EFFICIENT ALGORITHMS FOR MODULATION AND DEMODULATION IN OFDM-SYSTEMS

Anders Vahlin
Nera Networks, P.O. Box 7090, 5020 Bergen, Norway, ava@networks.nera.no

ABSTRACT

This paper presents algorithms for efficient implementation of modulator and demodulator in OFDM-systems with offset-QAM and pulse shaping in the subchannels. An algorithm for modulation and demodulation with one sample per symbol output are presented. The algorithm has a complexity comparable to previously presented algorithms but the structure is considerably different. The paper also presents a demodulation algorithm which generates two samples per symbol output without doubling the complexity compared to one sample per symbol.

1. INTRODUCTION

Orthogonal Frequency Division Multiplex (OFDM), also known as Multicarrier Modulation (MCM), has received an increased attention in the past decade. This is due to attractive features such as high bandwidth efficiency and simple equalisation of linear channel distortion [3].

There are two main classes of OFDM systems. The most common is the one using QAM and rectangular baseband pulse shaping. This method is attractive because linear channel distortion can easily be handled by using a guard interval between the OFDM symbols. The other class uses offset-QAM and more general baseband pulse shaping, as in e.g. [6]. With this method, a higher spectral containment can be achieved compared with a system using rectangular pulse shaping.

A direct implementation with individual modulation and demodulation of each sub-channel in the OFDM system will in most applications give a prohibitive complexity. For OFDM systems with rectangular pulse shaping and QAM modulation, the use of FFT to get an efficient implementation is straightforward [7]. With more general pulse shaping and offset-QAM, the situation is a bit more complicated. Algorithms for modulation and demodulation in this case are presented in [2, 5, 8]. The modulation algorithms in [2] and [8] generate a real passband signal but it can be modified to generate a complex baseband signal. With the algorithm in [2], the number of active carriers for a given number of samples per symbol interval is limited. The algorithm in [8] is based on an approximation that is valid when the number of carriers is large [9]. The modulation algorithm in [5] generates a complex baseband signal and has full flexibility in the number of active channels.

This paper presents a new algorithm for modulation and demodulation with complex baseband signals and full flexibility on the number of active channels. This algorithm and the one in [5] have comparable complexity but the structures are considerably different. The availability of different structures can be important when partitioning the algorithm in very high speed applications.

The new algorithm, and those in [5], all have demodulation with one output sample per symbol (baud spaced output). This paper presents an algorithm for demodulation that generates two samples per symbol (fractionally spaced output) without doubling the complexity compared to the demodulator with one sample per symbol. This is an important feature when using fractionally spaced equalisers to compensate for channel distortion.

The paper is organised as follows. In Section 2, the modulator for OFDM with offset-QAM and pulse shaping is described and two algorithms for efficient implementation are presented. The demodulator with one sample per symbol (baud spaced) output and corresponding efficient algorithms are presented in Section 3. Section 4 describes the demodulator with two samples per symbol (fractionally spaced) output and its algorithms for efficient implementation. The computational complexity of the different algorithms is derived in Section 5. Finally, the conclusions are presented in Section 6.

2. MODULATOR

Figure 1 shows a block diagram of the signal processing functions in the modulator of an offset-QAM OFDM system. In each channel, the real and imaginary parts of the complex symbols \( c_{k,n} \), are filtered by separate pulse shaping filters \( h(m) \) and \( h(m + \frac{N}{2}) \), respectively. The two branches are added and shifted to the desired frequency. All channels are then added to form the complex baseband signal. Figure 1 illustrates the function of the modulator but the structure is not suitable for direct implementation (except for a very small number of channels) as the complexity would be intolerable. The modulator structure presented in this paper has the same function but a considerably reduced complexity.
The output of the modulator is given by

\[
x(m) = \sum_{k=0}^{\infty} \sum_{n=0}^{N-1} [a_{k,n} e^{j2 \pi (m + \frac{N}{2}) n} h(m - kN) + j b_{k,n} h(m - kN + \frac{N}{2})] e^{j \frac{2 \pi}{N} m (m + \frac{N}{2})}
\]

where \( c_{k,n} = a_{k,n} + j b_{k,n} \). \( N \) is the number of samples per symbol interval and also the maximum number of channels. The symbols in some of the channels will be set to zero in a practical system to obtain a certain oversampling of the output of the modulator. The expression in (1) can be manipulated to obtain different efficient structures as shown below.

**Modulator with two N/2-point FFT**

Splitting the sum over \( n \), (1) can be rewritten as

\[
x(m) = \sum_{k=0}^{\infty} \sum_{n=0}^{N-1} [a_{k,n} e^{j2 \pi (m + \frac{N}{2}) n} h(m - kN) + j b_{k,n} h(m - kN + \frac{N}{2})] e^{j \frac{2 \pi}{N} m (m + \frac{N}{2})}
\]

The two sums over \( n \) can now be identified as N-point Inverse Fast Fourier Transforms (IFFT) which gives

\[
x(m) = \sum_{k=0}^{\infty} A_k(m + \frac{N}{2}) h(m - kN) + j B_k(m + \frac{N}{2}) h(m - kN + \frac{N}{2})
\]

where \( A_k(m) = IFFT(a_{k,n}) \) and \( B_k(m) = IFFT(b_{k,n}) \).

When computing \( A_k(m) \) and \( B_k(m) \) in (3), instead of using one N-point IFFT for both, it is possible to use one N/2-point IFFT for each [1]. To see that, first define

\[
Z_k(m) = IFFT(a_{k,2n} + ja_{k,2n+1})
\]

where the IFFT is of size N/2. Then let

\[
U_k(m) = \frac{1}{2}(Z_k(m) + Z_k(N-m))
\]

\[
V_k(m) = \frac{1}{2j}(Z_k(m) - Z_k(N-m))
\]

The first half of the IFFT of \( a_{k,n} \) can then be computed as

\[
A_k(m) = U_k(m) + e^{j \frac{\pi}{N} m} V_k(m),
\]

\( m = 0...\frac{N}{2} - 1 \)

and the second half as

\[
A_k(m + \frac{N}{2}) = U_k(m) - e^{j \frac{\pi}{N} m} V_k(m),
\]

\( m = 0...\frac{N}{2} - 1 \)

Exactly the same method can be used for computation of \( B_k(m) \).

Figure 2 shows the block diagram of this implementation with two N/2-point IFFT and some additional multiplications.

**3. BAUD SPACED DEMODULATOR**

Figure 3 shows a block diagram of the signal processing functions in the baud spaced demodulator. In each channel, the received signal is first shifted to get the desired signal at baseband and then filtered with separate receiver filters for real and imaginary branches. The signal is finally decimated to get one sample per symbol.
The real part of the received symbol in channel $n$ can be written

$$a_{k,n} = \text{Re} \left\{ \sum_m h(m) x(kN - m)e^{j(\frac{2\pi m}{N} - \frac{\pi}{2})} \right\}$$  \hspace{1cm} (9)$$

and correspondingly for the imaginary part

$$b_{k,n} = \text{Im} \left\{ \sum_m h(m - \frac{N}{2}) x(kN - m)e^{j(\frac{2\pi m}{N} - \frac{\pi}{2})} \right\}$$  \hspace{1cm} (10)$$

### Demodulator with N/2-point FFT

Consider the real part of the received symbols defined in (9) The symbols with even channel numbers can be written as

$$a_{k,2n} = \frac{1}{2} \sum_{m=0}^{N/2-1} \left( A_k(m) + A_k(m + \frac{N}{2}) \right) + \sum_{m=0}^{N/2-1} A_k^*(\frac{N}{2} - m) + A_k^*(N - m)e^{j2\pi m/2}$$  \hspace{1cm} (11)$$

and those with odd channel numbers as

$$a_{k,2n+1} = \frac{1}{2} \sum_{m=0}^{N/2-1} \left( A_k(m) - A_k(m + \frac{N}{2}) \right) - \sum_{m=0}^{N/2-1} A_k^*(\frac{N}{2} - m) + A_k^*(N - m)e^{j2\pi m/2}$$  \hspace{1cm} (12)$$

where $A_k(m)$ is defined as

$$A_k(m) = \sum_{m'} h(m + \frac{N}{4} + m'N) x(kN - m - \frac{N}{4} - m'N)$$  \hspace{1cm} (13)$$

Now define $\tilde{A}_k(m)$ as

$$\tilde{A}_k(m) = \frac{1}{2} \left( A_k(m) + A_k(m + \frac{N}{2}) \right) + \frac{1}{2} \left( A_k^*(\frac{N}{2} - m) + A_k^*(N - m) \right) - \sum_{m'=0}^{N/2-1} h(m - \frac{N}{4} + m'N) x(kN - m - \frac{N}{4} - m'N)$$  \hspace{1cm} (14)$$

The symbols with even channel numbers are then obtained from the real outputs of a size $N/2$ IFFT as

$$a_{k,2n} = \text{Re} \left\{ \text{IFFT} \left( \tilde{A}_k(m) \right) \right\}$$  \hspace{1cm} (15)$$

and those with odd channel numbers from the imaginary outputs of the same IFFT as

$$a_{k,2n+1} = \text{Im} \left\{ \text{IFFT} \left( \tilde{A}_k(m) \right) \right\}$$  \hspace{1cm} (16)$$

Now consider the imaginary part of the received symbols given by (10). The symbols with even channel numbers can be written as

$$b_{k,2n} = \frac{1}{2j} \sum_{m=0}^{N/2-1} \left( B_k(m) + B_k(m + \frac{N}{2}) \right) - \sum_{m=0}^{N/2-1} B_k^*(\frac{N}{2} - m) - B_k^*(N - m)e^{j\frac{2\pi m}{N/2}}$$  \hspace{1cm} (17)$$

and those with odd channel numbers as

$$b_{k,2n+1} = \frac{1}{2j} \sum_{m=0}^{N/2-1} \left( B_k(m) - B_k(m + \frac{N}{2}) \right) + \sum_{m=0}^{N/2-1} B_k^*(\frac{N}{2} - m) - B_k^*(N - m)e^{j\frac{2\pi m}{N/2}}$$  \hspace{1cm} (18)$$

where $B_k(m)$ is defined as

$$B_k(m) = \sum_{m'} h(m - \frac{N}{4} + m'N) x(kN - m - \frac{N}{4} - m'N)$$  \hspace{1cm} (19)$$

Define $\tilde{B}_k(m)$ as
\[ \tilde{B}_k(m) = \frac{1}{2} j \left\{ B_k(m) + B_k(m + \frac{N}{2}) - B_k^*(\frac{N}{2} - m) - B_k^*(N - m) \right\} + \frac{1}{2} \left\{ B_k(m) - B_k(m + \frac{N}{2}) + B_k^*(\frac{N}{2} - m) - B_k^*(N - m) \right\} e^{\frac{2\pi m}{N}} \]

(20)

The symbols with even channel numbers are then obtained from the real outputs of a size $N/2$ IFFT as

\[ b_{k,2n} = \text{Re}\left\{ \text{IFFT}(\tilde{B}_k(m)) \right\} \]

(21)

and those with odd channel numbers from the imaginary outputs of the same IFFT as

\[ b_{k,2n+1} = \text{Im}\left\{ \text{IFFT}(\tilde{B}_k(m)) \right\} \]

(22)

Figure 4 shows the block diagram of the demodulator based on two $N/2$-point IFFTs.

![Block diagram of demodulator with two $N/2$-point IFFTs.]

The output of channel $n$ of the demodulator is given by

\[ z_n(i) = \sum_m h(m) x(i \frac{N}{2} - m) e^{(\frac{j \pi}{N} m - \frac{j}{2}) n} \]

(23)

**4. FRACTIONALLY SPACED DEMODULATOR**

A receiver structure with separate equalisers after the demodulator requires more than one sample per symbol interval. Figure 5 shows a block diagram of a demodulator with two samples per symbol.

![Block diagram of fractionally spaced demodulator.]

The output of channel $n$ of the demodulator is given by

\[ z_{2n}(i) = \sum_{m=0}^{N/2-1} \sum_{m'} h(m + m' \frac{N}{2}) x(i \frac{N}{2} - m - m' \frac{N}{2}) e^{(\frac{j \pi}{N} m - \frac{j}{2}) n} \]

(24)

This can be computed with an $N/2$-point IFFT as

\[ z_{2n}(i) = \text{IFFT}(C_1(m) + C_2(m)) \]

(25)

where

\[ C_1(m) = \sum_{m'} h(m + m' \frac{N}{2}) x(i \frac{N}{2} - m - m' \frac{N}{2}) \]

(26)

and

\[ C_2(m) = \sum_{m'} h(m + m' N - \frac{N}{4}) x(i \frac{N}{2} - m - m' N + \frac{N}{4}) \]

(27)

The output of channels with odd numbers are given by
This can also be efficiently computed with an N/2-point IFFT as

\[ z_{2n+1}(i) = \sum_{m=0}^{N/2-1} (C_1(m) - C_2(m)) e^{-j\pi m} e^{j\pi m} \]  

where \( C_1(m) \) and \( C_2(m) \) are given by (26) and (27) respectively.

Figure 6 shows the block diagram of a structure for a fractionally spaced demodulator.

**5. COMPLEXITY**

There is no single figure that completely specifies the complexity of an algorithm. Multiplication usually requires significantly more power and silicon area in an ASIC than other operations and number of multiplications is therefore used to assess the complexity. More specifically the complexity is measured in number of real multiplications per complex baseband sample. The complexity of the modulator presented is the same as for the corresponding baud spaced demodulator. The complexity is therefore listed only for the demodulators.

In the calculations, it is assumed that the number of complex multiplication required in an N-point FFT is

\[ M_c = \frac{N}{2} \log_2 N \]  

This can be reduced by omitting trivial multiplications [6]. It is also assumed that a complex multiplication requires three real multiplications. Below, \( N \) is the number of samples per symbol interval and \( L \) is the length of the pulse shaping filter in number of symbol intervals. The filter coefficients \( h(m) \) are real.

**Baud spaced demodulator, direct implementation**

A direct implementation, as depicted in Figure 3, is usually not feasible but its complexity is included for comparison. Assume that \( N/2 \) channels are active. Mixing and filtering requires \( 3N + 2NL \) multiplications per channel per symbol interval. Thus, the number of real multiplications per complex input sample is

\[ M_r = N(\frac{3}{2} + L) \]  

**Baud spaced demodulator with N/2-point FFT**

The structure in Figure 4 processes blocks of \( N \) complex samples. For each block there are \( 2NL \) multiplications in each of the two filterbanks. For each block of \( N \) samples there are \( N \) complex multiplications for the phase shifters. adding these two to the complexity of the two N/2-point IFFT, the number of real multiplications per complex input sample is

\[ M_r = 4L + \frac{3}{2} + \frac{3}{2} \log_2 (N) \]  

**Fractionally spaced demodulator, direct implementation**

The fractionally spaced demodulator in Figure 5 has the same mixing as the baud spaced version but the filters operate with double rate. Thus the number of real multiplications per complex input sample is

\[ M_r = N(2L + \frac{3}{2}) \]  

**Fractionally spaced demodulator with N/2-point FFT**

The demodulator in Figure 6 operates on blocks of \( N/2 \) samples. For each block the filterbank requires \( 2LN \) real multiplications. The phase shifter in the lower branch requires \( N/2 \) complex multiplications per block. Using (31) for each of the two IFFT, the number of real multiplications per complex input sample is

\[ M_r = 4L + 3\log_2 (N) \]  

The complexity versus number of samples per symbol interval is plotted in Figure 7 for the different demodulators. The pulse length \( L \) is set to 4 in these examples [10].
Figure 7: Number of real multiplications per input sample for different demodulators. 1) Baud spaced, direct implementation, 3) Baud spaced with $N/2$-point FFT, 4) Fractionally spaced, direct implementation, 5) Fractionally spaced with $N/2$-point FFT.

6. CONCLUSION

Efficient algorithms, based on FFT\textsuperscript{1}, for modulation and demodulation in OFDM systems with offset-QAM has been presented. Demodulation algorithms for both one and two samples per symbol output were described. The algorithms with one sample per symbol are of the same complexity as previously presented algorithms but they offer a different structure which can be attractive in some applications. The algorithm with two samples per symbol offers, for reasonable pulse shapes, Nyquist-sampling of the subchannels without doubling the complexity of the demodulator.

7. REFERENCES


\textsuperscript{1} FFT and IFFT are basically the same algorithms, just with different numbering of the sequences