ITERATIVE DEMODULATION AND DECODING FOR LDPC CODED GENERALIZED FREQUENCY DIVISION MULTIPLEXING

A Thesis

by

DIGVEER DE

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Krishna Narayanan
Co-Chair of Committee,	Scott L. Miller
Committee Members,	Alex Sprintson
	Andrew Jiang
Head of Department,	Miroslav M. Begovic

August 2017

Major Subject: Electrical Engineering

ABSTRACT

Currently, there is a standardization process underway to design the fifth generation of wireless systems or 5G wireless systems. The ambitious targets set forth for 5G wireless systems call for novel approaches in all layers of the network. At the physical layer (PHY), Orthogonal Frequency Division Multiplexing (OFDM) has become a de facto standard for wireless systems such as 4G cellular and IEEE 802.11 (Wi-Fi) systems. However, the large peak to average power ratio of OFDM signals makes OFDM an unattractive candidate for some services envisioned in 5G systems, particularly in the uplink.

Recently, Generalized Frequency Division Multiplexing (GFDM), which is a member of the non-orthogonal multiple access technologies has been proposed as the modulation scheme for 5G wireless systems. GFDM has some advantages over OFDM, such as looser requirements on synchronization, a lower PAPR requirement, as well as a lower out-of-band spectral leakage. However, in GFDM the sub-channels are not orthogonal which results in inter-carrier interference and, hence, an increased uncoded bit error rate. While iterative receivers have been proposed for improving the bit error rate performance of uncoded GFDM, there are very few works that have studied the performance of coded GFDM systems. In this thesis, we investigate the performance of coded systems with GFDM. Using earlier results on soft interference cancellation based turbo equalization and turbo multi-user detection, we design an iterative receiver for GFDM with low density parity check codes. We show that the receiver is able to successfully combat the non-orthogonality of sub-channels in GFDM and provide performance similar to that of coded OFDM systems at an increased receiver complexity.

DEDICATION

To my family, who believed enough to make this happen

ACKNOWLEDGMENTS

I would foremost like to thank Prof. Krishna Narayanan, my thesis advisor and mentor, for the amount of time and effort he took in guiding me. He has always found time for me from his busy schedule. This thesis would not have seen the light of the day but for his continued support.

I also thank Prof. Miller, Prof. Sprintson and Prof. Jiang for agreeing to serve on my committee.

I would also like to thank my fellow group members, who guided me in the journey that is a graduate thesis.

I would like to thank my friends who helped make this journey seem shorter than it was. Neha, Parv, Shivani, Udit, thank you for being there at odd times.

In conclusion, I would like to thank my parents and my all so great sister who gives me hope and motivation from time to time, who supported me in this endeavour.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Dr. Krishna Narayanan, Dr. Scott Miller and Dr. Alex Sprintson of the Department of Electrical Engineering and Dr. Andrew Jiang of the Department of Computer Science & Engineering at Texas A&M, College Station.

All other work conducted for the thesis was completed by the student independently.

Funding Sources

There are no outside funding contributions to acknowledge related to the research and compilation of this document.

NOMENCLATURE

AWGN	Additive White Gaussian Noise
CP	Cylic Prefix
FBMC	Filter Bank Multi Carrier
FDM	Frequency Division Multiplexing
FFT	Fast Fourier Transform
GFDM	Generalized Frequency Division Multiplexing
ICI	Inter Carrier Interference
IFFT	Inverse Fast Fourier Transform
ISI	Inter Symbol Interference
LDPC	Low Density Parity Check
MF	Matched Filter
MMSE	Minimum Mean Squared Error
OFDM	Orthogonal Frequency Division Multiplexing
OOB	Out of Band
PAPR	Peak to Average Power Ratio
RC	Root Cosine
RRC	Root Raised Cosine
SIC	Successive Interference Cancellation
SNR	Signal to Noise Ratio
UFMC	Universal Filtered Multicarrier
ZF	Zero Forcing

α	Rolloff Factor of Prototype Filter
$\mathbf{A}$	Modulation Matrix
$\vec{b}$	Binary Data
$\vec{c}$	Encoded Binary Data
$\mathbf{B}$	Demodulation Matrix
$\vec{d}$	Modulated Data
$d_k[m]$	Data Symbol on m^{th} symbol , k^{th} subcarrier
$\vec{g}$	Prototype Filter
$\mathbf{H}$	Channel Matrix
K	Total Number of Subcarriers
M	Total Number of Subsymbols
N	Total Number of Samples Per Symbol
$\vec{x}$	Modulated Transmit Signal
$\vec{y}$	Received Signal
$\vec{z}$	Received Signal Post Equalization

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

1.1 Motivation

Orthogonal Frequency Division Multiplexing(OFDM) has so far been the most popular choice of modulation scheme for wireless communications, as well as for wired communications. [1]. For example, 4G (LTE), and IEEE 802.11 (WiFi) use OFDM as the physical layer Radio Access Technology [2]. This is mainly because of the low implementation complexity of OFDM in comparison to single carrier schemes. The low complexity is a consequence of the OFDM signal being a summation of multiple separated frequency tones which permits the use of inverse fast Fourier transform blocks for implementing the transmitter and the receiver. In addition, OFDM displays robustness against multipath dispersion.

While OFDM is the most popular choice in the downlink of most modern cellular networks, it is typically not used for the uplink in most cellular networks. This is because the orthogonality of OFDM is disrupted by the lack of frequency synchronization between the oscillators or clocks at the transmitter and receiver and the Doppler shift which results from the mobility of users. Multiple methods have been proposed [3] [4] which use feedback loops to achieve better synchronization. However, these methods come at the cost of added complexity and cost for the transceiver [5] and in general, become impractical and cumbersome for small low-cost devices, such as deployed in the Internet of Things (IoT), where devices are expected to operate on small batteries for 10-15 years. OFDM also displays spectral leakage due to its rectangular time domain pulse shape. The need for a cyclic prefix to be added before every block to be transmitted also adds to overhead [6]. Therefore, standard OFDM is not a suitable waveform for all use

cases of interest in 5G wireless for both uplink and the downlink.

A suitable waveform for 5G and IoT would possess the following desirable features[7]

- High spectral efficiency and low out of band leakage.
- Low PAPR, allowing for efficient power amplifier design.
- Robustness to Doppler shift in case of user mobility.
- Support for asynchronous transmission and reception.

1.1.1 Proposed Waveforms for 5G

Various physical layer multicarrier systems that can tolerate looser frequency synchronization targets and relaxed precision have been proposed for the 5^{th} Generation of Mobile Networks. These include

- Filter Bank Multicarrier (FBMC)[8]
- Generalized Frequency Division Multiplexing (GFDM) [9]
- Universal Filtered Multicarrier [2].

While the performance of these modulation schemes have been studied without coding, there are only a few works that have evaluated the performance of coded GFDM.

1.1.2 Contribution of this Thesis

In this thesis, we consider the design of a transmitter and receiver for a coded GFDM system and its implementation and evaluation in software. Particularly, we consider low density parity check codes and evaluate the performance of various iterative receivers that employ iterative demodulation and decoding based on

their bit error rate performance, among other metrics. We use existing methods from works by other researchers in iterative equalization and multi-user Other prior works which have evaluated coding for GFDM have looked into its application of coding for improved bit error rate performance . Prior work by Fettweis has looked into the use of a rate 1/3 Parallel Concatenated Convolutional Code [9]. Recently, an iterative MMSE receiver has been considered for LDPC-coded GFDM [10].

1.2 Organization of This Thesis

This thesis concentrates on the design of coding schemes for GFDM, and is organized into four separate chapters.

- Chapter 2 presents the core idea behind GFDM as well as its system model.
- Chapter 3 describes various linear receiver design considered for GFDM. A discussion of methods such as interference cancellation and coding schemes is also included.
- Chapter 4 describes our contribution, and talks about the iterative demodulator and decoder. Methods for complexity reduction for the iterative demodulator are discussed.
- Chapter 5 sums up the overall thesis, and compares GFDM with OFDM on certain key metrics, such as spectral efficiency and implementation complexity.

2. GENERALIZED FREQUENCY DIVISION MULTIPLEXING

The OFDM system design calls for rigid restrictions on both orthogonality as well as the frequency synchronization on individual subcarriers [11]. A much more efficient use of available bandwidth would be to permit a scheme where the spectrum of individual subcarriers is allowed to overlap [12]. The GFDM scheme (shown in Fig. 2.1) is based on the relaxation of these requirements, via modulation of independent blocks, where each block consists of subcarriers and subsymbols. In the GFDM scheme, orthogonality between subcarriers is not maintained and the data on the subcarriers is separately filtered through a prototype pulse shaped filter. Different filter responses can be used to filter the subcarriers and affect the choice of OOB radiation and the BER performance.

2.1 System Design

GFDM is a block filtered multicarrier modulation scheme [9]. It uses circular pulse shaping to transmit multiple subsymbols per subcarrier, and sacrifices orthogonality to achieve the requirements detailed in section 1.1.

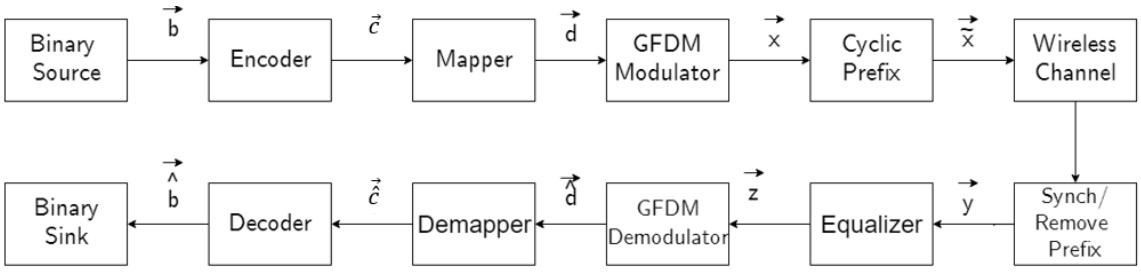

Figure 2.1: The GFDM Block Diagram

A binary data source first generates a binary data vector $\vec{\mathbf{b}}$, which is then encoded by an error correcting code to generate a new data vector $\vec{\mathbf{c}}$. We then map the bits to the symbols based on the modulation scheme in use. Let S denote the modulation symbol set with L complex symbols, such that $S \triangleq \{S_1, \ldots, s_L\}$. This vector, of complex symbols, say $\vec{\mathbf{d}}$, is the data to be transmitted. It is a data block that contains N elements, which will then get divided into K subcarrier and M subsymbols, as shown in Fig. 2.2. This reshapes the vector $\vec{\mathbf{d}}$ into $(\mathbf{d}_0^T, \ldots, \mathbf{d}_{M-1}^T)^T$, where $\mathbf{d}_m = (d_{0,m}, \ldots, d_{K-1,m})^T$. Then, $d_{k,m}$ corresponds to the data symbol on the k^{th} subcarrier and the m^{th} block. After this, each data symbol has circular pulse

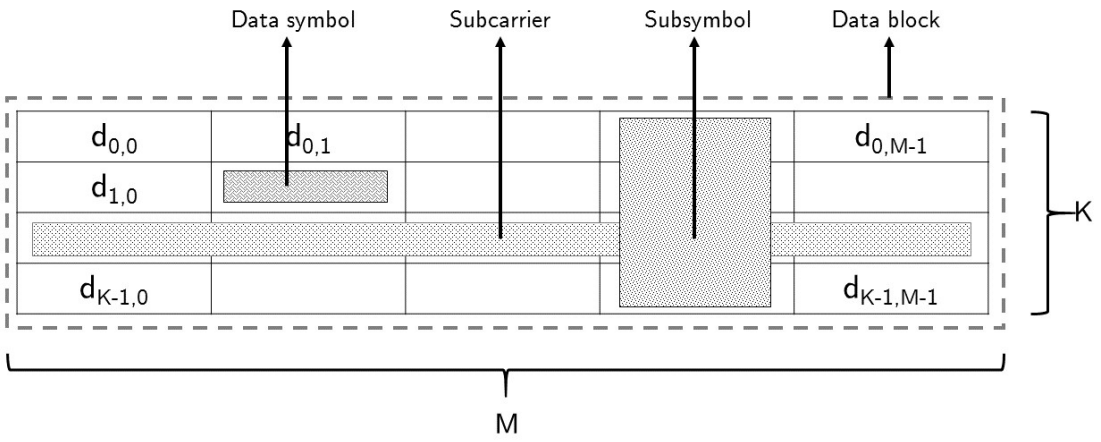

Figure 2.2: The GFDM Time Frequency Grid

shaping applied to it, where the pulse to be applied is a function of k and m, and is given by

$$g_{k,m}[n] = g[(n - mK) \mod N]\, e^{2\pi j \frac{k}{K} n}, \tag{2.1}$$

where k is the subcarrier index, m is the subsymbol index and n is the sampling

index. g here is the prototype filter, which is chosen from different designs based on the system to be designed. Each $g_{k,m}[n]$ is a shifted version of our prototype filter. Finally, summing all of them, the transmitted symbol at time n is given by

$$x[n] = \sum_{m=0}^{M-1} \sum_{k=0}^{K-1} d_{m,k} g_{0,0}[n - mK] e^{j2\pi \frac{kn}{K}}. \tag{2.2}$$

This is an overall superposition of all subcarriers in the system. To achieve this for one subcarrier, we use the initial prototype filter g_{00} , which then gets weighted or multiplied by the complex symbol values $d_{m,k}$. The overall transmit signal also delays each subsymbol by a length mN in time, and shifts it by $\frac{k}{N}$ in the frequency domain. This can be considered as a mapping from the time domain symbol to the frequency-time symbol , as shown in Fig. 2.3.

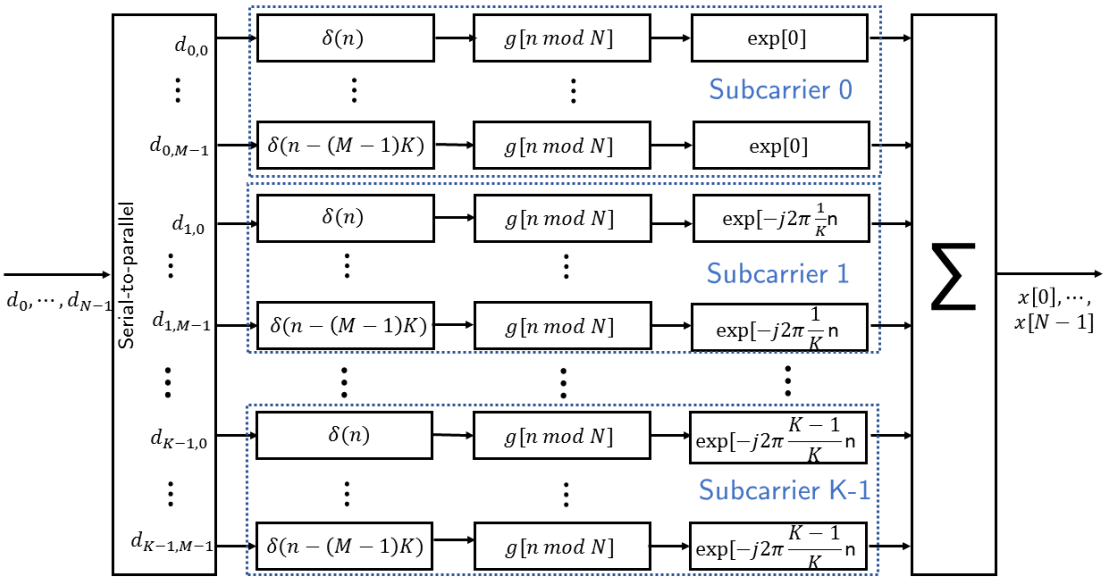

Figure 2.3: The GFDM Symbol Mapper

In order to better understand the GFDM modulation process, a representation of the impulse responses of the filters for each subcarrier during one time slot is shown in Fig. 2.4 . We can see that the impulse response of the subcarriers is shifted in the frequency domain.

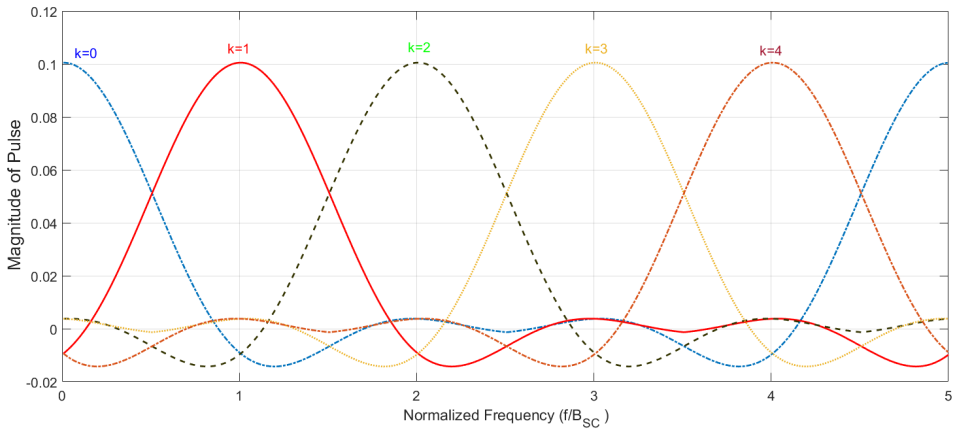

Figure 2.4: A normalized view of the frequency shifted subcarriers during one time slot

2.2 The Matrix Model for GFDM

We can also implement the above model as a multiplication of two matrices, as given in [13]. This allows us to simplify the transmitter structure.

In the first processing step, the bits are mapped to a 2^{μ} modulation grid, which in turn gets remapped to a the time-frequency matrix mentioned above. Thereafter, in order to ensure alias-free data shifting, zeros are inserted with a sampling matrix.

$$\mathbf{S}_N^M = \{s_{n,m}\}_{MN\times M}\ s_{n,m} = \begin{cases} 1 & n = (m-1)N+1 \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

This gives us the result $\mathbf{X}_D = \mathbf{S}_N^M\mathbf{D}$, where $\mathbf{X}_D$ are the upsampled data bits.

Next, a pulse shaping filter of length M symbols is sampled N times per symbol, which gives a vector $\mathbf{g}_{Tx}$.

$$G_{Tx} = \begin{bmatrix} g_1 & g_{MN} & \cdots & g_2 \\ g_2 & g_1 & \ddots & g_3 \\ \vdots & \ddots & \ddots & \vdots \\ g_{MN} & g_{MN-1} & \cdots & g_1 \end{bmatrix} \tag{2.4}$$

Using the previously derived $\mathbf{X}_G$,we get $\mathbf{X}_G = \mathbf{G}_{Tx}\mathbf{X}_D$. After this, we perform an inverse Fourier Transform. This operation can be represented by multiplication with a Fourier matrix containing the weights. Since there are K subcarriers in the system, to maintain subcarrier spacing, only every M^{th} column is selected. Adding all the processing steps together, we get our final expression as $\tilde{\mathbf{x}} = diag(\mathbf{G}_{Tx}\mathbf{S}_M^N\mathbf{D}(\mathbf{S}_M^N)^T\mathbf{W}^H)$

This form can be carried over to the more convenient

$$\vec{\mathbf{x}} = \mathbf{A}\vec{\mathbf{d}} \tag{2.5}$$

which is the model we shall use for the rest of this thesis.

2.3 Choosing the Prototype Filter

The choice of the prototype pulse has a strong effect on overall system performance. While OFDM uses a rectangular pulse in the time domain as the pulse

shaping filter, GFDM lets us select from a number of prototype filters for use. In GFDM, the k^{th} subcarrier is centered at the normalized frequency k/K, and hence the parameter α is a descriptor of the roll-off as well as the intercarrier interference suffered in the system. If the modulation matrix $\mathbf{A}$ can be succesfully inverted, then intersymbol interference can also be cancelled [14]. We choose our filters to be in such a way, that after the Matched Filter receiver is applied, the Nyquist property is met [15]. The following filters can be used for the prototype:

- **Raised Cosine:** The RC filter takes advantage of the roll off factor that decides the secondary roll off lobes.
- **Root Raised Cosine:** The RRC filter's frequency response is the square root of the frequency response of the RC filter.It is ideally suited when the receiver performs matched filtering, and helps in minimizing intersymbol interference.
- **Rectangular:** The Rectangular pulse shape is the OFDM pulse shape in the time domain.

2.4 Visualizing the A Matrix

An example transmitter matrix $\mathbf{A}$ helps us in showing the GFDM matrix model in more detail. To this end, we first begin by showing what an $\mathbf{A}$ looks like for a small system, where we choose a smaller K and M. Setting the number of subcarriers K to be 4 and the number of subsymbols M to be 7, and using a root raised cosine filter, we can plot the magnitude of the sample $\mathbf{A}$ matrix.

The $\mathbf{A}$ matrix also lets us visualize the crosstalk or intercarrier interference present in the system. Taking the absolute value of $\mathbf{A}^H\mathbf{A}$, we can see the ambiguity matrix 2.6 for the matrix $\mathbf{A}$ which we created in Fig. 2.5. Darker regions in the

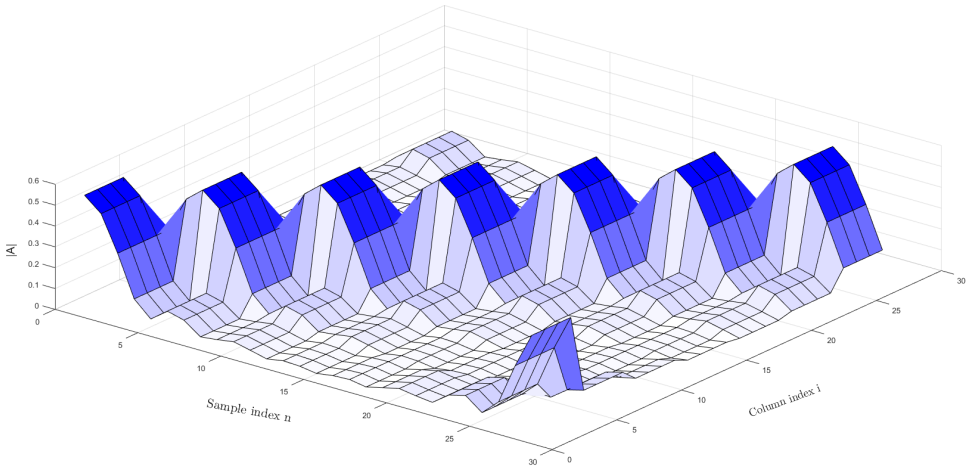

Figure 2.5: The Transmitter **A** matrix for N=28, K=4 and M=7 with an RRC filter and rolloff factor α=0.3

ambiguity matrix represent regions of greater interference.

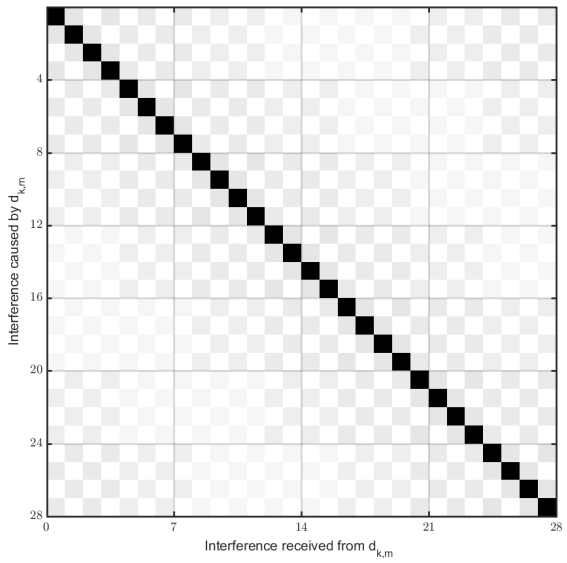

Figure 2.6: The ambiguity matrix for a matched filter demodulator for K=4 and M=7

2.5 Linear GFDM Receivers

$\vec{x}$ is the transmit symbols that correspond to the data given by the block $\vec{d}$. Transmission through a wireless channel can be modelled by $\vec{y}$, where $\vec{y} = \mathbf{H}\vec{x} + \vec{N}$, where $\vec{y}$ is the received version of $\vec{x}$. Here $\mathbf{H}$ represents the channel matrix for a Rayleigh multipath fading channel.

By performing time and frequency synchronization at the receiver side, we assume that the received signal is under perfect synchronization. The overall equation can now be written as $\vec{y} = \mathbf{HA}\vec{d} + \vec{N}$.We can perform channel equalization, assuming we have knowledge of the channel matrix, and recover an estimate, using

$$\vec{z} = \mathbf{H^{-1}HA}\vec{d} + \mathbf{H^{-1}}\vec{N} = \mathbf{A}\vec{d} + \vec{N} \tag{2.6}$$

Assuming, $\mathbf{H}$ is $\mathbf{I}$, that is , we are able to recover the channel state perfectly in (2.6).

After this channel equalization, we can perform linear demodulation on the signal using any of the multiple receivers. The equations for the linear receivers are straightforward, after we express the transmitter processing as a single multiplication of a complex valued matrix $\mathbf{A}$ with a vector of modulated symbols[15], as described in section 2.2. Using $\mathbf{B}$ to represent the receiver, we can express an estimate of the data as

$$\vec{\hat{d}} = \mathbf{B}\vec{z}. \tag{2.7}$$

Here $\mathbf{B}$ is a $MK \times MK$ receiver matrix.

The following receiver types are considered:

1. **Matched Filter Receiver** : This is the simplest way of reception, and as we

can see from (2.8), is the case where the receiver is the reciprocal of the transmitter. The matched filter is applied to each subcarrier individually. It maximizes the signal-to-noise ratio (SNR), but at the same time increases the self interference present in the system in the presence of a non-orthogonal pulse [9]. The matched filter receiver is given as

$$\vec{\hat{d}} = \mathbf{A}^H \vec{z} \tag{2.8}$$

2. **Zero Forcing Receiver**: The Zero Forcing Receiver removes self-interference from the received signal. However, this casues noise enhancement of the received signal.

$$\vec{\hat{d}} = (\mathbf{A^H A})^{-1} \mathbf{A^H} \vec{z} \tag{2.9}$$

3. **Minimum Mean Square Error Receiver**: The MMSE receiver makes a trade-off between the self-interference and noise enhancement. However, it has increased complexity, as computationally complex matrix inversions must be performed at every instance [16]. However, when the SNR or $Eb/N0$ is high, the term $\frac{\sigma_n^2}{\sigma_{\vec{d}}^2}$ tends to zero, and the receiver becomes equivalent to $(\mathbf{A^H A})^{-1} \mathbf{A^H} \vec{z}$, which is the Zero Forcing Receiver discussed above in (2.9). The MMSE receiver is given as

$$\vec{\hat{d}} = \left(\mathbf{A^H A} + \frac{\sigma_n^2}{\sigma_{\vec{d}}^2} \mathbf{I} \right)^{-1} \mathbf{A}^H \vec{z} \tag{2.10}$$

Post this, we simply make a hard decision on the received symbols $\vec{\hat{d}}$ and then demap them to produce the bit stream. Error curves for the standard receivers are shown in Fig. 5.1.

2.6 Interference Cancellation

In GFDM, we sacrifice the orthogonality of the subcarriers in favor of having reduced OOB radiation and a more flexible structure. Hence, we have inter carrier interference from adjacent subcarriers, as shown in Fig. 2.7. As we can see in Fig. 5.1, a comparison for similarly modelled receivers in GFDM and OFDM gives us a noticeable difference in bit error rates. Once again referencing results from Fig. 5.1, we notice that better performance can be achieved by shifting to a more complex receiver such as ZF or MMSE. This is because the prototype filters used for the subcarriers are not orthogonal to each other, and happen to interfere with the adjacent subcarriers. The more complex receivers perform tradeoffs between interference and noise.

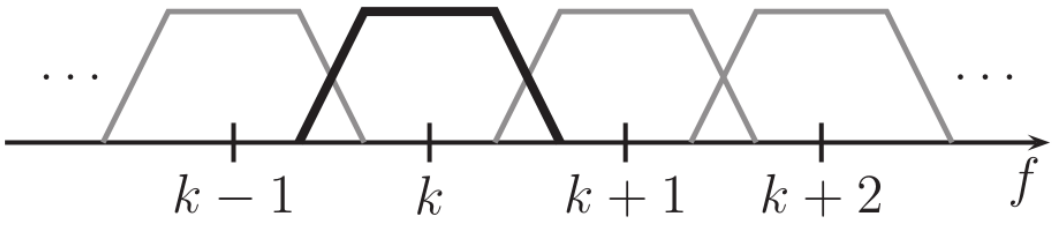

Figure 2.7: Inter carrier interference from adjacent subcarriers

2.6.1 Successive Interference Cancellation

One way to mitigate the effect of ICI is to perform successive interference cancellation on the received signal. The idea behind successive interference cancellation is to decode different users one after the other, that is, the interference due to other received decoded users is subtracted from the received signal before that

user is decoded [17]. In basic successive interference cancellation, we run the cancellation scheme K times, that is, once for every subcarrier in the system. In the subiteration, say $i = k$, we cancel out the interference from all the subcarriers excluding the i^{th} subcarrier.

The algorithm above first makes a soft decision for the data symbols on the k^{th} subcarrier. Then, the ICI is removed for each detected subcarrier. The pseudocode for the soft Successive Interference Cancellation algorithm is

Input: input signal $\vec{y}$

Output: Interference cancelled signal $\vec{\tilde{y}}$

initialization;

for *each iteration* **do**

  for *k=0 to K-1* **do**

    make soft decisions on $\vec{d}$;

    make $d_k[m] = 0 \; \forall \; m$;

    compute $z = y - Ad_{new}$;

    update $d_k[m]$ with the demodulated output of z ;

  end

end

Algorithm 1: The Soft Successive Interference Cancellation Algorithm

Another alternative is the Double Sided Interference Cancellation method. Most of the interference of the subcarriers is due to the adjacent subcarriers, as can be seen in the ambiguity matrix. Hence, instead of cancelling all the other $(k-1)$ subcarriers,we can also just cancel the adjacent two subcarriers to reduce computational complexity. The results for DSIC is similar to out SIC method, as most of the interference is contained within the adjacent two subcarriers. The BER performance for the interference cancellation scheme is most effective for the MF Re-

ceiver.

An alternative method proposed in [9], relied on symbol-by-symbol interference cancellation of the received signal $\vec{\mathbf{z}}$. A cancellation signal $\vec{u_{k,m}}$ was generated for each symbol, and $z_{k,m}$ was detected to obtain the k, m^{th} data symbol.

$$\vec{u_{k,m}} = \mathbf{A}\vec{\hat{d}} - \vec{g}_{k,m}\vec{d_{k,m}^{\wedge}} \qquad \forall k, m \tag{2.11}$$

$$\vec{z}_{k,m} = \vec{z} + \vec{u}_{k,m} \tag{2.12}$$

However, this approach comes at the expense of added complexity as compared to the previous approach, as we have to perform MK steps for every interference cancellation step, as opposed to the K in the previous method.

3. LDPC CODES AND MESSAGE PASSING DECODING

Every practical communication system uses error correction coding, especially at low SNR . Hence, receivers that are developed for GFDM should be able to work in conjunction with the decoders that will be used to decode the error correcting codes. The overall goal in this thesis is to build such a receiver for coded GFDM systems. Specifically, we focus on the class of low density parity check codes (LDPC) as the candidate for error correction coding since they are being considered for use in 5G wireless systems. In this chapter, we will provide a brief overview of LDPC codes and the message passing decoding algorithm used to decode LDPC codes.

3.1 Low Density Parity Check Codes

Low Density Parity Check Codes are a class of binary linear block codes with sparse parity check matrices. They were introduced and first studied by Gallager [18] and in the last 20 years, they have been used in a variety of wireless and wireline communication systems. In this thesis, we will restrict our attention to a sub class of LDPC codes called regular LDPC codes. The parity check matrix H of a (λ, ρ) regular LDPC code of length N and dimension K is a $(N-K) \times N$ binary matrix with exactly λ ones in each column and ρ ones in each row. Typically, λ, ρ are very small and N and K are large and, hence, H is a sparse matrix. While parity check matrix is sparse, the generator matrix is in general, not sparse. Some additional structure would usually have to be enforced on H to obtain a sparse generator matrix to facilitate encoding. However, in this thesis, we do not consider this. The generator matrix can be obtained through standard row operations on H, which can then be used to encode the data.

3.2 Iterative Decoding Algorithms

Tanner graphs are an effective way of representing LDPC codes in a graphical manner. They are bipartite graphs, where the left or variable nodes represents the coded bits and the right or check nodes represent the parity checks enforced by the code. An edge is drawn between check node c_i and variable node v_j if the corresponding entry h_{ij} in the **H** matrix is 1. An example of such a Tanner graph is shown in Fig. 3.1.

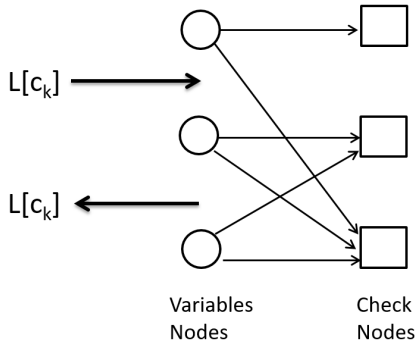

Figure 3.1: The Tanner Graph

LDPC codes can be efficiently decoded using a message passing algorithm called the sum-product algorithm that passes messages between the variable nodes and check nodes along the edges in the Tanner graph. We assume that the demodulator provides log-likelihood ratios for each of the variable nodes namely $L[c_k], \forall 1 \leq k \leq N$. Let $r_{i,j}$ denote the ith edge connected to the jth check node and let $L[r_{i,j}]$ denote the message passed along the edge $r_{i,j}$ from the right to left. Similarly, let $q_{i,j}$ denote the jth edge connected to the ith variable node and let $L[q_{i,j}]$ denote the message passed along the edge $r_{i,j}$ from the left to right. Further,

let $\mathcal{V}_j$ denote the set of variable nodes connected to the jth check node and let $\mathcal{C}_j$ denote the set of check nodes connected to the jth variable node. The following algorithm shows how $L[q_{i,j}]$ and $L[r_{i,j}]$ are updated iteratively.

Input: Vector of LLR's of bits $L_{ap}[c_k]$

Output: $L_{out}[c_i]$, an updated estimate of the LLR's

initialization $L_{out}[u_i] = L_{ap}[c_i]$;

for *number of iterations* **do**

  for *all i in N* **do**

    for *all j in* $(N - K)$ **do**

$$L[r_{i,j}] = 2\, atanh\left(\prod_{q_{i',j} \in \mathcal{V}_j \setminus q_{i,j}} \tanh(\frac{L(q_{i',j})}{2})\right);$$

$$L[q_{i,j}] = L[c_i] + \sum_{r_{i,j'} \in \mathcal{C}_i \setminus r_{i,j}} L(r_{j',i});$$

$$L_{out}[c_i] = L[c_i] + \sum_{r_{i,j'} \in \mathcal{C}_i} L(r_{j',i})\ ;$$

    end

  end

end

Algorithm 2: The LDPC Sum Product Algorithm

At end of the iterations, $L_{out}[c_i]$ represents the overall log likelihood of the variable node c_i. Often we are also interested in the *extrinsic* log likelihood of the variable node c_i which is given by

$$L_{ext}[c_i] = L_{out}[c_i] - L_{in}[c_i]. \tag{3.1}$$

Also, clipping of the extrinsic LLR's is a common practice to prevent numerical issues or 'overflow' [19].

In Chapter 3, the receiver we considered was a non-iterative decoder which performs one stage of interference canceller followed by one stage of hard decision or soft decision LDPC decoding as shown in Fig. 4.1. While this receiver is computationally less complex, the performance of such a receiver can be improved by iterative demodulation and decoding [20]. In this chapter, we consider a receiver that performs soft interference cancellation and LDPC decoding iteratively.

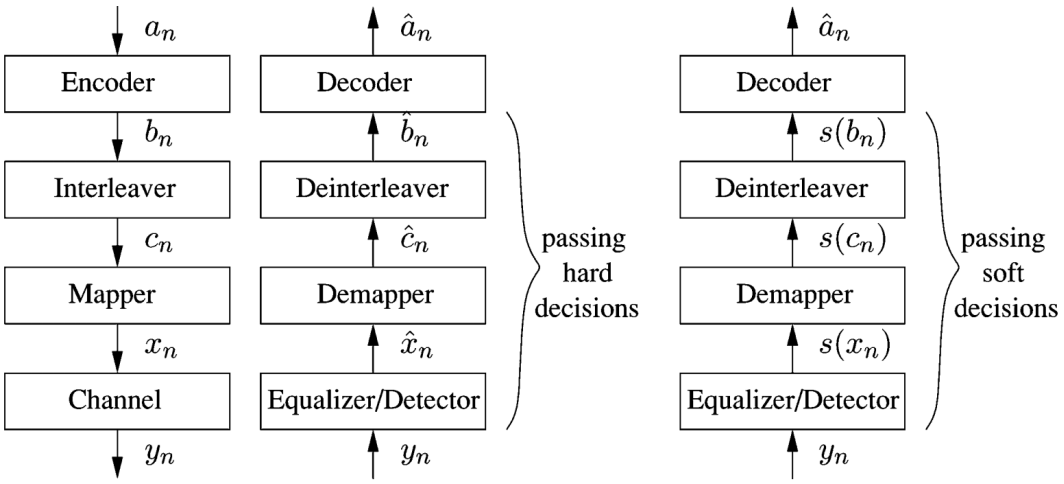

Figure 4.1: A non-iterative decoder structure [21]

4.1 Iterative Demodulation and Decoding

We first recall a few aspects of the system model before describing the receiver in order to clearly explain the workings of the iterative receiver. We assume that

$\vec{a}$, the data that is encoded is a sequence of i.i.d bits where the bits are equally likely *a priori*. The bit sequence $\vec{a}$ is encoded using an LDPC code to generate the bit sequence $\vec{b}$ which is then mapped to a vector of QAM symbols $\vec{d}$ and then modulated by the GFDM modulator into the sequence $\vec{x}$ and transmitted through an AWGN channel. The received data after transmission through the channel is $\vec{y}$. Fig. 4.2 shows this in a graphical manner.

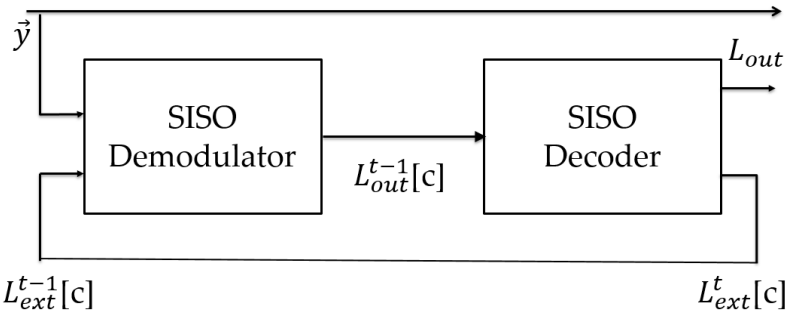

Figure 4.2: Single Stage of an Iterative Demodulator and Decoder

Our proposed receiver is based on the soft interference canceller originally proposed by Wang and Poor for iterative multiuser detection [22] and extended later to iterative equalization [23]. As shown in Fig. 4.2, the iterative demodulator and decoder consists of several stages (or iterations) with one soft-input soft-output (SISO) demodulator and one soft-input soft-output decoder in each stage. The demodulator is a soft interference cancellation based demodulator and the decoder is a message passing decoder for the LDPC code.

4.1.1 Soft-input Soft-output Demodulator

The inputs to the SISO demodulator at stage t in the receiver are $\vec{y}$ from the channel and the vector of log likelihood ratios $L_{ap}^{t}[c_k]$. We slightly abuse notation

here to let k denote the k^{th} QAM symbol. Let $c_{(K-1)M}, c_{(K-1)M+1}, \ldots, c_{KM-1}$ denote the bits associated with reconstructing our soft symbol estimate from our *a priori* LLR's.

For the AWGN channel, recall that

$$\vec{y} = \mathbf{A}\vec{d} + \vec{N}, \tag{4.1}$$

where $\vec{d}$ is the vector of QAM modulated symbols. For an AWGN channel, the post-equalization vector $\vec{z}$ is the same as the channel output $\vec{y}$.

Let the k^{th} QAM symbol d_k be given by $d_k = \mathcal{M}(c_{k,1}, c_{k,2}, \ldots, c_{k,M})$, where $\mathcal{M}$ denotes the mapper in Fig. 2.1. Let $\mathcal{M}_i^{-1}$ denote the ith component of the inverse mapper, i.e., $\mathcal{M}_i^{-1}(a)$ gives the i^{th} bit in the binary labeling of the QAM symbol a.

In the demodulator, the bit LLRs are first converted to bit probabilities using

$$\begin{aligned} P(c_{k,i} = 0) &= \frac{exp(L_{ap}^t[c_{k,i}])}{1 + exp(L_{ap}^t[c_{k,i}])} \\ P(c_{k,i} = 1) &= \frac{1}{1 + exp(L_{ap}^t[c_{k,i}])}. \end{aligned} \tag{4.2}$$

The bit probabilities are then converted to symbol probabilities according to

$$P(d_k = s_m) = \prod_{i=1}^{L} P(c_{k,i} = \mathcal{M}_i^{-1}(s_m)). \tag{4.3}$$

Using these bit probabilities, soft symbol estimates of d_k are computed using a weighted average of the constellation points taken from the set $\mathcal{S}$ and the *a priori* bit probabilities calculated in the above equation. The soft estimates are given by

$$\tilde{d_k} = \sum_{m=0}^{M-1} s_m \, P\left(d_k = s_m\right). \tag{4.4}$$

The soft estimates $\tilde{d_k}$'s are used to cancel the interference from the received signal $\vec{y}$. Specifically, for kth symbol, the interference estimate vector $\tilde{\hat{d}}_k$ is computed for each k as shown below and an estimate of the residual signal $\tilde{\vec{x}}_k$ (i.e., the residual in $\vec{y}$ after the interference is cancelled) is formed. More precisely,

$$\tilde{\hat{d}}_k = [\widetilde{d_1}, \ldots, \widetilde{d_{k-1}}, 0, \tilde{d}_{k+1}, \ldots, \widetilde{d_{MK}}] \tag{4.5}$$

$$\tilde{\vec{x}}_k = (\vec{y} - \mathbf{A}\hat{\tilde{d}}_k). \tag{4.6}$$

Then, an MMSE filter is applied to $\tilde{x_k}$ in order to form an estimate of d_k. The estimate is given by

$$u_k = w_k^H \tilde{x}_k, \tag{4.7}$$

where w_k is the filter. For each symbol, w_k is chosen so as to minimize the expected value of the mean square error. That is given the QAM symbol d_k and the filter output u_k, we choose a w_k such that

$$w_k = \arg\min_{w_k} E(\ |d_k - w^H \tilde{\vec{x}}_k|^2\ ) \tag{4.8}$$

This filter can be obtained as in [22] and is given by

$$w_k = (\mathbf{A}^H \Delta_k \mathbf{A} + \mathbf{\Sigma})^{-1} \mathbf{A} e, \text{ where} \tag{4.9}$$

$$\Delta_k \triangleq \text{diag}\left([2 - |\tilde{d_1}|^2, \ldots 2 - |\tilde{d}_{k-1}|^2, 2, 2 - |\tilde{d}_{k+1}|^2, \ldots, 2 - |\tilde{d}_{MK}|^2]\right). \tag{4.10}$$

Here $\vec{e}$ is an indicator vector with all zero entries, except the k^{th} position, where it is 1.

Once the MMSE estimate of d_k is obtained as in (4.7), these estimates need to be converted to log likelihood ratios of the bits. For this purpose, we can think of a

hypothetical side-information channel between d_k and u_k as shown in Fig. 4.3. The symbol u_k can be considered to be the output of an additive white Gaussian noise channel with input d_k, channel gain μ_k and the additive noise being $\sim \mathcal{N}(0, \nu^2)$.

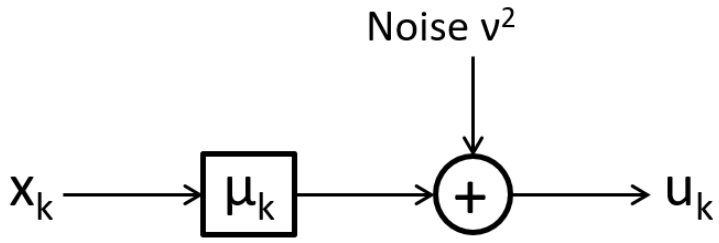

Figure 4.3: The MMSE Filter

Conditioned on the modulated symbol d_k, μ_k and ν_k can be calculated as follows [22].

$$\begin{aligned} \mu_k &\triangleq E(u_k|d_k) = e^T \mathbf{A}^H (\mathbf{A}\Delta\mathbf{A}^H + \mathbf{\Sigma})^{-1}\mathbf{A}\vec{e} \\ \nu_k^2 &\triangleq var(u_k|d_k) = \mu_k - \mu_k^2 \end{aligned}$$

It can then be seen that

$$P(u_k|d_k = s_m) \sim \mathcal{N}(\mu_k s_m|\nu_k^2), \tag{4.11}$$

where u_k is the MMSE output from the MMSE filter, and d_k belongs to the symbol alphabet S for the chosen constellation. Finally, we obtain LLRs for the bits $c'_{k,i}$s according to

$$L^t_{ext}[c_{k,i}] = \log \frac{\sum_{s_m:\mathcal{M}_i^{-1}=0} P(u_k|d_k = s_m)}{\sum_{s_m:\mathcal{M}_i^{-1}=1} P(u_k|d_k = s_m)}. \tag{4.12}$$

These LLRs are passed to the SISO decoder.

4.1.2 Soft-input Soft-output Decoder

The Soft-input Soft-output decoder is based on the message passing LDPC decoder discussed previously in chapter 3. In the t-th stage of the decoder, the SISO decoder takes in LLRs $L^t_{ext}[c]$ generated by the SISO demodulator and generates extrinsic LLRs $L^t_{ext}[c]$, which are fed to the SISO demodulator in the $(t+1)$th stage of the receiver. This iterative procedure continues for a fixed number of iterations or until a valid codeword is produced by the LDPC decoder.

4.2 Complexity Reduction for the Iterative Demodulator

It can be seen that the major bottleneck in terms of complexity in the receiver is given by the matrix inversion in (4.9), which needs to be calculated for each data symbol in the SISO demodulator. Taking the number of symbols N to be $M \times K$, the order of complexity for matrix inversion for a $N \times N$ is given by $\mathcal{O}(N^3)$, the overall complexity for a straight forward implementation of the algorithm as described is $\mathcal{O}(N^4)$. If N is large enough, say, greater than a 100, computational complexity is large. Since N, which is the product of the number of subcarriers(K) and the number of subsymbols (M) is expected to be large in a multicarrier system such as GFDM, we seek methods to reduce the complexity complexity. While there are methods which seek to calculate the inverse based on the properties and size of the matrix to be inverted [24], our aim is to have a more generalized method, based on the work done for iterative demodulation in [24] and [21]. As we will see below, the complexity can indeed be substantially decreased with some receivers requiring only $\mathcal{O}(N^2)$ operations. We will consider the following three approaches.

4.2.1 Woodbury's Matrix Inversion Lemma

From (4.9), we see that we that the the major complexity is due to the matrix inversion ($\mathcal{O}(N^3)$). However, we can compute the inverse for the $(k+1)$th symbol as a rank-2 update to the inverse for the kth symbol. To explain this further, let

$$G_k = (\mathbf{A}\Delta_k\mathbf{A}^H + \mathbf{\Sigma})^{-1}. \tag{4.13}$$

Then, w_k and w_{k+1} can be written as

$$w_k = (\mathbf{A}\Delta_k\mathbf{A}^H + \mathbf{\Sigma})^{-1}\mathbf{A} = G_k^{-1}\mathbf{A} \tag{4.14}$$

$$w_{k+1} = (\mathbf{A}\Delta_{k+1}\mathbf{A}^H + \mathbf{\Sigma})^{-1}\mathbf{A} = G_{k+1}^{-1}\mathbf{A}. \tag{4.15}$$

The matrix G_{k+1} can be written in terms of G_k and a difference matrix D as follows

$$\underbrace{(\mathbf{A}\Delta_{k+1}\mathbf{A}^H + \mathbf{\Sigma})}_{G_{k+1}} = \underbrace{(\mathbf{A}\Delta_k\mathbf{A}^H + \mathbf{\Sigma})^{-1}}_{G_k} + \underbrace{\mathbf{A}(\Delta_{k+1} - \Delta_k)\mathbf{A}^H}_{D} \tag{4.16}$$

Therefore,

$$G_{K+1}^{-1} = (G_K + D)^{-1} \tag{4.17}$$

Using Woodbury's Identity for the expansion, we can see that

$$G_{K+1}^{-1} = G_K^{-1} - \mathbf{A}((\Delta_{k+1} - \Delta_k)^{-1} + \mathbf{A}^H {G_k}^{-1}\mathbf{A})^{-1}\mathbf{A}^H G_K^{-1} \tag{4.18}$$

By setting $\tilde{\Delta}_k = \Delta_{k+1} - \Delta_k$, we obtain

$$G_{K+1}^{-1} = G_K^{-1} - \mathbf{A}(\tilde{\Delta} + {\mathbf{A_{k,k+1}}}^H G_k^{-1}\mathbf{A_{k,k+1}})^{-1}\mathbf{A}^H G_K^{-1} \tag{4.19}$$

where $\tilde{\Delta}_k = \Delta_{k+1} - \Delta_k$. To further reduce the complexity, we note that $\tilde{\Delta}_k$ is a very sparse matrix. This can be seen from the fact that Δ_k and Δ_{k+1} differ only in two positions, with all the other terms being identically $1 - \widetilde{d^2}$. Thus, their difference, $\tilde{\Delta}$ is a sparse matrix containing entries at only the positions (k, k) and $(k+1, k+1)$. $\mathbf{A}_k$ refers to selecting only the kth column of the matrix $\mathbf{A}$.

$$\Delta_k = [2 - \tilde{d}_1^2, \ldots, 2 - \tilde{d}_{k-1}^2, 2, 2 - \tilde{d}_{k+1}^2, \ldots, 2 - \tilde{d}_{MK}^2] \tag{4.20}$$

$$\Delta_{k+1} = [2 - \tilde{d}_1^2, \ldots, 2 - \tilde{d}_k^2, 2, 2 - \tilde{d}_{k+2}^2, \ldots, 2 - \tilde{d}_{MK}^2] \tag{4.21}$$

Therefore, taking $\widetilde{\Delta}$ to be the square matrix which has the the only non-zero entries along the diagonal matrix, we can we can reduce the complexity to a one time computation of $\mathcal{O}(N^3)$, with each update having the complexity of $\mathcal{O}(N^2)$, as given in (4.19). Only the columns of $\mathbf{A}$ corresponding to the k^{th} and $(k+1)^{th}$ entry are considered while computing this.

4.2.2 Hard Decision Feedback

In lieu of feeding soft symbol estimates from the LDPC decoder, we can feed hard decisions which are mapped onto the signal constellation S. For a QAM constellation, when hard decisions are fedback, since $|\tilde{d}_k|^2 = 2$, it can be seen that Δ_k in (4.10) becomes a diagonal matrix with zeros everywhere except at (k, k), where the value is 2. As a result, the inversion step in (4.9) becomes independent of the input $\widetilde{\hat{\Delta}_k}$ and, hence, can be precomputed for every k. Thus, we can reduce the computation complexity for each iteration to $\mathcal{O}(N^2)$, for an overall complexity of $\mathcal{O}(N^3)$.

4.2.3 Selecting only the Interfering Subcarriers

As we can see from the ambiguity matrix in Fig. 2.6, not all the subcarriers interfere onto each other. From Fig. 2.3, we note that individual streams within a single subcarrier are orthogonal to each other. Thus, cancelling out interference from within the streams present in a subcarrier presents no benefits. Alternately, we can sum up the streams in a subcarrier and perform interference cancellation between carriers. Since the streams within a carrier are orthogonal to each other, we can perform cancellation on a per subcarrier basis, this reducing the number of cancellations we need to do from KM to K. Further complexity reduction can be accomplished by using the previous method and cancelling out only those subcarriers which show up in the ambiguity matrix for the $\mathbf{A}$ matrix.

5. RESULTS AND CONCLUSIONS

To evaluate the performance of our iterative demodulator and decoder for GFDM, a simulator was implemented using MATLAB. The implementation was done as described in chapter 2 and chapter 4.

5.1 Uncoded GFDM Performance

Using the parameters in table 5.1, we ran simulations for the uncoded GFDM system. Uncoded GFDM BER curves for the three standard GFDM receivers are plotted in Fig. 5.1. The OFDM BER curve is shown as well for comparison. It can be observed in the figure that the GFDM MF receiver has a worse performance as compared to OFDM, and the GFDM MMSE and ZF receivers have half a dB gap in performance as compared to OFDM. It can be seen that this gap widens at higher E_b/N_0's for the MF receiver. Fig. 5.2 shows a plot for soft interference cancellation for the MF-GFDM receiver, as described in algorithm 1 in chapter 2. Soft estimates are used to cancel out interference and three iterations are used. We notice that there is still a performance gap between the interference cancelled signal BER and the standard OFDM BER curve. However, adding interference cancellation has enabled the MF performance to approach the same performance as displayed by the MMSE or ZF receivers for GFDM.

Description	Parameter	Value
Number of Subcarriers	K	64
Number of Subsymbols	M	5
Pulse Shaping Filter	g	RRC
Roll-Off Factor	α	0.5
Modulation Scheme	4	QAM

Table 5.1: Parameters for Bit Error Rate Simulation for Fig. 5.1 and 5.2

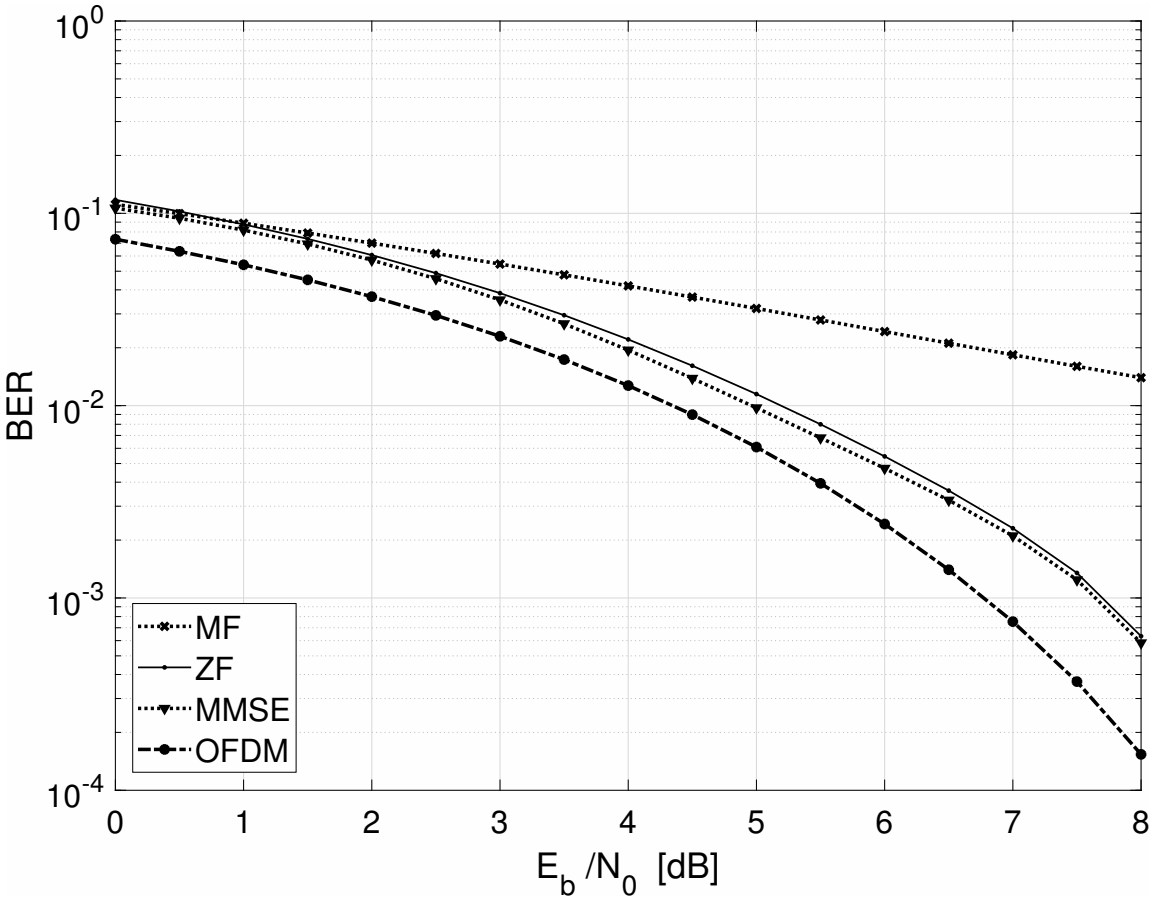

Figure 5.1: BER curve for the standard GFDM Receivers: MF, ZF, MMSE

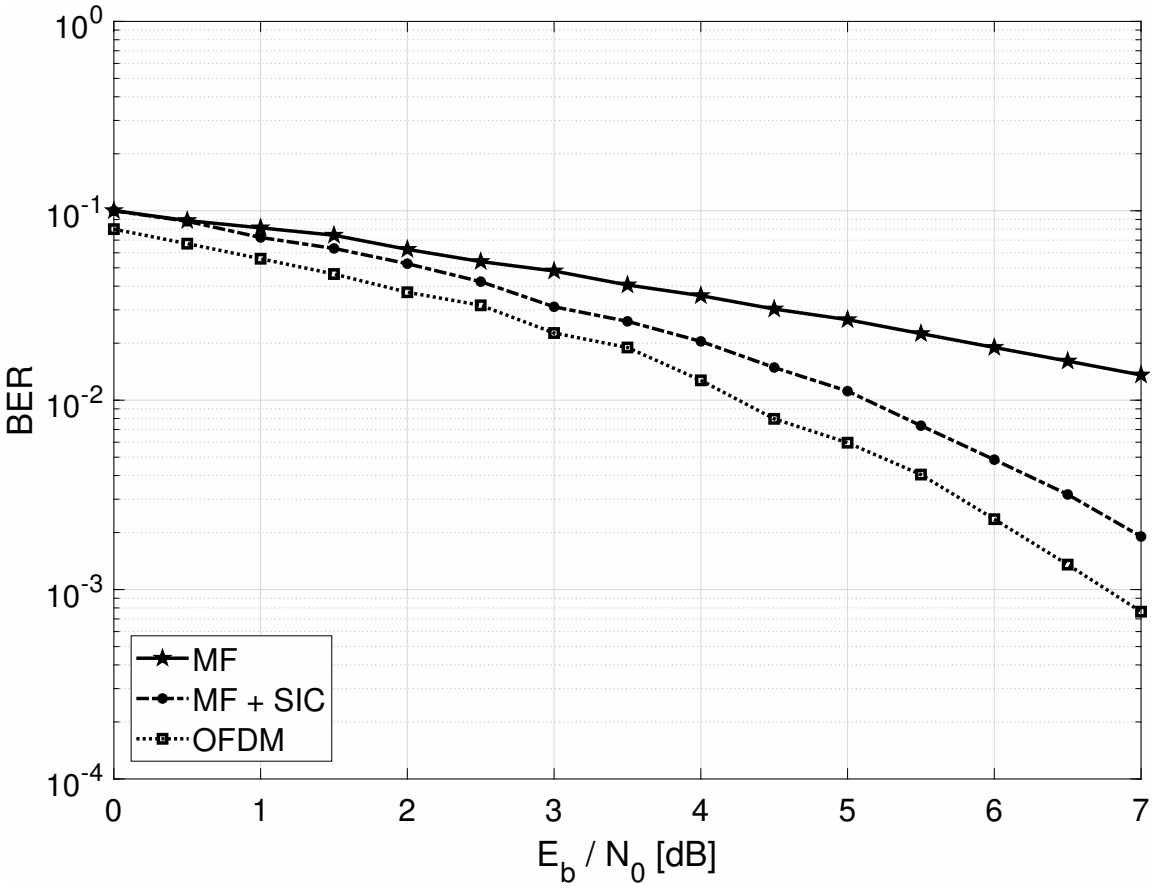

Figure 5.2: BER curve for Soft Interference Cancellation for GFDM

5.2 Coded Performance

In the uncoded scenario, the bit error rate of the OFDM receiver is better as compared to the BER of GFDM receivers. In the previous section 4, we built an iterative demodulator and decoder for GFDM to reduce this gap. In Fig. 5.3, we show a plot of the bit error rate versus E_b/N_0 for the iterative demodulator and decoder proposed in this thesis for $\alpha = 0.3$. The other parameters used in the simulation are given in Table 5.2. The performance of OFDM with the same LDPC code is also shown in the Figure. It can be seen from Fig. 5.3 that the gap between

our receiver after three iterations and coded OFDM is very small indicating that the iterative demodulator can nearly remove all the interference for $\alpha = 0.3$. It should also be noted that there is a diminishing performance gain from performing multiple iterations. However, the performance gap is larger as we increase the roll-off factor of the prototype filter in use. Increasing the roll-off factor to 0.9 creates more interference which our iterative demodulator and decoder is not able to fully cancel. This performance gap can be seen in Fig. 5.4.

One of the ideas proposed in chapter 4 to reduced receiver complexity was to feed hard estimates of the transmitted symbols to the SISO demodulator instead of the soft-symbol estimates proposed earlier. That is, given a symbol estimate, it would first be mapped to one of the points in the symbol constellation S. However, it should be noted that choosing hard decision feedback is a tradeoff between receiver complexity and BER performance. In Fig. 5.5, we see that the performance penalty for choosing hard decision decoding is around 0.5 dB, and thus hard decision feedback can be further explored as a method to reduced receiver complexity. Receiver complexities are compared in Table 5.3.

Description	**Parameter**	**Value**
Number of Subcarriers	K	32
Number of Subsymbols	M	5
Pulse Shaping Filter	g	RRC
Roll-Off Factor	α	{0.3,0.9}
Modulation Scheme	4	QAM

Table 5.2: Parameters for Bit Error Rate Simulation for Fig. 5.3 , 5.4 and 5.5

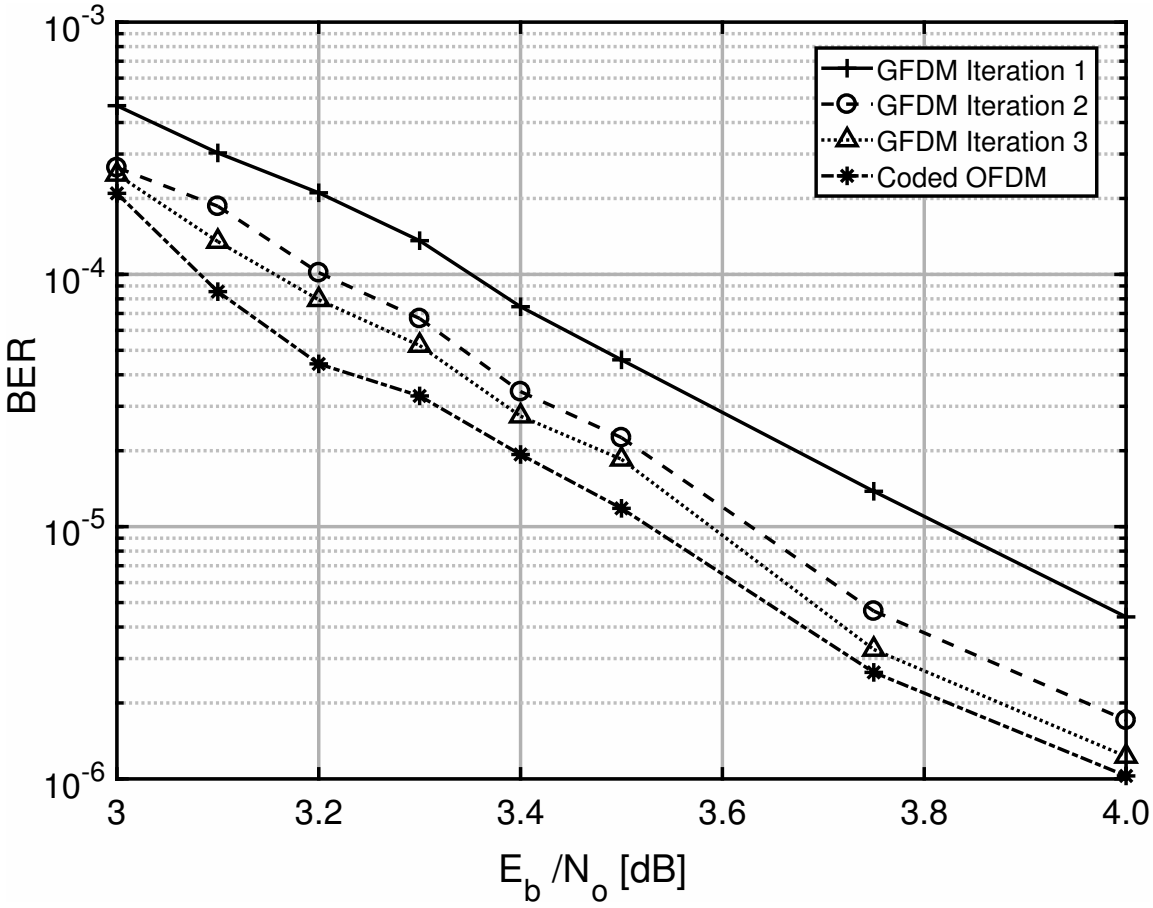

Figure 5.3: Iterative Demodulator and Decoder for GFDM with α=0.3

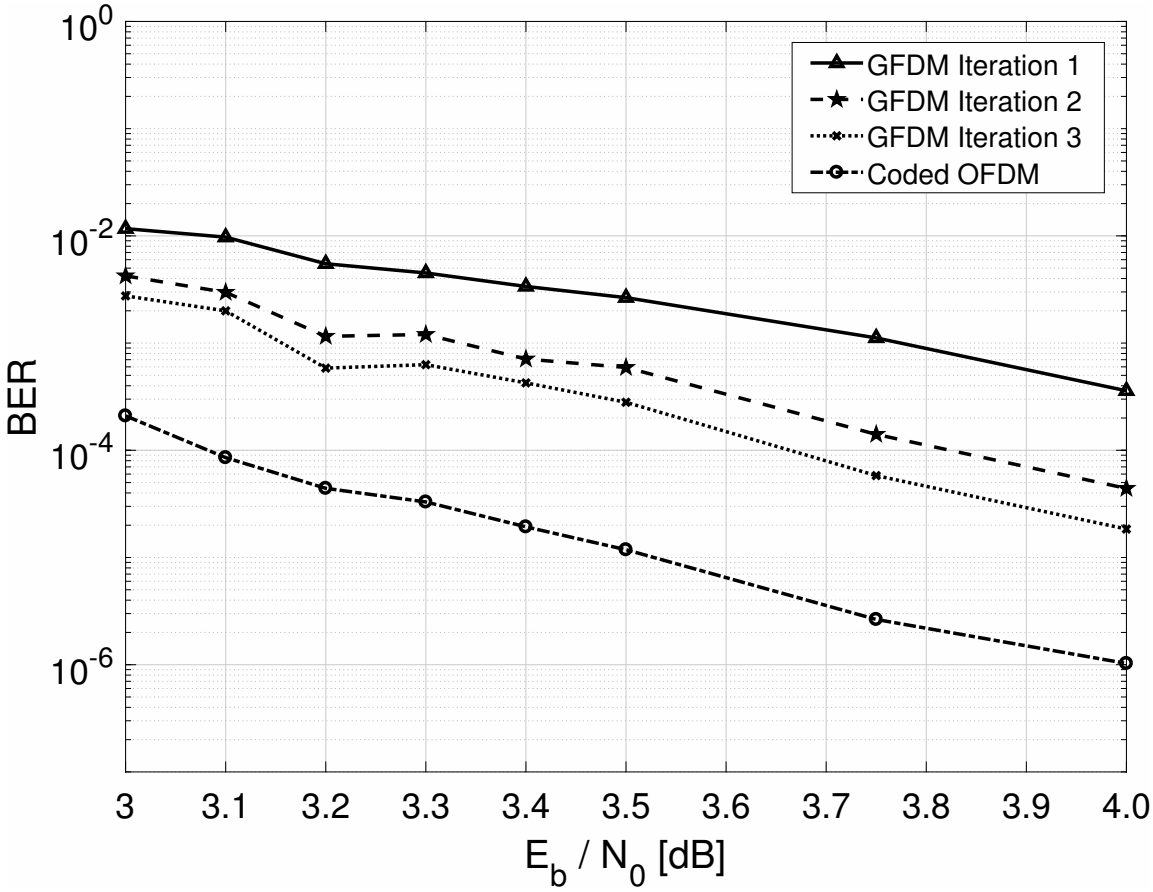

Figure 5.4: Iterative Demodulator and Decoder for GFDM with α=0.9

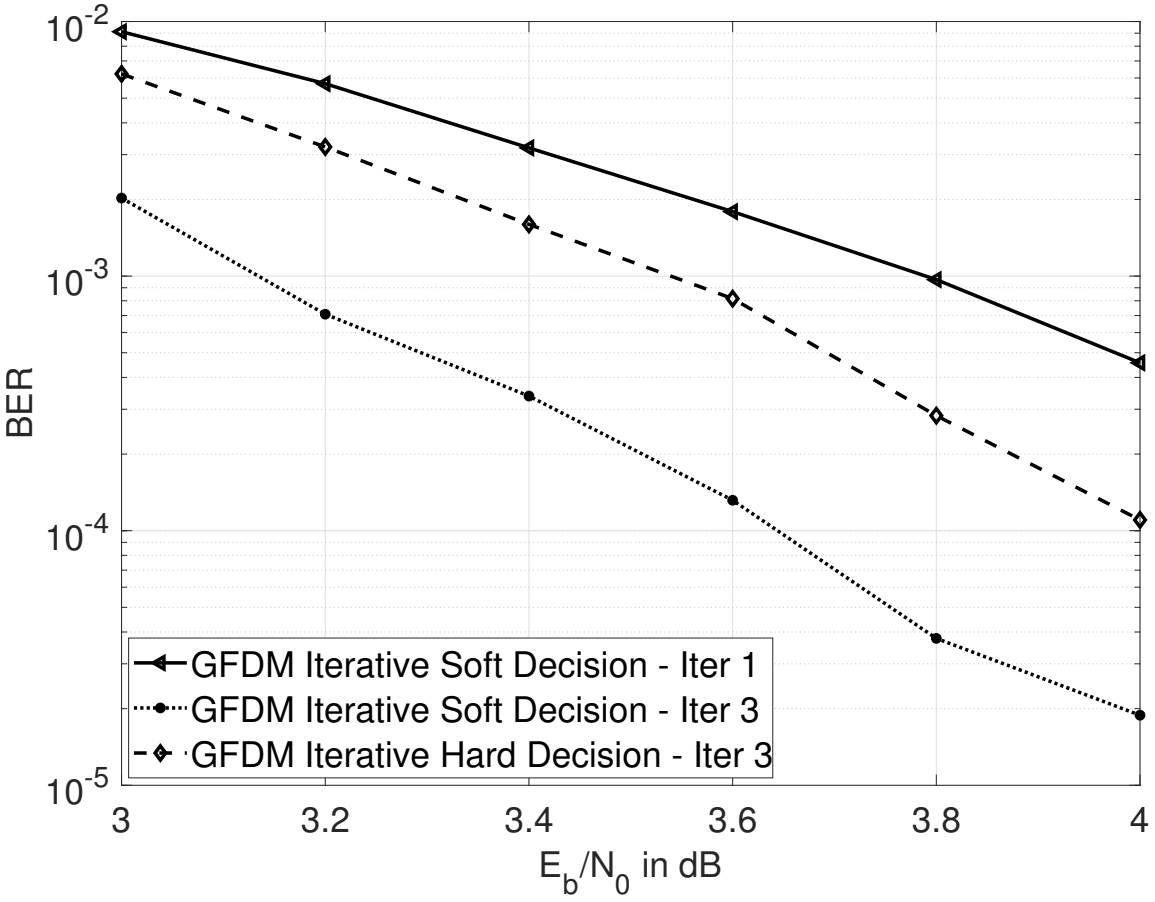

Figure 5.5: BER Curve for Soft Decision vs Hard Decision Feedback

Receiver Type	**Decoding Complexity**
OFDM	$\mathcal{O}(N \log N)$
GFDM-ZF	$\mathcal{O}(N^2)$
GFDM-MMSE	$\mathcal{O}(N^3)$
Iterative Receiver	$\mathcal{O}(N^4)$
Iterative Receiver + Woodbury's Inversion	$\mathcal{O}(N^3)$
Iterative Receiver + Hard Decision	$\mathcal{O}(N^2)$

Table 5.3: Receiver Complexities

5.3 Conclusion

In this work, we have designed an iterative demodulator and decoder for LDPC coded GFDM systems based on similar receiver used for multi-user detection and equalization in previous works. We have also proposed a few low complexity versions of the receiver. Finally, we have evaluated the performance of the proposed receivers through software simulations. Our main findings and conclusions are

1. Without coding or interference cancellation, the basic GFDM receiver performs worse than OFDM. However, at lower filter roll off factors, GFDM performs similar to OFDM.

2. For coded GFDM, with our proposed receiver, coded GFDM can perform close to that of coded OFDM even for moderate values of α. For large values of α, there is a gap between the performance of coded GFDM and coded OFDM.

3. It is possible to create a low complexity iterative receiver for GFDM which provide a reasonable tradeoff in performance versus complexity.

REFERENCES

[1] H.-H. Chen and M. Guizani, *Next generation wireless systems and networks*. John Wiley & Sons, 2006.

[2] F. Schaich, T. Wild, and Y. Chen, "Waveform contenders for 5G-suitability for short packet and low latency transmissions," in *Vehicular Technology Conference (VTC Spring), 2014 IEEE 79th*, pp. 1–5, IEEE, 2014.

[3] X. Zheng, Y. Jiang, and J. Li, "Efficient closed-loop schemes for MIMO-OFDM-based WLANs," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 202–202, 2006.

[4] J. Wang, Z.-X. Yang, C.-Y. Pan, M. Han, and L. Yang, "A combined code acquisition and symbol timing recovery method for TDS-OFDM," *IEEE Transactions on broadcasting*, vol. 49, no. 3, pp. 304–308, 2003.

[5] A. Tasic, W. A. Serdijn, and G. Setti, *Circuits and systems for future generations of wireless communications*. Springer Science & Business Media, 2009.

[6] R. Gerzaguet, N. Bartzoudis, L. G. Baltar, V. Berg, J.-B. Doré, D. Kténas, O. Font-Bach, X. Mestre, M. Payaró, M. Färber, *et al.*, "The 5G candidate waveform race: a comparison of complexity and performance," *EURASIP Journal on Wireless Communications and Networking*, vol. 2017, no. 1, p. 13, 2017.

[7] A. Roessler, "5G Waveform Candidates Application Note,"

[8] "Physical layer for future broadband radio systems, author=Bellanger, Maurice," in *Radio and Wireless Symposium (RWS), 2010 IEEE*, pp. 436–439, IEEE, 2010.

[9] G. Fettweis, M. Krondorf, and S. Bittner, "GFDM - Generalized Frequency Division Multiplexing," in *VTC Spring 2009 - IEEE 69th Vehicular Technology Conference*, pp. 1–4, April 2009.

[10] M. Matthe, D. Zhang, and G. Fettweis, "Iterative Detection using MMSE-PIC Demapping for MIMO-GFDM Systems," in *European Wireless 2016; 22th European Wireless Conference; Proceedings of*, pp. 1–7, VDE, 2016.

[11] G. Wunder, P. Jung, M. Kasparick, T. Wild, F. Schaich, Y. Chen, S. T. Brink, I. Gaspar, N. Michailow, A. Festag, L. Mendes, N. Cassiau, D. Ktenas, M. Dryjanski, S. Pietrzyk, B. Eged, P. Vago, and F. Wiedmann, "5GNOW: non-orthogonal, asynchronous waveforms for future mobile applications," *IEEE Communications Magazine*, vol. 52, pp. 97–105, February 2014.

[12] L. Cimini, "Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing," *IEEE transactions on communications*, vol. 33, no. 7, pp. 665–675, 1985.

[13] N. Michailow, S. Krone, M. Lentmaier, and G. Fettweis, "Bit error rate performance of generalized frequency division multiplexing," in *Vehicular Technology Conference (VTC Fall), 2012 IEEE*, pp. 1–5, IEEE, 2012.

[14] Q. Bodinier, C. F. Bader, F. Supelec, C. Moy, and H. Ahmadi, "Contribution to PHY and MAC layers for GFDM in the context of Cognitive Radio," 2016.

[15] I. Gaspar, N. Michailow, A. Navarro, E. Ohlmer, S. Krone, and G. Fettweis, "Low complexity GFDM receiver based on sparse frequency domain processing," in *Vehicular Technology Conference (VTC Spring), 2013 IEEE 77th*, pp. 1–6, IEEE, 2013.

[16] A. Farhang, N. Marchetti, F. Figueiredo, and J. P. Miranda, "Massive MIMO and waveform design for 5th generation wireless communication systems," in *5G for Ubiquitous Connectivity (5GU), 2014 1st International Conference on*, pp. 70–75, IEEE, 2014.

[17] X. Zhang and M. Haenggi, "The performance of successive interference cancellation in random wireless networks," *IEEE Transactions on Information Theory*, vol. 60, no. 10, pp. 6368–6388, 2014.

[18] R. Gallager, "Low-density parity-check codes," *IRE Transactions on information theory*, vol. 8, no. 1, pp. 21–28, 1962.

[19] B. K. Butler and P. H. Siegel, "Numerical issues affecting LDPC error floors," in *Global Communications Conference (GLOBECOM), 2012 IEEE*.

[20] C. Douillard, M. Jézéquel, C. Berrou, D. Electronique, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo equalization," *European transactions on telecommunications*, vol. 6, no. 5, pp. 507–511, 1995.

[21] M. Tuchler and A. C. Singer, "Turbo Equalization: An overview," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 920–952, 2011.

[22] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Transactions on communications*, vol. 47, no. 7, pp. 1046–1061, 1999.

[23] K. R. Narayanan, "Turbo equalization," *Encyclopedia of Telecommunications*, 2002.

[24] L. Fang, L. Xu, Q. Guo, D. Huang, and S. Nordholm, "A low complexity iterative soft-decision feedback MMSE-PIC detection algorithm for massive

MIMO," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pp. 2939–2943, IEEE.