A Hybrid Gross List Decoding and Chase-Like Algorithm of Reed-Solomon Codes

Wei Jin, HongJun Xu and Fambirai Takawira, *Member, IEEE*

Abstract—**This paper presents a hybrid Gross list decoding and Chase-like algorithm for Reed-Solomon (RS) codes. A tree scheme for Gross list decoding algorithm, which is used to decrease the complexity of the hybrid algorithm, is also proposed in this paper. Simulation results show that the performance of the hybrid algorithm for (15, 7) RS code is comparable with the hybrid algorithm proposed in [1], which combines KV list algorithm and Chase algorithm, but the complexity of the proposed hybrid algorithm is much less than the hybrid algorithm in [1] .**

Index Terms—**Gross List decoding algorithm, Chase algorithm, Tree scheme, Reed-Solomon codes.**

I. Introduction

Based on the extensive application of Reed-Solomon (RS) codes, how to improve the decoding performance of RS codes is always to be an attention-getting theme. Many researchers try to find the way to approach Maximum likelihood decoding (MLD) of RS codes through the soft information drawn from received sequence. Their contributions are proposed in [2-4]. However, Applying Chase algorithm grounded on bit reliability is still an efficient technique to approach the performance of MLD.

To apply the Chase algorithm to RS codes, we need an algebraic decoder. The Guruswami-Sudan (GS) list decoding algorithm that was discovered by Madhu Sudan [5], and developed by Guruswami and Sudan two years later [6], is one of the enhanced algebraic decoding algorithms for RS codes. The number of errors that can be corrected by GS list algorithm increases to $t_{GS} = n-1-\lfloor\sqrt{(k-1)n}\rfloor$ which is more than the classical $t=(d_{\min}-1)/2$ errors. With the increase of multiplicity that is one of the determinants of the complexity of GS decoding, the error-correction capability of GS list algorithm is also increased, and we denote it as t_m . Koetter-Vardy (KV) list decoding algorithm, proposed in [7], is a practical implementation for the GS algorithm. Gross list algorithm proposed in [8] is defined as a simplified KV list Algorithm. We use Gross list algorithm in this paper.

Although more performance is achieved after using Chase algorithm, it is unacceptable and not feasible in practice that the complexity unfavorably grows exponentially. A tree scheme is employed to reduce the complexity through cutting down the iterations of the interpolation step that is the most time-consuming component of list algorithm. The internal relationship between the candidates of received set created by Chase algorithm is the primary element for the inspiration of tree scheme.

The paper is organized as follows. Section II introduces KV soft-decision front end. Section III introduces Gross list decoding algorithm. Section IV expatiates how to combine Gross list decoding algorithm and Chase algorithm. A tree scheme is also given in this section. Simulation results are given in Section V. Section VI draws a conclusion for the paper.

This work was partially supported by the University competitive research grant, Alcatel Altech Telecoms, THRIP and Telkom SA as part of the Centre of Excellence Programme

II. Koetter-Vardy Soft-Decision Front End

Guruswami and Sudan hinted at a possible soft-decision extension to their algorithm by allowing each point on the interpolated curve to have its own multiplicity. Koetter and Vardy (KV) proposed a method to perform soft-decision decoding by assigning unequal multiplicities to different points according to their relative reliabilities. Their algorithm generates the multiplicity matrix from the reliability matrix $\prod$. A lower complexity algorithm for implementing KV front-end was proposed in [9]. But we use KV front end from [7] which is shown on the follows.

KV Algorithm for calculating Multiplicity Matrix from the reliability matrix $\prod$ subject to complexity constraint s

---

Choose a desired value $s=\sum_{i=0}^{q-1}\sum_{j=0}^{n-1} m_{i,j}$;

$\Pi^* \leftarrow \Pi$; $M \leftarrow 0$;

While $s>0$ do

Find the position (i, j) of the largest entry $\pi^*_{i,j}$ in Π^* ;

$$\pi^*_{i,j} \leftarrow \frac{\pi_{i,j}}{m_{i,j}+2}\text{ ; } m_{i,j} \leftarrow m_{i,j}+1\text{ ; } s \leftarrow s+1\text{ ;}$$

End while

Using this algorithm, we obtain the support set, received set and multiplicity set. The candidates for the codeword polynomial are obtained through interpolation and factorization steps.

III. GROSS ALGORITHM

Gross list decoding algorithm is given as follows.

We want to transmit a message $\left(f_0, f_1, \cdots, f_{k-1}\right)$ whose all elements come from the $GF\left(q\right)$.The RS $\left(n,k\right)$ codewords can be generated by a polynomial evaluation, that is:

$$c=\left(f\left(\alpha^0\right), f\left(\alpha^1\right), \cdots, f\left(\alpha^{n-1}\right)\right) \tag{1}$$

where $f\left(x\right)=f_0+f_1x+\cdots+f_{k-1}x^{k-1}$ and α is the primitive element of $GF\left(q\right)$. An alternate way is a systematic encoding where all the *k* input symbols explicitly appear in the encoded codeword. For systematic encoding where the information appears in fixed positions, we use $g\left(x\right)=\left(x-\alpha\right)\left(x-\alpha^2\right)\cdots\left(x-\alpha^{n-k}\right)$ to be the generator polynomial, and let $a\left(x\right)=f\left(x\right)x^{n-k}$ mod $g\left(x\right)$, then $c\left(x\right)=f\left(x\right)x^{n-k}+a\left(x\right)$, and the coefficients of $c\left(x\right)$ are the codeword *c*. For systematic encoding where the information appears in arbitrary positions, an efficient approach of implementation is to apply an erasures-only RS decoder which has been proposed in [10].

After transmitting codeword *c* over an AWGN channel, the hard decision is obtained as $r=c+e$, where *e* is the error sequence.

A. Reencoding scheme

The idea of reencoding is to reduce the number of received symbols based on their reliability. The first step is to partition the hard decision *r* into two sets, unreliable set *(U)* that includes $n-k$ unreliable symbols and reliable set *(R)* that includes *k* reliable symbols. Then apply the arbitrary-position systematic encoding to the *R* set to get a reencoded codeword, ψ. Taking the difference between *r* and ψ we get:

$$r^{'}=r-\psi=\left(c+e\right)-\psi=\left(c-\psi\right)+e \tag{2}$$

which is a codeword that has the same error pattern with *r*. However, the new codeword $r^{'}$ only includes at most $n-k$ non-zero symbols due to the application of systematic encoding, and the *k* reliable symbols in $r^{'}$ are zeros. These *k* reliable symbols with zero value contribute an interpolation polynomial $v\left(x\right)^m$, where *m* is the multiplicities of these symbols and $v\left(x\right)$ is a simple univariate interpolation polynomial as follows:

$$v\left(x\right)=\prod_{i\in R}\left(x-\alpha^i\right) \tag{3}$$

because these symbols correspond to *k* interpolation points with a zero *y*-component. In [8], Gross sets the multiplicities of these *k* reliable symbols with the maximum multiplicity.

B. The simplified interpolation step

To implement the simplified interpolation, consider the original set of polynomials $G=\left\{1, y, \cdots, y^{d_y}\right\}$ where d_y is the weighted degree of *y*. Based on the univariate polynomial created by the symbols in the set R, the starting polynomial set for simplified interpolation is :

$$\begin{aligned} G^{'} &= \left\{v\left(x\right)^m, v\left(x\right)^{m-1}y, \cdots, v\left(x\right)^{m-d_y}y^{d_y}\right\} \\ &= v\left(x\right)^m\left\{1, \frac{y}{v\left(x\right)}, \left(\frac{y}{v\left(x\right)}\right)^2, \cdots, \left(\frac{y}{v\left(x\right)}\right)^{d_y}\right\}. \end{aligned} \tag{4}$$

After changing variables $\tilde{y}=y/v\left(x\right)$, we get a new starting set $\tilde{G}=\left\{1, \tilde{y}, \cdots, \tilde{y}^{d_y}\right\}$ and the weighted degree of the new variable $\tilde{y}$ is:

$$\begin{aligned} \deg^{(1,k-1)}\left(\tilde{y}\right) &= \deg^{(1,k-1)}\left(y\right)-\deg^{(1,k-1)}\left(v\left(x\right)\right) \\ &= \left(k-1\right)-k \\ &= -1. \end{aligned} \tag{5}$$

Correspondingly, the new *y*-coordinates of the interpolation points are acquired as follows:

$$\tilde{y}_i=\frac{y_i^{'}}{v\left(x_i\right)} \tag{6}$$

where $y_i^{'}$ are the *y*- components in $r^{'}$. It is concluded that the simplified interpolation step starts from the set $\tilde{G}$ and applies the KV interpolation step to the $n-k$ unreliable symbols $\left(x, \tilde{y}\right)$ with respect to the $\left(1,-1\right)-$ weighted degree. After the simplified interpolation step the reduced interpolation polynomial is as follows:

$$\tilde{P}\left(x, \tilde{y}\right)=\sum_{j=0}^{d_y} w_j\left(x\right)\tilde{y}^j \ . \tag{7}$$

C. The simplified factorization step

The simplified factorization step is realized by applying Roth-Ruckenstein algorithm proposed in [11] directly to the reduced polynomial $\tilde{P}\left(x, \tilde{y}\right)$. After the factorization step, a sequence $\left\{s_0, s_1, \cdots, s_{l-1}\right\}$ which are the coefficients of

$$s\left(x\right)=\frac{f^{'}\left(x\right)}{v\left(x\right)} \tag{8}$$

is obtained, where $f^{'}\left(x\right)$ is the generator polynomial corresponding to $c^{'}=c-\psi$. From $r^{'}=c^{'}+e$ and $r_i^{'}=0\left(i\in R\right)$ we get:

$$f^{'}\left(\alpha^i\right)=e_i\left(i\in R\right). \tag{9}$$

If no error in position $i\in R$ then e_i =0 and $f^{'}\left(\alpha^i\right)$ =0. Therefore $\left(x-\alpha^i\right)$ is a root of $f^{'}\left(x\right)$, and for all non-error positions in *R* we get:

$$f^{'}\left(x\right)=\prod_{i\in R\ s.t.e_i=0}\left(x-\alpha^i\right)\Omega\left(x\right). \tag{10}$$

Therefore,

$$s\left(x\right)=\frac{f^{'}\left(x\right)}{v\left(x\right)}=\frac{\prod_{i\in R\ s.t.e_i=0}\left(x-\alpha^i\right)\Omega\left(x\right)}{\prod_{i\in R}\left(x-\alpha^i\right)}=\frac{\Omega\left(x\right)}{\prod_{i\in R\ s.t.e_i\neq 0}\left(x-\alpha^i\right)} \tag{11}$$

We denote the denominator as $\Lambda(x)$, which is an error-locating polynomial that can be efficiently reconstructed by the Berlekamp-Massey algorithm. The roots of $\Lambda(x)$ indicate the error positions in set *R*. Based on the fact that most errors are occurred in the $n-k$ unreliable positions, we only need to correct a few errors in *R*. Consequently fewer coefficients than the requirement of KV list algorithm are demanded. In Gross algorithm, only $l = 2\lceil (k/n)t \rceil$ coefficients are given after simplified factorization step.

To get error values we use L'Hôpital's rule to following equation:

$$f^{'}(x) = \frac{\Omega(x)v(x)}{\Lambda(x)} \quad (12)$$

and we get:

$$f^{'}(\alpha^i) = \frac{\Omega(\alpha^i)v^{(1)}(\alpha^i)}{\Lambda^{(1)}(\alpha^i)} \quad (13)$$

where $v^{(1)}(x)$ and $\Lambda^{(1)}(x)$ are the formal derivatives of $v(x)$ and $\Lambda(x)$. After putting error values in their respective positions, we get the error pattern for set *R*. The error pattern for the whole *n* positions is easily obtained by using the arbitrary-position systematic encoding and we denote it as $\hat{e}$. The finally estimated codeword can be found by adding $\hat{e}$ and ψ.

IV. The Hybrid Algorithm

The algorithm proposed in [1], which combines KV list algorithm and Chase algorithm, shows us that the performance is improved since more candidates of codeword are obtained. Although the application of Chase algorithm brings a large decoding gain, the complexity also unhappily increases exponentially because KV list algorithm has to produce interpolation polynomial after *C* iterations for each candidate of received set. Assume the received set has the simplex multiplicity as m_s, and then the number of constraints for the interpolation step is as follows if the *b* bits chosen by Chase-2 algorithm are distributed in different symbols:

$$C_{KV} = \frac{1}{2} \cdot 2^b \sum_{i=0}^{q-1}\sum_{j=0}^{n-1} m_{i,j}\left(m_{i,j}+1\right). \quad (14)$$

In this section we discuss how to combine Gross list decoding algorithm with Chase algorithm in order to reduce the number of iterations in interpolation step. The reencoding scheme in Gross algorithm may reduce the total iterations.

The hybrid Gross list decoding and Chase-like algorithm includes the following steps:

(1) Implement KV soft-decision front end to obtain support set, received set and multiplicity set.

(2) Apply the reencoding scheme to divide the received set into two sets: reliable set *(U)* that we also denote it as set 1 and unreliable set *(U)*.

(3) Apply Chase algorithm to the unreliable set. Then the unreliable set can be further subdivided into two sets: the set that only includes all bits that are chosen by Chase algorithm, we denote it as set 2; and the set includes the remaining entries in unreliable set, we denote it as set 3.

(4) Start the simplified interpolation step to acquire the interpolation polynomials for set 3 and store them. Accomplish the remaining iterations through Tree scheme that will be proposed later.

(5) Simplified Factorization step.

(6) Compare the probabilities of all candidate codewords created by different received sets and output the most likely one.

In the proposed algorithm, the hard decision *r* is substituted by received set since the situation that a single support symbol matches multi received symbols, which we denote it as multi-points, exists in practices.

In the proposed hybrid algorithm, we apply chase algorithm after the reencoding step. It means that chase algorithm is applied to $n-k$ symbols not all *n* received symbols. Actually, the test error patterns created in this method are less reliable than firstly using chase algorithm to find the unreliable bits among the overall received bits. If we chose the posterior method to produce the test error patterns, then we have to build $v(x)$ and ψ for every candidate received set. In the simulation results shown in section V, we see the performance gap between two methods is unobvious. So we give the reencoding step the priority to tradeoff performance with decoding complexity.

The Tree scheme applied in the proposed algorithm is a technique through using the memory to minimize the number of iterations. Consider Chase-2 algorithm is employed and set the number of bits that are chosen by Chase-2 algorithm to b, thereby, the 2^b subset for set 2 are created. We denote a ideal situation that assume these bits are distributed in diverse received symbols and the multiplicities for the symbols in unreliable set are all equal to the minimum value $m_{\min}$. Then the total cost of decoding is as follows if we simply joint the Chase-2 algorithm and Gross algorithm under the condition that the received set does not contain the multi-points:

$$C_{Gross} = 2^{b-1}(n-k)m_{\min}\left(m_{\min}+1\right). \quad (15)$$

Although C_{Gross} is much smaller than C_{KV} since the *k* reliable symbols do not get involved into creation the interpolation polynomial for each candidate received set, the cost still needs to be reduced if the rate of code is small or *b* is large. The proposed hybrid algorithm reduces the iterations in interpolation step through the intrinsic relations among 2^b candidates of received set. The polynomials produced by symbols in set 3 accommodate the all 2^b candidates of received set because set 3 is the same among all 2^b sets. So the hybrid algorithm firstly processes the set 3 and stores the polynomials created by interpolation iteration. The remaining iterations are compressed by tree scheme, which is visualized as Fig.1.

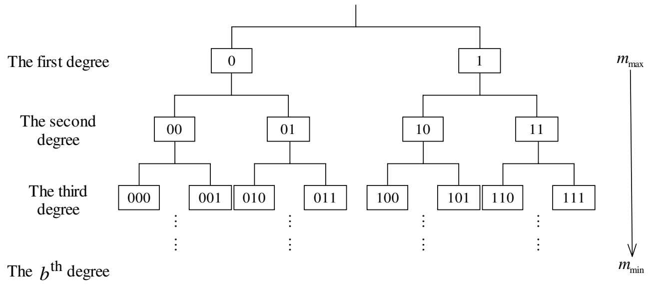

Fig.1. The Tree Scheme

In Fig.1, 0 means the original symbol in the received set, consequently the more reliable symbol obtained through changing one bit is expressed by 1. After the iterations for one symbol in set 2, we store d_y polynomials to prepare for the remaining candidates of the received set. Then the cost of interpolation step is reduced to:

$$C_{Gross+T} = \frac{1}{2}\left(\left(n-k-b\right)+\left(2^{b+1}-2\right)\right)m_{\min}\left(m_{\min}+1\right). \quad (16)$$

An example is necessary to explain the success of the proposed hybrid algorithm. Consider that an (255,127) RS code is adopted and 10-bit Chase 2 algorithm is realized by Gross list decoder. Under the ideal situation, $65536 m_{\min}\left(m_{\min}+1\right)$ iterations are required for simply jointing Chase-2 algorithm and Gross algorithm but only $1082 m_{\min}\left(m_{\min}+1\right)$ iterations are demanded in the proposed hybrid algorithm. The iterations for proposed hybrid algorithm take only 1.651% proportion for C_{Gross} in this case. It is obvious that the proposed hybrid algorithm largely reduce the iterations for interpolation step especial for code has low rate.

In practice, the multiplicities of symbols in set 2 are usually not identical, how to decide the order of symbols in tree scheme also need consideration. After checking the using frequency for symbols in degrees, the conclusion is drawn that the reduction for iterations under current symbols is reduced as the degree of tree scheme increases. The reduction for iterations is expressed as follows:

$$S_a = 2^{(b-1)} - 2^{(\deg-1)}. \quad (17)$$

Then it means that the iterations brought by the symbols in the first degree get the maximum reduction and no reduction for the symbols in lattermost degree. This opinion is tested through checking the iterations for every symbol in each degree. For each symbol in the first degree, it contributes single iteration for interpolation step for all 2^b candidates of received set, but $2^{(b-1)}$ iterations for each symbol in the last degree. In order to further reduce the iterations, the symbols in set 2 are placed into tree scheme in conformity to the regulation that the symbol with larger multiplicity is arranged into more anterior degree.

The costs for different hybrid algorithm compose for list decoding algorithm and *b*-bit chase 2 algorithm are shown in following table under the ideal situation.

TABLE I
COMPARISON FOR COSTS

Algorithm	Cost
The algorithm proposed in [1]	$C_{KV} = \frac{1}{2}\cdot 2^b \sum_{i=0}^{q-1}\sum_{j=0}^{n-1} m_{i,j}\left(m_{i,j}+1\right)$
Gross +Chase	$C_{Gross} = 2^{b-1}\left(n-k\right)m_{\min}\left(m_{\min}+1\right)$
The algorithm proposed in[12]	$C_{Xia} = \frac{1}{2}\left(\left(n-k-b\right)+2^b\cdot b\right)m_{\min}\left(m_{\min}+1\right)$
The hybrid algorithm	$C_{Gross+T} = \frac{1}{2}\left(\left(n-k-b\right)+\left(2^{b+1}-2\right)\right)m_{\min}\left(m_{\min}+1\right)$

The hybrid algorithm proposed in [12], which is composed by Gross algorithm and Chase algorithm, is similar with the proposed algorithm but without the tree scheme.

In practice, the cost for the proposed hybrid algorithm is greater than the value in Table I because the multiplicities of symbols in set 2 do not outright equal to $m_{\min}$. The hybrid algorithm spares more iterations as the multiplicities of symbols in set 2 increase from $m_{\min}$ since the tree scheme arranges the symbols based on the multiplicity order that from maximum to minimum.

V. THE SIMULATION RESULTS

The main focus in this section is to compare the performances of the proposed hybrid algorithm and the hybrid algorithm proposed in [1]. The complexity comparison is accomplished through showing the numerical data to cooperate with performances. The diversity of two hybrid algorithm in performance and complexity is easily observed. An (15,7)RS code is involved in the simulation results. We denote the hybrid KV list decoding and chase-like algorithm proposed in [1] as hybrid algorithm 1 and the proposed hybrid algorithm using Gross decoder is denoted as hybrid algorithm 2. The tunable parameter *s* is chosen to 12 for both hybrid algorithms.

For (15,7) RS code, we choose chase-2 algorithm to enhance the performance. We select 4 unreliable bits and then correspondingly, we attain 2^4 candidate received sets. Simulation results in Fig. 2 show that the hybrid algorithm 1 is able to correct more than 6 errors at 6dB, which exceeds the t_{GS} by one error. The performances of hybrid algorithm 2 are also shown in Fig.2 and two methods discussed in section IV are accomplished. Fig.2 shows that the performance of Gross list algorithm is exactly same as KV list algorithm. As mentioned in section IV and proved by simulation results, it can be neglected that the performance gap between the hybrid algorithm 2 that firstly use chase algorithm, which is denoted as hybrid 2(chase) in Fig.2, and the hybrid algorithm 2 that firstly use reencoding step, which is denoted as hybrid 2(reencoding) in Fig.2. The decoding costs for (15,7)RS code under the situation that no multi-points in received set and $s=12$ are calculated in the table shown on the following page.

TABLE II

COMPARISON FOR COSTS FOR (15,7)RS CODE

Algorithm	4-bit	3-bi	2-bi	1-bi
The hybrid algorithm 1	$(3\times10+5)\times2^4 = 560$	280	140	70
Gross+ Chase	$(3\times3+5)\times2^4 = 224$	112	56	28
The algorithm proposed in [12]	$(3\times3+1)+4\times2^4 = 74$	35	20	15
The hybrid algorithm 2	$(3\times3+1)+2^5-2 = 40$	25	18	15

Although the performance of 4-bit hybrid algorithm 2 is the same as the 3-bit hybrid algorithm 1, 40 iterations are required by the former algorithm but 280 iterations by the latter. After observing the performance, a strange situation is that the performance of hybrid algorithm 2 even the hybrid algorithm 2(chase) is worse than the hybrid algorithm 1 despite the Gross algorithm has the completely uniform performance with KV algorithm. Then why the performances are different after applying the same chase algorithm to two algebraic decoders that have the same performance? Actually, the difference of performances between the hybrid algorithm 1 and 2 is caused by the simplified factorization step. As mentioned in section III, only $l = 2\lceil (k/n)t \rceil$ coefficients produced by simplified factorization step, it is a hint that the number of errors that occur in reliable set must equal to $\lceil (k/n)t \rceil$ or less, otherwise Gross list algorithm can not successfully decoding the received sequence even though the amount of errors that occur in the whole received set does not exceed the classical decoding bound $t = \frac{d_{\min}-1}{2}$. Since the probability that more than $\lceil (k/n)t \rceil$ errors occur in k reliable symbols is very low and imaginably decreases as the SNR increases, then it is acceptable that the performance of Gross algorithm is same as KV algorithm. However if we apply chase algorithm to Gross algorithm with large erasures, the decoding failure because more than $\lceil (k/n)t \rceil$ errors occur in k reliable symbols take the weighty proportion for the reduction of performance. As SNR increases, the symbols in set 2 with increasing reliabilities conduce the performance of hybrid algorithm 2 to approaching the hybrid algorithm 1. This situation is proved by simulation result. To amend this situation, the simplified factorization step needs to unceasingly produce more coefficients, but the complexity also undoubtedly increases.

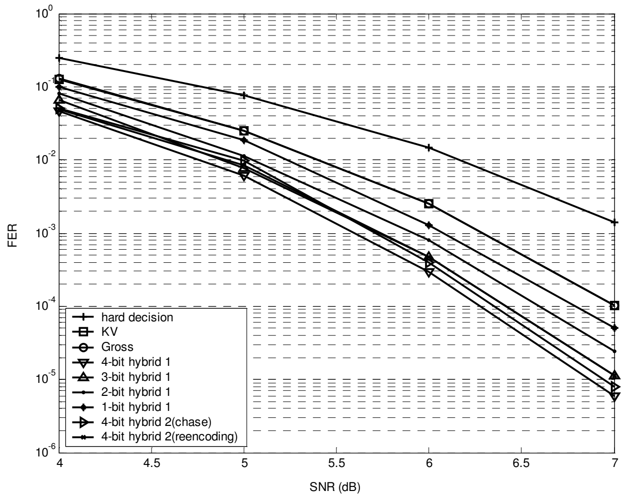

Fig.2. FER performance of (15,7)RS code

VI. CONCLUSION

A hybrid Gross list decoding and chase-like algorithm is proposed in this paper. The priority between chase algorithm and reencoding step is discussed. A tree scheme is applied to further reduce the number of iterations through utilizing the mutual relationship among 2^b candidates of received set. The costs for decoding for different hybrid algorithms are educed. From the comparisons of simulation results and the costs of decoding, our hybrid algorithm has the less complexity than the hybrid algorithm proposed in [1] and the hybrid algorithm proposed in [12] but with a slight loss in performance corresponding to the hybrid algorithm in [1].

REFERENCES

[1] Wei Jin, HongJun Xu and Fambirai Takawira, "A Hybrid List Decoding and Chase-Like Algorithm of Reed-Solomon Codes" in Proceedings of the 4th International Symposium on Information and Communication Technologies, *pp.* 87–92, Jan 3-6 2005.

[2] Vishakan Ponnampalam and Branka Vucetic, "Soft Decision Decoding of Reed-Solomon Codes". *IEEE Trans.Commu.vol.50, no.11 (Nov 2002), pp. 1758~1768.*

[3] Marc Fossorier and Antoine Valembois, "Reliability-Based Decoding of Reed-Solomon Codes Using Their Binary Image". *IEEE.Commu Letter.vol.8, no.7 (July 2004), pp. 452~454.*

[4] Ta-Hsiang Hu and Shu Lin, "An Efficient Hybrid Decoding Algorithm for Reed-Solomon Codes Based on Bit Reliability," *IEEE Trans.Commu.vol.51, no.7 (July 2003), pp. 1073~1081.*

[5] M.sudan, "Decoding of Reed Solomon codes beyond the error-correcting bound", J. Complexity, vol. 13(1997), *pp*.180~193.

[6] V.Guruswami and M. Sudan, "Improved decoding of Reed-Solomon codes and algebraic geometry codes*", IEEE Trans. Inform. Theory vol. 45 no. 6 (Sept. 1999), pp. 1757~1767.*

[7]R. Koetter and A. Vardy, "Algebraic soft-decision decoding of Reed-Solomon codes," *Proceedings of the IEEE International Symposium on Information* Theory, p. 61, 2000.

[8] Warren J. Gross, Frank R. Kschischang, Ralf Koetter, and P. Glenn Gulak, "Towards a VLSI Architecture for Interpolation-Based Soft-Decision Reed-Solomon Decoders", *Journal of VLSI Signal Processing Special Issue on SIPS 2002*.

[9] W. J. Gross, F. R. Kschischang, R. Koetter, and P. Gulak, Simulation results for algebraic soft-decision decoding of Reed-Solomon codes, in Proceedings of the 21 st Biennial Symposium on Communications, pp. 356 360, June 2-5 2002.

[10] S. B. Wicker, Error Control Systems for Digital Communication and Storage. Upper Saddle River, New Jersey: Prentice Hall, 1995.

[11] R. M. Roth and G. Ruckenstein, "Efficient decoding of Reed-Solomon codes beyond half the minimum distance," *IEEE Transactions on Information Theory*, *vol. 46, pp. 246–257, January 2000.*

[12] Haitao Xia and J. R. Cruz, "Application of soft-decision Reed-Solomon decoding on magnetic recording channels," *IEEE Trans. Magn., vol. 40, pp. 3419-3430, Sept. 2004.*

Principal Author: Mr. Wei Jin is currently doing his master degree in the Radio Access Technology Research Centre at the University of KwaZulu-Natal, Howard College Campus in Durban 4001, SA, (e-mail: Jin@ukzn.ac.za).

Co-author: Dr. Hong-Jun Xu is with in the School of Electrical, Electronic and Computer Engineering at the University of KwaZulu-Natal, Howard College Campus. He received the BSc degree in 1984 from the University of Guilin Technology and the MSc degree from the Institute of Telecontrol and Telemeasure in Shi Jia Zhuang, 1989, and the PhD degree from the Beijing University of Aerinautics and Astronautics in Beijing, 1995. His research interests are in the area of digital and wireless communications and digital systems.

Co-author: Professor Fambirai Takawira is full time Professor in the faculty of Electronic Engineering, and head of the School of Electrical and Electronic Engineering at the University of KwaZulu-Natal. He is also the director of the Centre for Radio Access and Rural Technologies at the same university. He holds a BScEng with first class honours in Electrical Engineering from Manchester University and a PhD degree from Cambridge, United Kingdom. His research interests are in the general areas of adaptive signal processing, digital and wireless communications and data networks.