

A RECURSIVE METHOD FOR CALCULATING ERROR PROBABILITIES FOR A REED-SOLOMON CODEWORD WITH BOUNDED DISTANCE ERRORS AND ERASURES DECODING

Troy C. Nolan* and Wayne E. Stark
University of Michigan
Dept. of Electrical Engineering and Computer Science
Ann Arbor, Michigan 48109
tcnolan@eecs.umich.edu stark@eecs.umich.edu

Abstract

This paper is the second in a series of two papers dealing with methods for recursively calculating codeword error and erasure probabilities for Reed-Solomon (RS) codewords. As with the previous paper [1], a single RS codeword is transmitted in a channel where each transmitted symbol experiences independent identically distributed (IID) noise or a channel where each symbol experiences independent differently distributed (IDD) noise. Each received symbol is decoded using mechanism where a symbol or an erasure is produced, and the entire codeword is decoded using a bounded-distance (BD) decoder. Specifically, this paper deals with an errors and erasures (EE) decoder, while the previous paper addressed errors only (EO) decoding. It is common practice to assume the probability of incorrect codeword decoding is negligible and assume the decoder either correctly decodes the received codeword or fails the decoding process. However, we develop an efficient, recursive, mechanism for generating the exact probabilities (correct decode, incorrect decode, and decoder failure) in the IID case and bounds on the probability of incorrect decode for the IDD case.

1 Introduction

It is well known that coding is an essential element of a well designed digital communication system. In some cases, the additional structure imposed by coding can lower the required signal power by as much as 30 to 40 dB. Specifically, in systems where parts of the

transmitted codewords experience interference or fading, coding can provide a mechanism to combat these effects. Many systems utilize Reed-Solomon codes as part of their error control structure. While the Reed-Solomon code might provide an attractive error control element due to its distance and burst error control properties, analysis of the probability of undetected error for these codes can be complex.

In this paper we develop an efficient algorithm for calculating the probabilities of correct decode, incorrect decode, and decode failure for an RS code which is decoded using an error and erasures bounded distance decoder. We start with the assumption that each symbol in the codeword has a similar interference level (IID), and extend the analysis and algorithm to the case where each symbol may experience a different interference level (IDD).

In Section 2 we will discuss the system model under consideration. Section 3 will detail the development of the algorithm for codewords whose symbols all experience the same fading, and Section 4 will generalize the algorithm to the case where each symbol can have different fading statistics. In Section 5 we will compare our results to standard algorithms and present some numerical results.

2 System Model

In the system under consideration, a singly extended (N, K) RS codeword with minimum distance $d_{min} = N - K + 1$ is transmitted using orthogonal modulation across an M-ary symmetric channel with errors and erasures. The receiver demodulates noncoherently and makes symbol-by-symbol decisions as to what symbol

*This research funded by the Department of Defense graduate fellowship program and Army Research Office grant DAAH04-96-1-0377

was transmitted. Interference in the channel results in the non-zero probability of symbol crossover. In addition, the receiver may declare a symbol erasure if some condition is satisfied (such as the ratio threshold test (RTT)).

The receiver attempts to decode the received vector into a codeword using a BD/RS/EE decoder. If the received vector has t errors and e erasures satisfying $2t + e < d_{min}$, the received vector decodes correctly. However, if the the received codeword lies within a d_{min} "radius" with respect to an incorrect codeword, the codeword decodes incorrectly. If neither of these conditions hold, the decoder declares a decoding failure.

While it is common practice to assume the probability of undetected error is negligible, there are some applications that require this knowledge. Since the channel interference induces the non-zero probability that the decoded codeword is not the same as the transmitted codeword, we wish to determine this probability. With out loss of generality we assume the all zeros codeword is the transmitted codeword for all cases.

2.1 Channel Interference

The type of interference considered in this paper is non-specific, that is, as long as the individual symbol transition probabilities are calculable the type of interference is irrelevant. However, we do require independence between symbols.

2.2 Zero Patterns

A zero pattern is a vector that flags the positions of zero symbols in a RS codeword. If we create a vector of length N that contains 0's where a RS codeword has zero symbols and 1's where the same RS codeword has non-zero symbols we have created a "zero pattern". In this manner, every codeword has a zero pattern, but not every codeword has a unique zero pattern.

3 Case I: Symbols Experience IID Interference

In this section we consider the case where each symbol experiences statistically the same interference. For sake of later notation we index the transition probabilities so each symbol may have a different transition probability, but for this case this information is extraneous. The channel interference induces a non-zero transition probability on each symbol. Define the following quantities,

- P_c^l : the probability that the l -th symbol is correctly received,

- P_f^l : the probability that the l -th symbol is erased,
- P_e^l : the probability that the l -th symbol is incorrectly received,
- P_s^l : the probability that the l -th symbol is in error to a *specific* incorrect symbol.

In our case we use an M -ary symmetric errors and erasures channel and we assume $P_s^g = P_s^h$ for all h and g , so we may drop the superscript index for now. Clearly $P_f = 1 - P_c - P_e$, and since the M -ary channel is symmetric $P_s = \frac{P_e}{M-1}$.

3.1 Probability of Correct Decode

To begin, we define the event $B_{i,j}^l$ to be the event that i erasures and j errors have occurred in l transmitted symbols. To build the density function for the event $B_{i,j}^N$ we start from the first transmitted symbol:

- $P\{B_{0,0}^1\} = P_c$,
- $P\{B_{1,0}^1\} = P_f$,
- $P\{B_{0,1}^1\} = P_e$.

For the second symbol, we continue as follows:

- $P\{B_{0,0}^2\} = P\{B_{0,0}^1\}P_c$,
- $P\{B_{1,0}^2\} = P\{B_{1,0}^1\}P_c + P\{B_{0,0}^1\}P_f$,
- $P\{B_{2,0}^2\} = P\{B_{1,0}^1\}P_f$,
- $P\{B_{0,1}^2\} = P\{B_{0,1}^1\}P_c + P\{B_{0,0}^1\}P_e$,
- $P\{B_{1,1}^2\} = P\{B_{0,1}^1\}P_f + P\{B_{1,0}^1\}P_e$,
- $P\{B_{0,2}^2\} = P\{B_{0,1}^1\}P_e$.

The general term then becomes

$$P\{B_{i,j}^l\} = [P\{B_{i,j}^{l-1}\} \times I_{\{l \neq i+j\}}]P_c + [P\{B_{i-1,j}^{l-1}\} \times I_{\{i \neq 0\}}]P_f + [P\{B_{i,j-1}^{l-1}\} \times I_{\{j \neq 0\}}]P_e, \quad (1)$$

where $i = 0, 1, \dots, l$ and $j = 0, 1, \dots, l-i$, and $I_{\{m\}} = 1$ if the event m is true and $I_{\{m\}} = 0$ otherwise. Thus, the probability of correct decoding is

$$P_{CD} = \sum_{i=0}^N \sum_{j=0}^{N-i} P\{B_{i,j}^N\} \times I_{\{2j+i < d_{min}\}}. \quad (2)$$

3.2 Probability of Incorrect Decode

The probability of incorrect decoding can be constructed in a similar fashion to that of correct decoding. For this analysis we define the event of interest in a slightly different manner. Let $B_{i,j|k}^l$ be the event that we have i erasures and are a Hamming distance j (in the non-erased positions) from a weight k codeword after l symbols have been transmitted. In addition we define transition probabilities as

$$P_{c^*}^i = \begin{cases} P_s & \text{if } Z_i = 1 \\ P_c & \text{if } Z_i = 0, \end{cases}$$

$$P_{e^*}^i = 1 - P_f - P_{c^*}^i,$$

and

$$P_{f^*}^i = 1 - P_{c^*}^i - P_{e^*}^i = P_f,$$

where Z is the zero pattern for any weight k codeword.

For this case, we skip directly to the general term, noting that the construction matches the construction of the correct decode probability except that the distance measure is from an erroneous codeword and the transition vector is used instead of a scalar measure. The significance of the different transition probabilities is important in this analysis but more so in the development of the probability of error with different fading levels of Section 4.

For the I.I.D. case the general term is

$$\begin{aligned} P\{B_{i,j|k}^l\} &= [P\{B_{i,j|k}^{l-1}\} \times I_{\{l \neq i+j\}}] P_{c^*}^l + \\ & [P\{B_{i-1,j|k}^{l-1}\} \times I_{\{i \neq 0\}}] P_{f^*}^l + \\ & [P\{B_{i,j-1|k}^{l-1}\} \times I_{\{j \neq 0\}}] P_{e^*}^l, \end{aligned} \quad (3)$$

which is independent of the weight k word decoded. The probability of error to all weight k codewords can be found by summing

$$P_{ID,k} = A_k \sum_{j=0}^{\lfloor \frac{d_{min}}{2} \rfloor} \sum_{i=0}^{d_{min}-2j} P\{B_{i,j|k}^N\}, \quad (4)$$

where A_k is the weight enumerator for an MDS code [2]. The overall probability of error is the sum over the possible weights of code words,

$$P_{ID} = \sum_{k=d_{min}}^N P_{ID,k}. \quad (5)$$

3.3 Probability of Decoding Failure

Once the probability of correct and incorrect decoding is known, the remainder of the probability space is occupied by the probability of decoder failure. BD decoders fail to decode to a codeword when the received codeword is not in the decoding sphere of any codeword [3]. Thus, the probability of decoding failure is as follows:

$$P_{DF} = 1 - P_{CD} - P_{ID} \quad (6)$$

4 Case II: Symbols Experience IDD Interference

While Section 3 assumed each symbol in the transmitted codeword experienced statistically the same interference, this section does not. In the new model, we assume each transmitted symbol can have statistically different interference. Thus the index l in the definition of P_c^l , P_f^l , and P_e^l is now relevant. This adds another degree of complexity to the calculation, and requires some changes in the recursion for the probability of codeword error.

4.1 Probability of Correct Decode

We start by using the same notation as in the correct decode calculations of Section 3. Noting this, (1) becomes:

$$\begin{aligned} P\{B_{i,j}^l\} &= [P\{B_{i,j}^{l-1}\} \times I_{\{l \neq i+j\}}] P_c^l + \\ & [P\{B_{i-1,j}^{l-1}\} \times I_{\{i \neq 0\}}] P_f^l + \\ & [P\{B_{i,j-1}^{l-1}\} \times I_{\{j \neq 0\}}] P_e^l, \end{aligned} \quad (7)$$

where the definition of $B_{i,j}^l$ is the same as in (1). Thus, the probability of correct decoding is

$$P_{cd} = \sum_{i=0}^N \sum_{j=0}^{N-i} P\{B_{i,j}^N\} \times I_{\{2j+i < d_{min}\}}. \quad (8)$$

4.2 Probability of Incorrect Decode

When we consider the incorrect decoding case we must first construct transition probability vectors with respect to a weight k codeword. Unfortunately, this vector is dependent upon the position of the zeros in the weight k codeword under consideration. In fact, there are $\binom{N}{N-k}$ combinations of zero positions for a weight k codeword [2]. Thus, the number of cases to consider grows exponentially. For a realistic code, this complexity growth is unacceptable. To address this problem, we lower bound the probability of incorrect

decoding by selecting L unique zero patterns from the most probable error codewords of a given weight, and we sum the probability contribution of *all* the codewords associated with these L zero patterns.

Since transmitted symbols experience statistically different interference, if after decoding we examine the zero patterns associated with the decoded codewords some zero patterns are more likely to occur than others. We denote the i -th symbol in the ζ -th most likely zero pattern Z_i^ζ . We define the correct probability vector to be the probability that the received symbol type (zero or non-zero symbol) matches the symbol type of the zero pattern of weight k under consideration. Thus the correct probability vector, P_C , is constructed of elements P_C^i where

$$P_C^i = \begin{cases} P_c^i & \text{if } Z_i^\zeta = 0 \\ P_s^i & \text{if } Z_i^\zeta = 1. \end{cases} \quad (9)$$

The erasure probability vector is given from the channel as

$$P_F = [P_f^1 P_f^2 \dots P_f^N]. \quad (10)$$

The error probability vector, P_E is constructed from the probability space that remains,

$$P_E^i = \begin{cases} 1 - P_f^i - P_c^i & \text{if } Z_i^\zeta = 0 \\ 1 - P_f^i - P_s^i & \text{if } Z_i^\zeta = 1. \end{cases} \quad (11)$$

The event $B_{i,j|k}^{l,\zeta}$ is defined as the event that after l symbols are transmitted the received codeword has i erasures and it's associated zero pattern is a Hamming distance j (in the $N - i$ non-erased positions) from the ζ 'th most likely weight k zero pattern. For the specific pattern ζ , the general term for the recursion is very similar to that of Section 3 and is given by

$$\begin{aligned} P\{B_{i,j|k}^{l,\zeta}\} &= [P\{B_{i,j|k}^{l-1,\zeta}\} \times I_{\{l \neq i+j\}}] P_C^l + \\ & [P\{B_{i-1,j|k}^{l-1,\zeta}\} \times I_{\{i \neq 0\}}] P_F^l + \\ & [P\{B_{i,j-1|k}^{l-1,\zeta}\} \times I_{\{j \neq 0\}}] P_E^l. \end{aligned} \quad (12)$$

Next, the probability contribution of the ζ 'th zero pattern is found by summing,

$$P_{id,k}^\zeta = A_\zeta \sum_{j=0}^{\lfloor \frac{d_{min}}{2} \rfloor} \sum_{i=0}^{d_{min}-2j} P\{B_{i,j|k}^{N,\zeta}\}, \quad (13)$$

where A_ζ is the number of codewords with zero pattern ζ [2].

Various methods can be employed at this stage to determine how many likely zero patterns to select, L .

Some methods include choosing L as a function of N , selecting the zero patterns that are within an order of magnitude as likely as the most likely, or selecting a constant number of zero patterns. We use a constant L for our results since, in general, the received vector is more likely to be closer to the origin than further. A constant number will emphasize the most likely zero/error patterns without undue additional complexity. Summing over the L most likely zero patterns gives us a lower bound on the contribution of the weight k error patterns. To upper bound we treat the contribution of the L most likely zero patterns as the average contribution. Next, we sum over the possible error weights to get the bounds due to all zero patterns:

$$P_{id}^{UB} = \sum_{k=d_{min}}^N \frac{A_k}{A_\zeta L} \sum_{\zeta=0}^L P_{id,k}^\zeta \quad (14)$$

and

$$P_{id}^{LB} = \sum_{k=d_{min}}^N \sum_{\zeta=0}^L P_{id,k}^\zeta. \quad (15)$$

The upper bound is tight when all code symbols experience IID interference. For IDD interference the upper and lower bounds grow tighter as L grows larger (more processing power).

4.3 Probability of Decoding Failure

In a similar manner to the failure probability of the same interference case we calculate the probability of decoding failure as follows:

$$P_{df} = 1 - P_{cd} - P_{id}. \quad (16)$$

5 Results

The result of our work is an efficient recursion which allows calculation of probability of error statistics for RS codewords, with an errors and erasures decoder, in a channel in which the interference can (potentially) vary from symbol to symbol. We have discussed 6 specific cases:

1. Probability of correct decode when each symbol experiences IID interference, P_{CD} .
2. Probability of in-correct decode when each symbol experiences IID interference, P_{ID} .
3. Probability of decode failure when each symbol experiences IID interference, P_{DF} .
4. Probability of correct decode when each symbol experiences IDD interference, P_{cd} .
5. Probability of in-correct decode when each symbol experiences IDD interference, P_{id} .

6. Probability of decode failure when each symbol experiences IDD interference, P_{df} .

The major result of this work is the reduction in calculation complexity. Standard methods, such as those found in [3], achieve their result by brute force searching. This method, while accurate, must consider all possibilities of received vectors within the decoding sphere around all codewords. In case 1, the complexity of our method grows as $O(N^2)$ and the brute force method also grows as $O(N^2)$, but our method can be applied to some interesting channel models like Markov channel models, where the brute force search cannot. Case 2 is where our methods starts to show improvement; a brute force search of possible codeword errors grows as $O(N^3)$ while our method maintains it's $O(N^2)$ growth rate. Cases 3 and 6 are trivial subtractions. In case 4 our method still has complexity on the order of $O(N^2)$, while the brute force method must consider all possible combinations of t errors and e erasures where $2t + e < d_{min}$, which grows as $O(3^N)$. The final case, 6, our bounds maintain their growth rate as $O(N^2)$ (potentially $O(N^3)$ if L is a function of N), and the brute force method grows as $O(N^3 3^N)$.

5.1 Numerical Results

To illustrate the tightness of the upper bound in the case where symbols experience IID noise we present the results in Figure 1. We generate the upper and

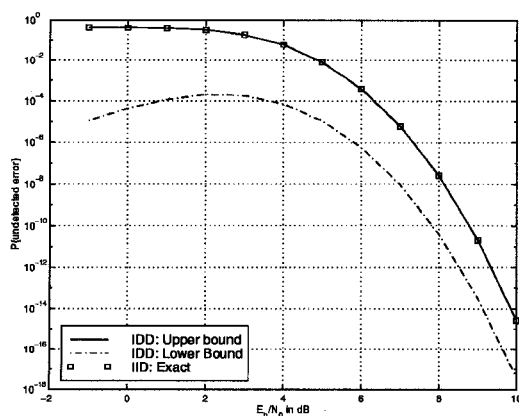


Figure 1: Bounds on and exact error probability for a (16,12) RS codeword with no symbols faded, $P_f = 10^{-4}$.

lower bound error probability curves using the IDD bounds from (14) and (15), and the tightness of the upper bound is shown by the IID error probability curve from (5).

Next we illustrate the tightening of the bounds by

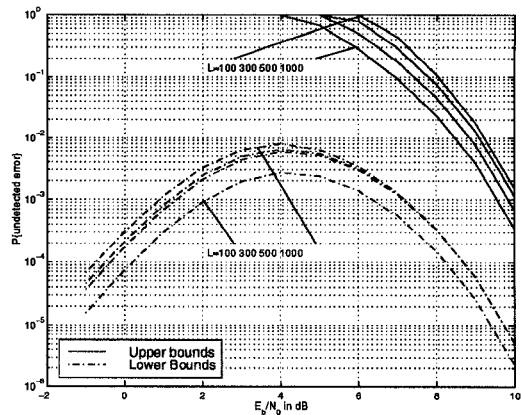


Figure 2: Tightening bounds on the error probability for a (16,12) RS codeword with 4 symbols faded, $P_f = 10^{-4}$.

fading 4 symbols by 5dB and increasing L from 100 to 1000. As the curves in Figure 2 show, they tighten with the increase in L . In the low SNR region the bound is relatively loose. This is a result of the transition probabilities of the non-faded symbols becoming closer to the transition probability of the faded symbols, hence the contribution of the first L zero patterns is reduced. Since we expect the probability of incorrect decode to be monotonic non-increasing, a first pass improvement to this bound would be to move the lower bound in the low SNR region up to the level of the maxima.

6 Conclusion

In this paper we give a method for calculating the probabilities of correct, incorrect, and decoder failure for correcting a single Reed-Solomon codeword with an errors and erasures bounded distance decoder in the presence of interference. The presented method holds computational complexity to polynomial time, while standard brute force methods compute in exponential time.

References

- [1] T. C. Nolan and W. E. Stark, "A Recursive Method for Calculating Error Probabilities for a Reed-Solomon Codeword," Proceedings of the IEEE Vehicular Technology Conference, May 1998.
- [2] R. E. Blahut, "Theory and Practice of Error Control Codes," Addison-Wesley, Reading, Mass., 1983.
- [3] A. Michelson and A. Levesque, "Error Control Techniques for Digital Communication," John Wiley and Sons, Inc., New York, 1985.
- [4] S. Wilson, "Digital Modulation and Coding," Prentice-Hall, Inc., New Jersey, 1996, p. 211-215.