Basis Transformation: There is an efficient algorithm by (Beelen and Brander, 2010) that solves the Key Equations for Theorem 1. A highlight of how their technique guarantees improved complexity is given. To see how they solved the Key Equations, one should read Section 3 in (Beelen and Brander, 2010). From the parametrization in Outcome 11, let

$$\begin{pmatrix} Q_0(x) \\ Q_1(x) \\ \vdots \\ Q_l(x) \end{pmatrix} = \mathcal{A} \begin{pmatrix} B_0(x) \\ B_1(x) \\ \vdots \\ B_l(x) \end{pmatrix} \qquad (8)$$

where $\mathcal{A} := [\mathcal{A}_1 | \mathcal{A}_2]$ such that

$$[\mathcal{A}_1]_{a,i} = G(x)^{m-i}(-R(x))^{i-a}\binom{i}{a},$$

with $(a, i) \in \{0,1,\cdots,l\} \times \{0,1,\cdots,l\}$ and

$$[\mathcal{A}_2]_{a,i} = (-R(x))^{i-a}\binom{i}{a},$$

with $(a, i) \in \{0,1,\cdots,l\} \times \{m, m+1,\cdots,l\}$.

By the weighted degree condition of $Q_a(x)$, the system Equation (8) is equivalent to

$$\begin{pmatrix} Q_0(x) \\ x^{k-1}Q_1(x) \\ x^{2(k-1)}Q_2(x) \\ \vdots \\ x^{l(k-1)}Q_l(x) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & x^{k-1} & 0 & \cdots & 0 \\ 0 & 0 & x^{2(k-1)} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & x^{l(k-1)} \end{pmatrix} \mathcal{A} \begin{pmatrix} B_0(x) \\ B_1(x) \\ B_2(x) \\ \vdots \\ B_l(x) \end{pmatrix}.$$

Define

$$\mathcal{B} := \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & x^{k-1} & 0 & \cdots & 0 \\ 0 & 0 & x^{2(k-1)} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & x^{l(k-1)} \end{pmatrix} \mathcal{A}. \qquad (9)$$

Basically, a solution to the system Equation (8) is any vector

$$(Q_0(x), x^{k-1}Q_1(x), x^{2(k-1)}Q_2(x), \cdots, x^{l(k-1)}Q_l(x))^{\mathrm{T}}$$

in the $\mathbb{F}_q[x]$-span of the columns of $\mathcal{B}$ and have maximum degree less than $m(n-\tau)$. Using the concept of degree of a vector as defined in (Beelen and Brander, 2010) - the maximum degree of a vector is the degree of its entry with the highest power, and the maximum degree of a collection of vectors is the sum of the maximum degrees of each vector - the following analysis is made.

For $0 \le i < m$, $i \le j \le l$, let $S^{(i,j)}(x)$ be the syndrome polynomial given by Outcome 12. In other words, there is some polynomial $E^{(i,j)}(x)$ with $R^{j-i}(x) = E^{(i,j)}(x)G(x)^{m-i} + S^{(i,j)}(x)$ such that deg $S^{(i,j)}(x) <$ deg $G(x)^{m-i} = (m-i)n$. Also, define a matrix $\mathcal{U}$ such that for $(i, j) \in \{0,1,\cdots,l\} \times \{0,1,\cdots,l\}$

$$[\mathcal{U}]_{i,j} = \begin{cases} 1 & \text{if } i = j \text{ and } j < m, \\ 0 & \text{if } i \ne j \text{ and } j < m, \\ \binom{j}{i} E^{(i,j)}(x) & \text{if } i < m \text{ and } j \ge m, \\ \binom{j}{i} R(x)^{j-i} & \text{if } i \ge m \text{ and } j \ge m. \end{cases}$$

(10)

The matrix $\mathcal{U}$ is a good matrix for change of basis since it is upper triangular with determinant 1. Thus, the module spanned by the columns of $\mathcal{B}$ and $\mathcal{BU}$ are the same. Therefore, if a vector spans the columns of $\mathcal{B}$ also spans the columns of $\mathcal{BU}$ but with a lower maximum degree, a reduced complexity is acheived. Below a result from (Beelen and Brander, 2010) that proves that a basis transformation resulting in an improved complexity is attained by their method of solving the GS interpolation step is given.

Theorem 7: *Let $\mathcal{B}$ be as in Equation (9), $\mathcal{U}$ be as in Equation (10). Then for $0 \leq a \leq l$ and $0 \leq i \leq l$,*

$$[\mathcal{BU}]_{a,i} = \begin{cases} x^{a(k-1)}G(x)^{m-i}(-R(x))^{i-a}\binom{i}{a} & \text{if } a \leq i \text{ and } 0 \leq i < m, \\ -x^{a(k-1)}\sum_{h=a}^{m-1}(-R(x))^{h-a}\binom{h}{a}\binom{i}{h}S^{(h,i)}(x) & \text{if } a < i \text{ and } m \leq i \leq l, \\ x^{a(k-1)} & \text{if } a = i \text{ and } m \leq i \leq l, \\ 0 & \text{if } a > i. \end{cases}$$

Moreover, the maximum degree of $\mathcal{BU}$ is less than $(l+1)mn$.

Proof. See the proof on the the original paper (Beelen and Brander, 2010). Q.E.D.

The method of writing the GS interpolation problem in terms of modules over a univariate polynomial ring resulted in a complexity gain. (Beelen and Brander, 2010) reformulated and showed how to solve the problem, and presented an algorithm for improved complexity. Their algorithm can complete the interpolation step of the GS problem in $O(ml^4 n\log^2 n\text{loglog}n)$ time. Note that the basis transformation results in a gain only if $l > m$ (it could be assumed that $m/l \approx \kappa$).

Theorem 8: *Applying both polynomial reconstruction and basis transformation to solve Theorem 1 using the GS algorithm attains no significant complexity gain. Moreover, the process completes the interpolation step of the GS algorithm in $O(ml^4 n^{r+1} log^2 nloglogn)$ time.*

Proof. There are n^r choices, where r is the number of reductions, for a sufficient reduction and the transformation completes the interpolation step in $O(ml^4 n\log^2 n\text{loglog}n)$ time. It follows that a coupled decoding process will complete the interpolation step of the GS algorithm in $O(ml^4 n^{r+1}\log^2 n\text{loglog}n)$ time .

CONCLUSION

The polynomial reconstruction technique ensures an improved complexity on the decoding radius. Codes with large minimum distance are desired - this unequivocally points to the fact that a large decoding radius is preferred for better decoding. The gain in radius is at the expense of other complexities like the decoding time and the number of polynomials listed by the interpolation step. It is shown in Theorem 5 that the list size and the time taken to enumerate the polynomials make no improvement to their counterparts in the GS algorithm. In the case of basis transformation, the GS problem is reformulated in terms of modules over a univariate polynomial ring and the basis of the modules are replaced accordingly for improved complexity (decoding time) in the interpolation step. This is a nice technique in the sense that its gain is not at the expense of other complexities like the decoding radius and list size. For the polynomial reconstruction and the basis transformation coupled to decode Reed-Solomon codes using the GS algorithm, no newer improvement in the complexities is ensured. In fact, attaining a good reduction, the coupled system will complete the interpolation step in $O(ml^4 n^{r+1}\log^2 n\text{loglog}n)$ time (greater than $O(m^6 n^3)$ time, the complexity for the interpolation step of the GS algorithm), where r is the number of reductions as in Theorem 5.

ACKNOWLEDGEMENT: The author will like to thank Pal Hegedus of Central European University for his guidance and support in scrutinizing this work and also acknowledges the University of The Gambia for the per diem and conference fee which lead to this paper.

REFERENCES

Beelen, P., & Brander, K. (2010). Key equations for list decoding of Reed–Solomon codes and how to solve them. *Journal of Symbolic Computation*, *45*(7), 773–786. https://doi.org/10.1016/j.jsc.2010.03.010

Berlekamp, E. R. (2015). *Algebraic Coding Theory* (Revised). Retrieved from http://gen.lib.rus.ec/book/index.php?m d5=91c7be245968be68412977b2d67c9832

Cassuto, Y., Bruck, J., & McEliece, R. J. (2013). On the Average Complexity of Reed–Solomon List Decoders. *IEEE Transactions on Information Theory*, *59*(4), 2336–2351. https://doi.org/10.1109/TIT.2012.2235522

Guruswami, V., & Sudan, M. (1999). Improved decoding of Reed-Solomon and algebraic-geometry codes. *IEEE Transactions on Information Theory*, *45*(6), 1757–1767. https://doi.org/10.1109/18.782097

McEliece, R. J. (2003). *The Guruswam-sudan Decoding Algorithm for Reed-solomon Codes*.

Muralidhara, V. N., & Sen, S. (2009). Improvements on the Johnson bound for Reed–Solomon codes. *Discrete Applied Mathematics*, *157*(4), 812–818. https://doi.org/10.1016/j.dam.2008.06.014

Peterson, W. (1960). Encoding and error-correction procedures for the Bose-Chaudhuri codes. *IRE Transactions on Information Theory*, *6*(4), 459–470. https://doi.org/10.1109/TIT.1960.1057586

Roth, R. (2006). *Introduction to Coding Theory*. Cambridge University Press.

Sudan, M. (1997). Decoding of Reed Solomon Codes beyond the Error-Correction Bound. *Journal of Complexity*, *13*(1), 180–193. https://doi.org/10.1006/jcom.1997.0439

Zeh, A., Gentner, C., & Augot, D. (2011). An Interpolation Procedure for List Decoding Reed–Solomon Codes Based on Generalized Key Equations. *IEEE Transactions on Information Theory* *57*(9), 5946–5959.

Cite as: **Amadou Keita (2019). Improved Complexity In Decoding Reed-Solomon Codes. In: Asuquo, Francis E. (Editor) on *Harnessing African Potentials for Sustainable Development, Calabar, Nigeria, September 2019;* UNICAL Press & GIS Publishers. *U6CAU Proceedings (Maiden Edition) Volume 1, Number 1, 58 - 66.***