REDUCED-COMPLEXITY IMPLEMENTATION OF ALGEBRAIC SOFT-DECISION DECODING OF REED-SOLOMON CODES

Haitao Xia and J. R. Cruz

School of Electrical and Computer Engineering
The University of Oklahoma
202 West Boyd, Room 219, Norman, OK 73019
email: {haitao_xia, jcruz}@ou.edu

ABSTRACT

A reduced complexity implementation of a soft Chase algorithm for algebraic soft-decision decoding of Reed-Solomon (RS) codes based on the recently proposed algorithm of Koetter and Vardy is presented. The reduction in complexity is obtained at the algorithm level by integrating the re-encoding and Chase algorithms and at the architecture level by considering a backup mode which sharply reduces the average computational complexity of the hybrid decoder.

1. INTRODUCTION

Reed-Solomon (RS) codes have been widely used in a variety of communication systems, such as wireless local area networks (LANs), deep-space communications, data storage systems, etc., because of their ability to correct burst errors. The rediscovery of low-density parity-check (LDPC) codes [1], which use channel soft information and iterative decoding to deliver impressive coding gains, has stimulated interest in considering them as replacements for RS codes. However, their higher decoding complexity and large performance degradation on bursty channels, is partially responsible for the continuing dominance of RS codes as the error-correcting codes (ECCs) of choice. Recently, Koetter and Vardy [2] proposed an algebraic soft-decision RS decoding algorithm, which brings the error correction capability of RS codes much beyond half the minimum distance. The availability of a long sought soft-decision decoding algorithm for RS codes has spurred an interest in revisiting the conventional ECC architectures of current communication systems with the objective of using this soft-decision algorithm in place of the traditional hard-decision RS decoders. The main obstacle to its practical implementation is its large computational complexity.

In this paper we attack the complexity reduction problem in two ways. First we discuss the implementation of a soft-decision RS decoder as the decoder of last resort and determine the necessary system modifications required to utilize the soft channel output information. Given the higher complexity of soft-decision RS decoding, this back up mode provides a good tradeoff between performance and complexity, since the soft-decision algorithm is only used when the hard-decision algorithm fails. This hybrid scheme exhibits a large coding gain without substantially increasing the average decoding complexity. Secondly, we combine the soft Chase algorithm proposed in [3] with a re-encoding algorithm [4] to improve decoding performance and reduce the decoding complexity of the soft-decision decoder, respectively.

The paper is organized as follows. In Section II, the concept of algebraic soft-decision RS decoding is briefly described. Computation complexity of the hybrid system and architectures using the combined soft Chase and re-encoding algorithms are given in Section III. Performance evaluation results are presented in Section IV.

2. SOFT-DECISION REED-SOLOMON DECODING

Let $GF(q)$ be a finite field with $q = n+1$ elements and denote the information sequence as $(f_0, f_1, \cdots, f_{k-1})$. The $RS(n,k)$ codewords can be generated systematically by multiplying the information polynomial $f(x) = f_0 + f_1 x^1 + \cdots + f_{k-1} x^{k-1}$ by the generator polynomial $g(x) = (x-\alpha)(x-\alpha^2)\cdots(x-\alpha^{n-k})$, where α is the primitive element of $GF(q)$; the coefficients of $c(x) = f(x)g(x)$ are the codeword $\mathbf{c}$. An alternate way of generating RS codes consists of evaluating $f(x)$ over the distinct elements of $GF(q)$. The RS codes generated by the two methods can be transformed from one to the other by simple multiplication by a nonzero matrix. Hereafter, we will always consider the systematic RS codeword $\mathbf{c}$, unless specified otherwise.

2.1. Soft-decision decoding algorithms

The soft-decision decoding algorithm proposed in [2] consists of four steps: multiplicity matrix calculation, soft interpolation, polynomial factorization and list decoding. Suppose a codeword $\mathbf{c}$ is sent and the hard-decision channel output is $\mathbf{y} = \mathbf{c} + \mathbf{e} = (y_0, y_1, \cdots, y_{n-1})$, with $\mathbf{e}$ being the error vector. In fact, the output of a noisy channel is "soft" information, which can be converted into a reliability matrix $\Pi[i, j]$, where each entry represents the probability of channel output $y_j = i$, $j = 0, \cdots, n-1$; $i = 0, \cdots, q-1$. Using this reliability matrix and a given parameter, the total multiplicity s, we can generate a multiplicity matrix $M[i, j]$. Each nonzero entry in $M[i, j]$ can be taken as a point $(x_t = i, y_t = j)$, whose value is m_t. The channel output information has been translated into a sequence of points $\{(x_t, y_t), m_t\}_{t=0}^{N-1}$, where N is not necessary to be equal to n. By finding a bivariate polynomial $Q_M(x, y)$ that passes through $\{x_t, y_t\}_{t=0}^{N-1}$ with variable multiplicity m_t [2] or constant multiplicity m [5], and identifying all its factors of the form $y - f(x)$ with $deg(f(x)) < k$, the coefficients of the polynomials $f(x)$ will provide the decoding answers. Using the same notation as in [2], it has been shown that if the score of the multiplicity matrix $S_M(c)$ and the $(1, k-1)$-weighted degree of $Q_M(x, y)$ satisfy:

$$S_M(c) > \delta = \deg_{1,k-1} Q_M(x, y), \qquad (1)$$

then the correct codeword can be found by list decoding. The sum of all multiplicities m_i for each point pair $\{x_t, y_t\}_{t=0}^{N-1}$ is the total multiplicity s, which strongly determines decoding performance and complexity.

Notice that the most computational complex step of the soft-decision decoding algorithm is the soft interpolation step, which has been the focus of most of the research on reducing decoding complexity. One such technique is the re-encoding (RE) scheme proposed in [4], which simplifies the polynomial interpolation by finding a modified polynomial $Q'_M(x, y)$ that passes through at least k modified point pairs $\{x_i, y'_i = 0\}$, which can be implemented with a lower complexity.

2.2. Chase-type algorithms

A soft version of the Chase algorithm using simple bit-flipping based on the Koetter-Vardy (KV) soft-decision algorithm has been recently proposed in [3]. By flipping the p least reliable bits in a received sequence and generating a set of test patterns, we can obtain a set of multiplicity matrices for the soft-interpolation step, expanding the decoding list, and leading to improved decoding performance. Since, among the 2^p multiplicity matrices, only a few columns in each matrix are different, we only need to store one sparse matrix, and the difference information to generate the remaining $2^p - 1$ matrices.

3. REDUCED-COMPLEXITY SYSTEM ARCHITECTURES

3.1. Backup mode and its complexity

Common concatenated coding systems use a Viterbi algorithm (VA) as the inner decoder and an algebraic hard-decision RS decoding algorithm with error-correction capability no more than $(n-k)/2$ and complexity near $O(n \log n)$. The soft-decision decoding algorithm provides an error-correction capability larger than $n - \sqrt{nk}$ [5], or even larger for low-rate codes [2]. However, the complexity of the soft-decision decoding algorithm is much higher. The complexity of interpolation-based RS decoding algorithms is $O(\delta^6 / k^3)$ [5], and since

$$min(\delta) < \sqrt{2(k-1)C_M} = \sqrt{(k-1)\sum_{i=0}^{q-1}\sum_{j=0}^{n-1} m_{ij}(m_{ij}+1)}$$

the complexity of the KV algorithm is approximately

$$O\left(\frac{\delta^6}{k^3}\right) < O\left(\left(\sqrt{(k-1)\sum_{i=0}^{q-1}\sum_{j=0}^{n-1} m_{ij}(m_{ij}+1)}\right)^6 / k^3\right)$$

$$\approx O\left(\left(\sum_{i=0}^{q-1}\sum_{j=0}^{n-1} m_{ij}(m_{ij}+1)\right)^3\right), \qquad (2)$$

where the m_{ij}'s are the nonzero entries in the multiplicity matrix. From (2) we can see that the complexity of the KV algorithm is $O(n^3)$. If we use the re-encoding algorithm, (2) can be simplified to

$$O\left(\left(\sum_{i=0}^{q-1}\sum_{j=0}^{n-1} m_{ij}(m_{ij}+1) - \sum_{t \in L} m_t(m_t+1)\right)^3\right),$$

where L represents the set of the k largest entries in the multiplicity matrix M, and the complexity is reduced to $O((n-k)^3)$. Although the re-encoding algorithm provides a large complexity reduction, soft-decision decoding complexity is still substantially larger than hard-decision decoding, especially when the code length n is large. However, for applications such as magnetic recording systems, most of the errors in a given frame are less than half the minimum distance because high-rate RS codes are used at very high signal-to-noise ratios, as illustrated by the example in Fig. 1, and therefore it is not necessary to use the more complex soft-decision algorithm on every frame. A hardware or software implementation of the backup mode shown in Fig. 2 is much more attractive than a strictly soft-decision decoding system. In such a system, the soft-decision RS decoder is invoked only if the hard-decision RS decoder

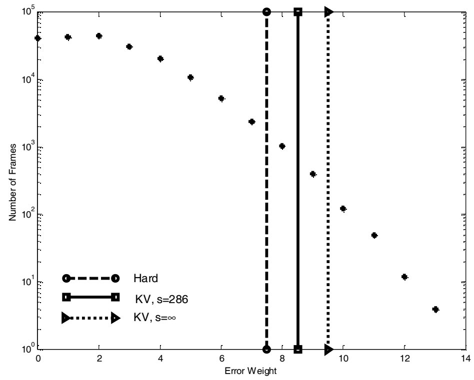

Fig. 1. Error distribution of RS (143,129) on an MEEPR4 channel with SNR=14.5dB.

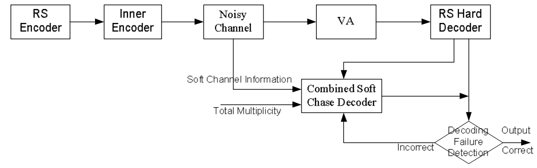

Fig. 2. Block diagram of a backup soft-decision RS decoding algorithm.

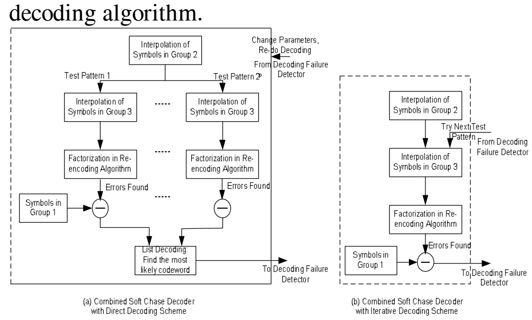

Fig. 3. Block diagram of two different types of combined soft Chase decoders.

fails, which can be determine in some way, e.g., a CRC check, and the coding gain of the soft decoder can be adjusted by changing the total multiplicity. A possible implementation of this system could have a hardware hard-decision decoder and a hardware/software (depending on system latency requirements) backup soft-decision decoder being executed at the same time, and if the output of the hard-decision decoder is correct, the operation of the soft-decision decoder is halted.

3.1. Combined soft Chase and re-encoding algorithms

Although the soft Chase algorithm described above provides a large decoding gain compared to the original KV algorithm, the decoding complexity is also very large, since the interpolation and factorization steps need to be executed for every test pattern. In order to reduce the complexity while maintaining the decoding gain, we propose a novel scheme, which combines the soft Chase algorithm with the re-encoding algorithm. The concept stems from the following observations: 1) In the soft Chase algorithm, we divide the sequence of polynomial interpolation points $\{(x_t, r_t), m_t\}_{t=0}^{N-1}$ into two groups: one includes $N-k'$ points $\{(x_t, r_t), m_t\}$, whose symbols do not contain any bit involved in bit-flipping; another includes k' points $\{(x_t, r_t), m_t\}$, whose symbols contain bits involved in bit-flipping (k' is related to p). 2) In the re-encoding algorithm, we also divide the sequence of points $\{(x_t, r_t), m_t\}_{t=0}^{N-1}$ into two groups: one includes k points $\{(x_t, y_t), m_t\}$, with the largest multiplicity m_t, which will be used to generate the modified set $\{(x_t, y'_t = 0), m_t\}$; another includes the remaining $N-k$ points $\{(x_t, y_t), m_t\}$, (For details on the re-encoding algorithm, please refer to [3], [4]).

It is interesting that the points in the first group of the re-encoding algorithm can be made to belong to the first group of interpolation points for the soft Chase algorithm, if the parameter k' is properly chosen. So for the soft Chase algorithm, we first generate a set of multiplicity matrices in accordance to the parameters s and k', then the multiplicities are divided into two groups: one consisting of those multiplicities that are common to all matrices, another consisting of the multiplicities that are generated by each particular test pattern. The first group can be further subdivided into two groups: the k entries with the largest value will be labeled as Group 1, the rest as Group 2. The pattern dependent multiplicities will be labeled as Group 3, which consists of 2^p subsets. With this grouping of the interpolation points, it is easy to see that we can calculate the bivariate polynomial passing through all the points in Groups 1 and 2 only once, which significantly reduces the soft Chase algorithm complexity without loss of performance. Since k' is usually very small, the complexity of finishing the interpolation step, by passing through each subset of points in Group 3, 2^p times, is not much larger than implementing the soft interpolation step just once in the KV algorithm. When combined with the re-encoding algorithm [4], the decoding complexity can be further reduced. The additional cost incurred by the soft Chase algorithm is that we need to perform 2^p factorizations. A factorization algorithm given in [6, p. 32] which utilizes the conventional hard-decision RS decoder to help perform the factorizations can be used here to reduce the factorization complexity. The implementation of 2^p partial interpolation and factorization steps can be realized in parallel, so the decoding can be achieved with a small delay (Fig. 3(a)). We can also trade off complexity for decoding delay by using only a single hardware/software core which implements the partial interpolation and factorization steps sequentially (Fig. 3(b)). The significant performance improvement makes this combined algorithm attractive. A summary of the combined algorithm is as follows:

Initialize: Channel output probabilities for each bit of received codeword.

Step 1: Find the p least reliable bits by searching the output probability sequence, and generate a set of reliability and multiplicity matrices corresponding to the test patterns;

Step 2: For each multiplicity matrix, generate a sequence of points $\{(x_t, y_t), m_t\}_{t=0}^{N-1}$, and assign them to their respective groups;
Step 3: Generate the intermediate polynomial $Q^{(1)}{}_M(x, y)$, which passes through points in Group 2, and store $Q^{(1)}{}_M(x, y)$;
Step 4: Finish the polynomial interpolation step by making $Q^{(1)}{}_M(x, y)$ pass through points in Group 3 to get $Q^{(2)}{}_M(x, y)$; finish soft-decision decoding using the re-encoding algorithm. If it fails, go to Step 5; else go to End;
Step 5: For the next test pattern, read $Q^{(1)}{}_M(x, y)$ from memory, then go to Step 4;
End.

It should be mention here, that soft-decision decoding failure can be determined by checking if the decoded codeword satisfies (1).

4. PERFORMANCE EVALUATIONS

For our simulations, we used the proposed backup mode described in Section 3. All the errors beyond half the minimum distance in one block were decoded using the soft-decision RS decoding algorithm. Practical consideration dictates the use of a fairly small total multiplicity, which in turn, leads to a modest performance gain.

The performance of the soft Chase algorithm was evaluated, and the results are given in Figs. 4 and 5 for a Rayleigh fading channel and a magnetic recording channel, respectively. In Fig. 4, the RS (63, 47) code with $p = 6$, i.e., $2^p = 64$ test patterns is used on an uncorrelated fading channel with total multiplicity s=158. The soft Chase algorithm provides almost a 1-dB gain over the Chase II algorithm [7], a 2-dB improvement over the KV algorithm and a 4-dB gain over traditional hard-decision decoding algorithms. It is also shown that increasing the total multiplicity will increase the error correction capability of both the soft Chase and KV algorithms. Similar results can be observed for magnetic recording systems, and are illustrated in Fig. 5.

5. CONCLUSIONS

An implementation of a soft-decision RS decoding algorithm combining a soft Chase algorithm with a re-encoding algorithm in a backup mode was proposed and the modification of current RS-coded ECC systems outlined. This ECC architecture has the advantage of using soft-decision decoding to improve current system performance without largely increasing hardware complexity.

REFERENCES

[1] R.G. Gallager, "Low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 8, pp.21-28, Jan. 1962.

[2] R. Koetter and A. Vardy, "Algebraic soft-decision decoding of Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2809-2825, Nov. 2003.

[3] H. Xia, and J.R. Cruz, "Application of soft-decision Reed-Solomon decoding to magnetic recording channels," submitted to *IEEE Trans. Magn*., Feb. 2003.

[4] W.J. Gross, F.R. Kschischang, R. Koetter and P.G. Gulak, "A VLSI architecture for interpolation in soft-decision list decoding of Reed-Solomon codes," in *Proc. IEEE Workshop Signal Processing Syst.*, pp. 39-44, Oct. 2002.

[5] V. Guruswami and M. Sudan, "Improved decoding of Reed-Solomon and algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. 45, pp.1755-1764, Sept. 1999.

[6] A. Ahmed, R. Koetter and N. R. Shanbhag, "VLSI architecture for soft-decision decoding of Reed-Solomon codes," submitted to *IEEE Trans. VLSI Systems,* Feb. 2003.

[7] D. Chase, "A class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inform. Theory*, vol. 18, pp.170-182, Jan. 1972.

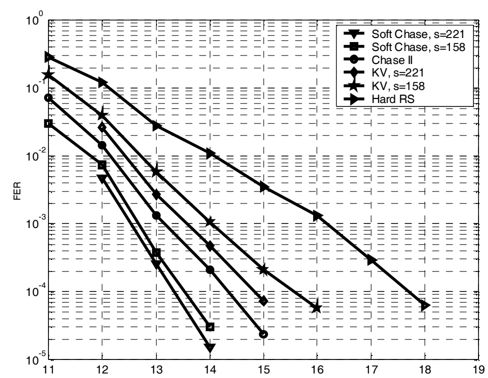

Fig. 4. Performance of a soft Chase algorithm on an RS (63, 47) code over an uncorrelated fading channel with different total multiplicities, p = 6.

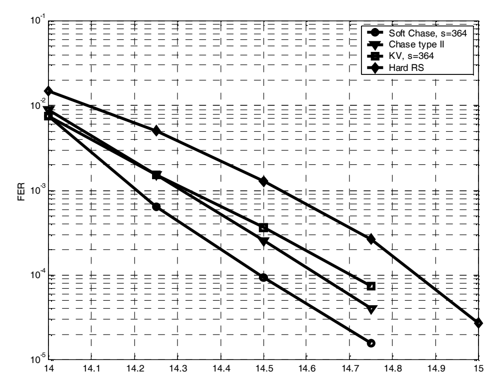

Fig. 5. Performance of a soft Chase algorithm on an RS code (137, 117) over an equalized MEEPR4 channel, S_c=2.967, p=4.