TDA Progress Report 42-114

August 15, 1993

Exploiting the Cannibalistic Traits of Reed-Solomon Codes

O. Collins[1]
Johns Hopkins University

In Reed-Solomon codes and all other maximum distance separable codes, there is an intrinsic relationship between the size of the symbols in a codeword and the length of the codeword. Increasing the number of symbols in a codeword to improve the efficiency of the coding system thus requires using a larger set of symbols. However, long Reed-Solomon codes are difficult to implement and many communications or storage systems cannot easily accommodate an increased symbol size, e.g., M-ary frequency shift keying (FSK) and photon-counting pulse-position modulation demand a fixed symbol size. This article describes a technique for sharing redundancy among many different Reed-Solomon codewords to achieve the efficiency attainable in long Reed-Solomon codes without increasing the symbol size. The article presents techniques both for calculating the performance of these new codes and for determining their encoder and decoder complexities. These complexities are usually found to be substantially lower than conventional Reed-Solomon codes of similar performance.

I. Introduction

This article examines a new class of codes derived from Reed-Solomon codes that captures the essential power of long Reed-Solomon codes even though the symbol size remains small. In most cases, the decoding complexity also remains small. One explanation of the increased performance of longer Reed-Solomon codes in a communications or storage system is that for a given rate they have a higher minimum distance. However, this rationale applies rigorously only when the symbol error probability is minute. Another explanation for the performance of long Reed-Solomon codes is that as sequences become longer the law of large numbers begins to take hold, e.g., very long codewords are likely to have the typical number of errors. An elaboration of this statement will serve as a good introduction to the fundamental concept in this article.

A Reed-Solomon code that experiences independent errors with symbol error probability e and erasure probability f must have a rate less than $1 - 2e - f$, the expected fraction of parity check symbols required, if the probability of its failing to decode is to be low. For a given decoder failure probability, the longer the code, the more closely its redundancy can approach this value. Any size code will have a 50 percent chance of correct decoding if its distance is exactly equal to one more than the expected number of erasures plus twice the expected number of errors. A very long code will operate in the region where the law of large numbers applies and experience a sudden, precipitous drop

[1] Work supported under a contract between Johns Hopkins University and JPL.

in the probability of decoder failure as the amount of redundancy provided increases from less than $1 - 2e - f$ to greater than $1 - 2e - f$. A shorter code will experience a more gradual drop. Figure 1 shows an example of a 5-bit extended Reed-Solomon code operating over the independent symbol erasure channel with 1 percent symbol erasure probability.

If a 5-bit Reed-Solomon code experiences an independent symbol erasure rate of 1 percent, and if the probability of decoder failure must be less than 5×10^{-6}, then the code must able to correct 5 erasures. The probability of having six or more erasures, $E_{32}(0.01, 06)$, is actually 7.24×10^{-7}. There is, however, less than a 0.4 percent chance of having more than two erasures in a codeword. Thus, there is the possibility of sharing among many different codewords the three parity check symbols that are needed only 0.4 percent of the time. One way of achieving this would be through a return channel over which the receiver informs the transmitter of the particular codewords that require the extra parity checks. The transmitter computes all five parity checks, but, in the beginning, sends over the channel only the first two. The receiver attempts decoding of the shortened codewords by declaring the symbols that were not sent to be erasures. Then, $E_{29}(0.01, 3) = 0.0030$ of these shortened codewords will fail to decode. (Since this example is using the erasure channel, the probability of incorrect decoding is undefined.) The receiver then requests the extra parity checks for those few words that failed to decode. The decoder failure probability is unchanged from the original system, but only 2.009 parity checks per codeword are required on average, i.e., the average number of parity checks per codeword has been cut by more than half.

The essential concept in this article is that the same type of parity check sharing can be accomplished without the reverse channel. The next section will show how this is accomplished by completing the introductory example and will explain techniques for computing the performance of codes used on the erasure channel.

II. Codes for Erasure Correction

The parity check symbols will be shared among a set of codewords by using the same Reed-Solomon encoder that produced them. The third, fourth, and fifth parity symbols from 27 different codewords will be fed back into the encoder to form the vertical codewords shown in Fig. 2.

None of the symbols intersected by both Reed-Solomon codewords is sent over the channel. In fact, only the first four parity check symbols of the vertical codewords will be sent over the channel; the fifth symbol of each of the vertical codewords will be discarded.

Now that the scheme for sharing parity check symbols among the codewords has been precisely described; its operation will be proven by showing that the decoder failure probability of the horizontal codewords has decreased. Consider a horizontal codeword that has more than two erasures and so has failed to decode. The probability that three or more of the remaining horizontal codewords have also failed to decode is $E_{26}(0.0030, 3) = 6.67 \times 10^{-5}$. Should this event occur, decoder failure can be declared since doing so will contribute only $(0.003)(6.67 \times 10^{-5}) = 2 \times 10^{-7}$ to the overall probability of failure to decode. The probabilities multiply since the initial failure of a single codeword is independent of the success or failure of any or all of the others.

If the 26 other codewords have only two failures among them, then each of the vertical codewords will decode if no more than one channel erasure has occurred in its set of four parity symbols, i.e., the symbols actually sent over the channel. The probability of more than one channel erasure in a set of four is $E_4(0.01, 2) = 0.0006$. Thus, if three horizontal codewords have failed, the vertical codewords will fail independently with probability 0.0006; this number is an upper bound on the erasure rate experienced by the rightmost three symbols of each horizontal codeword if failure has not already been declared.

An upper bound on the failure of each horizontal codeword is thus obtained by assuming independent erasures on all symbols with a 0.01 rate for the first 29 and a 0.0006 rate for the last three and then adding 2×10^{-7} to account for the probability of declaring vertical codeword failure. This bound is negligibly less than the original failure probability based on a uniform 1 percent erasure rate. The complexity increase at the encoding end is 3/27, since for every set of 27 codewords, three more will be needed. The decoder complexity increase will never be more than 6/27, the fractional contribution of the vertical codewords plus the fractional contribution of the three possible horizontal redecodings. The average work performed by the decoder can, of course, be less, e.g., often complete decoding can be accomplished without using the vertical codewords and so their information may simply be discarded.

III. Single Field Codes for Error Correction

This section will explain the design of single stage combined Reed-Solomon codes for the independent symbol error channel by presenting an example based on the

NASA Standard (255,223) code. The channel error rate is 2 percent which produces a decoder failure probability of $E_{255}(0.02, 17) = 1.9 \times 10^{-5}$; this is almost equal to the design symbol error rate in the Voyager spacecraft communications system. For implementation economy, the vertical and horizontal codewords will again be identical. The first 16 redundant symbols of each horizontal codeword will be sent over the channel in the conventional manner. The remaining 16 will be incorporated into vertical codewords as shown in Fig. 3. The probability that one of the horizontal codewords will have more than eight errors, and so fail, is $E_{239}(0.02, 9) = 0.0528$. The structure of Reed-Solomon codes guarantees that almost all of these excess error patterns can be recognized, i.e., the probability of incorrect decoding of one of the horizontal codewords is negligible [1]. The 223 information symbols of the vertical codewords will thus experience an erasure rate of 0.0528.

Consider a codeword that has experienced 9 or more errors. The probability of 28 or more of the other codewords failing is $E_{222}(0.0528, 28) = 2.00669 \times 10^{-5}$. If 29 or more of the horizontal codewords have failed, the decision not to attempt decoding of the vertical codewords contributes $(0.0528)(2.00669 \times 10^{-5}) = 1 \times 10^{-6}$ to the failure probability of each of the horizontal codewords. The decision never to attempt decoding of the vertical codewords with more than 28 declared erasures is quite sound in this case since it keeps the error probability of the vertical codewords negligible [1].

If a vertical codeword has 28 erasures, then the probability of its not decoding successfully is $E_{32}(0.02, 3) = 0.0257$. Thus, each of the horizontal codewords will experience an independent symbol erasure rate of no more than 0.0257 on the last 16 symbols if an excess of erasures has not already caused vertical codeword failure to be declared. The failure probability of the horizontal codewords has again improved since the drop in the error (now erasure) rate experienced by the last 16 symbols more than offsets the 10^{-6} chance of decoder failure caused by an excess of erasures in the vertical codewords.

The average redundancy of each codeword in the block is $16 + 16(32/223) = 18.29$, which gives a code rate for the block of $223/(223 + 18.29) = 0.9242$. The rate of a 10-bit Reed-Solomon code that experiences a 2 percent symbol error rate and is able to achieve the same failure probability is 0.917.

The encoder complexity increase produced by using the scheme shown in Fig. 3 will be the fractional increase in the number of codewords required, 16/223, just as it was in the case of erasure. The decoder may now, however, experience a decrease in complexity since it never needs to cope with the situation where more than 28 of the horizontal codewords have more than eight errors. The design of Reed-Solomon decoders often exploits the typically small number of errors per codeword by making the decoding time a random variable and employing a buffer. However, the scheme presented here allows an explicit upper bound on the required buffer size. This decoder simplification is another reason for declaring failure before exhausting all possibility of success.

Sections II and III have presented techniques for analyzing the decoder failure probabilities of single-stage arrays of codes when the symbol error rate is known. The essential technique used in those sections was the division of the vertical codeword failure mechanisms into two groups: those that affect the entire block of codewords and those that affect each codeword independently. For economy of expression, the division was made complete by the use of a union bound. A more refined horizontal decoder failure probability estimate would have to consider all different possible numbers of erasures in the vertical codewords; the methods involved are straightforward but lengthy extensions of the arguments in Sections II and III.

IV. Optimum-Distance Single Field Codes

The approach of Sections II and III was to take an existing coding system and improve its rate without substantially increasing its complexity. The total distance of the entire code block never factored into the design because the channel error and erasure rates were substantial. However, the designs developed did maintain the free distance of the entire block. This section considers to what extent the number of redundant symbols in a block of crossed maximum distance separable (MDS) codes can be reduced if the only requirement is that a minimum distance be maintained for the entire block. This section will present optimum constructions for single- and double-error correcting codes. In addition to quiet communications channels, such low distance codes are important for disk drive arrays and computer memory applications. The techniques used can be extended to higher distances; however, different constructions not based on crossed sets of MDS codes can yield higher rates when the distance is greater than five. Nevertheless, [3] shows that other types of multilevel codes may still be useful for high distance applications because of their economy of implementation and adaptability to channel error statistics.

Figure 4 shows a means of constructing a single-error or double-erasure correcting code with $(N-2)^2$ information symbols and three redundant symbols out of length

N Reed-Solomon codes. The distance of the code is most easily demonstrated by the erasure decoding algorithm. If the two possible erasures are in different horizontal codewords, then the first vertical codeword will experience two erasures and will decode. Each of the two horizontal codewords with an erasure will receive one symbol of redundancy from the first vertical word and, so, will be able to decode. The second vertical codeword provides for the case where both erasures occur in the same horizontal codeword. Thus, since the code can correct two erasures, it has distance three.

The procedure for correcting a single error is slightly more complex. As before, the first step involves calculating the information symbols of the vertical codewords by re-encoding the information in the horizontal codewords. If there has only been one error, then all of the symbols in the first vertical codeword except one are correct. Furthermore, the symbol in the first vertical codeword coming from the horizontal codeword containing the error is guaranteed to be incorrect since the first redundant symbol together with the information symbols of each horizontal codeword form a code of distance 2, which is single-error detecting. Thus, the decoding of the first vertical codeword pinpoints the horizontal codeword with the error. The second vertical codeword can now be decoded by declaring an erasure in the marked position and, so, both redundant symbols will be available to the damaged horizontal codeword. The construction in Fig. 4 is asymptotically close to being a perfect code since $[(N-2)^2(N-1)]/N^3$ approaches one as N becomes large. Figure 5 shows a construction for a distance five code. Its efficiency is, however, no better than can be achieved by concatenating symbols and using double-length Reed-Solomon codes; its advantage is easy encoding and decoding.

V. Summary and Discussion

The technique presented in this article allows a clear improvement in code rate of symbol-error-correcting and symbol-erasure-correcting codes for any given decode failure probability. Moreover, the computational cost to both the encoder and decoder is negligible. The only price to be paid is in interleaving and buffering, i.e., decoding cannot be completed until all of the codewords in an entire array are received.

An encoder will customarily use interleaving anyway to make symbol errors independent. Some number of Reed-Solomon codewords, e.g., eight 255 symbol codewords for the Galileo S-band, will be interleaved to form an interleaving block. All codewords in an interleaving block must come from different arrays to preserve independence. The amount of storage required at the encoder is not large, however, since only the vertical redundancy needs to be saved until the entire interleaved collection of arrays has been sent.

The overall encoding and decoding cost of the class of codes presented in this article proves to be substantially superior to that of longer Reed-Solomon codes even if large and small symbols experience the same error rate. Real communications channels, e.g., those employing determinate-state decoding [2], favor small symbols. For these channels, cannibalistic Reed-Solomon codes can offer very large improvements.

References

[1] R. J. McEliece and L. Swanson, "On the Decoder Error Probability for Reed-Solomon Codes," *IEEE Transactions on Information Theory*, IT-32, pp. 701–703, September 1986.

[2] O. Collins and M. Hizlan, "Determinate State Convolutional Codes," *The Telecommunications and Data Acquisition Progress Report 42-107, vol. July–September*, Jet Propulsion Laboratory, Pasadena, California, pp. 36–56, November 15, 1991.

[3] K. Abdel-Ghaffar and M. Hassner, "Multilevel Codes for Data Storage Channels," *IEEE Transactions on Information Theory*, IT-37, pp. 735–741, May 1991.

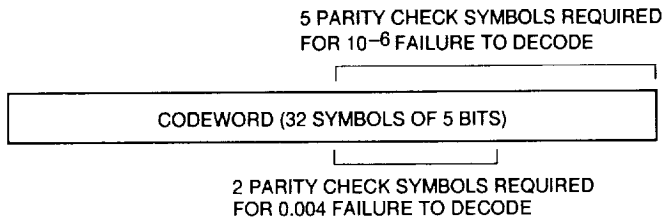

Fig. 1. Five-bit extended Reed-Solomon code.

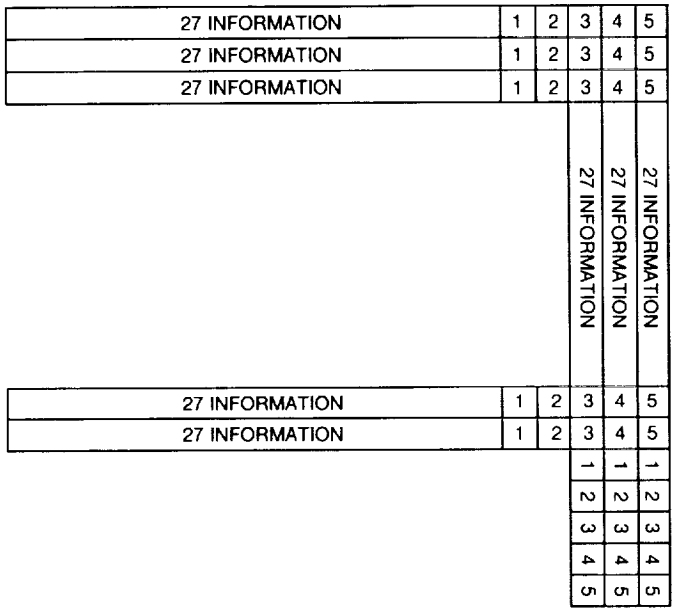

Fig. 2. Redundancy sharing.

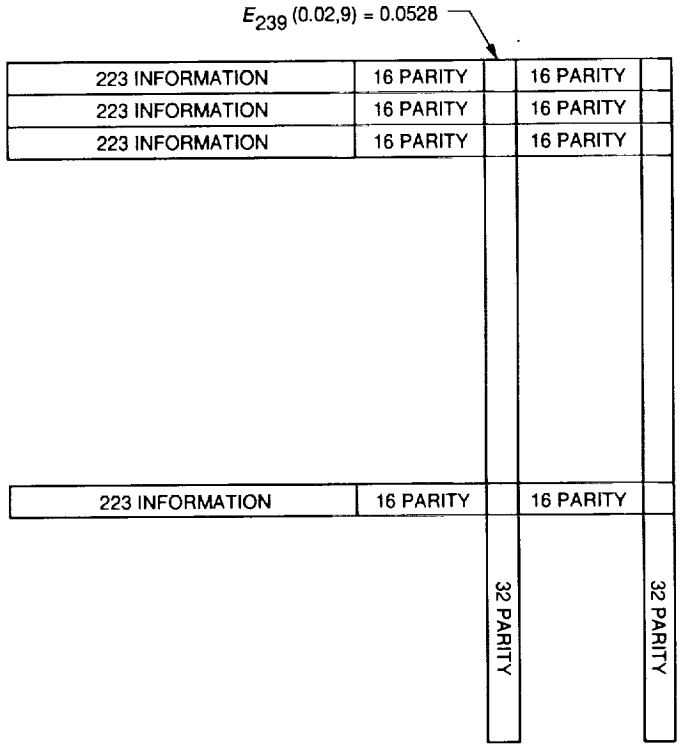

Fig. 3. Using the NASA Standard (255,223) code.

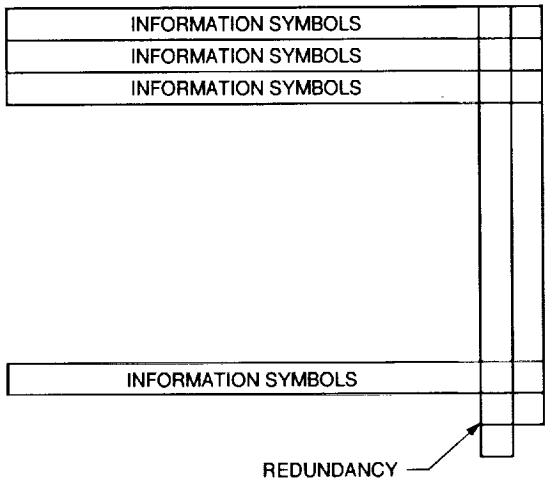

Fig. 4. A means of constructing a single-error or double-erasure correcting code.

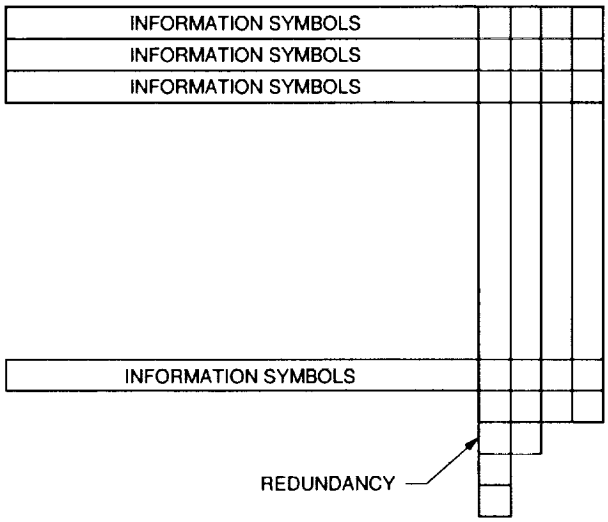

Fig. 5. A construction for a distance five code.