A FAST OFDM-CDMA USER DEMULTIPLEXING ARCHITECTURE

P. Martí-Puig, J. Sala-Álvarez(*)

Department of Signal Theory and Communications
University of Vic, Sagrada Família 7, 08500, Vic (Catalunya)
(*)Polytechnic University of Catalunya, Campus Nord, UPC, Barcelona (Catalunya)

ABSTRACT

A fast algorithm based on a butterfly structure is presented that demultiplexes the symbols of a particular type of MC-MA (Multi-Carrier Multiple-Access) modulation previously proposed for indoor radio communications [1]. A special transform is used to packetize the different symbols on the down-link such that the code sequences associated with the different symbols are transmitted synchronously at the base station. The input to the FFT bins is defined as a symbol-dependent combination of Walsh-Hadamard codes [2][3][4]. We derive a fast architecture that can combine the common redundancy found in the Walsh-Hadamard and the Inverse Fourier Transform. In the direct implementation, the receiver would compute the IFFT (as in OFDM) before the Walsh Hadamard transform. The proposed algorithm evaluates both transforms in a single step. A general expression for the butterfly weights is derived and the savings in computational complexity in terms of the frame length is evaluated.

1. INTRODUCTION

In the last few years we have witnessed a growing interest in developing spectrally efficient multiple access techniques for high bit rate data communications: wide-band mobile communications, wireless LAN's, and twisted-pair line communications. This modulation is required to be resistant to channel effects and amenable to simple implementation. In those cases where the symbols of each user can be transmitted synchronously –i.e. a cellular system base station to mobile users–, the system can use perfectly aligned orthogonal codes. OFDM-CDMA [1]..[4] constitutes one such strategy. The user symbols are grouped in vector $\mathbf{x}$ and combined via the Walsh-Hadamard transform $\mathbf{W}_{(N)}$ into the vector $\mathbf{u}=\mathbf{W}_{(N)}\mathbf{x}$. Therefore, this modulation can be thought of as OFDM on the vector $\mathbf{u}$: $\mathbf{s}=\mathbf{F}_{(N)}\mathbf{u}$. This scheme provides frequency diversity in the transmission of each different symbol in frequency selective channels.

In practice, OFDM and OFDM-CDMA systems, use a circular prefix greater than the channel impulse response length with the objective of minimizing the channel effect and mitigate ISI (Inter Symbol Interference). Some symbols can be used as pilots to aid the receiver in channel estimation, timing and frequency synchronization. OFDM-CDMA systems prove advantageous, but also specially sensitive to frequency synchronization errors as reported in [5][6].

A fast butterfly structure algorithm has been reported [7] for the calculation of the FFT/IFFT in terms of the Walsh-Hadamard Transform. In this paper we present a fast algorithm to accomplish the joint $\mathbf{A}_{(N)}=\mathbf{F}_{(N)}^H\mathbf{W}_{(N)}$ transformation directly.

The signal model and the mathematical properties used in this paper are presented in section number two: Algorithm Derivation. It is shown that $\mathbf{A}_{(N)}$, save a permutation matrix, displays a block diagonal structure. All blocks share a common structure, so that the effort is centered on developing an efficient way to perform its calculation. In section number three: Butterfly Structure, we show that a single butterfly only needs one complex multiplication and three complex additions. The inter-block connection paths are also defined. Computational cost analysis and comparisons are provided in section number four.

2. ALGORITHM DERIVATION

2.1 Signal model

The bit set $\mathbf{x} = [x_1, \dots, x_M]^T$ generated for the N users is combined using N orthogonal Walsh-Hadamard codes of length N to form the resulting symbols $\mathbf{u} = [u_1, \dots, u_M]^T = \mathbf{W}_{(N)}\mathbf{x}$. Each Walsh-Hadamard code appears as a column in matrix $\mathbf{W}_{(N)}$. The resulting symbols $\mathbf{u}$, $\mathbf{u} = \mathbf{W}_{(N)}\mathbf{x}$, are modulated into N sub-carriers via IDFT, so that base-band signal $\mathbf{s}$ at the base station can be written as:

$$1. \quad \mathbf{s} = \mathbf{F}_{(N)}^H \mathbf{W}_{(N)} \mathbf{x}$$

The samples in $\mathbf{s}$ are parallel to series converted, a cyclic prefix is added and the resulting signal is up-converted and transmitted to the receiver.

2.2 Mathematical Properties

The following criteria have been used for the sake of simplicity:

- All matrices are square and referred to in uppercase. Their size is identified as a bracketed subscript: i.e. $\mathbf{F}_{(N)}$ will stand for the $N \times N$ Discrete Fourier Transform.
- Blanks in the block structure of matrices are assumed to contain null matrices of a suitable dimension.

The unitary 2×2 Walsh-Hadamard matrix is defined as:

$$2. \quad \mathbf{W}_{(2)} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

The recursion used for the Walsh-Hadamard matrix $\mathbf{W}_{(N)}$ is the following [8][9]:

$$3. \quad W_{(N)} = \frac{1}{\sqrt{2}} \begin{bmatrix} I_{\left(\frac{N}{2}\right)} & I_{\left(\frac{N}{2}\right)} \\ I_{\left(\frac{N}{2}\right)} & -I_{\left(\frac{N}{2}\right)} \end{bmatrix} \begin{bmatrix} W_{\left(\frac{N}{2}\right)} \\ W_{\left(\frac{N}{2}\right)} \end{bmatrix}$$

Here $I_{(N/2)}$ stands for the $N/2 \times N/2$ identity matrix. The $W_{(N)}$ matrices are real.

In a way similar to (3), the known recursion properties of the unitary inverse Fourier transform are shown to be:

$$4. \quad F_{(N)}^H = \frac{1}{\sqrt{2}} P_{(N)}^H \begin{bmatrix} F_{\left(\frac{N}{2}\right)}^H & \\ & F_{\left(\frac{N}{2}\right)}^H D_{\left(\frac{N}{2}\right)}^H \end{bmatrix} \begin{bmatrix} I_{\left(\frac{N}{2}\right)} & I_{\left(\frac{N}{2}\right)} \\ I_{\left(\frac{N}{2}\right)} & -I_{\left(\frac{N}{2}\right)} \end{bmatrix}$$

$F_{(N)}^H$ is the unitary $N \times N$ inverse Fourier matrix and $F_{(N/2)}^H$ is the $N/2 \times N/2$ unitary inverse Fourier matrix. $P_{(N)}$ is the $N \times N$ row even-odd permutation matrix. Every row of matrix $P_{(N)}$ is formed by all zero elements but one. The position of the one is:

$$5. \quad P_{i,j} = 1 \quad \begin{matrix} j = \text{even} - \text{odd}(i) \\ i = 1..N \end{matrix}$$

As $P_{(N)}$ is real its Hermitian matrix, $P_{(N)}^H$ is its transpose. $I_{(N/2)}$ is the identity matrix of size $N \times N$. $D_{(N/2)}^H$ is a diagonal matrix.

A general matrix $D_{(N)}^H$ can be defined as follows:

$$6. \quad D_{(N)}^H = \text{Diag} \left\{ e^{j \frac{\pi}{N} k} \right\} \quad k = 0, 1, \dots, N-1$$

Where $D_1^H = 1$. Here, it is interesting to observe that the matrix $D_{(N)}^H$ can be written using a new diagonal matrix $\Phi_{(N/2)}^H$ of smaller dimension. The recursion is:

$$7. \quad D_{(N)}^H = \begin{bmatrix} \Phi_{\left(\frac{N}{2}\right)}^H & \\ & e^{j \frac{\pi}{2}} \Phi_{\left(\frac{N}{2}\right)}^H \end{bmatrix}$$

Finally, it is trivial to show that matrices $D_{(N)}^H$ and $\Phi_{(N)}^H$ are related in the following fashion:

$$8. \quad D_{(N)}^H = \left(\Phi_{(N)}^H \right)^2$$

2.3 Block diagonal structure

From the relationship (3) and (4) we can observe that, save a permutation matrix $P_{(N)}^H$, the matrix $F_{(N)}^H W_{(N)}$ has a diagonal block structure [10]. That is:

$$9. \quad F_{(N)}^H W_{(N)} = P_{(N)}^H \begin{bmatrix} F_{\left(\frac{N}{2}\right)}^H W_{\left(\frac{N}{2}\right)} & \\ & F_{\left(\frac{N}{2}\right)}^H D_{\left(\frac{N}{2}\right)}^H W_{\left(\frac{N}{2}\right)} \end{bmatrix}$$

The upper left block resolution from the iteration has the same form that the original one. So it admits an iteration of the same

type. This process can be repeated until the block diagonal structure is completed. The general expression of these blocks is:

$$10. \quad F_{(N)}^H D_{(N)}^H W_{(N)}$$

A new permutation matrix appears each time an iteration is accomplished. $Q_{(N)}^H$ is the product of all those permutation matrices as we can see in expression (11). Its effect is a reordering of the vector elements.

$$11. \quad Q_{(N)}^H = P_{(N)}^H \begin{bmatrix} P_{\left(\frac{N}{2}\right)}^H & \\ & I_{\left(\frac{N}{2}\right)}^H \end{bmatrix} \begin{bmatrix} P_{\left(\frac{N}{4}\right)}^H & \\ & I_{\left(\frac{3N}{4}\right)}^H \end{bmatrix} \dots I_{(N)}$$

Then, the block diagonal structure of $Q_{(N)}^H F_{(N)}^H W_{(N)}$ has two 1×1 upper blocks. The next blocks have dimensions 2×2 , 4×4 , and so on up to $N/2 \times N/2$.

2.4 Blocks Displaying a Butterfly Structure

We begin with the general expression of each block. Let us consider a general block of size N :

$$12. \quad F_{(N)}^H D_{(N)}^H W_{(N)}$$

Using the relationships (3), (4) and (6), reordering in the appropriate way the matrix products and using (13), we get:

$$13. \quad \begin{bmatrix} (1 + e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} & (1 - e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} \\ (1 - e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} & (1 + e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} \end{bmatrix} \begin{bmatrix} W_{\left(\frac{N}{2}\right)} \\ W_{\left(\frac{N}{2}\right)} \end{bmatrix} = \begin{bmatrix} W_{\left(\frac{N}{2}\right)} & \\ & W_{\left(\frac{N}{2}\right)} \end{bmatrix} \begin{bmatrix} (1 + e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} & (1 - e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} \\ (1 - e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} & (1 + e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} \end{bmatrix}$$

We arrive at the following expression at the first iteration:

$$14. \quad F_{(N)}^H D_{(N)}^H W_{(N)} = P_{(N)}^H \begin{bmatrix} F_{\left(\frac{N}{2}\right)}^H \Phi_{\left(\frac{N}{2}\right)}^H W_{\left(\frac{N}{2}\right)} & \\ & F_{\left(\frac{N}{2}\right)}^H D_{\left(\frac{N}{2}\right)}^H \Phi_{\left(\frac{N}{2}\right)}^H W_{\left(\frac{N}{2}\right)} \end{bmatrix} \times \begin{bmatrix} (1 + e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} & (1 - e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} \\ (1 - e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} & (1 + e^{j \frac{\pi}{2}}) I_{\left(\frac{N}{2}\right)} \end{bmatrix}$$

Here, the problem has been reduced to a permutation matrix $P_{(N)}^H$, a two-block diagonal matrix with its blocks having the same structure than the block they preceded and a butterfly matrix (including factor $1/2$). The butterfly matrix uses only two elements from the input vector to calculate two elements to the output vector. The $P_{(N)}^H$ matrix multiplication does not need any

addition or multiplication operation, it is a reordering of the output vector

We have reduced the problem of computing one $N \times N$ block into the problem of computing two $N/2 \times N/2$ blocks as we can see in (15) and (16). Hence, new and successively smaller blocks, can be always written into the structure, using the same recursion in terms of an inverse Fourier matrix, a diagonal matrix and a Walsh-Hadamard matrix.

The diagonal matrix constitutes the difference between the two new blocks. This diagonal matrix will determine the butterfly weights in the following iteration step. With the objective of identifying the butterfly matrix and consequently, the weights for each stage, it is necessary to identify the way the diagonal matrix is transformed in the iteration process. To do this, it is necessary to use the relationship in (8) to write these new blocks as:

$$15. F_{\left(\frac{N}{2}\right)}^H \Phi_{\left(\frac{N}{2}\right)}^H W_{\left(\frac{N}{2}\right)}^H = F_{\left(\frac{N}{2}\right)}^H \left(D_{\left(\frac{N}{2}\right)}^H \right)^{\frac{1}{2}} W_{\left(\frac{N}{2}\right)}^H$$

$$16. F_{\left(\frac{N}{2}\right)}^H D_{\left(\frac{N}{2}\right)}^H \Phi_{\left(\frac{N}{2}\right)}^H W_{\left(\frac{N}{2}\right)}^H = F_{\left(\frac{N}{2}\right)}^H \left(D_{\left(\frac{N}{2}\right)}^H \right)^{\frac{3}{2}} W_{\left(\frac{N}{2}\right)}^H$$

When the same process is repeated, using expressions (3), (4), (8) and (13) on each new block, the full decomposition is completed. The set of permutation matrices P_0^H accumulating on the left-hand product is denoted in the form:

$$17. R_{1,N} = P_{1,N}^H \left[P_{\left(\frac{N}{2}\right)}^H \left[P_{\left(\frac{N}{4}\right)}^H \left[P_{\left(\frac{N}{4}\right)}^H \left[P_{\left(\frac{N}{4}\right)}^H \right] \right] \right] \right] \cdot I_{1,N}$$

$R_{1,N}$ is a product of permutation matrices, and hence, a permutation matrix itself. $R_{1,N}$ reduces, in this particular case, to a bit-reversed reordering. The same result is obtained for the remaining blocks of smaller dimension.

Let us write the general butterfly matrix $B_{m,k(N)}(\alpha_{m,k})$ as:

$$18. B_{m,k(N)}(\alpha_{m,k}) = \frac{1}{2} \begin{bmatrix} (1 + \alpha_{m,k}) I_{\left(\frac{N}{2}\right)} & (1 - \alpha_{m,k}) I_{\left(\frac{N}{2}\right)} \\ (1 - \alpha_{m,k}) I_{\left(\frac{N}{2}\right)} & (1 + \alpha_{m,k}) I_{\left(\frac{N}{2}\right)} \end{bmatrix}$$

The subscript m indicates which stage each matrix $B_{m,n}$ belongs to and subscript n its order within this stage. We denote by $\alpha_{m,n}$ a weight of the butterfly, so that it depends on the corresponding stage and the place where this matrix is found inside the stage. Note that all butterflies within a butterfly matrix have the same weight. We can see this in expression (18) and (20). A matrix of butterflies is univocally defined with its size and its weight. The general weight expression is:

$$19. \alpha_{m,k} = e^{j(2^m) \frac{\pi}{N} (2k'+1)} \quad \begin{matrix} m=0,1,\dots, \text{LOG}_2(N)-1 \\ k' = \text{bitreversed}(k=0,1,\dots, \frac{N}{2^m}-1) \end{matrix}$$

Where m denotes the stage and k the order inside the stage matrix, as we can see in next example. We consider the transform $F_{(16)}^H W_{(16)}$. Its larger block of size $N/2 \times N/2$ is reduced as follows:

$$20. F_{(8)}^H D_{(8)}^H W_{(8)} = R_{(8)} \begin{bmatrix} B_{1,0(2)} \left(e^{j\frac{\pi}{8}} \right) & & & \\ & B_{1,1(2)} \left(e^{j\frac{5\pi}{8}} \right) & & \\ & & B_{1,2(2)} \left(e^{j\frac{9\pi}{8}} \right) & \\ & & & B_{1,3(2)} \left(e^{j\frac{13\pi}{8}} \right) \end{bmatrix} \times \begin{bmatrix} B_{2,0(4)} \left(e^{j\frac{\pi}{4}} \right) & & \\ & B_{2,1(4)} \left(e^{j\frac{3\pi}{4}} \right) & \\ & & B_{2,2(4)} \left(e^{j\frac{5\pi}{4}} \right) \end{bmatrix}$$

3. BUTTERFLY STRUCTURE

The butterfly operation relates two elements of the entry stage with two elements of the exit stage. For a minimal block size of 2×2 , a single butterfly can be written in matrix form as:

$$21. \begin{bmatrix} o_1 \\ o_2 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 + \alpha_{mk} & 1 - \alpha_{mn} \\ 1 - \alpha_{mn} & 1 + \alpha_{mn} \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix}$$

The graphical representation of the operation (21) can be seen in figure 1.

Figure 1. Block diagram of a single butterfly. The entry arrows connect i_1 and i_2 of the input stage with o_1 and o_2 of the following stage through the output arrows.

Figure 2. Block diagram of a single butterfly. Only three complex additions and one complex multiplication are necessary. It is easy to show that the value of the multiplication is related to the weight of the butterfly according to expression: $\Omega_{m,k} = 0.5(1 - \alpha_{mk})$

The same operation (21) can be accomplished with a minimum of one complex multiplication and three complex additions such as it is also observed in [9].

Figure 3.Depiction the interconnection of the butterflies in each stage in a 16x16 transformation. The result reordering imposed by the permutation matrices is not shown in the picture. The diagonal block structure can be observed: a block of 8x8 (three stages), a block 4x4 (2 stages), a block 2x2 (a stage).

From figure 2 and using the relationship in (19) it is immediate to obtain a general expression for the butterfly weight $\Omega_{m,k}$:

$$22. \quad \Omega_{m,k} = \frac{1}{2} \left(1 - e^{j(2^m) \frac{\pi}{N} (2k'+1)} \right) \quad \begin{matrix} m = 0, 1, \dots, \text{LOG}_2(N) - 1 \\ k' = \text{bitrev}(k = 0, 1, \dots, \frac{N}{2^m} - 1) \end{matrix}$$

4. SUMMARY

We have seen in section number three that a simple butterfly can be made with a complex multiplication and three complex additions. The studied transform accomplishes full block decomposition. These blocks can be calculated separately. Using the presented technique, the order of operations for the calculation of a block of size $N_b \times N_b$ is proportional to $N_b \text{LOG}_2(N_b)$. Once the block has been computed a bit reversed ordering is applied and no extra operation must be effectuated. As the size of the blocks is different in each block the calculation of smaller blocks take much less computing cost. Once all the blocks have been calculated, a $\mathbf{Q}_{(N)}^H$ reordering must be applied.

The permutation operations can be done with appropriate operand addressing. The mathematical operations needed to calculate all blocks, for an input vector of length N , are:

$$23. \quad \begin{aligned} N_{\text{mult}} &= \frac{1}{2} [N \log_2(N) - 2(N - 2)] \\ N_{\text{add}} &= \frac{3}{2} [N \log_2(N) - 2(N - 2)] \end{aligned}$$

Where N_{mult} is the number of multiplications and N_{add} it is the number of additions. The fact of needing less multiplications than additions proves advantageous in bit-level implementations of the algorithm.

This algorithm outperforms the concatenation of the Fourier and the Walsh-Hadamard Transforms in terms of computational complexity. This second option, for a input vector of length N , requires $0.5N \text{LOG}_2(N)$ multiplications and $3N \text{LOG}_2(N)$ additions. To obtain this number we take the operations to perform an efficient IFFT algorithm ($0.5N \text{LOG}_2(N)$ multiplications and $2N \text{LOG}_2(N)$ additions) and we add the number of additions to perform the fast Walsh-Hadamard algorithm ($N \text{LOG}_2(N)$). No reordering operations are considered.

$$24. \quad \begin{aligned} N_{\text{mult}} &= \frac{1}{2} N \log_2(N) \\ N_{\text{add}} &= 3N \log_2(N) \end{aligned}$$

This algorithm works with input vectors whose size is a power of two and we can find a similar algorithm to develop the inverse transform $\mathbf{A}_{(N)}^{-1} = \mathbf{A}_{(N)}^H = \mathbf{W}_{(N)}^H \mathbf{F}_{(N)}$.

5. REFERENCES

- [1] Yee N., Linnartz J.-P., Fettweis G. "Multi-Carrier CDMA in Indoor Wireless Radio Networks". In *Proc. IEICE Transaction on Communications*, volume E77-B7, No. 7, Japan, 1996.
- [2] Kaiser S. "On the Performance of Different Detection Techniques for OFDM-CDMA in Fading Channels". In *Proc. IEEE Global Telecommunication Conference (GLOBECOM)*, pages 2059-2063, Singapore, November, 1995.
- [3] Kaiser S. "OFDM-CDMA versus DS-CDMA: performance evaluation for fading channels". In *Proc. ICC'95 Seattle*, pp. 1722-1726, June 1995.
- [4] Fazel K. "Performance of CDMA/OFDM for mobile communication system". In *Proc. ICUPC, Ottawa* 1993.
- [5] Tomba L., Krzymien W.A. "Effect of carrier phase noise and frequency offset on the performance of multicarrier CDMA systems". In *Proc. IEEE Int. Conf. Commun.* 1996.
- [6] Pollet T., Van Bladel M., Moeneclaey M. "BER sensitivity of OFDM systems to carrier frequency offset and wiener phase noise". *IEEE Trans. Commun.*, Vol.43, N° 2/3/4 pp.191/193, Feb./Mar./Apr. 1995.
- [7] S. Boussakta and A.G.J. Holt. *Fast Algorithm for Calculation of Both Walsh-Hadamard and Fourier Transforms* Electronics Letters Vol. 25 No. 20 September 89
- [8] Ray Liu K. J. "A Simple and Unified Proof of Dyadic Shift Invariance and the Extension to Cyclic Shift Invariance". *Transactions on education*, Vol. 36, N° 4, November 1993.
- [9] Falkowski B. J. and Rahardja S. "Properties and Applications of Unified Complex Hadamard Transforms". *IEEE*, 1997.
- [10] Ja-Ling Wu. "Block diagonal structure in discrete transforms". *IEE Proc.*, Vol. 136, Pt. E, N° 4, July 1989.