

Real-Number DFT Codes for Estimating a Dispersive Channel

Jiun Shiu and Ja-Ling Wu

Abstract—The utilization of real-number DFT codes for channel equalization is studied in this letter. As shown below, through real-number DFT codes, it is possible to deterministically calculate the dispersive parameters of a channel by introducing some redundancies into the transmitting data.

I. INTRODUCTION

IN 1981, Marshall showed the possibility to define error control codes for real or complex data (the so-called real-number codes) [1]. In a later paper [2], he suggested that real-number codes could have applications similar to those of Reed-Solomon codes and could be implemented with increasingly available VLSI processors and digital image processors. Wolf, with a different view, took real-number codes as a new technique for solving signal processing problems such as impulsive noise cancellation in information transmission [3]. Many classes of real-number codes and their corresponding decoding algorithms have been studied; for example, DFT codes [1] and DHT codes [4] with modified error trapping decoding algorithm, DCT codes [5] with the modified Berlekamp-Massey algorithm, and some other unique decoding techniques for real-number codes such as Reed-Solomon voting [3] and rank reduction [6]. A common feature of the above real-number codes is that the channel-error model is assumed to be additive. In this paper, the real-number decoding method for the convolutional channel-error model (which corresponds to the situation of transmitting over a dispersive channel in practice) will be investigated.

II. ENCODING AND DECODING SCHEMES FOR A DISPERSIVE CHANNEL

When a discrete-time signal y_i is transmitted through a communication channel, the received baseband signal r_i is usually modeled as

$$r_i = y_i \otimes e_i + n_i \quad (1)$$

where e_i is the impulse response of the channel, n_i is the background noise, and \otimes is the convolution operation [7, Section 4.1]. The most intuitive way to transform a convolution into an addition is by taking the complex cepstrum [8, Section 10.4]. However, since the signal y_i cannot be assumed finite, one

Manuscript received September 17, 1994. The associate editor coordinating the review of this letter and approving it for publication was Dr. P. Duhammel.

The authors are with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, Republic of China. IEEE Log Number 9415884.

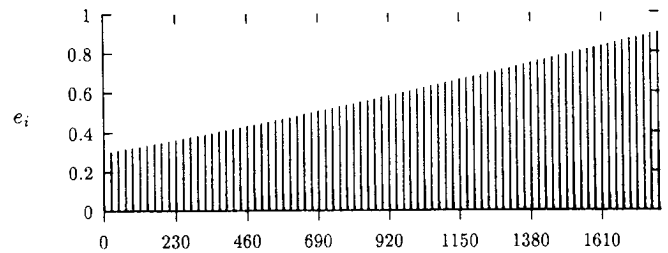


Fig. 1. Impulse response of the simulated channel.

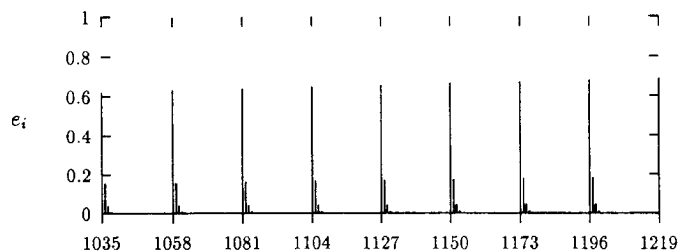


Fig. 2. Enlarged part of the impulse response of the simulated channel.

cannot simply take DFT of (1). In the following derivation, the overlap-and-save technique is used to cope with this difficulty.

Assume that the channel impulse response e_i is causal and decays significantly for $i \geq M$. In addition, suppose that the transmitted signal y_i is encoded such that it can be divided into blocks of $N + L$ symbols with the last L symbols essentially 0 and $L > M$. Under these assumptions, the convolution operation in (1) can be accomplished by block-wise cyclic convolutions of an N -point finite signal and an M -point finite signal. To simplify the notation, these two signals are still denoted as y_i and e_i , respectively. The result of the cyclic convolution will be a finite signal with length at most $N + M + 1$. Therefore, by the overlap-and-save technique, the equivalent convolution is achieved.

Now, we derive an encoding procedure that will produce a proper signal y_i . Suppose that y_i is computed as

$$y_i = \sum_{j=0}^{N+L-1} X_j \cdot e^{-2\pi \frac{ij}{N}} \quad (2)$$

where $X_j = 1$, for $j = 0, 1, \dots, N - K - 1$ are the appended redundancies X_j for $j = N - K, N - K + 1, \dots, N - 1$ are a block of the transmitted data, and X_j , for $j = N, N + 1, \dots, N + L - 1$ are free variables to meet the previously mentioned encoding constraint on Y_i , i.e., $y_i = 0$ for $i = N, N + 1, \dots, N + L - 1$. Let matrix A be the $(N + L)$ -point

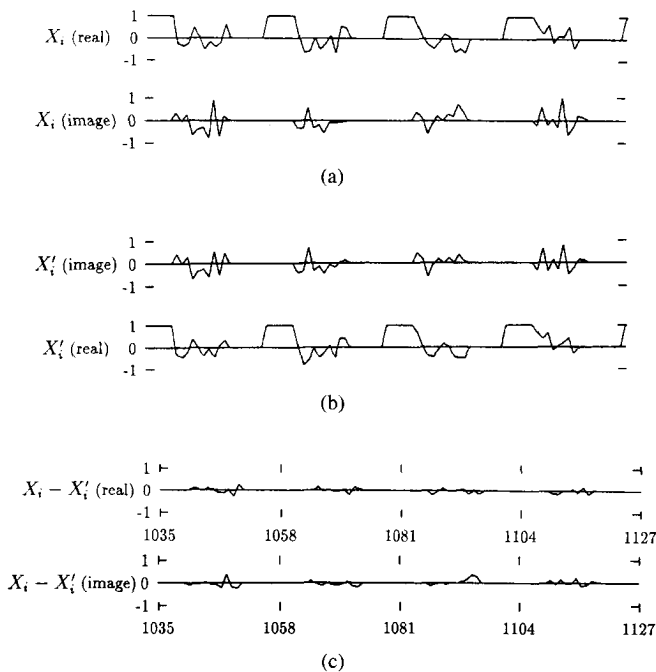


Fig. 3. Enlarged part of the simulation: (a) Randomly generated complex data; (b) estimated data; (c) estimation error.

inverse DFT matrix and

$$A = \begin{bmatrix} A_{N,N} & A_{N,L} \\ A_{L,N} & A_{L,L} \end{bmatrix}. \quad (3)$$

It is easy to derive the following computing procedure of y_i for $i = 0, \dots, N - 1$:

$$[y_0, \dots, y_{N-1}]^T = (A_{N,N} - A_{N,L}A_{L,L}^{-1}A_{L,N}) [1, \dots, 1, X_{N-K}, \dots, X_{N-1}]^T. \quad (4)$$

At the decoder, the received data are first divided into blocks of length $N + L$. Then, the $(N + L)$ -point DFT of each block is computed. Under the above assumptions and the property of cyclic convolution, we get the following equation from (1):

$$\log_c R_j = \log_c Y_j + \log_c E_j + \log_c \left(1 + \frac{N_j}{Y_j \cdot E_j} \right) \quad (5)$$

where R_j , Y_j , E_j , and N_j are, respectively, the $(N + L)$ -point DFT of r_i , y_i , e_i , and N_i . Since we have assumed that e_i decays significantly for $i \geq M$, we could safely assume that \hat{e}_i , which is the cepstrum of e_i or the inverse DFT of $\log_c E_j$, are mostly zero [8, sec. 10.5]. In other words, we are assuming that

$$\log_c E_j = \sum_{l=1}^{\nu} \hat{e}_{il} \cdot e^{-j2\pi \frac{jil}{N+L}} \quad (6)$$

where k_l is the location of a cepstrum, and \hat{e}_{il} is the corresponding magnitude. By (2), the first term $\log_c Y_j$ in (5) will be 0 for $j = 0, \dots, N - K - 1$, which provides the desired syndromes for calculating the unknown locations and magnitudes in (6). These noisy syndromes can be input to one of the readily available decoding algorithms for the DFT codes, such as the modified error trapping decoding algorithm for the cyclic DFT codes [2] or the modified Berlekamp–Massey algorithm [5] for the BCH DFT codes, to compute \hat{e}_i , provided

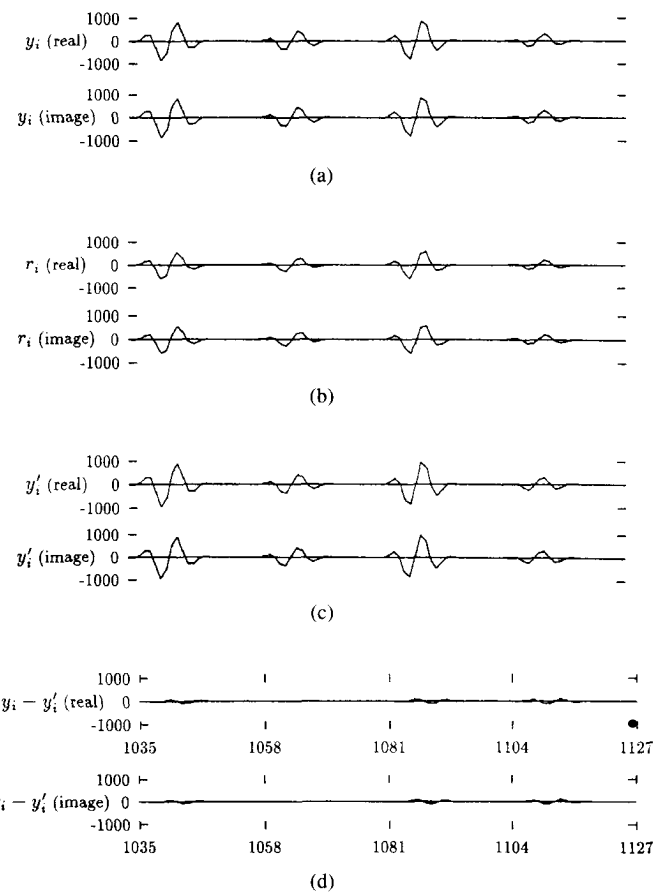


Fig. 4. Enlarged part of the simulation: (a) Codewords; (b) received codewords; (c) estimated codewords; (d) estimation error.

that the number of nonzero terms in (6), i.e., ν , does not exceed the designed correcting capacity. After \hat{e}_i is computed $\log_c E'_j$, which is the estimation of $\log_c E_j$, can be calculated from (6). An estimation of Y_j can then be derived by

$$Y'_j = \frac{R_j}{E'_j} = \frac{Y_j \cdot E_j}{E'_j} + \frac{N_j}{E'_j}.$$

According to (2), X'_j equals Y'_j for $j = N - K, N - K + 1, \dots, N - 1$. In this way, the channel parameters e_i (from E'_j) and, at the same time, the transmitted discrete-time signal X_j can be estimated.

III. A CONCRETE EXAMPLE AND SIMULATION RESULT

In this section, a time-varying channel represented by the following state-variable model

$$\begin{aligned} z(i) &= \alpha(i)z(i-1) + y(i) \\ r(i) &= \beta(i)z(i) + n(i) \end{aligned}$$

was used in the simulation where $\alpha(i) = 0.1 + 0.3 \sin(i/2000)$; $\beta(i) = 0.7 + 0.8 \cos(i/3000) - 0.4 \cos(i/1990)$, and $n(i)$ is normally distributed with mean 0 and variance 0.000001.

Note that this channel is actually a one-pole channel and that the larger $|\alpha(i)|$ is, the longer the associated channel length. The impulse response of the simulated channel is plotted in Fig. 1. An enlarged part is also given in Fig. 2.

It is shown in Fig. 3(a) that the randomly generated complex data are divided into blocks of length 10 and appended by six redundant 1's. These blocks are then encoded according to (4) into codeword blocks of length 23, which is plotted in Fig. 4. These codewords are then transmitted on the simulated channel and are received as shown in Fig. 4(b). Some useful syndromes are calculated and are then input to a modified BCH decoding algorithm that is used to estimate the impulse response of the simulated channel. The two thresholds in this modified decoding algorithm are $TH1 = 0.01$ and $TH2 = 0.1$ [5]. The estimated codewords are plotted in Fig. 4(c), whereas the estimation errors are given in Fig. 4(d). In Fig. 3(b) and (c), the estimated information words and their estimation error, respectively, are plotted. Under this coding scheme, the SNR of the estimation X'_i is 22.743. It is clearly worthwhile to encode a discrete-time signal to combat this kind of channel interference because one will get a transmitted signal with very low SNR if no coding or estimating schemes are applied.

REFERENCES

- [1] T. G. Marshall, Jr. "Real number transform and convolutional codes," in *Proc. 24th Midwest Symp. Circuits Syst.*, S. Karne, Ed., Albuquerque, NM, June 1981, pp. 650–653.
- [2] ———, "Coding of real-number sequences for error correction: A digital signal processing problem," *IEEE J. Selected Areas Commun.*, vol. SAC-2, no. 2, pp. 381–391, Mar. 1984.
- [3] J. K. Wolf, "Redundancy, the discrete fourier transform, and impulse noise cancellation," *IEEE Trans. Commun.*, vol. COM-31, no. 3, pp. 458–461, Mar. 1983.
- [4] J.-L. Wu and J. Shiu, "Discrete Hartley transform in error control coding," *IEEE Trans. Signal Processing*, vol. 39, no. 10, pp. 2356–2359, Oct. 1991.
- [5] ———, "Real-valued error control coding by using DCT," *Proc. Inst. Elec. Eng.*, vol. 139-I, no. 2, pp. 133–139, Apr. 1992.
- [6] R. Kumaresan, "Rank reduction techniques and burst error-correction decoding in real/complex fields," in *Proc. 19th Asilomar Conf. Circuits Syst.*, Pacific Grove, CA, Nov. 1985, pp. 457–461.
- [7] R. E. Blahut, *Digital Transmission of Information*. Reading, MA: Addison-Wesley, 1990.
- [8] A. V. Oppenheim and R. W. Schaffer, *Digital Signal Processing*. Boston: Rainbow-Bridge, 1975.