Decoding Real Numbered Block and Convolutional Codes with Erasure and Impulsive Noise Channels

S. Jafarzadeh, M. Khatami, and F. Marvasti, Senior Member IEEE

Abstract

In this paper, a class of real numbered block and convolutional codes are encoded and decoded using various iterative methods to remove channel impulsive noise and erasures. In order to recover the received vector, two iterative algorithms are used for erasure and impulsive noise distortions. Similarities of real number convolutional codes to linear block codes are discussed. An iterative method with adaptive thresholding is described for the reconstruction of impulsive noise using sparse signal processing.

Index Terms—Real number codes, Impulsive noise cancellation, Erasure channel, Convolutional codes, Iterative technique, IMAT method.

1. INTRODUCTION

Generally, the channel encoding is performed in finite Galois fields as opposed to real/complex fields. The reason is the simplicity of logic circuit implementation and insensitivity to the pattern of errors. On the other hand, the real/complex field implementation of error correction codes has stability problems with respect to the pattern of impulsive, quantization and additive noise [1]-[3]. Nevertheless, such implementation has found applications in fault tolerant computer systems [4]-[6] and impulsive noise removal from 1-D and 2-D signals [7], [8]. Similar to finite Galois fields, real/complex field codes can be implemented in both block and convolutional fashions. These coding methods are over the real or complex number fields, and can be implemented with standard digital signal processors. The possibility of utilizing real numbered codes permits the codes to be implemented with operations normally available in standard programmable digital signal processors.

Also, many of the well-known algebraic principles of error correction codes hold over the fields of real number and these principles are therefore directly applicable. The signal processing techniques which are introduced in this paper are appropriate for erasure and impulsive noise channels.

The paper is organized in the following manner. In Section II, a brief introduction of real numbered linear block codes is given by defining DFT codes and summarizing the important properties. In Section III, we define real numbered convolutional codes, and show the similarities between these codes and linear block codes. Section IV introduces an iterative algorithm to compensate the distortion of the erasure channel. Section V deals with a non-linear iterative technique named the IMAT method[1]. This method is presented to reconstruct the impulsive noise in the code vector, thus it can be omitted from the received vector. Some simulation results are presented in section VI and section VII concludes the paper.

2. REAL NUMBER LINEAR BLOCK CODES

The (N, K) Complex field linear block codes are a class of error correcting codes similar to finite field codes which consist of message and code blocks of K and N symbols, respectively. The generator matrix G is a K×N matrix consisting of K independent vectors which form the code space. The parity check matrix is a (N-K) ×N matrix including N-K independent vectors which are orthogonal to the code space. In order to form such a generator matrix, K rows of a unitary matrix can be chosen for matrix G and the remaining rows form the parity check matrix. Since rows of a unitary matrix are orthogonal, the following equation is satisfied [9]:

$$GH^H=0 \tag{1}$$

By using the inverse DFT matrix as a unitary matrix, the generator matrix consists of any K rows of the IDFT matrix. The parity check matrix of the code consists of the remaining rows. Thus, each code word is zero in N-K parity frequencies. If the parity frequencies are not zero, the presence of error in the code vector will be indicated. In order to form DFT codes in the field of real numbers, the frequencies are selected in such a way that the complex conjugate of each row also belongs to the generator matrix [10]. DFT codes have a minimum distance equal to N-K+1, which it implies they can correct up to $\left\lfloor\frac{N-K}{2}\right\rfloor$ sample errors or N-K sample erasures.

We first consider the message signal is sent through an erasure channel. We use the signal space projection method to compute the projection of the received vector in the code space [11]. In other words, the code word which has the minimum distance to the received vector is computed in order to approximate the message signal. This method can be used for all linear block codes with the generator matrix G. The matrix which gives us the message signal is called the pseudo inverse matrix.

$$\hat{x} = y\,\underbrace{G^H(G.G^H)^{-1}}_{pseudo\ inverse} \tag{2}$$

where $\hat{x}$ is the minimum distanced code to the received

[1] Iterative Method with Adaptive Thresholding.

vector and y is the received code word.

In the case of DFT codes, the pseudo inverse matrix simply becomes G^H. Because of the erased samples of the received signal, the approximated message is distorted. In order to recover the distorted message, an iterative technique is used.

3. Convolutional Codes

Convolutional codes can be represented from two points of view. First, they can be considered as linear block codes. Then the generator matrix and the parity check matrix are as follows:

$$G=\begin{bmatrix} G_0 & G_1 & \cdots & G_m & 0 & 0 & \cdots & 0 & 0 \\ 0 & G_0 & G_1 & \cdots & G_m & 0 & \cdots & 0 & 0 \\ 0 & 0 & G_0 & \cdots & G_{m-1} & G_m & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & G_0 & G_1 & \cdots & G_m \end{bmatrix} \tag{3}$$

$$H=\begin{bmatrix} H_0 & 0 & 0 & \cdots & 0 & 0 & 0 & \\ H_1 & H_0 & 0 & \cdots & 0 & 0 & 0 & \\ H_2 & H_1 & H_0 & \cdots & 0 & 0 & 0 & \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \\ H_m & H_{m-1} & H_{m-2} & \ldots & H_0 & 0 & 0 & \\ 0 & H_m & H_{m-1} & \cdots & H_1 & H_0 & 0 & \\ 0 & 0 & H_m & \cdots & H_2 & H_1 & H_0 & \\ & & & & & & & \ddots \end{bmatrix} \tag{4}$$

Where m is the constraint length.

G_i and H_i i = 0,…,m are K×N and (N-K)×N matrices, respectively. In real numbered codes, each element of these matrices is real. For the convolutional codes we have:

$$C_{1\times N\cdot(p+m)}=U_{1\times k\cdot p}G_{k\cdot p\times N\cdot(p+m)} \tag{5}$$

where C, U, (k×p) and N×(p+m) are the message, code vector, message length and code length, respectively. Thus convolutional codes can be decoded similar to the linear block codes. From the second point of view, convolutional codes are considered as the response of message vector to two or more filters; these responses are multiplexed to form the code vector.

Because of the lost samples in erasure channels, the responses of the decoders are distorted. In the next section, an iterative technique is applied to compensate for this type of distortion to achieve the original signal.

4. Iterative Method for Erasure Channels

Iterative methods can be applied to compensate for the distortion of an erasure channel. In order to obtain the original signal from the distorted one, we use a recursive relation as follows [12]:

$$x_{k+1}=\lambda[y-f(x_k)]+x_k \tag{6}$$

where λ and k are the relaxation parameter and k[th] iteration, respectively. x_0 is replaced by $\hat{x}$ in equation (2). $f(\cdot)$ can be considered as the distorting operator and y is the distorted signal, which must be recovered. This equation converges to x (original signal) if k goes to infinity.

Figure 1 demonstrates the block diagram for the iterative algorithm.

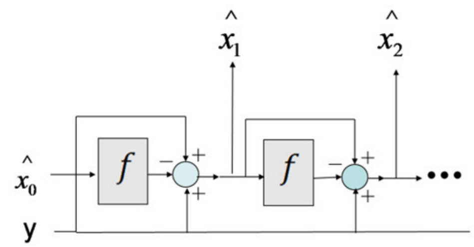

Fig. 1. Block diagram of the iterative algorithm.

4.1 *Decoding linear block codes using iterative technique*

The iterative decoding scheme for linear block codes is to substitute the block f with a distorting function which, in this case, consists of the generator matrix, known erasure channel and the pseudo inverse of the generator matrix. Figure 2 depicts the block diagram of the described distorting function. If the rate of erasure does not exceed the encoder capacity, which is $\frac{N-K}{N}$ in case of erasure channel, the iteration represented in Fig. 1 converges to the actual signal.

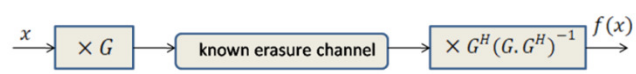

Fig. 2. Block diagram of the distorting function in iterative method for linear block codes decoder.

4.2 Decoding of convolutional codes using an iterative technique

Real numbered convolutional codes can be assumed as linear block codes; thus they can be decoded using the method described in the previous subsection. The generator matrix of these codes is depicted in (3). On the other hand, considering convolutional codes as linear filters, an approximation of the message code can be computed by averaging the responses of these filters. Figure 3 shows the block diagram of encoding and decoding of convolutional codes.

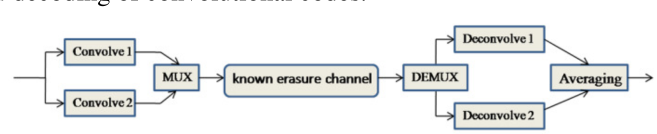

Fig. 3. Block diagram of the averaging method for convolutional codes decoding.

This block diagram can also be considered as the distorting function $f(\cdot)$in the iterative block diagram in order to compensate for the distortion. Figure 3 is designed for the rate $\frac{1}{2}$ convolutional encoder. At each stage of decoding, the results of the two branches are averaged in order to recover the message signal.

5. THE IMAT METHOD FOR IMPULSIVE NOISE CANCELATION

In this section, a non-linear iterative technique is proposed to reconstruct the impulsive noise, which is called the IMAT method. This method was first proposed in [13]. The goal of this technique is to reconstruct the impulsive noise in order to remove it from the received signal. In order to separate noise from signal, parity check matrix is used; denoting the observation vector at the receiver by $\hat{y}$, we have:

$$\hat{y} = y + n \tag{7}$$

where n is the impulsive noise. Multiplying $\hat{y}$ by the Hermitian of the parity check matrix, we have:

$$\hat{y}.H^H = (y+n).H^H = x.G.H^H + n.H^H = n.H^H \tag{8}$$

Using the pseudo inverse of H^H, we obtain:

$$\hat{y}.H^H.(H.H^H)^{-1}.H = n.H^H.(H.H^H)^{-1}.H = \tilde{n} \tag{9}$$

$\tilde{n}$ is an approximation of n. In the case of DFT codes, $n.H^H$ defines the amplitudes of the parity check frequencies of the noise which are available. Thus, the goal is to compute the whole noise from its known parameters using the fact that it is sparse. The non-linear function in this method is thresholding. According to the sparsity of impulsive noise, a thresholding block is used in the process to keep the sparse characteristic of the impulsive noise in consecutive iterations. The threshold value is decreased exponentially through the iterations in order to find every impulse. The following block diagram represents the IMAT method.

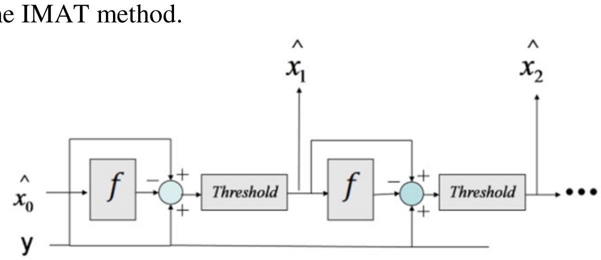

Fig. 4. Block diagram of the IMAT method.

where f is the distorting function.

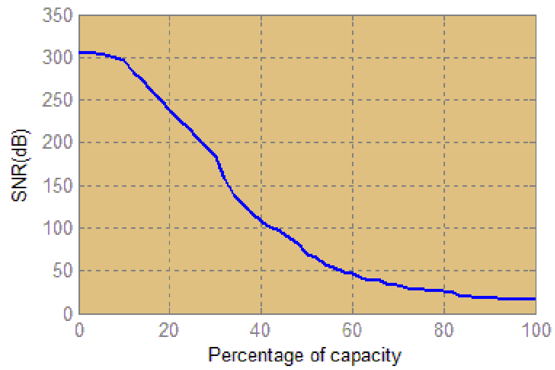

Fig. 5. SNR vs. the percentage of erasure for the linear block code decoder.

In both cases of real numbered linear block and convolutional codes, the distorting function relates n to $\tilde{n}$ according to (9):

$$f(x) = x.H^H.(H.H^H)^{-1}.H \tag{10}$$

Thus, we can obtain better approximation of the noise vector through the iterations; thus, it can be removed from the code vector which results in the correct decoding of the message signal. If the rate of the erasure does not exceed the encoder capacity, which is$\frac{N-K}{2N}$ in the case of a channel with impulsive noise, the iteration represented in Fig. 4 converges to the actual signal with a proper choice of the relaxation parameter. The threshold level is reduced exponentially in each iteration.

6. SIMULATION RESULTS

6.1 Real numbered linear block code (DFT code) results:

The input signal is taken from a uniform random distribution of block length 50 and the simulations are run 1000 times and then averaged. The SNR value ,which is mentioned in these figures, is the ratio of the original signal power to the difference of the original and the recovered signal power. The following subsections describe the simulation results for erasure and impulsive noise channels.

6.1.1) Decoding for Erasure Channels: The iterative method which is shown in Fig. 1 is used for the decoding of DFT codes for erasure channel. The encoder rate is $\frac{1}{2}$ and the relaxation parameter is set to 0.01. The SNR improvement versus the relative rate of erasures with respect to the theoretical maximum rate of correction capability (full capacity) is shown in Fig. 5.

6.1.2) Decoding for Impulsive Noise Channels: In this figure, the locations of the impulsive noise samples are generated randomly and their amplitudes have Gaussian distributions with zero mean and variance equal to 1, 2, 5. The SNR values versus the percentage of the channel capacity is shown in Fig. 6.

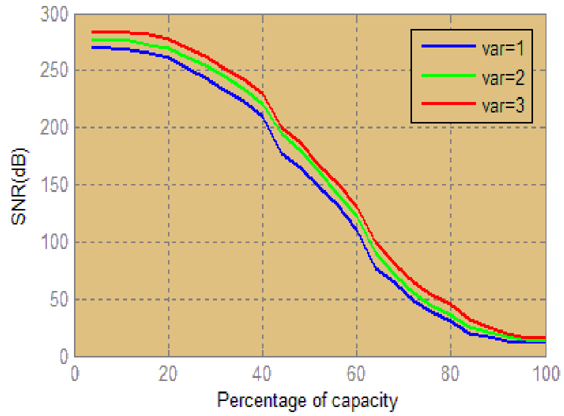

Fig. 6. SNR vs. percentage of channel capacity using the IMAT method for detecting the location and amplitude of the impulsive noise.

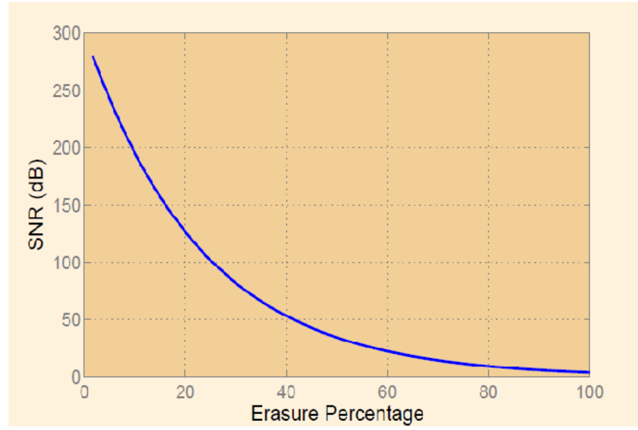

Fig. 7. SNR vs. the percentage of erasure for the convolutional decoder with averaging method after 50 iterations.

6.2 Real numbered convolutional codes results:

The performance of convolutional decoders depends on the coding rate, the number and values of FIR taps for the encoders, and the type of the decoder. Let us take the convolutional encoder of rate $\frac{1}{2}$ of Fig. 3 as our platform for simulations. For simulation results, the taps of the filters in the encoder of Fig. 3 are:

$$\begin{aligned} h1 &= [1\ 2\ 3\ 4\ 5\ 16] \\ h2 &= [16\ 5\ 4\ 3\ 2\ 1] \end{aligned} \tag{11}$$

6.2.1) Decoding for Erasure Channels: For the erasure channels, we employ two methods as described below:

a) Iterations with Averaging: The averaging method to decode for erasures in the convolutional code is shown in Fig. 3. This figure is designed for the rate $\frac{1}{2}$ convolutional encoder. At each stage of decoding, the results of the two branches are averaged. For the rate $\frac{1}{2}$ and specific FIR structure, the SNR improvement versus the relative rate of erasures is shown in Fig.7. This figure shows that the SNR values gradually decrease as the channel erasure rate increases.

b) Decoding Using the Generator Matrix: The generator matrix of a convolutional encoder of the type depicted in Fig. 2 with taps given in (11) can be shown to be:

$$G = \begin{bmatrix} 1 & 16 & 2 & 5 & 3 & 4 & 4 & 3 & 5 & 2 & 16 & 1 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 1 & 16 & 2 & 5 & 3 & 4 & 4 & 3 & 5 & 2 & 16 & 1 & 0 & \cdots \\ 0 & 0 & 0 & 0 & 1 & 16 & 2 & 5 & 3 & 4 & 4 & 3 & 5 & 2 & 16 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 16 & 2 & 5 & 3 & 4 & 4 & 3 & 5 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 16 & 2 & 5 & 3 & 4 & 4 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{12}$$

With a proper choice of the relaxation parameter, the iteration represented in Fig. 1 converges to the actual signal. By using the above operator G in our iterative simulations, better results can be obtained in comparison with the averaging method of Fig. 7. Figure 8 shows that the SNR values gradually decrease as the rate of erasure reaches its maximum (capacity). This figure shows that the generator matrix approach for decoding using the iteration matrix performs much better than the averaging method represented in Figs. 7 and 8. However, the complexity of the matrix approach is higher than the averaging method.

6.2.2) Decoding for Impulsive Noise Channels: For simulation results, we use the generator matrix shown in (12). Its parity check matrix can be calculated from [14] and is given below:

$$H = \begin{bmatrix} -1 & 0.063 & 0 & 0 & 0 & 0 & \cdots \\ -0.313 & 0.125 & -1 & 0.063 & 0 & 0 & \cdots \\ -0.25 & 0.188 & -0.313 & 0.125 & -1 & 0.063 & \cdots \\ -0.188 & 0.25 & -0.25 & 0.188 & -0.313 & .0125 & \cdots \\ -0.125 & 0.313 & -0.188 & 0.25 & -0.25 & 0.188 & \cdots \\ -0.063 & 1 & -0.125 & 0.313 & -0.188 & 0.25 & \cdots \\ 0 & 0 & -0.063 & 1 & -0.125 & 0.313 & \cdots \\ 0 & 0 & 0 & 0 & -0.063 & 1 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}_{60\times 110} \tag{13}$$

In our simulations, the locations of the impulsive noise samples are generated randomly and their amplitudes have Gaussian distributions with zero mean and variance equal to 1, 2, 5 and 10 times the variance of the encoder output. The results are shown in Fig. 9 after 300 iterations. This figure shows that the high variance impulsive noise has a better performance.

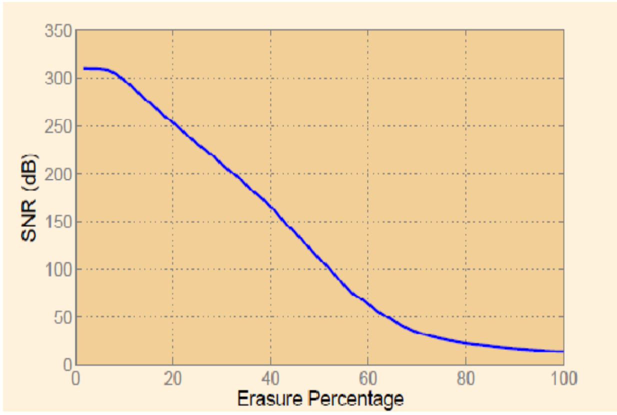

Fig. 8. SNR vs. the relative rate of erasures in an erasure channel applying the iterative method with the generator matrix.

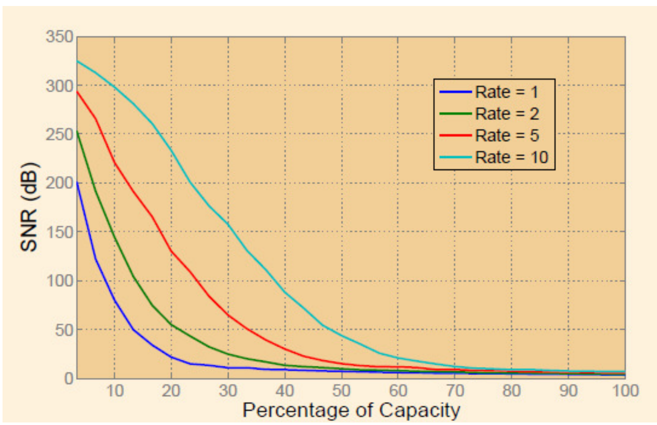

Fig. 9. SNR vs. percentage of channel capacity using the knock out method for detecting the location and amplitude of the impulsive noise, $\lambda = 1.9$.

7. Conclusion

Real numbered block and convolutional codes can be useful in fault tolerant systems. We have developed decoding methods for removing erasure and impulsive noise using various novel algorithms. Two iterative algorithms are introduced to recover the received signal from erasure and impulsive noise channels. A non-linear method (IMAT) to reconstruct the impulsive noise is simulated for both block and convolutional codes. The results are quite impressive.

References

[1] F. Marvasti, M. Hasan, M. Eckhart, and S. Talebi, “Efficient algorithms for burst error recovery using FFT and other transform kernels,” *IEEE Trans. on Signal Proc.*, vol. 47, no. 4, April 1999.

[2] F. Marvasti, “Applications to error correction codes,” in *Nonuniform Sampling: Theory and Practice*, F. Marvasti, Ed. Springer, formerly Kluwer Academic/Plenum Publishers, 2001, pp. 689–738.

[3] J.Wolf, “Redundancy, the discrete Fourier transform, and impulse noise cancellation,” *IEEE Trans. on Comm.*, vol. 31, no. 3, pp. 458–461, Mar 1983.

[4] C. N. Hadjicostis and G. C. Verghese, “Coding approaches to fault tolerance in linear dynamic systems,” *IEEE Trans. on Info. Theory*, vol. 51, no. 1, pp. 210–228, Jan. 2005.

[5] C. J. Anfinson and F. T. Luk, “A linear algebraic model of algorithm based fault tolerance,” *IEEE Trans. on Computers archive*, vol. 37, no. 12, pp. 1599–1604, Dec. 1988.

[6] V. S. S. Nair and J. A. Abraham, “Real-number codes for fault-tolerant matrix operations on processor arrays,” *IEEE Trans. on Computers*, vol. 39, no. 4, pp. 426–435, Apr. 1990.

[7] S. Zahedpour, S. Feizi, A. Amini, and F. Marvasti, “Impulsive noise cancellation based on soft decision and recursion,” *IEEE Trans. On Instrumentation and Measurement*, vol. 67, no. 12, Dec. 2008.

[8] S. Feizi, S. Zahedpour, M. Soltanolkotabi, A. Amini, and F. Marvasti, “Salt and pepper noise removal for images,” *IEEE Proc. on Int. Conf. on Telecommunications ICT’08, St. Petersburg, Russia*, pp. 1–5, June 2008.

[9] T.G. Marshall, Jr. "Coding of Real-Number Sequences for Error Correction: A Digital Signal Processing Problem" *IEEE Journal on selected areas in communications, vol. sac-2, No. 2, March 1984.*

[10] J.K. Wolf, "Analog Codes", IEEE Int. Conf. on Comm. (ICC ’83), Boston, MA, USA, June 19-22, 1983, Vol. 1, pp. 310-12.

[11] Wolf, J.K., “Redundancy, the Discrete Fourier Transform, and Impulse Noise Cancellation", IEEE Trans. on Communications, Vol. COM-31, No. 3, pp. 458-461,March 1983.

[12] F. Marvasti, “An Iterative Method to Compensate for the Interpolation Distortion” , IEEE trans on ASSP

[13] F. Marvasti, A. Amini, F. Haddadi, M. Soltanolkotabi, B. H. Khalaj, A. Aldroubi, S. Holm, S. Sanei, J. Chambers, “A Unified Approach to Sparse Signal Processing,” Feb 2009 http://arxiv.org/abs/0902.1853v1

[14] S. Lin and D. J. Costello, *Error control coding*. Prentice-Hall, Englewood Cliffs, NJ, 1983