Extended GFDM Framework: OTFS and GFDM Comparison

Ahmad Nimr, Marwa Chafii, Maximilian Matthé, Gerhard Fettweis
Vodafone Chair Mobile Communication Systems, Technische Universität Dresden, Germany
{first name.last name}@ifn.et.tu-dresden.de

***Abstract*—Orthogonal time frequency space modulation (OTFS) has been recently proposed to achieve time and frequency diversity, especially in linear time-variant (LTV) channels with large Doppler frequencies. The idea is based on the precoding of the data symbols using symplectic finite Fourier transform (SFFT) then transmitting them using orthogonal frequency division multiplexing (OFDM) waveform. Consequently, the demodulator and channel equalization can be coupled in one processing step. As a distinguished feature, the demodulated data symbols have roughly equal gain independent of the channel selectivity. On the other hand, generalized frequency division multiplexing (GFDM) modulation also employs the spreading over the time and frequency domains using circular filtering. Accordingly, the data symbols are implicitly precoded in a similar way as applying SFFT in OTFS. In this paper, we present an extended representation of GFDM which shows that OTFS can be processed as a GFDM signal with simple permutation. Nevertheless, this permutation is the key factor behind the outstanding performance of OTFS in LTV channels, as demonstrated in this work. Furthermore, the representation of OTFS in the GFDM framework provides an efficient implementation, that has been intensively investigated for GFDM, and facilitates the understanding of the OTFS distinct features.**

***Index Terms*—GFDM, OTFS**

I. Introduction

In the contention between 5G waveform candidates, several modulation techniques were proposed to attain the requirements for different use cases. Some of the waveforms focus on providing very low out-of-band (OOB) emission, e.g. filter bank multicarrier (FBMC) [1], which allows asynchronous multiple access. Other designs concern about the implementation complexity, which leads to several proposals based on orthogonal frequency division multiplexing (OFDM), such as windowed-OFDM [2], filtered-OFDM [3]. Furthermore, generalized frequency division multiplexing (GFDM) [4] was first proposed as an alternative to OFDM to improve the spectral efficiency by reducing the cyclic prefix (CP) overhead of OFDM. However, due to the flexibility in tuning the different parameters, such as the number of subcarriers and subsymbols, the prototype pulse shape and the active subsymbol set, GFDM can be reconfigured to meet different requirements [5]. This flexibility inspires the design and implementation of unified multicarrier framework based on the GFDM model.

The work presented in this paper has been performed in the framework of the ORCA project [https://www.orca-project.eu/].This project has received funding from the Eropean Union's Horizon 2020 research and innovation programme under grant agreement No 732174.

Orthogonal time frequency space modulation (OTFS) modulation technique has been recently proposed in [6] to deal with high mobility scenarios. In this approach, the data symbols are spread in the time and frequency domains using symplectic finite Fourier transform (SFFT)-based precoding. As a consequence, high diversity gain is achieved. The spread data symbols are then transmitted with OFDM waveform. Compared to OFDM, a very significant gain in terms of frame error rate (FER) is shown in high mobility case [6]. Furthermore, the processing is of relatively low cost, where the channel equalization and demodulation are coupled at the receiver. Interestingly, all the estimated symbols have the same signal-to-noise ratio (SNR) at the output of the equalizer. This feature can not be achieved with OFDM unless power allocation techniques are applied. However, this requires the channel knowledge at the transmitter, which is infeasible under higher Doppler frequencies.

Because GFDM is based on circular filtering in the time and frequency domains, the data symbols are implicitly spread in a similar way as in OTFS. This motivates us to investigate the structure of GFDM and reveal a relation between both systems. As we show in this paper, the OTFS samples can be generated from a permutation of GFDM signal, which exchanges the role of the number of subcarriers in OTFS to be the number of subsymbols in the GFDM terminology and vice versa. This observation is relevant for two reasons; first, a slight modification of a real-time flexible implementation of GFDM transceiver [7], enables the processing of OTFS. Second, the properties of OTFS can be clearly derived from the structure of GFDM. For example, we show that the coupled equalizer and demodulator are equivalent to circular filtering of the received OTFS signal using a GFDM receive filter computed from the transmit pulse and the estimated channel.

The remainder of the paper is organized as follows: in Section II we provide a short overview of the conventional GFDM. Section III is dedicated to the advanced representation of GFDM and extended framework. The relation between the conventional GFDM and OTFS is discussed in Section IV. Numerical comparisons through simulation are introduced in Section V. Finally, Section VI concludes the paper.

II. GFDM overview

In the GFDM modulation, a time-frequency resource block of duration T and bandwidth B is used to convey a message of maximum N data symbols. For that, the frequency band

is divided into K equally spaced subcarriers with subcarrier spacing $\Delta f = \frac{B}{K}$, and the time duration is divided into M equally spaced subsymbols with subsymbol spacing $T_{\text{sub}} = \frac{T}{M}$ such that $\Delta f T_{\text{sub}} = 1$. Each pair of subcarrier-subsymbol (k, m) is used to modulate one data symbol $d_{k,m}$ using a pulse shape $g_{k,m}(t)$, and thus, $N = MK$. The pulse shapes are generated by a shift in the time and frequency domains of a periodic prototype pulse shape $g_T(t)$, in addition to windowing to confine the block within the duration T, thus

$$g_{k,m}(t) = w_T(t) g_T(t - mT_{\text{sub}}) e^{j2\pi k \Delta f t}, \tag{1}$$

where $w_T(t)$ is a rectangular window of duration T. Accordingly, one GFDM block in the time domain is expressed as

$$x(t) = w_T(t) \sum_{k=-K/2}^{K/2-1} \sum_{m=0}^{M-1} d_{k,m} g_T(t - mT_{\text{sub}}) e^{j2\pi k \Delta f t}. \tag{2}$$

based on this representation, GFDM can be seen as generalization of OFDM, where $g_T(t) = 1$ and $M = 1$. The discrete-time representation can be achieved by using a sampling frequency $F_s = B$. The samples of one GFDM block can be represented in a vector $\boldsymbol{x} \in \mathbb{C}^{N \times 1}$ such that [4],

$$[\boldsymbol{x}]_{(n)} = x(\tfrac{n}{B}) = \sum_{k=0}^{K-1} \sum_{m=0}^{M-1} d_{k,m} g[< n - mK >_N] e^{j2\pi \frac{k}{K} n}. \tag{3}$$

Here, $< . >_N$ denotes the modulo-N operator. The corresponding representation in the frequency domain is given by

$$[\tilde{\boldsymbol{x}}]_{(n)} = \sum_{k=0}^{K-1} \sum_{m=0}^{M-1} d_{k,m} \tilde{g}[< n - kM >_N] e^{-j2\pi \frac{m}{M} n}. \tag{4}$$

The GFDM demodulator applies circular filtering on the received signal $y[n]$ with a receiver pulse $\boldsymbol{\gamma}[n]$ such that,

$$\hat{d}_{k,m} = \sum_{n=0}^{N-1} y[n] \gamma^*[< n - mK >_N] e^{-j2\pi \frac{k}{K} n}. \tag{5}$$

In the conventional matrix representation, the GFDM block can be expressed using a matrix $\boldsymbol{A} \in \mathbb{C}^{N \times N}$ as

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{d}, [\boldsymbol{A}]_{(n,k+mK)} = g[< n - mK >_N] e^{j2\pi \frac{k}{K} n}, \tag{6}$$

where $\boldsymbol{d} = \text{vec}\{\boldsymbol{D}\}$ with $[\boldsymbol{D}]_{(k,m)} = d_{k,m}$. Moreover, the demodulation matrix $\boldsymbol{B} \in \mathbb{C}^{N \times N}$ has a similar structure and is generated from the receiver pulse γ as

$$[\boldsymbol{B}]_{(n,k+mK)} = \gamma[< n - mK >_N] e^{j2\pi \frac{k}{K} n}, \ \hat{\boldsymbol{d}} = \boldsymbol{B}^H \boldsymbol{y}. \tag{7}$$

III. GFDM ALTERNATIVE REPRESENTATION

In this section, we introduce an alternative matrix model by arranging the GFDM block $\boldsymbol{x}$ in a matrix instead of a vector. Initially, we define the following auxiliary matrices for a generic vector $\boldsymbol{a} \in \mathbb{C}^{PQ \times 1}$ as visualized in Fig. 1,

$$\boldsymbol{V}_{P,Q}^{(\boldsymbol{a})} = \text{unvec}_{Q \times P}\{\boldsymbol{a}\}^T \Leftrightarrow \left[\boldsymbol{V}_{P,Q}^{(\boldsymbol{a})}\right]_{(p,q)} = [\boldsymbol{a}]_{(q+pQ)}, \tag{8}$$

$$\boldsymbol{Z}_{P,Q}^{(\boldsymbol{a})} = \boldsymbol{F}_P \boldsymbol{V}_{P,Q}^{(\boldsymbol{a})}, \ \bar{\boldsymbol{Z}}_{Q,P}^{(\tilde{\boldsymbol{a}})} = \frac{1}{Q} \boldsymbol{F}_Q^H \boldsymbol{V}_{Q,P}^{(\tilde{\boldsymbol{a}})}, \tag{9}$$

where $\tilde{\boldsymbol{a}} = \boldsymbol{F}_{PQ}\boldsymbol{a}$, $\text{unvec}_{Q \times P}\{\boldsymbol{a}\}$ denotes the inverse of vectorization operation, $\boldsymbol{F}_P$ is the P-point discrete Fourier transform (DFT) matrix, defined by $[\boldsymbol{F}_P]_{(i,j)} = e^{-j2\pi \frac{ij}{P}}$. The matrix $\boldsymbol{Z}_{P,Q}^{(\boldsymbol{a})}$, $\bar{\boldsymbol{Z}}_{Q,P}^{(\tilde{\boldsymbol{a}})}$ are known as the discrete Zak transform (DZT) [8] of $\boldsymbol{a}$ and $\tilde{\boldsymbol{a}}$, respectively.

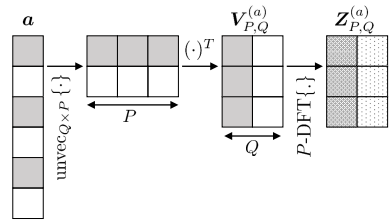


Fig. 1: Visualization of (8) and (9).

A. Time domain representation

The GFDM block equation (3) can be reformulated by representing n with two indexes $q = 0, \cdots, K-1$ and $p = 0 \cdots, M-1$, such that $n = q + pK$. Thereby,

$$[\boldsymbol{x}]_{(q+pK)} = \sum_{m=0}^{M-1} \sum_{k=0}^{K-1} d_{k,m} g[< q + pK - mK >_N] e^{j2\pi \frac{k}{K} q}$$

Using the notations in (8), then

$$\begin{aligned} \left[\boldsymbol{V}_{M,K}^{(\boldsymbol{x})}\right]_{(p,q)} &= \sum_{m=0}^{M-1} \left[\boldsymbol{V}_{M,K}^{(\boldsymbol{g})}\right]_{(<p-m>_M, q)} \sum_{k=0}^{K-1} [\boldsymbol{D}]_{(k,m)} e^{j2\pi \frac{k}{K} q} \\ &= \sum_{m=0}^{M-1} \left[\boldsymbol{V}_{M,K}^{(\boldsymbol{g})}\right]_{(<p-m>_M, q)} \left[\boldsymbol{D}^T \boldsymbol{F}_K^H\right]_{(m,q)}. \end{aligned}$$

The second line defines a circular convolution between the q-th column of $\boldsymbol{V}_{M,K}^{(\boldsymbol{g})}$ and the q-th column of $\boldsymbol{D}^T \boldsymbol{F}_K^H$, which can be expressed in the frequency domain with M-DFT as

$$\left[\boldsymbol{F}_M \boldsymbol{V}_{M,K}^{(\boldsymbol{x})}\right]_{(p,q)} = \left[\boldsymbol{F}_M \boldsymbol{V}_{M,K}^{(\boldsymbol{g})}\right]_{(q,q)} \cdot \left[\boldsymbol{F}_M \boldsymbol{D}^T \boldsymbol{F}_K^H\right]_{(p,q)}. \tag{10}$$

Using the notation in (9), we get

$$\boldsymbol{V}_{M,K}^{(\boldsymbol{x})} = \frac{1}{MK} \boldsymbol{F}_M^H \left(K \boldsymbol{Z}_{M,K}^{(\boldsymbol{g})} \odot \left[\boldsymbol{F}_M \boldsymbol{D}^T \boldsymbol{F}_K^H\right]\right). \tag{11}$$

Here, $\odot$ denotes the element-wise multiplication operator.

B. Frequency domain representation

Following similar derivation on $\tilde{\boldsymbol{x}}$ defined in (4), we get

$$\left[\boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{x}})}\right]_{(q,p)} = \sum_{k=0}^{K-1} \left[\boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{g}})}\right]_{(<q-k>_K, p)} \sum_{m=0}^{M-1} [\boldsymbol{D}]_{(k,m)} e^{-j2\pi \frac{m}{M} p}.$$

$$\text{Hence, } \boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{x}})} = \frac{1}{K} \boldsymbol{F}_K \left(K \bar{\boldsymbol{Z}}_{K,M}^{(\tilde{\boldsymbol{g}})} \odot \left[\boldsymbol{F}_K^H \boldsymbol{D} \boldsymbol{F}_M\right]\right). \tag{12}$$

C. New interpretation of GFDM

Based on (11) and (12), GFDM modulation can be split onto four steps, as shown in Fig. 2

1) Data spreading : the spreading is achieved by applying DFT on the rows and inverse discrete Fourier transform (IDFT) on the columns of $\boldsymbol{D}$. Therefore, we get the spread data matrix

$$\boldsymbol{D}_s = \frac{1}{K} \boldsymbol{F}_K^H \boldsymbol{D} \boldsymbol{F}_M. \tag{13}$$

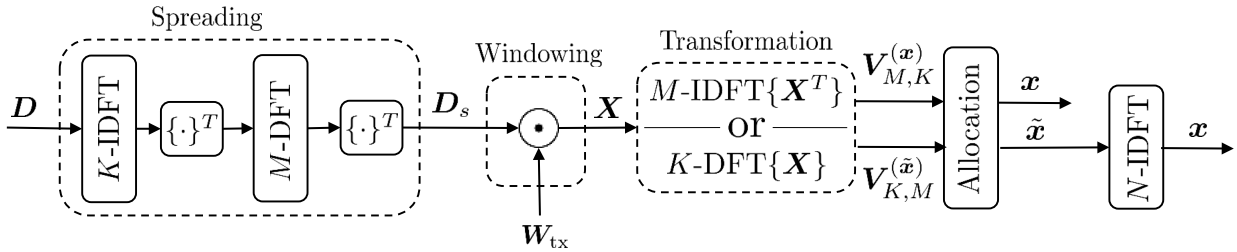


Fig. 2: GFDM modulator in 4 steps

2) Windowing: the spread data matrix $\boldsymbol{D}_s$ is element-wise multiplied with a transmitter windowing matrix $\boldsymbol{W}_{\text{tx}} \in \mathbb{C}^{K\times M}$, which is generated based on the prototype pulse shape: $\boldsymbol{X} = \boldsymbol{W}_{\text{tx}} \odot \boldsymbol{D}_s$. The entries of $\boldsymbol{W}_{\text{tx}}$ depends on the implementation domain,

$$\boldsymbol{W}_{\text{tx}}^{(\text{TD})} = K\boldsymbol{Z}_{M,K}^{(\boldsymbol{g})}{}^T \quad , \boldsymbol{W}_{\text{tx}}^{(\text{FD})} = K\boldsymbol{Z}_{K,M}^{(\tilde{\boldsymbol{g}})}. \tag{14}$$

3) Transformation: the matrix $\boldsymbol{X}$ can be seen as frequency domain blocks of length M in rows, or as time domain symbols of length K in columns. Thus, a final M-DFT or K-IDFT is applied to obtain the block samples, as follows

$$\boldsymbol{V}_{M,K}^{(\boldsymbol{x})} = \tfrac{1}{M}\boldsymbol{F}_M^H\boldsymbol{X}^T, \quad \boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{x}})} = \boldsymbol{F}_K\boldsymbol{X} \ . \tag{15}$$

4) Allocation: the final vector is achieved by allocating the generated samples to the corresponding indexes. Specifically,

$$\boldsymbol{x} = \text{vec}\left\{\boldsymbol{V}_{M,K}^{(\boldsymbol{x})}{}^T\right\}, \quad \tilde{\boldsymbol{x}} = \text{vec}\left\{\boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{x}})}{}^T\right\}. \tag{16}$$

Additionally, N-IDFT is required for the frequency domain implementation so the time domain signal is $\boldsymbol{x} = \frac{1}{N}\boldsymbol{F}_N^H\tilde{\boldsymbol{x}}$. The demodulator performs the inverse steps. First, the matrix $\boldsymbol{Y} \in \mathbb{C}^{K\times M}$ is constructed from the received signal $\boldsymbol{y}_{\text{eq}}$,

$$\boldsymbol{Y}^{(\text{TD})} = \left[\boldsymbol{F}_M\boldsymbol{V}_{M,K}^{(\boldsymbol{y}_{\text{eq}})}\right]^T, \quad \boldsymbol{Y}^{(\text{FD})} = \tfrac{1}{K}\boldsymbol{F}_K^H\boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{y}}_{\text{eq}})}. \tag{17}$$

Then a receive window $\boldsymbol{W}_{\text{rx}}$ is applied to $\boldsymbol{Y}$ followed by despreading, so that $\hat{\boldsymbol{D}} = \frac{1}{M}\boldsymbol{F}_K\left(\boldsymbol{W}_{\text{rx}} \odot \boldsymbol{Y}\right)\boldsymbol{F}_M^H$, which is an alternative representation of the circular demodulation in (5). Table I summarizes all the processing steps.

D. Extended flexibility of GFDM

In addition to the main configuration parameters K, M and the prototype pulse shape, further degrees of freedom can be exploited to extend the flexibility of GFDM modem, namely:

- Flexible spreading: the spreading can be altered to enable or disable the spreading matrices, as in [9].
- Flexible transformation: the DFT or IDFT transformation can be turned on or off to get more options.
- Flexible allocation: the conventional allocation preserves the spectral characteristics of the GFDM signal. However, the mapping can be customized by changing the distribution of the samples in the transmitted signal. This extended flexibility is exploited in the next section to generate the OTFS signal.

IV. OTFS in the GFDM framework

A. OTFS overview

In the OTFS modulation [6], a packet burst of duration $T = N_oT_o$ and bandwidth $B = M_o\Delta f_o$ is used to transmit the data symbols $\boldsymbol{D}_o \in \mathbb{C}^{N_o\times M_o}$. Although OTFS is originally designed in the delay-Doppler domain, it can be simply introduced in the time-frequency representation with three-step processing. First, the data symbols are spread with the inverse of SFFT to generate the spread data matrix $\boldsymbol{D}_{s_o}$ defined by [1]

$$\boldsymbol{D}_{s_o} = \frac{1}{N_o}\boldsymbol{F}_{N_o}^H\boldsymbol{D_o}\boldsymbol{F}_{M_o} \in \mathbb{C}^{N_o\times M_o}. \tag{18}$$

Then, a transmit window $\boldsymbol{W}_{\text{tx}} \in \mathbb{C}^{N_o\times M_o}$ can be applied,

$$\boldsymbol{X}_o = \boldsymbol{W}_{\text{tx}} \odot \boldsymbol{D}_{s_o}. \tag{19}$$

Finally, the time-domain signal is generated with a multicarrier modulation of M_o subcarriers and N_o subsymbols as

$$s(t) = \sum_{m=-\frac{M_o}{2}}^{\frac{M_o}{2}-1}\sum_{n=0}^{N_o-1}\left[\boldsymbol{X}_o\right]_{(n,m)} g_{\text{tx}}(t-nT_o)e^{j2\pi m\Delta f_o(t-nT_o)}.$$

Despite of its general representation, it is implicitly considered, as stated in the works related to OTFS, e.g. [10], that $g_{\text{tx}}(t)$ is a rectangular pulse of duration T_o and $T_o\Delta f_o = 1$. Therefore, the discrete OTFS block is given by

$$[\boldsymbol{s}]_{(l)} = \sum_{m=0}^{M_o-1}\sum_{n=0}^{N_o-1}\left[\boldsymbol{X}_o^T\right]_{(m,n)} g_{\text{tx}}[< l-nM_o >_{(N_oM_o)}]e^{j2\pi\frac{ml}{M_o}}.$$

This shows that the OTFS block is composed of N_o consecutive OFDM symbols of length M_o. Here, the CP is not considered. Let $\boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{x}_o)} = \frac{1}{M_o}\boldsymbol{F}_{M_o}^H\boldsymbol{X}_o^T$, with $\boldsymbol{x}_o = \text{vec}\left\{\{\boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{x}_o)}\}^T\right\}$, then

$$\boldsymbol{s} = \text{vec}\left\{\boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{x}_o)}\right\}. \tag{20}$$

B. OTFS relation to GFDM

Comparing the OTFS samples (20) with the GFDM samples (15), we reveal that the OTFS samples can be generated using the first three steps of the GFDM modulator with the parameters $M = M_o$, $K = N_o$. The difference between both blocks is in the allocation step. Actually, the OTFS block is a

[1]The subscript o is added to [6] to distinguish the OTFS terminology.

TABLE I: The four steps of conventional GFDM modulator.

	Modulation		Demodulation	
	TD implementation	FD implementation	TD implementation	FD implementation
Spread.	$\boldsymbol{D}_s = \frac{1}{K}\boldsymbol{F}_K^H \boldsymbol{D}\boldsymbol{F}_M$		$\hat{\boldsymbol{D}} = \frac{1}{M}\boldsymbol{F}_K \hat{\boldsymbol{D}}_s \boldsymbol{F}_M^H$	
Wind.	$\boldsymbol{W}_{\text{tx}} = K\{\boldsymbol{Z}_{M,K}^{(\boldsymbol{g})}\}^T$	$\boldsymbol{W}_{\text{tx}} = K\bar{\boldsymbol{Z}}_{K,M}^{(\tilde{\boldsymbol{g}})}$	$\boldsymbol{W}_{\text{rx}}$ generated from $\boldsymbol{W}_{\text{tx}}$	
	$\boldsymbol{X} = \boldsymbol{W}_{\text{tx}} \odot \boldsymbol{D}_s$		$\hat{\boldsymbol{D}}_s = \boldsymbol{W}_{\text{rx}} \odot \boldsymbol{Y}$	
Trans.	$\boldsymbol{V}_{M,K}^{(\boldsymbol{x})} = \frac{1}{M}\boldsymbol{F}_M^H \boldsymbol{X}^T$	$\boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{x}})} = \boldsymbol{F}_K \boldsymbol{X}$	$\boldsymbol{Y} = \left[\boldsymbol{F}_M \boldsymbol{V}_{M,K}^{(\boldsymbol{y}_{\text{eq}})}\right]^T$	$\boldsymbol{Y} = \frac{1}{K}\boldsymbol{F}_K^H \boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{y}}_{\text{eq}})}$
Alloc.	$\boldsymbol{x} = \text{vec}\left\{\{\boldsymbol{V}_{M,K}^{(\boldsymbol{x})}\}^T\right\}$	$\tilde{\boldsymbol{x}} = \text{vec}\left\{\{\boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{x}})}\}^T\right\}$	$\left[\boldsymbol{V}_{M,K}^{(\boldsymbol{y}_{\text{eq}})}\right]_{(m,k)} = [\boldsymbol{y}_{\text{eq}}]_{(k+mK)}$	$\left[\boldsymbol{V}_{K,M}^{(\tilde{\boldsymbol{y}}_{\text{eq}})}\right]_{(k,m)} = [\tilde{\boldsymbol{y}}_{\text{eq}}]_{(m+kM)}$

permutation of the GFDM block $\boldsymbol{x}_o$ with square the commutation matrix $\boldsymbol{P}_{N_o,M_o} \in \Re^{N_oM_o \times N_oM_o}$, i.e. $\boldsymbol{s} = \boldsymbol{P}_{N_o,M_o}\boldsymbol{x}_o$,

$$\text{where } \boldsymbol{x}_0 = \text{vec}\left\{\boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{x}_o)}{}^T\right\} = \boldsymbol{P}_{N_o,M_o}^T \text{vec}\left\{\boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{x}_o)}\right\}. \quad (21)$$

C. Received signal model

The CP insertion is performed per block in the case of GFDM, while a CP is added to each OFDM symbol in OTFS. The received OTFS block $\boldsymbol{r}$ after removing the CP from each OFDM symbol can be expressed in a matrix form as

$$\boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{y}_o)} = \text{unvec}_{M_o \times N_o}\{\boldsymbol{r}\},\ \boldsymbol{y}_0 = \text{vec}\left\{\boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{y}_o)}{}^T\right\} = \boldsymbol{P}_{N_o,M_o}^T \boldsymbol{r}.$$

Consider a linear time-variant (LTV) channel with the response $h(l,n)$ of L delay taps. Following the derivation steps in the appendix on each OFDM symbol, we get

$$\boldsymbol{F}_{M_o}\boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{y}_o)} = \tilde{\boldsymbol{H}}_o^{(e)} \odot \boldsymbol{X}_o^T + \boldsymbol{E}_{d_o} + \boldsymbol{V}_o \quad (22)$$

where $\tilde{\boldsymbol{H}}^{(e)} \in \mathbb{C}^{M \times K}$ is the equivalent channel defined by

$$\left[\tilde{\boldsymbol{H}}_o{}^{(e)}\right]_{(p,q)} = \sum_{m=0}^{M_o-1}\sum_{l=0}^{L-1} h(l, N_{cp}+m+q(M_o+N_{cp}))e^{-j2\pi\frac{lp}{M_o}},$$

$\boldsymbol{E}_{d_o}$ denotes the interference terms resulting from Doppler spread and $\boldsymbol{V}_o$ is the additive noise samples. Furthermore, let

$$\boldsymbol{W}_{\text{tx}}^{(e)} = \tilde{\boldsymbol{H}}_o^{(e)T} \odot \boldsymbol{W}_{\text{tx}},\ \boldsymbol{X}_o^{(e)} = \boldsymbol{W}_{\text{tx}}^{(e)} \odot \boldsymbol{D}_{s_o}, \quad (23)$$

$$\begin{aligned} \text{then } \boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{y}_o)} &= \frac{1}{M_o}\boldsymbol{F}_{M_o}^H \boldsymbol{X}_o^{(e)T} + \boldsymbol{E}_{d_o} + \boldsymbol{V}_o \\ &= \boldsymbol{V}_{M_o,N_o}^{(\boldsymbol{x}_o^{(e)})} + \boldsymbol{E}_{d_o} + \boldsymbol{V}_o \end{aligned} \quad (24)$$

where, $\boldsymbol{x}_o^{(e)}$ is an equivalent time-domain GFDM block generated by the window $\boldsymbol{W}_{\text{tx}}^{(e)}$ (15). Furthermore, the relation

$$\boldsymbol{y}_o = \boldsymbol{P}_{N_o,M_o}^T \cdot \boldsymbol{r} = \boldsymbol{x}_o^{(e)} + \boldsymbol{\epsilon}_{d_o} + \boldsymbol{v}_o, \quad (25)$$

shows that the received OTFS block can be processed with a ready implemented real-time GFDM receiver, e.g. [7], under additive noise channel. In this case, the minimum mean square error (MMSE) receive pulse shape can be efficiently calculated based on the inverse Zak transform of $\boldsymbol{W}_{\text{tx}}^{(e)}$ [11]. Moreover, due to the circular filtering in the combined equalization and demodulation, the symbols at the output achieve roughly the same SNR considering the additional interference. The received GFDM block after removing the CP is:

$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{h}}^{(e)} \odot \tilde{\boldsymbol{x}} + \tilde{\boldsymbol{\epsilon}}_d + \tilde{\boldsymbol{v}}, \quad (26)$$

$$\text{where } \left[\tilde{\boldsymbol{h}}^{(e)}\right]_{(q)} = \frac{1}{N}\sum_{n=0}^{N-1}\sum_{l=0}^{L-1} h(l, N_{cp}+n)e^{j2\pi\frac{lq}{N}}. \quad (27)$$

Thereby, additional channel equalization processing is required. However, the overall equalization and demodulation is not necessarily equivalent to circular filtering and the data symbols at the output of the demodulator have different SNRs.

$$\begin{aligned} \boldsymbol{Y}_o &= \boldsymbol{H} \odot \boldsymbol{W}_{\text{tx}} \odot \boldsymbol{D}_{s_o} + \boldsymbol{V} \\ \boldsymbol{s} &= \left(\boldsymbol{F}_N^H \otimes \boldsymbol{I}_M\right)\boldsymbol{\Lambda}^{(g_t x)}\boldsymbol{\Pi}\boldsymbol{x} \end{aligned} \quad (28)$$

were $\boldsymbol{\Pi}$ is row-column permutation matrix, and $\boldsymbol{\Lambda}^{(g_t x)} = \text{diag}\left\{\text{vec}\left\{\boldsymbol{Z}^{(g_{tx})T}\right\}\right\}$.

V. Numerical simulation

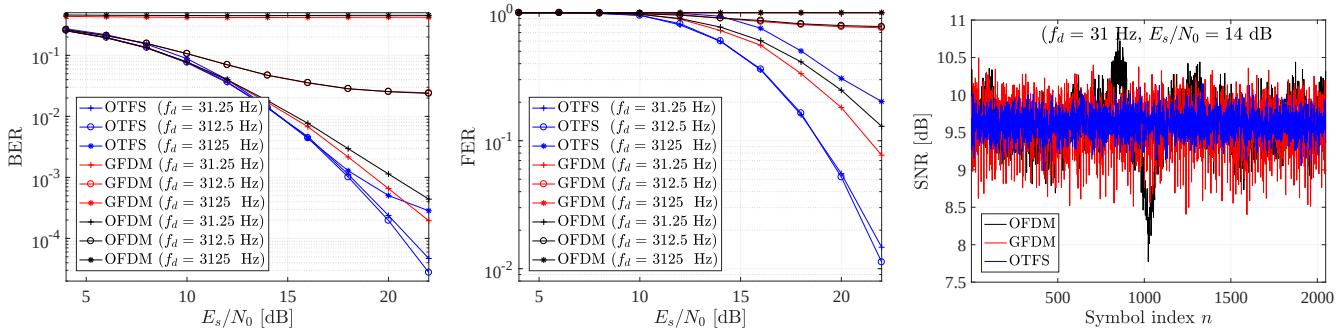
The purpose of this section is first to compare the performance of the conventional GFDM[2] and OTFS systems generated from the same parameters. The simulation parameters are listed in Table II. The data bits are encoded with Long Term Evolution (LTE)-Turbo code of code rate $1/2$ and mapped to 16-QAM symbols. The transmit window is generated from a prototype pulse g, which is a periodic raised-cosine (RC) with roll-off factor $\alpha = 0$. The samples in both systems are generated from the GFDM modulator depicted in Fig. 2 using the parameters $K = N_o$ and $M = M_o$. The MMSE receiver in the case of OTFS is performed using MMSE pulse shape generated based on the equivalent window $\boldsymbol{W}_{\text{tx}}^{(e)}$ (23), while the GFDM-MMSE receiver considers the linear model [5]

$$\tilde{\boldsymbol{y}} = \text{diag}\left\{\tilde{\boldsymbol{h}}^{(e)}\right\}\tilde{\boldsymbol{A}}\boldsymbol{d} + \tilde{\boldsymbol{\epsilon}}_d + \tilde{\boldsymbol{v}}. \quad (29)$$

The demodulated symbols are fed to soft-input decoder. Both modulation techniques are evaluated in the extended vehicular A channel model (EVA) [12] with the consideration of ideal channel estimation per block as in (37) in the appendix. The SNR is defined by the ratio E_s/N_0, where E_s is the average symbol power and N_0 is the additive white noise power.

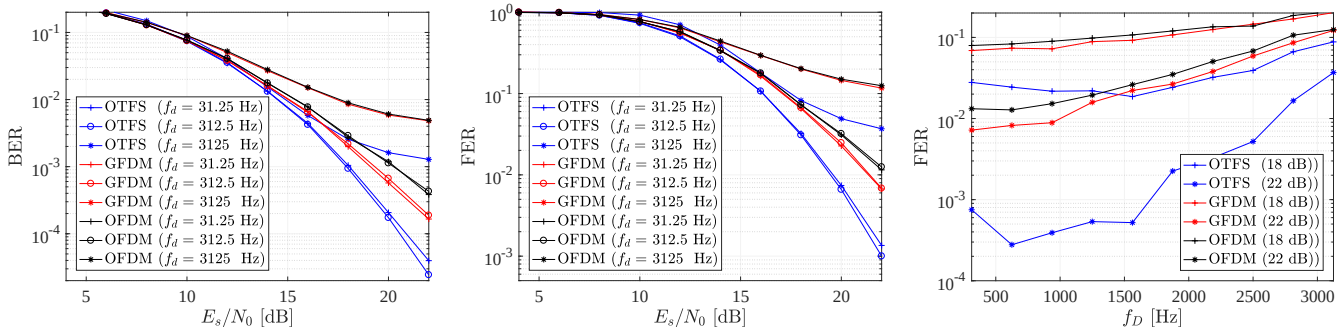
$$\text{NMSE} = \frac{\mathbb{E}\left[\|\hat{\boldsymbol{d}} - \boldsymbol{d}\|^2\right]}{\mathbb{E}\left[\|\boldsymbol{d}\|^2\right]},\ \text{SNR}[n] = \frac{\mathbb{E}\left[|\hat{d_n} - d_n|^2\right]}{\mathbb{E}\left[|d_n|\right]^2}. \quad (30)$$

[2]In this section, GFDM refers to the conventional GFDM waveform [4].



(a) BER vs. SNR. (b) FER vs. SNR. (c) SNR per symbol.

Fig. 3: Performance evaluation of OFDM, GFDM and OTFS with long frames.



(a) BER vs. SNR. (b) FER vs. SNR. (c) FER vs. Doppler frequency.

Fig. 4: Performance evaluation of OFDM, GFDM and OTFS with short frames.

TABLE II: Simulation parameters

OTFS	$N_o = 16,\ M_o = 128$
Long-GFDM	$K = 16,\ M = 128$
Long-OFDM	$N = 2048$
Short-GFDM	$K = 16,\ M = 8$
Short-OFDM	$N = 128$
CP length N_{CP}	32
Bandwidth $B = F_s$	8 MHz
OTFS subcarrier spacing Δf_o	62.5 KHz
Modulation and coding	16-QAM, LTE-Turbo (1/2)

Fig. 3 shows a comparison between OTFS, conventional GFDM and OFDM using long block. A frame is represented in a code word corresponding to one GFDM block. It is shown that, the Doppler has less influence on OTFS compared to the GFDM and OFDM. As discussed in the appendix, the is because of the accuracy of the equivalent channel estimation, which increases with the decrease of the product $P\nu_d$, where $\nu_d = \frac{f_d}{B}$ and P is the DFT size. In OTFS, $P = M_o$, while in the long GFDM and OFDM $P = KM_o$, and thus, GFDM and OFDM obviously suffer from higher Doppler interference. This is reflected in the bit error rate (BER) performance as shown in Fig. 3a. However, the achieved gain is at the cost of increased CP overhead in the case of OTFS. Although GFDM and OFDM achieve similar performance in term of BER for smaller Doppler shift, OTFS significantly outperforms them in term of FER as depicted in Fig. 3b. For instance, for $f_d = 31.5$Hz the FER gain is 3dB at SNR > 14dB. This can be explained, as illustrated in Fig. 3c, by the almost equal SNR per-symbols at the output of the OTFS demodulator unlike in GFDM and OFDM, where the SNR per symbol experiences significant variations. Additionally, these variations are even higher in OFDM such that GFDM achieves slightly lower FER than OFDM. Based on that, the corresponding bit errors in OFDM and GFDM can be burst, which increases the number of frame errors. But in the case of OTFS, it is more likely that an erroneous frame has higher number of bit errors. On the other hand, it can be observed that, with increased Doppler shift, e.g. $f_d = 312.5$Hz, OTFS maintains smaller FER through the exploitation of time and frequency diversity, while the GFDM and OFDM links are almost broken. Furthermore, the Doppler interference increases with the increase of the transmit power, which explains the BER floor.

However, the design of OFDM and GFDM can be updated with respect to the channel mobility conditions. Therefore, another configuration considering shorter block length of GFDM and OFDM is evaluated in Fig. 4. Moreover, the frame length in OTFS, i.e. the code word is shortened accordingly. In this scenario, GFDM and OFDM attain better performance due to the lower Doppler interference but less robust than OTFS against increased Doppler frequencies as illustrated in Fig. 4a. Moreover, due to the spreading technique in OTFS, the short frame is spread over larger time duration. As a result, the FER performance gain of OTFS is appealing as shown in Fig. 4b,

especially at high Doppler shift. For instance, the gain is higher than 4 dB at $f_d > 1500$Hz an SNR $= 18$dB, as can bee seen in Fig. 4c. On the other hand, considering additional low-latency and throughput constraints, the conventional GFDM provides acceptable performance under moderate mobility conditions.

VI. Conclusion

In this paper, we have proposed an extended GFDM framework that provides unified four-step implementation structure in the time and frequency domains. The OTFS modulation, which is a powerful transmission technique in LTV channels, can be processed with the GFDM framework with simple modification. Namely, it is shown that the OTFS block results from a permutation of the corresponding conventional GFDM block and the received OTFS block can be represented as a GFDM block in additive noise channel. Thus, the received OTFS signal can be fed to the conventional GFDM demodulator with a reconfigurable receive pulse shape that depends on the channel estimation. As a result, the OTFS equalization and demodulation are combined in a circular filtering that produces data symbols with equal SNR. In addition, the MMSE-OTFS receiver can be implemented with low complexity. The simulation results show that OTFS outperforms the counterparts long-GFDM and long-OFDM in terms of BER and FER. Furthermore, OTFS significantly outperforms the short blocks of GFDM and OFDM in high mobility scenarios, due to the exploitation of time and frequency diversity. However, GFDM can be a reasonable choice for moderate mobility and low-latency use cases as it outperforms OFDM in term of FER.

In the future work, the extended GFDM framework is used to develop alternative forms of OTFS with applications for multiuser scenarios.

References

[1] B. Farhang-Boroujeny, "OFDM versus filter bank multicarrier," *SPM*, vol. 28, no. 3, pp. 92–112, 2011.

[2] Y. Medjahdi *et al.*, "Wola processing: A useful tool for windowed waveforms in 5G with relaxed synchronicity," in *ICC Workshops*. IEEE, 2017, pp. 393–398.

[3] T. Wild *et al.*, "5G air interface design based on universal filtered (UF-) OFDM," in *Digital Signal Processing (DSP), 2014 19th International Conference on*. IEEE, 2014, pp. 699–704.

[4] N. Michailow *et al.*, "Generalized Frequency Division Multiplexing for 5th Generation Cellular Networks," *IEEE Trans. Commun.*, vol. 62, no. 9, pp. 3045–3061, Sep. 2014.

[5] D. Zhang, *et al.*, "A study on the link level performance of advanced multicarrier waveforms under MIMO wireless communication channels," *TSP*, vol. 16, no. 4, pp. 2350–2365, 2017.

[6] R. Hadani, *et al.*, "Orthogonal time frequency space modulation," in *IEEE WCNC*. IEEE, 2017, pp. 1–6.

[7] M. Danneberg *et al.*, "Flexible GFDM implementation in FPGA with support to run-time reconfiguration," in *IEEE VTC Fall*, 2015, pp. 1–2.

[8] H. Bölcskei and F. Hlawatsch, "Discrete Zak transforms, polyphase transforms, and applications," *IEEE Trans. Signal Process.*, vol. 45, no. 4, pp. 851–866, apr 1997.

[9] M. Matthé *et al.*, "Precoded GFDM transceiver with low complexity time domain processing," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, p. 138, 2016.

[10] A. Farhang, *et al.*, "Low complexity modem structure for OFDM-based orthogonal time frequency space modulation," *IEEE Commun. Lett.*, 2017.

[11] M. Matthé *et al.*, "Generalized frequency division multiplexing in a Gabor transform setting," *IEEE Commun. Lett.*, vol. 18, no. 8, pp. 1379–1382, 2014.

[12] 3GPP, "Base Station (BS) radio transmission and reception," Technical Specification (TS) 36.104, 05 2008, v8.2.0.

Appendix

Consider a signal $x^{(cp)}[n],\ n = 0 \cdots N + N_{\text{cp}} - 1$ with $x^{(cp)}[n] = x[< n - N_{\text{cp}} >_N]$ transmitted through LTV channel defined by the delay-Doppler response $H(l, \nu)$, where ν is the normalized Doppler frequency, i.e. $\nu = \frac{f}{F_s}$, the received signal can be expressed as

$$r[n] = \int_\nu \sum_{l=0}^{L-1} H(l,\nu) x^{(cp)}[n-l] e^{j2\pi n\nu} d\nu, \tag{31}$$

with $L \leq N_{\text{cp}} - 1$ is the maximum delay spread. Then we extract the samples $y[n]$ as $y[n] = r[N_{\text{cp}} + n],\ n = 0 \cdots N-1$,

$$\begin{aligned} \text{then}\ \ y[n] &= \int_\nu \sum_{l=0}^{L-1} H(l,\nu) x[< n-l >_N] e^{j2\pi[N_{\text{cp}}+n]\nu} d\nu \\ &= \frac{1}{N} \int_\nu \sum_{k=0}^{N-1} \tilde{H}(k,\nu) \tilde{x}[k] e^{j2\pi \frac{kn}{N}} e^{j2\pi[N_{\text{cp}}+n]\nu} d\nu. \end{aligned} \tag{32}$$

Here, $\tilde{H}(k,\nu) = \sum_{l=0}^{L-1} H(l,\nu) e^{-j2\pi \frac{kl}{N}}$. This corresponds to removing the CP and using N-DFT representation for circular convolution, which is attained due to the CP. The DFT of the received signal is

$$\begin{aligned} \tilde{y}[q] &= \sum_{n=0}^{N-1} y[n] e^{-j2\pi \frac{nq}{N}} \\ &= \frac{1}{N} \int_\nu \sum_{k=0}^{N-1} \tilde{H}(k,\nu) \tilde{x}[k] e^{j2\pi N_{\text{cp}}\nu} \sum_{n=0}^{N-1} e^{j2\pi \frac{n(k-q+N\nu)}{N}} d\nu. \end{aligned}$$

Assuming that the maximum Doppler spread f_d satisfies $N|\nu_D| \ll 1$, where $\nu_d = \frac{f_d}{F_s}$, then,

$$\sum_{n=0}^{N-1} e^{j2\pi \frac{(k-q)n}{N}} e^{j2\pi n\nu} \approx \sum_{n=0}^{N-1} e^{j2\pi n\nu} \delta(q-k), \tag{33}$$

$$\begin{aligned} \tilde{y}[q] &\approx \frac{1}{N} \sum_{n=0}^{N-1} \int_\nu \tilde{H}(q,\nu) \tilde{x}[q] e^{j2\pi N_{\text{cp}}\nu} e^{j2\pi n\nu} d\nu \\ &= \frac{1}{N} \sum_{n=0}^{N-1} \tilde{h}(q, N_{\text{cp}} + n) \tilde{x}[q]. \end{aligned} \tag{34}$$

Thus, the equivalent channel vector $\tilde{\boldsymbol{h}}^{(e)}$ in the frequency domain is

$$\begin{aligned} \left[\tilde{\boldsymbol{h}}^{(e)}\right]_{(q)} &= \frac{1}{N} \sum_{n=0}^{N-1} \tilde{h}(q, N_{\text{cp}} + n) \\ &= \frac{1}{N} \sum_{n=0}^{N-1} \sum_{l=0}^{L-1} h(l, N_{\text{cp}} + n) e^{-j2\pi \frac{ql}{N}} \end{aligned}, \tag{35}$$

where $h(l,n)$ is the channel impulse response. This corresponds to averaging over the time instance. As a result,

$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{h}}^{(e)} \odot \tilde{\boldsymbol{x}} + \boldsymbol{\epsilon}_d, \tag{36}$$

where $\boldsymbol{\epsilon}_d$ represents the interference due to the Doppler spectrum and depends on the accuracy of the approximation in (33). The equivalent channel impulse response $h^{(e)}[n]$ can be estimated by sending a Dirac pulse through the channel as

$$h^{(e)}[n] = \int_\nu \sum_{l=0}^{L-1} H(l,\nu) \delta(n-l) e^{j2\pi n\nu} d\nu. \tag{37}$$

This model can be used as an ideal channel estimation.