AN ADAPTIVE HYBRID LIST DECODING AND CHASE-LIKE ALGORITHM FOR REED-SOLOMON CODES

W. Jin, H. Xu and F. Takawira

School of Electrical, Electronic and Computer Engineering, University of KwaZulu-Natal, Howard College Campus, King George V Avenue, Durban 4041, South Africa

Abstract: This paper presents an adaptive hybrid Chase-like algorithm for Reed-Solomon codes, which is based on the list decoding algorithm. The adaptive hybrid algorithm is based on the reliability threshold to exclude the more reliable bits from being processed by the list decoding algorithm and reduce the complexity of the hybrid algorithm. Simulation results show that the decoding complexities of the adaptive hybrid algorithm for both (15.7) and (31.21) Reed-Solomon codes are almost the same as those of the list decoding algorithm (without Chase algorithm) at high signal-to-noise ratios, but there is a significant improvement in FER performance.

Key words: The list decoding algorithm, Bit reliability, Adaptive scheme, Hybrid algorithm.

1. INTRODUCTION

How to approach the performance of the Maximum likelihood decoding (MLD) with less complexity is a subject which has been researched extensively, especially for Reed-Solomon (RS) codes which are powerful error-correcting codes in digital communications and digital data-storage systems. Applying the bit reliability obtained from the channel to the conventional decoding algorithm is always an efficient technique to achieve the performance of the MLD, although the exponential increase of complexity is always concomitant. In [1]-[2], the authors also use the bit reliability to improve the performance of Bose-Chaudhuri-Hocquenqhem (BCH) codes and RS codes, respectively. It is undoubted that improved performance can be achieved if we apply the bit reliability to an enhanced algebraic decoding algorithm that is more powerful than the conventional algebraic decoding algorithms.

The Guruswami-Sudan (GS) list decoding algorithm [3] that was discovered by Madhu Sudan in 1997 and developed by Guruswami and Sudan two years later [4] is one of the enhanced algebraic decoding algorithms for RS codes. In the GS list decoding algorithm, the number of errors that can be corrected increases to $t_{GS} = n-1-\left\lfloor \sqrt{(k-1)n} \right\rfloor$ for (n,k) RS codes, where $\lfloor x \rfloor$ is the integer of x. It is easy to show that the GS list decoding algorithm is able to correct more errors than the conventional algebraic decoding algorithms. The fundamental idea of the GS algorithm is to take advantage of an interpolation step to get an interpolation polynomial which is produced by the support symbols, the received symbols and their corresponding multiplicities. The GS algorithm then implements a factorisation step to find the roots of the interpolation polynomial. After comparing the reliability of these codewords, which are obtained from the output of factorisation, the GS algorithm outputs the most likely one. The support set, the received set and the multiplicity set are created by the Koetter-Vardy (KV) algorithm [5] that is a practical implementation of the GS algorithm.

To further improve the performance of the GS list decoding algorithm, [6] has proposed a hybrid list decoding and Chase-like algorithm. Simulation results in [6] show that the performance of the hybrid algorithm for the (7.5) RS code can approach that of the MLD, and the performance of the (15.7) RS code can correct one more symbol error than the GS list decoding algorithm**.** The complexity of the hybrid algorithm in [6] depends on the number of bits which are used in the Chase-like algorithm, but the complexity is exponential with the number of bits. Actually, as signal-to-noise ratio (SNR) increases the received bits are more reliable, and it is not necessary to apply the Chase-like algorithm in the GS list decoding algorithm. To further reduce the complexity at high SNRs, we propose an adaptive hybrid algorithm which is based on the GS list decoding and the adaptive Chase-like algorithm in this paper. The adaptive hybrid algorithm is based on the reliability threshold to exclude the more reliable bits from being processed by the GS list decoding algorithm.

This paper is organised as follows. Section 2 introduces the KV soft-decision front end along with the corresponding algorithm. Section 3 gives a brief description of the adaptive Chase-Generalised Minimum Distance (Chase-GMD) algorithm. Section 4 explains how the list decoding algorithm and the Chase algorithms can be combined with further incorporation of the adaptive idea. Simulation results are given in Section 5. Section 6 draws conclusions for this paper.

2. THE KOETTER-VARDY SOFT-DECISION FRONT END

Guruswami and Sudan hinted at a possibility of a 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. An algorithm that generates the multiplicity matrix from the reliability matrix Π was presented in [5]. A lower complexity algorithm for implementing the KV front-end was proposed in [7], but we still use the KV algorithm from [5] which is shown as follows.

The KV Algorithm for calculating Multiplicity Matrix M from the reliability matrix Π subject to complexity constraint s.

Definition: $m_{i,j}$ is an entry at the position (i, j) in multiplicity matrix M.

Algorithm:

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^* ;

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

End while

Using this algorithm, we obtain the support set, the received set and the multiplicity set. The candidates of the codeword polynomial are obtained through an interpolation step and a factorisation step.

3. AN ADAPTIVE CHASE-GMD ALGORITHM

Mahran and Benasissa proposed an adaptive Chase-GMD algorithm for linear block codes in [8]. In the adaptive Chase-GMD Algorithm, an l-bit quantizer is used to classify the received bits by their reliability. A brief overview of the adaptive Chase-GMD algorithm is given as follows.

As errors are more likely to occur in the first η least reliable positions of the received bits R, the Chase-GMD algorithm is firstly considered to be applied in those positions. A reliability threshold function of confidence level, r, is used in the adaptive Chase Algorithm. The higher the confidence level the higher the possibility of selecting more chase-like erasures. The threshold function T is given by:

$$T = S \times r \times \sqrt{\frac{0.5}{\left(\frac{E_b}{N_o}\right)\cdot R_c}} \qquad (1)$$

where R_c is the rate of the RS code. E_b / N_0 is the bit signal-to-noise ratio. S given in (2) is a scalar constant that depends on the number of quantisation levels.

$$\frac{0.45}{2^{l-3}} \le S \le \frac{0.7}{2^{l-3}} \qquad (2)$$

The threshold can be used to decide which bit should be processed by the Chase algorithm. Let the reliability of received sequence be α. The bits will be used in the adaptive Chase algorithm only if their reliabilities satisfy the following condition:

$$-T \le \alpha_j \le +T \qquad j = 1, 2, \ldots, \left\lceil \frac{d_{\min}}{2} \right\rceil \qquad (3)$$

If a bit does not satisfy the above condition then it can be ignored by the Chase algorithm even if it is the most unreliable bit in the received bits.

4. THE ADAPTIVE HYBRID ALGORITHM

The application of the Chase algorithm to the KV soft-decision front end based on the bit reliability can improve the performance of the list decoding algorithm, with an adaptive scheme reducing complexity.

Before the presentation of the adaptive hybrid algorithm, there are some definitions which should be made clear. We can obtain the support set, the received set and the multiplicity set through the KV front end. We use 'multi-points' to define the received symbols whose support symbols are the same, 'low-multiplicity-points' to define the received symbols whose multiplicities are less than the maximum multiplicity except for multi-points, and 'high-multiplicity-points' to define the received symbols whose multiplicities are equal to the maximum multiplicity. We refer to the high-multiplicity-points as reliable points, and other received symbols as unreliable points.

Now, the adaptive hybrid list decoding and the Chase-like algorithm contain the following steps:

i. Implement the KV soft-decision front end to obtain the support set, the received set and the multiplicity set.
ii. Use Equation (1) to obtain the threshold value with appropriate scale constant S and confidence level r.
iii. Calculate the number of multi-points in the output of the KV soft-decision front end. If more than one received symbol is found to have the same support symbols, the number of multi-points increases by one. We denote it as N_{multi}.

iv. Calculate the number of low-multiplicity-points in the output of the KV soft-decision front end. We denote it as N_{low}.

v. If $N_{multi} + N_{low} \le t_{GS}$

use the chase-like algorithm for high-multiplicity-points. The threshold can be used to finally decide if those unreliable bits are picked up by the Chase algorithm or not.

Else

use the Chase-like algorithm for both low-multiplicity-points and high-multiplicity-points. The unreliable bits selected by the Chase algorithm must also satisfy the condition mentioned above.

t_{GS} is the number of errors that can be corrected for (n,k) RS codes.

vi. Output all the received sets, the corresponding support set and multiplicity set to interpolation step.

vii. Interpolation step proposed in [5].

viii. Factorisation step proposed in [5].

ix. Compare the probability of all candidates of the codeword created by different received sets and output the most likely one.

In the above proposed algorithm, we do not consider the multi-points because the list decoding algorithm has already taken them into account. In other words, the list decoding algorithm pays more attention to multi-points than other points. When the list algorithm fails, the errors coming from multi-points are not the significant source of failure for the list decoding.

In the above algorithm, we classify the output of the KV soft-decision front end into two different cases, $N_{multi} + N_{low} \le t_{GS}$ and $N_{multi} + N_{low} > t_{GS}$. We will discuss them separately.

If $N_{multi} + N_{low} \le t_{GS}$, it means that the number of unreliable points does not exceed the error-correcting ability. Even if all unreliable symbols are incorrect, the list decoding algorithm can still generate the right codeword polynomial. In this case, the errors coming from reliable symbols are the main reason for the failure of the list decoding algorithm, so we apply the Chase algorithm to reliable symbols in order to obtain more reliable received sets corresponding to the same multiplicity.

If $N_{multi} + N_{low} > t_{GS}$, it means that the number of unreliable symbols exceeds the error-correcting ability. If all these symbols are incorrect, the list decoding algorithm can not generate the right codeword polynomial. In this case, we have to concentrate on both low-multiplicity-points and high-multiplicity-points. Because the list decoding algorithm has already taken the multi-points into account, we do not take the multi-points into account. Before we apply the Chase algorithm to both kinds of points, we must make it clear which kind of points we should take into account first, low-multiplicity-points or high-multiplicity-points. We can extend the search scope into high-multiplicity-points by changing the unreliable bits. In order to improve the search scope, changing the unreliable bits in high-multiplicity-points is better than in low-multiplicity-points. This implies that we can obtain more candidate codeword polynomials if we choose high-multiplicity-points. It seems that we should change unreliable bits in high-multiplicity-points, but at high SNRs, we draw a different conclusion. As the SNR increases, the high-multiplicity-points (reliable points) become more and more 'reliable'. The probability that reliable points are received correctly is very large. The performance improvement is marginal even if we invert these bits which are in the reliable points. In this paper, we only take into account low-multiplicity-points first at high SNRs.

There are several threshold values that are shown in Figure 1 and Figure 2 for $R_c = 0.467$. The scalar constant in Figure 1 is 0.225, which is the minimum of a 4-bit quantizer. The scalar constant in Figure 2 is 0.35, which is the maximum of the same 4-bit quantizer.

It is obvious that we can change the confidence level to get different thresholds. As the confidence level increases, the number of bits that can be ignored by the Chase algorithm decreases. The confidence level can be adjusted to fit the list decoding algorithm. It is expected that the performance of the adaptive hybrid algorithm is comparable with the performance of the hybrid Chase-list algorithm in [6], but with lower complexity.

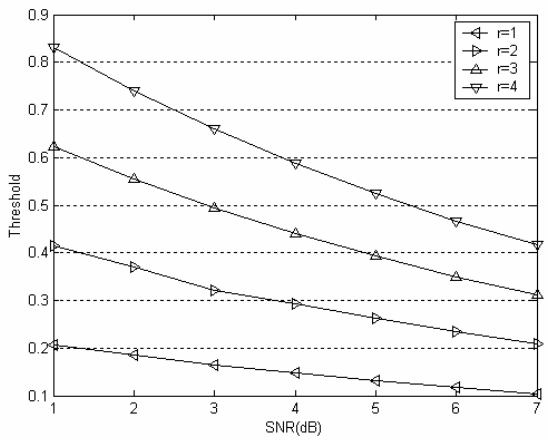

Figure1: The thresholds with S=0.225 for $R_c = 0.467$.

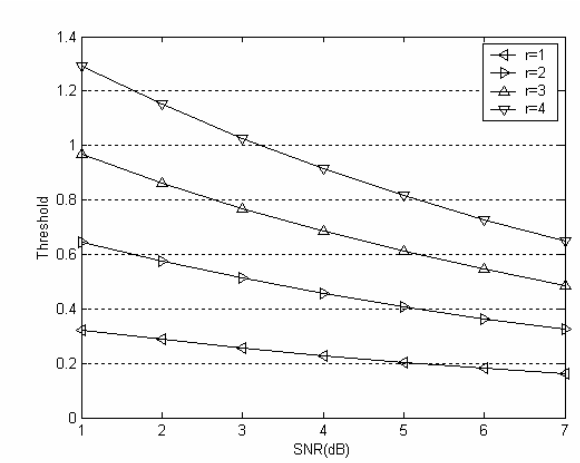

Figure 2: The thresholds with S=0.35 for $R_c = 0.467$.

5. SIMULATION RESULTS

All simulations are performed in an AWGN channel and BPSK transmission is assumed. For comparison purposes, we simulate the conventional decoding, the list decoding, the hybrid algorithm in [6] and the adaptive hybrid algorithm. We use the Chase-2 algorithm to improve the performance for the (15.7) RS code. We select 4 unreliable bits according to the least reliable positions based on the hybrid algorithm. A 4-bit quantizer with the confidence level 1 to 8 is used. In the simulation of the (15.7) RS code we choose S=0.26 and r=3, which are suitable for the list decoding algorithm with maximum multiplicity 2. The frame error rate (FER) performance is shown in Figure 3, and the corresponding complexity is shown in Figure 5. The simulation results of 2-bit hybrid algorithm, 3-bit hybrid algorithm and 4-bit hybrid algorithm are also shown in those figures for comparison. Based on the fact that one interpolation step and one factorisation step take almost 95% of total decoding time, we define a unit of the decoding complexity as the time taken by one interpolation step and one factorisation step in the list decoding algorithm. The hybrid algorithm can correct one more symbol error than the list decoding algorithm. Figure 5 also shows that the complexity of the adaptive hybrid algorithm decreases as the SNR increases. The complexity of the adaptive hybrid algorithm at 7dB is almost 2, which is the complexity with the 1 bit Chase-2 algorithm applied to the list decoding algorithm, but the gap between the FER performance of the adaptive hybrid algorithm and the hybrid's is negligible.

We still use the Chase-2 algorithm to improve the performance for the (31.21) RS code. A 4-bit quantizer is also used with S=0.225 and r=3. The FER performance is shown in Figure 4, and the corresponding complexity is shown in Figure 6. The simulation results of the 1-bit hybrid algorithm and the 2-bit hybrid algorithm are also shown in those figures for comparison. The simulation results in Figure 4 and Figure 6 show that the adaptive algorithm can reduce the complexity with small or marginal performance penalty. The complexity of the adaptive hybrid algorithm in Figure 6 can approach the list decoding algorithm without the Chase algorithm at high SNRs.

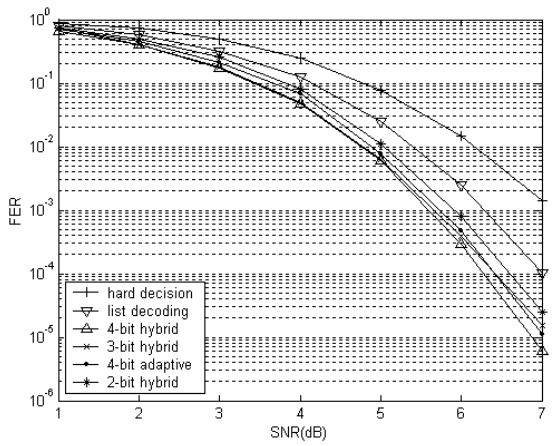

Figure 3: FER performance of the (15.7) RS code.

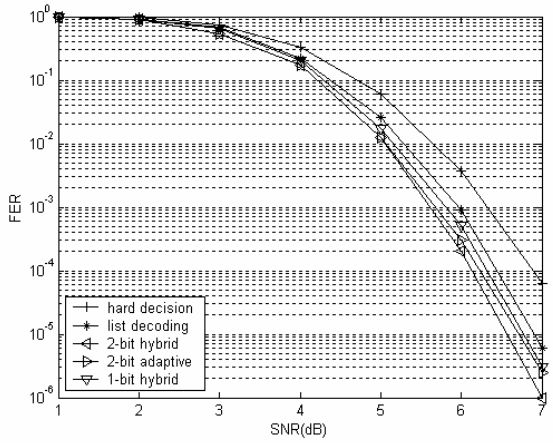

Figure 4: FER performance of the (31.21)RS code.

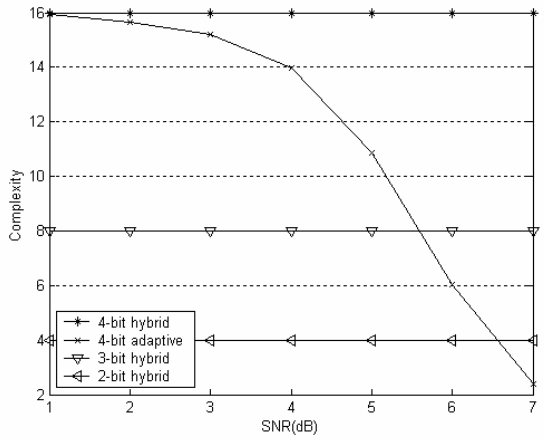

Figure 5: The complexity of the (15.7) RS code.

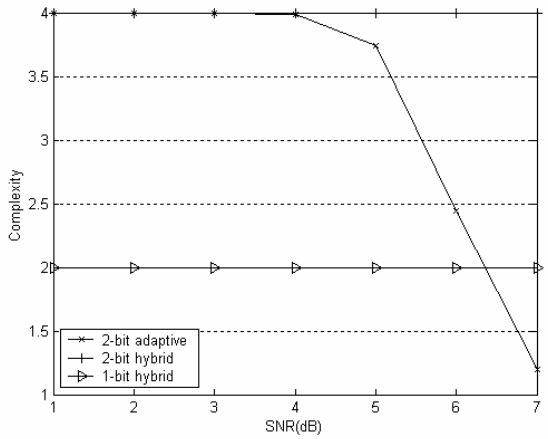

Figure 6: The complexity of the (31.21) RS code.

6. CONCLUSION

In this paper, an adaptive hybrid list decoding and Chase-like algorithm is presented. The adaptive hybrid algorithm is based on the reliability threshold to exclude the more reliable bits from being processed by the list decoding algorithm. In the above steps we obtain more received sets and accordingly obtain more candidate codeword polynomials. As the search scope is extended, the transmitted codeword is easily obtained. Simulation results show that the FER performance of the proposed adaptive hybrid algorithm for both (15.7) and (31.21) RS codes can be comparable with the performance of the hybrid algorithm in [6], but the complexity is much lower, especially at high SNRs.

7. REFERENCES

[1] M. P. C. Fossorier and S. Lin: "Complementary reliability-based decoding of binary block codes", *IEEE Transactions on Information Theory*, Vol.43, pp. 1667~1672.

[2] T. H. Hu and S. Lin: "An efficient hybrid decoding algorithm for Reed-Solomon codes based on bit reliability", *IEEE Transactions on Communications*, vol.51, no.7, pp. 1073~1081, July 2003.

[3] M. Sudan: "Decoding of Reed Solomon codes beyond the error-correcting bound", *J. Complexity*, Vol. 13, pp.180~193, 1997.

[4] V.Guruswami and M. Sudan: "Improved decoding of Reed-Solomon codes and algebraic geometry codes", *IEEE Transactions on Information Theory*, vol. 45 no. 6 , pp. 1757~1767, Sept. 1999.

[5] R. Koetter: "Fast generalized minimum-distance decoding of algebraic-geometry and Reed-Solomon codes", *IEEE Transactions on Information Theory*, vol. 42 no. 3, pp. 721–736, May 1996.

[6] W. Jin, H. Xu and F. Takawira: "A hybrid list decoding and Chase-like algorithm of Reed-Solomon codes", *Proceedings of the 4th International Symposium on Information and Communication Technologies*, pp. 87–92, Jan 3-6 2005.

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

[8] A. Mahran and M. Benaissa: "Adaptive combined Chase-GMD algorithms for block codes", *IEEE Communications Letter*, Vol.8, No.4, pp.235-237, April 2004.