



HAL
open science

UFMC Transceiver Complexity Reduction

Majed Saad, Ali Al-Ghouwayel, Hussein Hijazi

► **To cite this version:**

Majed Saad, Ali Al-Ghouwayel, Hussein Hijazi. UFMC Transceiver Complexity Reduction. IEEE 25th International Conference on Telecommunications (ICT 2018), Jun 2018, Saint-Malo, France. 10.1109/ict.2018.8464863 . hal-01834666

HAL Id: hal-01834666

<https://hal.science/hal-01834666>

Submitted on 10 Jul 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UFMC Transceiver Complexity Reduction

Majed Saad, Ali Al-Ghouwayel and Hussein Hijazi

LIU, Department Computer and Communication Engineering, Beirut- Lebanon
E-mail: majed.saad@liu.edu.lb, ali.ghouwayel@liu.edu.lb, hussein.hijazi@liu.edu.lb

Abstract—UFMC is a candidate waveform technology for 5G wireless systems and beyond. It combines the simplicity of OFDM with the advantages of FBMC. However, these advantages come together with an increase in the complexity at the transmitter caused by the implementation of a filter and applying an FFT for each sub-band, whereas at the receiver it is due doubling the size of the FFT being implemented. Then a low-complexity solutions must be found. UFMC waveform and FFT pruning have been widely studied recently but separately. In this paper, the computational complexity of different UFMC implementation methods with FFT pruning is evaluated. Depending on the number of sub-bands, it is shown that a complexity reduction up to 50% of the UFMC transmitter can be obtained. Also, the UFMC receiver complexity can be reduced to be similar or even less than OFDM. This complexity reduction of the UFMC transceiver comes without performance degradation since no computational approximation is introduced.

Index Terms—5G, UFMC, Computational Complexity, pruned Fast Fourier Transform (FFT), Power consumption.

I. INTRODUCTION

Universal Filtered Multi-Carrier (UFMC) or UF-OFDM is a novel multi-carrier modulation technique, which can be seen as a generalization of filtered OFDM, and Filter Bank Multi-Carrier (FBMC). While the former filters the entire band and the latter filters each sub-carrier, UFMC filters sub-band blocks, thus groups of sub-carriers.

Briefly, UFMC provides promising advantages [1] such as good spectral efficiency similar to FBMC with less overhead, and lower Out-Of-Band (OOB) leakage than for OFDM [2].

The sub-carrier filtering in FBMC systems enhances the robustness against Inter-Carrier Interference (ICI) effects. However, typical FBMC filters have lengths multiple times of the sub-carriers number, make it disadvantageous for communication in short up-link bursts [3] like low latency communication or energy-efficient Machine Type Communication (MTC).

On the contrary, the sub-band filtering allows reducing the filter length considerably, compared to FBMC. Furthermore, QAM is still efficient for UFMC, in contrast to the FBMC case [4] where OQAM is needed, making UFMC compatible with all kinds of Multiple Input Multiple Output (MIMO) systems.

Therefore, while UFMC maintains the advantages of OFDM and avoids its drawbacks such as the strict synchronicity and orthogonality requirements and high OOB, it increases the computational complexity as UFMC employs a filter to achieve this effect. Hence, the implementation of the filter and FFT for each sub-band at the transmitter increases the complexity. Similarly, the double size of FFT at the receiver leads to approximately double complexity relative to OFDM.

The computational complexity reduction of any system such as 5G and beyond is an important goal since it directly affects the speed, the power consumption and the cost of any new device or base station. The most recent paper [5] discussing the UFMC transmitter complexity proposed an approximation that led to 3.7 OFDM complexity.

In this paper, we investigate how FFT input/output pruning can reduce the computational complexity for all existing UFMC implementations. In the literature, it was stated that FFT pruning does not lead to a notable reduction. However, for UFMC system, FFT pruning is very useful due to many FFT blocks with a high percentage of zero-inputs and unused outputs. In addition, the results show that UFMC system can have a comparable power consumption to OFDM system.

This paper is organized as follows. In Section II, FFT pruning is described and applied to existing implementation methods of UFMC, whereas computational complexity expressions are derived in section III. Section IV illustrates and interprets the different results. Finally, section V concludes the paper.

The notations adopted are as follows. We use small letter x for vectors in the time domain and capital X for the frequency domain. All matrices are in capital bold \mathbf{X} , $(\cdot)^T$ denotes transpose, and $0_{N \times M}$ denotes the all-zero matrix of size $N \times M$. $x \odot y$ denotes the Hadamard product. $(I)FFT_N$ denotes (I)FFT of length N and $(I)FFT_N^{NZ,RO}$ denotes pruned (I)FFT (PFFT) of length N with NZ nonzero inputs and RO required outputs. The symbol ‘*’ denotes linear convolution operator. $\mathcal{C}_{CM}(I-FFT)$ and $\mathcal{C}_{CA}(I-FFT)$ denote the computational complexity of $(I)FFT$ in terms of the complex multiplier and complex adder respectively.

II. PRUNING-BASED COMPLEXITY REDUCTION

In this section, FFT pruning is described and applied to existing implementation methods for UFMC transceiver to reduce the computational complexity when it is possible.

Before describing the existing methods, we will introduce FFT pruning technique at input and output which is the key point to the complexity reduction.

The complexity of FFT can be decreased by removing operations related to zero inputs and unused outputs. When the number of nonzero inputs and/or used outputs is less than the FFT size, the number of complex multipliers and adders can be reduced. This technique is referred to input/output FFT pruning and it was described by Markel [6], Skinner [7], Alves et al [8] and many others.

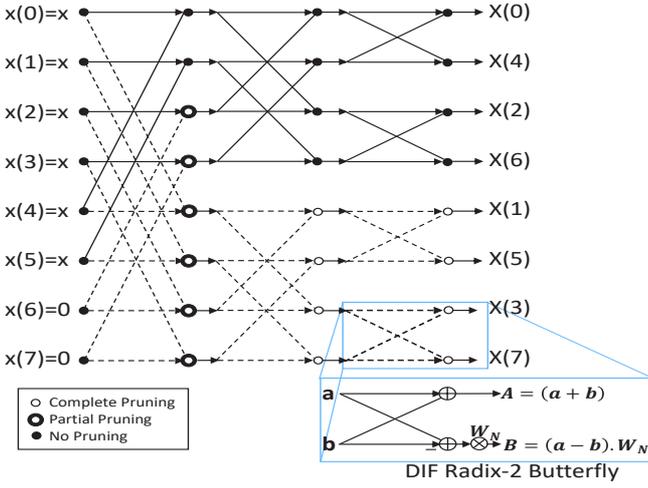


Figure 1. DIF FFT Flowchart With Input and Output Pruning

In all pruning algorithms, the basic concept is to identify the butterflies to be computed and those to be discarded. In addition, some methods can have some disadvantages in implementation such as control overhead or additional memory. However, in this paper, we are focusing on calculating the computational complexity after applying FFT pruning.

In addition, FFT can be implemented using Decimation-In-Frequency (DIF) or Decimation-In-Time (DIT). The latter can be seen as the transpose of the former. In this paper, we used DIF FFT radix-2 that contains $N/2$ butterflies per stage and $\log_2(N)$ stages, as shown in Fig. 1. The target is to have an optimum reduction by applying Complete Pruning (CP) or Complete-Partial Pruning (CPP). CP is applied when butterfly inputs are all zeros or butterfly outputs are all unused, whereas partial butterfly pruning is when some inputs are zeros or some outputs are unused. Therefore each complete butterfly pruning reduces the complexity by 1 Complex Multiplier (CM) and 2 Complex Additions (CA). However, partial input pruning can save only 2 CA and partial output pruning can save 1 CA if the used output is from the descending edge of butterfly, and 1 CA and 1 CM in the other case.

In [8], Alves et al proposed a method that takes into consideration butterflies CP only. We upgrade this method in order to include partially pruned butterflies. The new complexity of FFT CPP\CP is calculated using the MATLAB flowchart tracing method.

A. Reduced Complexity TX with Time Domain Filtering - Baseline (RC-TDF TX)

The system model of UPMC is shown in Fig. 2, where N is the total number of sub-carriers. The overall bandwidth is divided into B sub-bands and each sub-band can be allocated with N_B consecutive sub-carriers. The sub-band may correspond to Physical Resource Block (PRB) in LTE (1 PRB = 12 sub-carriers). Note that $B \cdot N_B \leq N$ is explicitly allowed and neither B nor N_B needs to divide N . An N -point IFFT operation is performed for every sub-band i where N_B data symbols X_i are modulated in the allocated sub-carrier positions for sub-band i and zeros are padded elsewhere. Then,

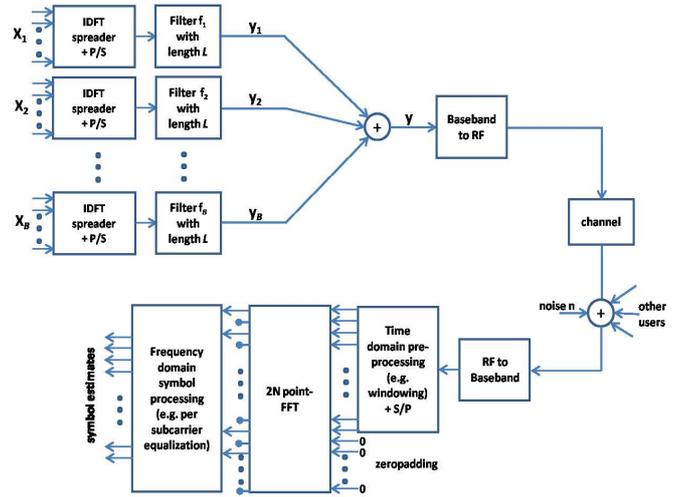


Figure 2. Baseline UPMC chain with time domain filtering at TX [2]

the output signal x_i is filtered by an FIR-filter f_i of length L which results in a sub-band signal y_i of length $N + L - 1$. Afterwards, all sub-band signals are summed-up to generate the transmitted UPMC signal y [2].

$$x_i = IFFT_N \{ [0_{[1 \times K_0 + i \cdot N_B]}, X_i^T, 0_{[1 \times (N - (i+1)N_B - K_0)]]}]^T \} \quad (1)$$

where K_0 denotes the starting frequency of the lowest sub-band.

$$y = \sum_{i=0}^{B-1} x_i * f_i = \sum_{i=0}^{B-1} y_i \quad (2)$$

The sub-band FIR filters f_i are obtained by frequency shifting the prototype filter f to i^{th} sub-band center and given by:

$$f_i[n] = f[n] \cdot e^{j2\pi \frac{0.5N_B}{N} n} \cdot e^{j2\pi \frac{K_0 + i \cdot N_B}{N} n} \quad (3)$$

The complexity of this method can be reduced by applying FFT input pruning because each N -point IFFT exhibits $(N - N_B)$ zero inputs. Hence, the described system is enhanced by implementing Eq. 1 but with IFFT input pruning $IFFT_N^{N_B, N}$.

B. TX Textbook One-Shot Implementation (OSI TX)

The UPMC transmitter can be modeled as a matrix multiplication known as Textbook One-Shot implementation where a particular UPMC symbol can be represented by:

$$y_{(N+L-1) \times 1} = \sum_{i=0}^{B-1} F_i_{(N+L-1) \times N} \cdot V_i_{N \times N_B} \cdot X_i_{N_B \times 1} \quad (4)$$

where V_i is the truncated IFFT matrix of the i^{th} sub-band which includes the relevant columns of the IFFT matrix according to the respective sub-band position within the overall available frequency range. F_i is a Toeplitz matrix, composed of the FIR filter impulse response f_i , performing the linear convolution [9].

The Eq. 4 can be adjusted to deduce the transmitted UPMC signal y by stacking all sub-bands data symbols in one vector X of size $B \cdot N_B$:

$$\Rightarrow y_{(N+L-1) \times 1} = \beta_{(N+L-1) \times B \cdot N_B} \cdot X_{B \cdot N_B \times 1} \quad (5)$$

where β can be seen as $(N + L - 1) \times B.N_B$ the matrix representation for the overall UPMC TX system.

Note that for a specific standard, the matrix β combining IFFT and filtering can be pre-computed and the pruning technique in this case cannot be applied.

C. Reduced Complexity TX with Time Domain Filtering Approximation (RC-TDFA TX)

According to [5], the idea behind this approximation is to modify the filtering process for each sub-band to be a filtering for each sub-carrier, then approximating the transmitted signal by dividing adjacent sub-carriers into groups that use a common effective filter for modulation. This approximation is done based on the assumption that the frequency response $F(f)$ of $f[n]$ is designed far wider than a single sub-carrier so that the amplitude difference between adjacent sub-carrier filters is small or even negligible. Note that the phase difference between adjacent sub-carriers is linear and can be corrected by one-tap equalizer at the receiver side which is necessary to compensate the channel effect and sub-band filter phase rotation. This extra phase correction does not require any additional complexity.

The UPMC transmitted signal before the approximation can be written as follows:

$$y[n] = \sum_{i=0}^{B-1} \sum_{k=0}^{N_B-1} X_{i,k} \cdot f_k[n] \cdot e^{j2\pi \frac{K_0+i.N_B}{N} n} \quad (6)$$

where $n = 0, \dots, N + L - 2$ and $f_k[n]$ can be understood as the effective filter used to modulate the k^{th} sub-carrier in each sub-band and is given by

$$f_k[n] = f[n] \cdot e^{j2\pi \frac{0.5N_B}{N} n} * e^{j2\pi \frac{k.N_B}{N} n} \cdot R_N[n] \quad (7)$$

where $R_N[n]$ denotes a rectangular window ranging from $n = 0, \dots, N - 1$. After some approximations, the simplified UPMC transmitted signal is given by:

$$y[n] \approx \sum_{j=1}^Q f_{k_j}[n] \underbrace{\sum_{i=0}^{B-1} \sum_{k \in Q_j} X_{i,k} \cdot e^{j2\pi \frac{K_0+i.N_B+(k-k_j)}{N} n}}_{x_j[n]} \quad (8)$$

where Q denotes the number of sub-carrier groups, $Q_j, j = 1, \dots, Q$ is the set of sub-carriers that belongs to the j^{th} sub-carrier group in each sub-band and associated by the effective filter $f_{k_j}[n]$. Note that $x_j[n]$ is effectively the expression of an N -point IFFT. The model given by Eq. 8 does not contain a costly convolution but merely a simple time-domain multiplication and hence offers the possibility for a reduced complexity implementation according to [5]. However, an additional complexity reduction is possible by using IFFT input pruning in computing $x_j[n]$ because each subgroup IFFT exhibits $B.N_B/Q$ nonzero inputs. Moreover, a method was proposed in [10] based on Eq. 6 and taking advantage of this sub-carrier processing to suggest a simplified UPMC filtering.

D. Reduced Complexity TX with Frequency Domain Filtering (RC-FDF TX)

The frequency domain implementation of sub-band signals would consist of frequency domain filtering and IFFT. Then, UPMC signal is obtained by summing up all sub-band signals.

The frequency domain signal \tilde{X}_i is oversampled by a factor N_{OS} , typically $N_{OS} = 2$, by zero-padding x_i of Eq. 1 and then applying FFT of size $N_{OS}.N$ as given by:

$$\tilde{X}_i = FFT_{N_{OS}.N} \{ [x_i^T, 0_{1 \times (N_{OS}-1)N}]^T \} \quad (9)$$

The sub-band filtering can be performed in the frequency domain by Hadamard multiplication $\tilde{Y}_{fil_i} = \tilde{F}_i \odot \tilde{X}_i$ where $\tilde{F}_i = FFT_{N_{OS}.N} \{ f_i \}$ is the oversampled frequency response sub-band filter. Then the time domain UPMC signal is obtained by summing up all sub-bands and applying $N_{OS}.N$ -point IFFT as shown by:

$$\tilde{Y}_{total} = \sum_{i=0}^{B-1} \tilde{Y}_{fil_i} \quad (10)$$

$$\tilde{y}_{total} = IFFT_{N_{OS}.N} \{ \tilde{Y}_{total} \} \quad (11)$$

where the first $N + L - 1$ samples corresponds to the UPMC transmitted signal y and the remaining ones are zeros.

In this method, (I)FFT input pruning can be applied in Eq. 1 and Eq. 9 so that the (I)FFTs in these equations are replaced by $IFFT_N^{N_B, N}$ and $FFT_{N_{OS}.N}^{N, N_{OS}.N}$ respectively.

Moreover, IFFT output pruning can be applied in Eq. 11 since only the first $N + L - 1$ outputs are required, and an $IFFT_{N_{OS}.N}^{N_{OS}.N, N+L-1}$ will be implemented.

E. Reduced Complexity TX with Frequency Domain Approximation (RC-FDA TX)

According to [9], the idea for reducing the frequency domain implementation complexity exploits the fact that sub-band signals have to be generated with a lower sample rate, compared to the full band.

This method can be summarized as follows: the first N -point IFFT for X_i in Eq. 1 with N_B data sub-carrier is replaced by N_0 -IFFT where $N_B \leq N_0 \leq N$. All further steps are adjusted to match the size $N_{OS}.N_0$. Before summing up all the filtered sub-band-component \tilde{Y}_{fil_i} of size $N_{OS}.N_0$ and proceeding with the same steps, \tilde{Y}_{fil_i} are placed at their respective frequency positions in the large $N_{OS}.N$ -point FFT.

The cut-off of frequency domain filter is one source of the approximation errors. The cut of the last $N - L + 1$ samples in this method generates a further (small) approximation error. Moreover, the error of this approximation decreases as N_0 increases and it reaches 0 when $N_0 = N$.

By applying the pruning technique, the IFFT/FFT in Eq. 1 and Eq. 9 will be $IFFT_{N_0}^{N_B, N_0}$ and $FFT_{N_{OS}.N_0}^{N_0, N_{OS}.N_0}$ respectively.

In addition, the IFFT in Eq. 11 is changed to include FFT input and output pruning as follow $IFFT_{N_{OS}.N}^{N_{OS}.N_0, N+L-1}$.

Table I
FFT COMPLEXITY WITH\WITHOUT PRUNING, FFT_N AND CONTIGUOUS NONZERO INPUT\REQUIRED OUTPUT X

Complexity	Without pruning	With Input OR Output pruning
Complex Multiplication	$\frac{N}{2} \cdot \log_2(N)$	$\frac{N}{2} \cdot \lfloor \log_2(X) \rfloor + \frac{N}{2} - X + \frac{\frac{N}{2} \cdot X}{2^{\lfloor \log_2(X) \rfloor}}$
Complex Addition	$N \cdot \log_2(N)$	$N \cdot \lfloor \log_2(X) \rfloor + N - 2X + \frac{N \cdot X}{2^{\lfloor \log_2(X) \rfloor}}$
Total Real Operations ($1CM = 4M2A$)	$5N \cdot \log_2(N)$	$5N \cdot \lfloor \log_2(X) \rfloor + 5N - 10X + \frac{5N \cdot X}{2^{\lfloor \log_2(X) \rfloor}}$

F. Reduced Complexity TX with Re-sampling

According to [11] the baseline UPMC complexity can be reduced by decreasing the IFFT size followed by a convenient up-sampling and moving the frequency shift operation to the end of transmission chain so that the filtering remains with real coefficients. The small IFFT size N' instead of N is the main design parameter to reduce the computational complexity. In addition, the value of N' must be chosen carefully with the upsampling factor N/N' . However this method can approximate the UPMC spectrum with reducing the system complexity if only a few sub-bands (e.g 1-3PRB) shall be generated as in the case of IoT MTC.

Similarly to previous methods, the IFFT pruning can have sometimes more complexity reduction by using pruned small size IFFT, i.e., $IFFT_{N'}^{N_B, N'}$.

G. Reduced Complexity RX - Baseline (RC-Baseline RX)

After a windowing pre-processing of the received signal according to [2], a $2N$ -point FFT is applied to transform the $N+L-1$ received samples into the frequency domain, where a zero-padding is required. Note that all $N+L-1$ samples are used for detection rather than only N samples as in OFDM. Finally, only even sub-carriers are selected for data detection while odd sub-carriers are discarded as shown in Fig. 2.

The UPMC receiver complexity is slightly higher than a factor 2 of OFDM, as a $2N$ -point FFT is used for sub-carrier demodulation to take the sub-band filter tails into consideration. This complexity can be highly reduced by using FFT output pruning since only half of FFT output (even-indices) is selected. An additional enhancement can be realized by making input FFT pruning due to input padding by $N-L+1$ zeros. Hence, the FFT will be replaced by PFFT $FFT_{2N}^{N+L-1, N}$.

III. COMPUTATIONAL COMPLEXITY ANALYSIS

In this section, the computational complexity is derived for all suggested methods of implementation with\without FFT pruning. We consider the number of required CM\CA then real multiplication\addition, and total real operations can be deduced when needed.

Note that each complex multiplication requires 4 real multiplications and 2 real additions ($4M2A$) (6 real operations) and each complex addition requires 2 real additions. However, complex multiplication can also be computed using 3 real multiplications and 3 real additions ($3M3A$) when one complex factor is known in advance as in FFT. The complex multiplication computed with $3M3A$ is as follows:

$$(a + bi)(c + di) = (k_1 - k_3) + j(k_1 + k_2) \quad (12)$$

$$k_1 = c(a + b); k_2 = a(d - c); k_3 = b(c + d)$$

where the 2 factors $(d - c)$ and $(c + d)$ are precomputed.

The complexity for radix-2 FFT without pruning and with only CP for input or output is shown in Table I. The CP-FFT complexity is according to [12]. These equations are for contiguous nonzero inputs or required outputs and they represent the FFT maximum complexity after pruning. In the following, we will show that CPP may lead to an additional complexity reduction. The complexity after applying CPP is estimated using the flowchart tracing function based on the number and the indices of nonzero\required inputs\outputs.

The complexity of the existing methods without using FFT pruning technique can be deduced from the equations derived in this section by replacing the P-FFT\IFFT $\mathcal{C}(I - FFT_N^{NZ, RO})$ by the un-pruned FFT (UFFT) $\mathcal{C}(I - FFT_N)$.

A. Reduced Complexity UPMC Transmitter

The equations of computational complexity in terms of complex multiplication and complex addition for all reduced complexity transmitter methods are derived and shown in Table II. The complexity of pre-computations β in OSI TX, the subgroup filter in TDFA TX and the FFT of the filter in RC-FDF TX are not included in these equations.

Note that no complexity reduction is achieved with CP when the number of required outputs (or nonzero inputs) is greater than or equal to half the FFT size. In such case only the partial pruning permits to reduce the complexity of the (I)FFTs. As example, the complexity $\mathcal{C}_{CM}(IFFT_{N_{OS}.N}^{N_{OS}.N, N+L-1})$ and $\mathcal{C}_{CA}(IFFT_{N_{OS}.N}^{N_{OS}.N, N+L-1})$ in equations of RC-FDF TX can be replaced by $N_{OS}.N/2 \cdot \log_2(N_{OS}.N) - (N - L + 1)$ and $N_{OS}.N \cdot \log_2(N_{OS}.N) - (N - L + 1)$ respectively.

Similarly in equations of RC-FDA TX, the complexity $\mathcal{C}_{CM}(IFFT_{N_{OS}.N}^{N_{OS}.N_0, N+L-1})$ and $\mathcal{C}_{CA}(IFFT_{N_{OS}.N}^{N_{OS}.N_0, N+L-1})$ was reduced to $\mathcal{C}(IFFT_{N_{OS}.N}^{N_{OS}.N_0, N_{OS}.N}) - (N - L + 1)$, where the latter part in the last equation represents the reduction obtained by partial output pruning and the former part represents the complexity after CPP at FFT input.

Concerning the method RC TX with re-sampling, the authors [11] suggested to use FFT radix-4. In order to be more general and fair in our comparison, all methods will be evaluated with an FFT radix-2. Moreover, the filter impulse response is considered real in all subbands because the frequency shifting is moved to the end of chain and it is performed only on nonzero samples after up-sampling by N/N' . Note that the complexity of filtering convolution by real coefficients is not explicitly shown in Table II. This is due to the presence of

Table II
COMPUTATIONAL COMPLEXITY IN TERMS OF CM AND CA FOR UPMC TRANSMITTER METHODS

Methods	Complex Multiplication	Complex Addition
RC-TDF TX	$B[\mathcal{C}_{CM}(IFFT_N^{N_B, N}) + N \cdot L]$	$B[\mathcal{C}_{CA}(IFFT_N^{N_B, N}) + (N-1)(L-1)]$ $+ (B-1) \cdot (N+L-1)$
OSI TX	$B \cdot N_B \cdot (N+L-1)$	$(B \cdot N_B - 1) \cdot (N+L-1)$
RC-TDFA TX	$Q[\mathcal{C}_{CM}(IFFT_N^{B \cdot N_B / Q, N}) + (N+L-1)]$	$Q[\mathcal{C}_{CA}(IFFT_N^{B \cdot N_B / Q, N})] + (Q-1)(N+L-1)$
RC-FDF TX	$B[\mathcal{C}_{CM}(IFFT_N^{N_B, N}) + \mathcal{C}_{CM}(FFT_{N_{OS} \cdot N}^{N, N_{OS} \cdot N})$ $+ (N_{OS} \cdot N)] + \mathcal{C}_{CM}(IFFT_{N_{OS} \cdot N}^{N_{OS} \cdot N, N+L-1})$	$B[\mathcal{C}_{CA}(IFFT_N^{N_B, N}) + \mathcal{C}_{CA}(FFT_{N_{OS} \cdot N}^{N, N_{OS} \cdot N})]$ $+ \mathcal{C}_{CA}(IFFT_{N_{OS} \cdot N}^{N_{OS} \cdot N, N+L-1}) + (B-1)(N_{OS} \cdot N)$
RC-FDA TX	$B[\mathcal{C}_{CM}(IFFT_{N_0}^{N_B, N_0}) + \mathcal{C}_{CM}(FFT_{N_{OS} \cdot N_0}^{N_0, N_{OS} \cdot N_0})$ $+ (N_{OS} \cdot N_0)] + \mathcal{C}_{CM}(IFFT_{N_{OS} \cdot N}^{N_{OS} \cdot N_0, N+L-1})$	$B[\mathcal{C}_{CA}(IFFT_{N_0}^{N_B, N_0}) + \mathcal{C}_{CA}(FFT_{N_{OS} \cdot N_0}^{N_0, N_{OS} \cdot N_0})]$ $+ \mathcal{C}_{CA}(IFFT_{N_{OS} \cdot N}^{N_{OS} \cdot N_0, N+L-1}) + (B-1)(N_{OS} \cdot N_0)$
RC TX with Re-sampling	$B[\mathcal{C}_{CM}(IFFT_{N'}^{N_B, N'}) + (N+L-1)$ $+ \mathcal{C}_{filtering}]$	$B[\mathcal{C}_{CA}(IFFT_{N'}^{N_B, N'}) + \mathcal{C}_{filtering}]$ $+ (B-1)(N+L-1)$

multiplications of real numbers by complex numbers instead of two complex numbers. Then, these multiplications require 2 real multiplications instead of 6 real operations. For this reason, the filtering complexity will be added directly to the total complexity in terms of real operations to include this reduction. By taking into consideration the operation on nonzero samples only, the complexity of filtering in terms of real operations is $2L \cdot N' + 2N'(L - \alpha) - (L - 1)$ where $\alpha = N/N'$ is the up-sampling factor according to [11].

B. Reduced Complexity UPMC Receiver

At the receiver, the N_1 -points FFT ($N_1 = 2N$) has $N + L - 1$ nonzero inputs and N used outputs (even index outputs). Hence, input and output pruning can be applied by $FFT_{2N}^{N+L-1, N}$.

1) *Output Pruning*: The required outputs are located on even indices. Thus, the equation in Table I expressing the complexity of the output pruning, defined for contiguous required outputs, cannot be applied. Instead, the DIF flowchart depicted in Fig. 1 will be used to determine the number of multipliers/adders that will be saved using CPP.

The special distribution of required N outputs at even indices leads to a great reduction since, as shown in Fig. 1 with $N = 4$ and $L = 3$, the even index outputs are the results of the upper $N_1/4$ butterflies in the last stages except the first one. Thus, the lower $N_1/4$ butterflies in the last $\log_2(N_1) - 1$ stages can be pruned completely. Moreover, only the sum result (ascending edge) from the outputs of the first stage butterflies contribute in the final result so no need to compute the descending edge (Subtraction and Multiplication) for all butterflies in first stage. Hence, the number of complex multipliers after applying output pruning is given by:

$$\mathcal{C}_{RX}^{CM} = \frac{N_1}{4} [\log_2(N_1) - 1] = \frac{N}{2} [\log_2(2N) - 1] \quad (13)$$

and the number of complex adders is given by:

$$\mathcal{C}_{RX}^{CA} = \underbrace{2CA \cdot \left[\frac{N_1}{4} [\log_2(N_1) - 1] \right]}_{\text{Last stages}} + \underbrace{1CA \cdot \frac{N_1}{2}}_{\text{1st stage}} \quad (14)$$

$$= N \log_2(2N)$$

Note that similar results could be obtained when DIT flowchart is used.

2) *Input Pruning*: The first $N+L-1$ inputs of an $N_1 = 2N$ point FFT are nonzero. Therefore, the complexity reduction is only from partial input pruning. Consequently, the saving per butterfly for the last $N - L + 1$ butterflies is only 2 complex additions because the last $N - L + 1$ inputs (b in butterfly section of Fig. 1) are zeros and joined with nonzero inputs (a in butterfly section of Fig. 1) in the same butterfly.

$$\mathcal{C}_{RX}^{CA} = 2N \log_2(2N) - 2(N - L + 1) \quad (15)$$

3) *Input and Output Pruning*: Note that the operations pruned by considering input pruning alone are different than those pruned by output pruning. Therefore, the saving by using both pruning is the sum of individual savings. The number of the complex multiplications is as in Eq. 13 while the number of complex additions is given by:

$$\begin{aligned} \mathcal{C}_{RX}^{CA} &= \underbrace{2N \log_2(2N)}_{\text{Without Pruning}} - \underbrace{N \log_2(2N)}_{\text{Saving By Output Pruning}} \\ &\quad - \underbrace{2(N - L + 1)}_{\text{Saving By Input Pruning}} \quad (16) \\ &= N \log_2(2N) - 2(N - L + 1) \end{aligned}$$

IV. NUMERICAL ANALYSIS

In this section, we evaluate, using numerical values of the system parameters, the complexity reduction induced by the (I)FFT pruning applied to the different UPMC implementation methods. First, the Complexity Saving defined by Eq. 17, is computed in terms of complex multiplications/additions as a function of the percentage of nonzero contiguous inputs or required outputs according to equations of Table I for CP-FFT and according to MATLAB function for CPP-FFT. Thus, CP-FFT permits to obtain a reduction ranging from 100 % when all the inputs are zeros down to 0 % when the number of nonzero contiguous inputs reaches 50 %. Additional reduction is obtained using CPP-FFT especially in terms of complex additions.

$$\mathcal{C}_{SFFT} = 1 - \frac{\mathcal{C}(PFFT)}{\mathcal{C}(UFFT)} \quad (17)$$

The system simulation parameters in this comparison are: $N = 1024$, $N_B = 12$ (1 PRB in LTE), $Q = 3$, $N_0 = 128$, $L = 73$, B from 1 to 85, the oversampling factor $N_{OS} = 2$. The small I-FFT size $N_0 = N/16$ or greater [9], and the number of sub-carriers groups in each sub-band is $Q = 3$ to have

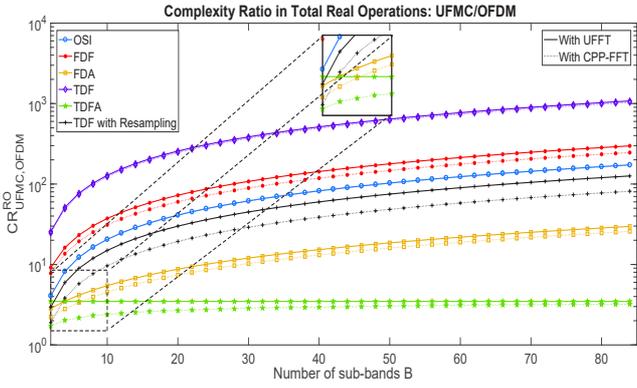


Figure 3. Transmitter Computational Complexity relative to OFDM vs number of sub-bands

negligible approximation errors [5]. In addition, the small I-FFT size in RC TDF with re-sampling method N' has been chosen equal to 64 as suggested in [11] when $N = 1024$.

In our analysis in this section, we express the complexity in terms of total real operations (multiplications and additions) while knowing that this complexity is dominant by the number of multipliers, and which in our case has the same behavior when only the multiplications are considered.

A. UFMC Transmitter

The computational complexity for all methods of implementation of UFMC transmitter is estimated in terms of total real operations relative to OFDM before and after pruning, where we evaluate a Complexity Ratio (CR) metric called $CR_{UFMC-OFDM}^{RO}$, defined by Eq. 18, in function of the number of sub-bands.

$$CR_{UFMC,OFDM}^{RO} = \frac{\mathcal{C}(UFMC)}{\mathcal{C}(OFDM)} \quad (18)$$

The simulations results are shown in Fig. 3, where each method, except OSI, is evaluated in two versions: (1) without FFT pruning (solid lines); (2) with CPP-FFT pruning (dotted lines). We will refer to dotted lines by prefix RC, as example RC-TDFA means TDFA with CPP FFT pruning. As shown, the complexity is increasing with the number of sub-bands except for TDFA TX without FFT pruning which is constant whatever the number of sub-bands. Moreover, the TDF TX has the highest complexity while its approximation RC-TDFA TX represents the best solution in terms of reduced complexity, and the RC-FDA TX comes as second best solution for large number of subbands. While the method RC-TDF with re-sampling have lower complexity with small number of sub-bands (1 to 3) which is convenient for MTC application and IoT devices.

Moreover, when no approximation is tolerated, the OSI TX, FDF TX in its both versions (pruned and un-pruned) constitutes the less costly solution. However, both methods requires additional memory to store the pre-computed matrix β in OSI and the filter response in RC-FDF, knowing that their complexities are much higher than OFDM complexity especially with large B.

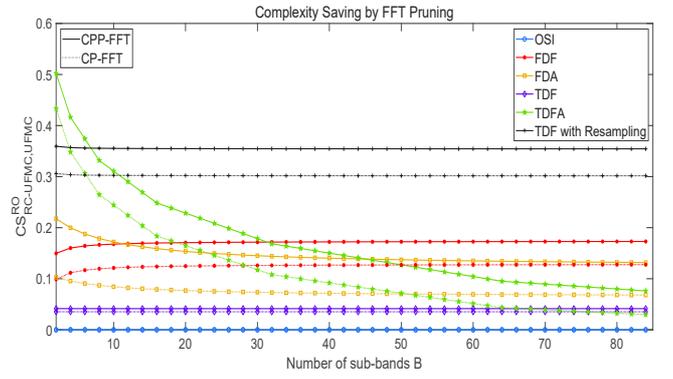


Figure 4. Complexity saving vs number of sub-bands: 1-RC-UFMC/UFMC

Note that according to [5], the complexity of TDFA TX with 3 subgroups filter was around 3.7 OFDM complexity for any number of sub-bands, whereas with the proposed reduction method it is reduced to around 1.7 OFDM complexity for 2 sub-bands, 3 for 50 sub-bands and to 3.2 OFDM complexity for maximum number of sub-bands by respecting the error vector magnitude (EVM) requirement of 22dB for LTE.

In order to illustrate the amount of saving obtained when using the pruning technique applied to all the UFMC methods, we have evaluated a new CS metric $CS_{RC-UFMC,UFMC}^{RO}$ defined in Eq. 19 indicating the amount of saving in terms of real operations of RC-UFMC as compared to UFMC. Fig. 4 shows the evolution of this metric as function of number of sub-bands. We notice that the saving becomes approximately constant after certain number of sub-bands. It is worth mentioning that the TDFA CP and TDFA CPP with low B permits to obtain a very high amount of complexity saving when compared to their un-pruned version TDFA TX that has the lowest complexity as previously shown in Fig. 3. This complexity saving ranges from 7.5% up to 50.2% with CPP and from 3% up to 43.4% with CP to depending on the number of sub-bands. As compared to the CP technique, the CPP offers additional saving ranging from 4.5% to 10% for all the methods except the TDF TX and OSI TX.

$$CS_{RC-UFMC,UFMC}^{RO} = 1 - \frac{\mathcal{C}(RC-UFMC)}{\mathcal{C}(UFMC)} \quad (19)$$

Moreover, according to [5] all MSE of time domain filtering approximation for several number of sub-carriers with Filter of 1 and 3 subgroups are below $-30dB$, which means that the approximations deviate on average less than 0.1% from the exact signal. This value is clearly below e.g. the EVM requirement of 22dB for LTE. It is important to note that all the proposed RC-UFMC techniques do not have any impact on the performance, in terms of Bit Error Rate (BER), since they do not introduce any further processing approximation.

Finally, the (I)FFT pruning techniques have a great impact on power consumption and speed. For example, the FFT energy saving is 73.7% and the speedup is around a factor of 3 as compared to (I)FFT without pruning in the case of 64 nonzero contiguous inputs for 2048-point (I)FFT [13].

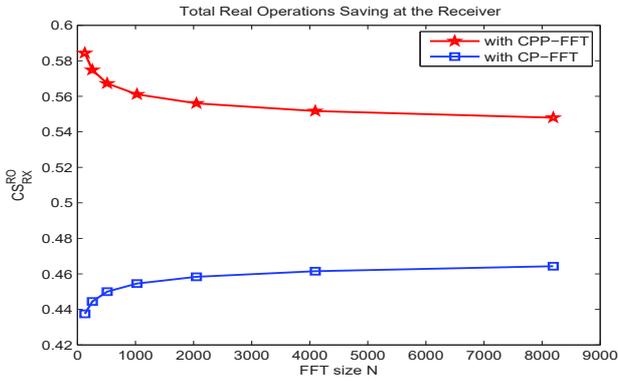


Figure 5. Total real operations saving by CP and CPP at the UPMC Receiver

B. UPMC Receiver

The complexity of the receiver is related only to the parameter N (FFT size at the transmitter) and the filter length L as shown in Eq. 13 and Eq. 16. In order to evaluate the complexity reduction at the receiver, we define three new complexity saving metrics that express the saving in terms of CM, CA and total RO for PFFT as compared to the UFFT:

$$CS_{RX}^{CM} = 1 - \frac{C_{RX}^{CM}}{N \cdot \log_2(2N)} = 1 - \frac{\log_2(2N) - 1}{2 \log_2(2N)} \quad (20)$$

$$CS_{RX}^{CA} = 1 - \frac{C_{RX}^{CA}}{2N \cdot \log_2(2N)} = 0.5 + \frac{N - L + 1}{N \cdot \log_2(2N)} \quad (21)$$

$$CS_{RX}^{RO} = 1 - \frac{6C_{RX}^{CM} + 2C_{RX}^{CA}}{10N \cdot \log_2(2N)} \quad (22)$$

The total RO saving as a function of FFT size N with filter length $L = L_{CP} + 1 \approx 7\% \cdot N$ is shown in Fig. 5, where L_{CP} is the OFDM cyclic prefix length in LTE, N is chosen to be a power of 2 and ranges between 128 and 8192.

As shown in Fig. 5, the total RO saving is around 56% for CPP-FFT and 46% for CP-FFT which approaches 50% for large values of N . In addition, the complexity ratio CR in terms total real operations of baseline UPMC receiver and OFDM before pruning is 2 for $N \rightarrow \infty$. However, the UPMC receiver complexity with CPP-FFT is always less than that of OFDM system, and it approaches OFDM as $N \rightarrow \infty$. While in case of CP-FFT, the UPMC complexity decreases from 1.28 till 1.15 OFDM complexity as N increases from 128 to 8192. In both UPMC cases, their complexities tend to OFDM complexity for $N \rightarrow \infty$. Therefore, the UPMC receiver complexity becomes similar or less than OFDM system.

V. CONCLUSION

In this paper, we discussed the complexity reduction techniques for the different UPMC implementation methods using (D)FFT pruning at both the transmitter and receiver sides. Without any performance degradation and by respecting EVM requirements of LTE system and beyond, the proposed complexity reduction methods permits to reduce the complexity of the UPMC TX chains, in terms of real operations, to reach 3.2 down to 1.7 OFDM complexity depending on the number

of sub-bands with the TDFA TX using the CPP technique and $Q=3$, while the most simplified UPMC TX proposed in the literature is of the order of 3.7 with any B and $Q=3$. As for the receiver, we showed that the UPMC RX complexity can be reduced from around 2.2 to less than OFDM complexity. Also, we emphasized the importance of the complete-partial pruning that gives additional simplification of the FFT.

Therefore, the main challenge of UPMC high computational complexity for transceivers is tackled in this paper by using FFT pruning techniques. In addition, it is worth mentioning that the proposed UPMC transceivers have the lowest computational complexity among the 5G candidates FBMC, GFDM and Filtered-OFDM that have complexity overheads of the order of 5 times the OFDM [14].

Finally, the complexity reduction of UPMC transceiver have a great impact on the system speed, the implementation cost and the power consumption for future mobile devices and base stations.

Future work will consider the hardware implementation of the proposed RC-TDFA TX/RX chain in order to determine the post-synthesis hardware complexity.

REFERENCES

- [1] F. Schaich and T. Wild, "Waveform contenders for 5G — OFDM vs. FBMC vs. UPMC," 2014 6th International Symposium on Communications, Control and Signal Processing (ISCCSP), 2014.
- [2] X. Wang, T. Wild, and F. Schaich, "Filter Optimization for Carrier-Frequency- and Timing-Offset in Universal Filtered Multi-Carrier Systems," IEEE 81st Vehicular Technology Conference, 2015.
- [3] G. Wunder, M. Kasparick, S. T. Brink, F. Schaich, T. Wild, I. Gaspar, E. Ohlmer, S. Krone, N. Michailow, A. Navarro, G. Fettweis, D. Ktenas, V. Berg, M. Dryjanski, S. Pietrzyk, and B. Eged, "5G NOW: Challenging the LTE Design Paradigms of Orthogonality and Synchronicity," 2013 IEEE 77th Vehicular Technology Conference (VTC Spring), 2013.
- [4] B. Farhang-Boroujeny, "OFDM Versus Filter Bank Multicarrier," IEEE Signal Processing Magazine, vol. 28, no. 3, pp. 92–112, 2011.
- [5] M. Matthe, D. Zhang, F. Schaich, T. Wild, R. Ahmed, and G. Fettweis, "A Reduced Complexity Time-Domain Transmitter for UF-OFDM," 2016 IEEE 83rd Vehicular Technology Conference (VTC Spring), 2016.
- [6] J. Markel, "FFT pruning," IEEE Transactions on Audio and Electroacoustics, vol. 19, no. 4, pp. 305–311, 1971.
- [7] D. Skinner, "Pruning the decimation in-time FFT algorithm," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 24, no. 2, pp. 193–194, 1976.
- [8] R. Alves, P. Osorio, and M. Swamy, "General FFT pruning algorithm," Proceedings of the 43rd IEEE Midwest Symposium on Circuits and Systems, vol. 3, pp. 1192 – 1195, Aug. 2000 (Cat.No.CH37144).
- [9] T. Wild and F. Schaich, "A Reduced Complexity Transmitter for UF-OFDM," IEEE 81st Vehicular Technology Conference, 2015.
- [10] N. Jeremy, C. Abdel Nour, A. Baghdadi, "Novel UF-OFDM transmitter: significant complexity reduction without signal approximation," IEEE Transactions on Vehicular Technology, 2017, pp. 1–1.
- [11] R. Knopp, F. Kaltenberger, C. Vitiello, and M. Luise, "Universal filtered multicarrier for machine type communications in 5G," in European Conference on Networks and Communications (EUCNC), Athens, Greece, June 2016. [Online]. Available: <http://www.eurecom.fr/publication/4910>
- [12] H. Sorensen and C. Burrus, "Efficient computation of the DFT with only a subset of input or output points," IEEE Transactions on Signal Processing, vol. 41, no. 3, pp. 1184–1200, 1993.
- [13] R. Airolidi, O. Anjum, F. Garzia, A. Wyglinski, and J. Nurmi, "Energy-Efficient Fast Fourier Transforms for Cognitive Radio Systems," IEEE Micro, vol. 30, no. 6, pp. 66–76, 2010.
- [14] M. V. Eeckhaute, A. Bourdoux, P. D. Doncker, and F. Horlin, "Performance of emerging multi-carrier waveforms for 5G asynchronous communications," EURASIP Journal on Wireless Communications and Networking, vol. 2017, no. 1, Jul. 2017.