THE APPLICATION OF WALSH TRANSFORM FOR FORWARD ERROR CORRECTION

M. Hung, F. Marvasti, M.R. Nakhai

Dept of Electronics, King's College London
Strand, London WC2R 2LS
UK

ABSTRACT

In this paper, we present a novel class of forward error correcting codes constructed using the discrete Walsh transform. They are a class of double-error correcting codes defined on the field of real numbers. An iterative decoding algorithm for Walsh transform codes is developed and implemented. The error correcting performance of Walsh transform codes over an AWGN channel is evaluated. Selected Walsh transform code parameters are compared to those of the well-known BCH and RS codes.

1. INTRODUCTION

FFT has been used for error detection and correction of real or complex numbers [1-3]. Efficient techniques for decoding FFT codes can be devised under erasure channels [4]. However, error detection and correction of real numbers are very sensitive to noise [5]; there are methods to get around this sensitivity issue [6-7]. The purpose of this paper is to present a class of forward error correcting codes using Discrete Walsh Transform (DWT) on the fields of real numbers. The advantage of DWT is that its elements are either 1 or –1; hence its transform and inverse transform are the same, and can be implemented very efficiently. Below we shall discuss a single and a double error correcting codes using DWT.

2. ENCODING OF DWT

In order to get a block code of size n, k information samples are chosen such that

$$k = n - \log_2(n) - 1, \qquad (1)$$

where n is a power of 2. To encode these k information samples, n-k zeros are inserted in the code of length n in the following positions:

$$i = 2^m - 1 \text{ for } m = 0,1,\ldots, \log_2(n) \qquad (2)$$

The remaining symbols of the bock of size n consist of the k information samples. The DWT of this block of size n is the transmitted code. For example, for $n=8$, the 0^{th}, 1^{st}, 3^{rd}, and 7^{th} positions are set to zero and in the remaining positions, 4 information samples are inserted. The DWT of size 8 of this block is the (8,4) Walsh code. The position of zeros (2) is chosen for the possibility of detection and correction of single and almost all double errors.

3. DECODING THE WALSH CODES

Like any error correcting codes, the decoding consists of syndrome calculation, detecting the number and position of errors, and finding the magnitude of the errors. We shall discus each one in the following:

Syndrome calculation- If the channel noise is additive, the Walsh transform[1] of the received code will yield the syndrome at the position of zeros (2). Thus the syndrome vector has a length of n-k = $\log_2(n)+1$. If the syndrome vector is all zero, then there is no

[1] Since the Inverse Walsh is equivalent to the Walsh transform divided by n.

error. In case of a single error, the absolute value of all the elements of the syndrome vector are equal to the absolute value of the error magnitude. If the elements of the syndrome vector do not show any pattern as mentioned above, there are two or more errors. The analysis for error detection and correction is as follows:

Let X_n be the transmitted code vector of block size n and Y_n be the received code vector. Since noise is assumed to be additive, we have

$$Y_n = X_n + E_n , \quad (3)$$

where E_n is the error signal. The syndrome is defined by the set of n-k equations:

$$s_i = \sum_{j=0}^{n-1} Y_j * WAL(i, j) = \sum_{j=0}^{n-1} E_j * WAL(i, j)$$

$$i = 2^m - 1 \text{ for } m = 0,1,\ldots, \log_2(n) \quad (4)$$

i is the position of zero as given in (2). If there are no errors, $s_i = 0$, if there is only one error at position p, then the syndrome becomes:

$$s_i = \sum_{j=0}^{n-1} E_j * WAL(i, j) = E_p * WAL(i, p) \quad (5)$$

For all values of i defined in (2).
The above equation implies that, depending on i and p, the syndrome is equal to $\pm E_p$; s_0 is always equal to E_p- the magnitude of the error at position p. A surprisingly simple algorithm can be used to determine the position p. If we normalise the syndrome vector by s_0, and then convert 1's into 0's and –1's into 1's, a binary representation of the syndrome yields the position p. For example, if $n = 8$, the normalised syndrome matrix for a single error at positions $p = 0, 1, \ldots, 7$ is in the following form:

$$S = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{matrix} \quad (6)$$

where the first column of S represents the normalised and converted syndrome vector when p = 0, and the second column represents the syndrome vector when p=1, etc. As it can be seen in (6), the binary representation of each column determines the position p. Once p is known, the single error $s_0 = E_p$ is subtracted from the received code vector Y_n at position p to get the actual transmitted code vector X_n.
If there are two errors, the syndrome will be unique for any pattern of loss provided that the absolute values of the two errors are not the same, i.e., $|E_p| \neq |E_q|$. The syndrome of a DWT code (4,1) is given in Table 1.

This table shows that if the magnitudes of the errors are not equal, the syndrome uniquely represents any pattern of two losses. In general, a systematic algorithm can be used to detect the position of the two errors, p and q; the algorithm is given below:

Algorithm

Initialise error positions p and q.
*/*Loop 1*/*
For m=1 to log(n)-1
Calculate $i=2^m-1$,
If $s_0= s_i$, then no operation,
Else if $s_0= -s_i$, then add ($n/2^m$) to p and q,
Else add ($n/2^m$) to q and exit Loop_1
End
Calculate $c=2^m-1$
If m=log(n), add 1 to q,
Else
*/*Loop 2*/*
For m=(m+1) to log(n)
Calculate $i=2^m-1$,
If $s_0= s_i$ and $s_c \neq s_i$, $_t$ then no operation
Else if $s_0 \neq s_i$ and $s_c = s_i$, add ($n/2^m$) to q,
Else if $s_0 \neq s_i$ and $s_c = - s_i$, add ($n/2^m$) to p,
Else if $s_0 = -s_i$ and $s_c \neq s_i$, add ($n/2^m$) to p and q,
Else declare error positions not found, and exit
End
Return error positions p and q.

4. ERASURE CHANNELS

For erasure channels, the positions of errors are known and there are no ambiguities in case the absolute values of double errors are equal. Therefore, DWT codes can always correct for two erased (lost) samples. Many triple and more erasures may also be corrected depending on *n* and *k* of the *(n,k)* DWT code.

5. PERFORMANCE EVALUATION

The performance of DWT codes is evaluated over AWGN channel using 32ary MFSK modulation with non-coherent detection and hard-decision decoding. The bit error rates with respect to Signal-to-Noise ratio (E_b / N_0) of different DWT codes are shown in Figure 1. The code rates of DWT codes compared to those of double error correcting BCH codes are better as can be seen in Table 2.

6. CONCLUSION

A novel class of forward error correcting codes has been constructed using DWT. These codes can detect and correct all patterns of single errors and almost all patterns of double errors, and definitely all patterns of double erasures. The block size of the DWT codes is $n = 2^m$ for any integer $m \geq 2$, the number of information symbols is $k = n\text{-}m\text{-}1$, and the code rate is $1 - \frac{(m+1)}{2^m}$. The fast DWT is more efficient than FFT since the operations are all real with additions and subtractions only.

7. REFERENCES

[1] R.E. Blahut, "Transform Techniques for error control codes," IBM J of Res. Develop., vol.23, pp. 299-315, May 1979.

[2] K. Wolf, "Redundancy, the discrete Fourier transform, and impulse noise cancellation," IEEE Trans. Commun., vol. COM-31, no.3, March 1983.

[3] F. Marvasti, "FFT as an alternative to error correction codes," IEE Colloquium on DSP Applications in Communication Systems, March 1993.

[4] F Marvasti, M Hasan and M Echhart, "Speech recovery with bursty losses", in Proc of ISCAS'98, Monterey, CA, May31-June 3, 1998.

[5] Wong, F. Marvasti, and W.C. Chambers, "Implementation of recovery of speech with impulsive noise on a DSP chip," IEE Electronic Letters, vol.31, no.17, pp.1412-1413, 17 Aug., 1995.

[6] M Hasan and F Marvasti, "Noise sensitivity analysis for a novel error recovery technique", in Proc of ICT'98, Halkidiki, Greece, 21-25 June 1998.

[7] F. Marvasti, M. Hasan, M. Echhart, S. Talebi, "Efficient techniques for burst error recovery," accepted for publication in the IEEE Trans on SP.