Performance Evaluation of Low-Complexity Algorithms for Orthogonal Time-Frequency Space Modulation

Dayse G. C. Bandeira, Didier Le Ruyet, Mylene Pischella, and João Cesar M. Mota

Abstract—Data transmission in wireless systems brings numerous challenges, especially when it involves propagating in a multipath scenario over rapidly time-varying channels. In this context, Orthogonal Time-Frequency Space (OTFS) modulation has been recently proposed to work with time-frequency selective channels with high Doppler. In this modulation, the symbols are first multiplexed in a delay-Doppler domain rather than in the time-frequency domain used by Orthogonal Frequency Division Multiplexing (OFDM). The studies point out advantages of OTFS performance over OFDM in many aspects, such as data rate increase in high mobility. Another advantage is the sparsity of the channel produced by OTFS that allows using low-complexity algorithms for the detection of the data. In this paper, the performance of OTFS modulation in a doubly dispersive channel is evaluated with several low-complexity variants of the message passing algorithm (MPA) in terms of complexity and Bit Error Rate (BER) performance. The results show that MPA and Approximate Message Passing simplified by Expectation Propagation (AMP-EP) algorithms achieve higher performance. However, when taking into account both complexity and BER performance, AMP simplified by First-Order (AMP-FO) achieves the best performance-complexity tradeoff.

Index Terms—OTFS, delay-Doppler domain, message passing.

I. INTRODUCTION

The increased demand for data rate and User Equipments (UE) restricted to a limited electromagnetic spectrum for wireless communications, make the project requirements along the evolution of the generations more challenging. Fourth generation (4G) networks have achieved a big success, due to their ability to provide high data rates for a large number of users using the Orthogonal Frequency Division Multiplexing (OFDM) modulation [1][2].

OFDM is a special form of multicarrier modulation which is particularly suited for transmission over a dispersive channel, where different subcarriers are orthogonal to each other. OFDM is a wideband modulation scheme that is designed to cope with the problems of the multipath channels. Here, the

D. G. C. Bandeira is with the GTEL from the Federal University of Ceará (UFC), Campus Fortaleza, Ceará, Brazil and with the CEDRIC from the Conservatoire National des Arts et Métiers, Paris, France (e-mail: {dayse.correia}@ifce.edu.br).

D. Le Ruyet is with the CEDRIC, Conservatoire National des Arts et Métiers, Paris, France (e-mail: {leruyet}@cnam.fr).

M. Pischella is with the CEDRIC, Conservatoire National des Arts et Métiers, Paris, France (e-mail: {mylene.pischella}@cnam.fr).

J. C. M. Mota is with the GTEL, from the Federal University of Ceará (UFC), Campus Fortaleza, Ceará, Brazil (e-mail: {mota}@gtel.ufc.br).

wideband frequency selective fading channel is divided into many narrow-band subchannels. If the number of subchannels is high enough, each subchannel may be considered as flat.

Although OFDM has been implemented in the 4G of mobile systems, it is not robust to time-varying channels with high Doppler spread (such as, high-speed rail mobile communications) [3]. The Orthogonal Time-Frequency Space (OTFS) modulation proposed by Hadani and et al. appears as a solution [4], targeting for example applications in the fifth generation (5G) of mobile systems, which requires a higher data rate and a higher UE speed.

OTFS shows significant advantages in doubly dispersive channels over OFDM [5], [6], [7]. The delay-Doppler domain is an alternative representation of a Linear Time-varying (LTV) channel modeling due to moving objects in the multipath scenario. For this, the OTFS modulator spreads each information (e.g., QAM) symbol over a set of two-dimensional orthogonal basis functions, which span across the frequency-time resources required to transmit a burst. The basis function set is specifically projected to deal with the dynamism of the time-varying multipath channel.

Using the technique of transformations in two dimensions, OTFS converts a doubly-dispersive channel into an almost non-fading channel in the delay-Doppler domain [8]. Hence, each symbol in a frame suffers an almost constant fade, thus achieving significant performance gains over existing modulation schemes that are not robust to strong Doppler such as OFDM. Moreover, since there are typically a small number of physical reflectors with Dopplers and associated delays in a multipath channel, few parameters are required for channel modeling and estimation in the delay-Doppler domain.

The impulse response of the channel in delay-Doppler domain is a sparse matrix [9], thus allowing to use low-complexity detector algorithms, such as message passing algorithm (MPA). It also has important implications for channel estimation/prediction and tracking [4].

The aim of this paper is to study, evaluate and compare different low-complexity MPA-based detectors for OTFS systems over time-frequency selective channel with high Doppler. Both bit error rate (BER) performance and complexity analysis are evaluated. The considered algorithms are the following: i) Factor Graph with Gaussian Approximation of Interference (FG-GAI) proposed in [10] whose linear complexity is attractive for detection in large-dimension channels, ii) Approximate Message Passing using Gaussian Approximation (AMP-GA) in which the calculations of the probability messages are

updated by the calculations of means and variances between the nodes, iii) AMP simplified by Expectation Propagation (AMP-EP) and iv) AMP simplified by First-Order (AMP-FO), proposed in [11].

The paper is organized as follows. Section II presents the characteristics of OTFS system and the adopted channel model. In section III the different detector algorithms based on message passing are detailed. In section IV is presented complexity analysis algorithms. In section V the coupling of OTFS with MIMO systems is considered. In section VI the results of the performance of algorithm simulations are presented and discussed and section VII is dedicated to the conclusions and perspectives.

II. GENERAL DESCRIPTION OF OTFS

Traditional OFDM modulation operates in the frequency-time domain. An OFDM resource elements (RE) occupies one subcarrier on one particular OFDM symbol. In contrast, OTFS modulation operates in the Delay-Doppler domain, which is related to frequency and time by the Symplectic Finite Fourier Transform (SFFT), a two-dimensional Discrete Fourier Transform (DFT) [4], [12]. The OTFS modulation framework can be understood as a time-frequency multicarrier modulation with an additional pre-processing transformation from delay-Doppler domain to the time-frequency domain of the information symbols, by Inverse Symplectic Finite Fourier Transform (ISFFT). Hence, OTFS can be implemented as a pre-processing step on top of an underlying OFDM signal [13].

In OTFS, the quadrature amplitude modulation (QAM) symbols are indexed by points on a grid in the Delay-Doppler domain. Through ISFFT, each QAM symbol weights a 2D basis function defined in the Time-Frequency domain. The size of the delay-Doppler resource grid is related to the size of the frequency-time plane by the signal properties, i.e. bandwidth (B), Transmission Time Interval (TTI), pulse time duration (T), sub-carrier spacing (Δf), number of subcarriers (M) and symbol block length (N).

Then, the delay-Doppler grid consists of M points (number of subcarriers) along delay with spacing $\Delta \tau = \frac{1}{M\Delta f}$ and N points (number of symbols) along Doppler with spacing $\Delta \nu = \frac{1}{NT}$. The reciprocal time-frequency grid consists of M points along frequency with spacing $\Delta f = \frac{B}{M}$ and N points along time with spacing $T = \frac{TTI}{N}$ [3]. Hence, the time-frequency grid can be interpreted as a sequence of N multicarrier symbols each consisting of M subcarriers, i.e. the bandwidth of the transmission B is the inverse to delay resolution $\Delta \tau$ and the duration of the transmission TTI is inverse to the Doppler resolution $\Delta \nu$. The two grids are shown in Figure 1.
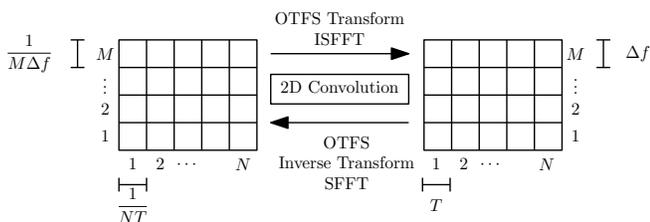

Fig. 1. OTFS Transform: Delay-Doppler grid x Time-Frequency

In summary, based on these definitions, it can be seen that the time-frequency plane is discretized in the grid by sampling the time and frequency axes in intervals of T (seconds) and Δf (Hz), as:

$$\Lambda = \{(m\Delta f, nT), \ m = 0, ..., M - 1; \ n = 0, ..., N - 1\}, \quad (1)$$

Consequently, the delay-Doppler plane is discretized, as:

$$\Gamma = \left\{ \left(\frac{k}{NT}, \frac{l}{M\Delta f} \right), k = 0, ..., N - 1; l = 0, ..., M - 1 \right\}. \quad (2)$$

Basically, a 2D ISFFT maps the information symbols $x[k, l]$ of a grid $\Gamma(2)$ in the delay-Doppler domain on a sequence of complex numbers $X[m, n]$ mapped in the grid Λ (1) in the time-frequency domain, as follows:

$$X[m, n] = \frac{1}{\sqrt{MN}} \sum_{l=0}^{M-1} \sum_{k=0}^{N-1} x[k, l] \ e^{j2\pi\left(\frac{nk}{N} - \frac{ml}{M}\right)}. \quad (3)$$

After this step of pre and post-processing blocks, one can implement the conventional OFDM modulation/demodulation. Figure 2 shows the OTFS system diagram.
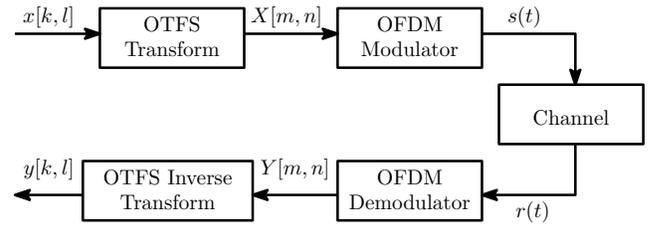

Fig. 2. Block diagram of OTFS System

The OFDM modulator is applied to time-frequency symbols $X[m, n]$ to convert the time-frequency modulated symbols to the time domain signal $s(t)$ for transmission over the channel. Hence, the signal output of OTFS transmitter will be:

$$s(t) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X[m, n] \ e^{j2\pi m\Delta f(t-nT)} \ g_{tx}(t - nT), \quad (4)$$

where $g_{tx}(t)$ is pulse shaping used in the transmitter. In the following we will consider rectangular pulses of amplitude equal to one and duration T.

From equation (4), it is observed that every OTFS QAM symbol is spread over the full time-frequency grid and hence it is possible to exploit all the channel diversity.

The $s(t)$ transmitted signal propagates through a time-varying channel with complex baseband channel impulse response $h(\tau, \nu)$ and noise $w(t)$. After passing through the channel, the received signal $r(t)$ is given by:

$$r(t) = \int_{\tau} \int_{\nu} h(\tau, \nu)s(t - \tau) \ e^{j2\pi\nu(t-\tau)} d\tau d\nu + w(t). \quad (5)$$

The channel modeling can be characterized by obtaining the Delay-Doppler Profile (DDP) of the channel [14], that contains the delay and Doppler paths associated with each multipath reflector. Given the sparsity of the channel representation, it is convenient to express the response $h(\tau, \nu)$ in the form:

$$h(\tau, \nu) = \sum_{i=1}^{P} h_i \delta(\tau - \tau_i)\delta(\nu - \nu_i), \qquad (6)$$

where P is the number of propagation paths, h_i, τ_i and ν_i represent the path gain, delay, and Doppler shift (or frequency) associated with i^{th} path, respectively, and $\delta(\cdot)$ denotes the Dirac delta function. The delay and Doppler taps for the ith path are equal to:

$$\tau_i = \frac{l_{\tau_i} + \tilde{l}_{\tau_i}}{M\Delta f}, \quad \nu_i = \frac{k_{\nu_i} + \tilde{k}_{\nu_i}}{NT}, \qquad (7)$$

where $\tilde{l}_{\tau_i}$ denotes the fractional delay, $\tilde{k}_{\nu_i}$ denotes the fractional Doppler shift, NT and $M\Delta f$ denote the total duration and bandwidth of the transmitted signal frame, respectively, with $T.\Delta f = 1$. The interference caused by fractional delay and fractional Doppler shift is effectively suppressed when M and N are sufficiently large to approximately achieve ideal OTFS resolution. Then we can consider $\tilde{l}_{\tau_i} = \tilde{k}_{\nu_i} = 0$ [15].

The received signal $r(t)$ is sampled at a rate $f_s = M\Delta f = \frac{M}{T}$ and a signal $r[n]$ is formed, whose entries, from eq. (5) and eq. (6) are equal to:

$$r[n] = \sum_{i=1}^{P} h_i \ e^{j2\pi \frac{k_i(n - l_i)}{MN}} (s[n - l_i]_{MN}) + w[n]. \qquad (8)$$

Then, at the receiver, the time domain received signal can be mapped to the time-frequency domain by an OFDM demodulator, and then to the delay-Doppler domain by SFFT.

Based on this mathematical description, in discrete domain the authors in [16] have used properties and identities between vectors to process the OTFS system. Following the same notation as $\mathbf{a}$ for vector, $\mathbf{A}$ for matrix, and $\mathbf{A}^H$ to represent the Hermitian transpose, the transmitted signal can be written as:

$$\mathbf{S} = \mathbf{G}_{tx}\mathbf{F}_M^H(\mathbf{F}_M\mathbf{X}\mathbf{F}_N^H) = \mathbf{G}_{tx}\mathbf{X}\mathbf{F}_N^H, \qquad (9)$$

where $\mathbf{X} \in \mathbb{C}^{M \times N}$ denote the two-dimensional information symbols transmitted in the delay-Doppler domain; $\mathbf{F}_n = \left\{ \frac{1}{\sqrt{n}}e^{2\pi jkl/n} \right\}_{k,l=0}^{n-1}$ and $\mathbf{F}_n^H = \mathbf{F}_n^{-1}$ is the n-point DFT and the Inverse Discrete Fourier Transform (IDFT) matrices, respectively, and $\mathbf{G}_{tx}$ is the diagonal matrix that has transmission pulse samples with duration $[0, T]$ repeated N times in the frame. Then, the column-wise vectorization of the $M \times N$ matrix $\mathbf{S}$ in eq. (9) yields the $MN \times 1$ vector $\mathbf{s} = vec(\mathbf{S}) = (\mathbf{F}_N^H \otimes \mathbf{G}_{tx})\mathbf{x}$, where $\mathbf{x} = vec(\mathbf{X})$ and $\otimes$ is the Kronecker product.

Consequently, the signal received vector $\mathbf{r}$ of size $MN \times 1$ can be written using samples of equation (8), $\mathbf{r} = \{r[n]\}_{n=0}^{MN-1}$, as:

$$\mathbf{r} = \mathbf{Hs} + \mathbf{w}, \qquad (10)$$

where $\mathbf{w}$ is the noise vector and $\mathbf{H}$ is the following $MN \times MN$ matrix:

$$\mathbf{H} = \sum_{i=1}^{P} h_i \mathbf{\Pi}^{l_i} \mathbf{\Delta}^{k_i}, \qquad (11)$$

with $\mathbf{\Pi}$ the permutation matrix (forward cyclic shift),

$$\mathbf{\Pi} = \begin{bmatrix} 0 & \dots & 0 & 1 \\ 1 & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 1 & 0 \end{bmatrix} \qquad (12)$$

and $\mathbf{\Delta}$ the $MN \times MN$ diagonal matrix:

$$\mathbf{\Delta} = diag[z^0, z^1, ..., z^{MN-1}], \qquad (13)$$

where $z = e^{\frac{j2\pi}{MN}}$. The matrices $\mathbf{\Pi}$ and $\mathbf{\Delta}$ model the delays and the Doppler shifts in eq. (5), respectively.

At the receiver, the received signal samples $\mathbf{r}$ are transformed into the time-frequency domain symbols $\mathbf{R} = vec^{-1}(\mathbf{r})$, then into the delay-Doppler domain symbols $\mathbf{Y} = \mathbf{F}_M^H(\mathbf{F}_M\mathbf{G}_{rx}\mathbf{R})\mathbf{F}_N$. To do this, an M-point FFT followed by an SFFT is applied. Here, $\mathbf{G}_{rx} \in \mathbb{C}^{M \times M}$ is the diagonal matrix of the receiver pulse. In vectored form the received signal in the delay-Doppler domain can be written as:

$$\mathbf{y} = (\mathbf{F}_N \otimes \mathbf{G}_{rx})\mathbf{r}. \qquad (14)$$

After substituting the transmitted signal vector $\mathbf{s}$ in eq. (10), we obtain:

$$\begin{aligned} \mathbf{y} &= (\mathbf{F}_N \otimes \mathbf{G}_{rx})(\mathbf{Hs} + \mathbf{w}) \\ &= (\mathbf{F}_N \otimes \mathbf{G}_{rx})\mathbf{H}(\mathbf{F}_N^H \otimes \mathbf{G}_{tx})\mathbf{x} + (\mathbf{F}_N \otimes \mathbf{G}_{rx})\mathbf{w} \\ &= \mathbf{H}_{\text{eff}}\mathbf{x} + \widetilde{\mathbf{w}}, \end{aligned} \qquad (15)$$

where $\mathbf{H}_{\text{eff}} = (\mathbf{F}_N \otimes \mathbf{G}_{rx})\mathbf{H}(\mathbf{F}_N^H \otimes \mathbf{G}_{tx})$ is a sparse matrix that denotes the effective channel matrix, $\widetilde{\mathbf{w}} = (\mathbf{F}_N \otimes \mathbf{G}_{rx})\mathbf{w}$ is the noise vector with variance σ_0^2.

Due to the sparsity of $\mathbf{H}_{\text{eff}}$, it is possible to implement low-complexity detector algorithms to obtain the estimated symbols [17]. The algorithms based on message passing are well-adapted for this. These algorithms use a representation of the matrix $\mathbf{H}_{\text{eff}}$ by a factor graph. The next section explains the different variants of message passing algorithms used in this paper.

III. DETECTOR ALGORITHMS BASED ON MESSAGE PASSING

Most of the estimation and inference problems in the field of digital communications can be described using a graphical representation such as the bayesian networks or the factor graphs [18]. A factor graph is a bipartite graph that specifies the joint distribution of the random variables x_i taking value in a given domain. It is composed of two sets of vertices or nodes and a set of branches or edges. The two sets of nodes are :

- the variables nodes x_i, graphically represented by circles on Fig. 3
- the function nodes f_j , represented by squares on Fig. 3

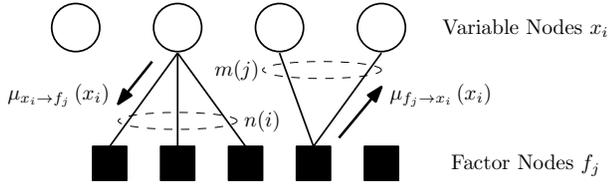An example of factor graph is given in Figure 3

Fig. 3. Factor graph to MPA

$m(j)$ is the set of variable neighbors connected to factor node f_j and $n(i)$ is the set of factor nodes connected to the variable node x_i.

Since the matrix $\mathbf{H_{eff}}$ of size $MN \times MN$ is sparse, we can represent the equation (15) using a factor graph of $MN = O$ variables nodes and O factor nodes.

The number of branches $d_f(j) = |m(j)|$ that converge on a given function node is called the degree of the function node f_j. Similarly, the number of branches $d_x(i) = |n(i)|$ that converge on a given variable node is called the degree of the variable node x_i. In our case, since the number of non zero element of the lines and columns of $\mathbf{H_{eff}}$ is equal to the number of propagation paths P, the factor graph is regular and we have $d_f(j) = d_x(i) = P \; \forall i, j$.

In the next subsections, we will detail the following algorithms based on MPA: the original MPA, the Approximate Message Passing (AMP) in the version Factor Graph using Gaussian Approximation of Interference (FG-GAI) [10], the AMP using Gaussian Approximation (AMP-GA), AMP using expectation propagation (AMP-EP) and AMP using first order (AMP-FO) introduced in [11].

A. Message Passing Algorithm (MPA)

The aim of message passing algorithm is to estimate the marginal probabilities μ_{x_i} for all variables x_i.

In the MPA, at each iteration, the algorithm computes messages or beliefs from the variable nodes to the factor nodes and then messages from factor nodes to the variable nodes. The messages are propagated usually in parallel, to the next factor node. This order, translated into the factor graph context and the SPA (Sum-Product Algorithm), results in the message update schedule.

The notations used to describe MPA based algorithms are the following: $\mu_{f_j \to x_i}$ are the messages from factor node f_j toward variable node x_i and $\mu_{x_i \to f_j}$ are the messages from the variable nodes x_i towards the factor nodes f_j.

These messages or beliefs, are a function of a variable node x_i either in one direction or the other. The message from x_i to f_j represents the probability that x_i has a certain value, given the observed value of this variable and given the values it received from the other factor nodes linked to x_i, except f_j. Using a Z-QAM (Z = constellation size), the messages have Z distinct values, one for each possible value from x_i.

Then, to initialize this algorithm, firstly the probability mass functions to each factor node based on corresponding variables node to each value of $\alpha_\mathbf{s}$ is computed as:

$$f_j(y_j|x_i = \alpha_s) = \prod_{l \in m(j)\setminus i} P(y_j|x_l = \alpha_s, \mathbf{H_{eff}})$$
$$\propto \prod_{l \in m(j)\setminus i} \exp\left(\frac{-|y_j - h_{j,l} \; \alpha_s|^2}{\sigma_0^2} \right), \quad (16)$$

where α_s belongs to alphabet $\mathcal{A}$ ($|\mathcal{A}| = Z$), $h_{j,l}$ is the element of the jth row and l^{th} column of matrix $\mathbf{H_{eff}}$ (channel transfer matrix), σ_0^2 is the noise variance, and y_j is the correlated received signal term.

The computation of all messages of the factor node f_j to the corresponding variables node x_i starts considering that the chances of the symbols of alphabet $\mathcal{A}$ are equality probable. Then, each message is computed according to the sum-product rule [18], where the previous product of all the messages sent from the variable node x_i are summarized for each associated factor node f_j, as follows:

$$\mu_{f_j \to x_i}^t(x_i = \alpha_s) = \sum_{\mathbf{x}\setminus x_i} \left(f_j(y_j|\mathbf{x}) \prod_{l \in m(j)\setminus i} \mu_{x_l \to f_j}^{t-1}(x_l) \right). \quad (17)$$

Then, the messages from variable node to factor node are updated by the resulting product of messages from factor node to variable node, as follows:

$$\mu_{x_i \to f_j}^t(x_i = \alpha_s) = \prod_{b \in n(i)\setminus j} \mu_{f_b \to x_i}^t(x_i). \quad (18)$$

During message exchanges, a normalization process followed by the application of the damping factor (Δ) is used. In this process, the values of the transferred messages from variables node to factor node are normalized ($\mu_{x_i \to f_j}$) by the addition of the corresponding x_i for each QAM symbol. Then, the damping factor is calculated at each iteration (t) by total the normalized messages with applying a weight, as shown by eq. (19).

The application of the damping factor is a technique that helps minimizing the BER when evaluating the best number of iterations for the decoding algorithm, depending on the density on $\mathbf{H_{eff}}$ matrix.

$$\mu_{x_i \to f_j}^t(x_i = \alpha_s) = (1 - \Delta).\mu_{x_i \to f_j}^{t-1}(x_i) + \Delta.\mu_{x_i \to f_j}^t(x_i). \quad (19)$$

At iteration t we can estimate the marginal distribution $\mu_{x_i}^t(x_i = \alpha_s)$ using the set of incoming messages $\mu_{f_b \to x_i}^t(x_i)$

$$\mu_{x_i}^t(x_i = \alpha_s) = \frac{\prod_{b \in n(i)} \mu_{f_b \to x_i}^t(x_i = \alpha_s)}{\sum_{x_i \in A} \prod_{b \in n(i)} \mu_{f_b \to x_i}^t(x_i)}. \quad (20)$$

Next, the Log Likelihood Ratio (LLR) calculation is applied to perform a test based on the probabilities ratio and thus infer about the detection of the received bit sequence. The LLR calculation $\Lambda_{b \to l}$ from the transferred messages from the bth factor node to the lth variable node ($b, l \to f_j, x_i$) is based on the fundamentals of [20] and considering QAM symbols. The MPA is detailed in pseudo-code in Algorithm 1.

Algorithm 1 MPA algorithm

Initialization

Set $\mu_{f_j \to x_i}(x_i = \alpha_s) = 0, \forall x_i \in \mathcal{A}$

Compute $f_j(y_j|x_i)$ using (16), $\forall i \in m(j)$, $\forall x_i \in \mathcal{A}$

for $t = 1$ to T **do** {Number of iterations}

    for $j = 1$ to O **do** {Computation of messages from FN (Factor Node) to VN (Variable Node)}

        Compute $\mu_{f_j \to x_i}^t(x_i)$ using (17), $\forall i \in m(j)$, $\forall x_i \in \mathcal{A}$

    end for

    for $i = 1$ to O **do** {Computation of messages from VN to FN}

        Compute $\mu_{x_i \to f_j}^t(x_i)$ using (18), $\forall j \in n(i)$, $\forall x_i \in \mathcal{A}$

    end for

    for $i = 1$ to O **do** {Normalisation of messages from VN to FN}

        $\mu_{x_i \to f_j}(x_i) = \frac{\mu_{x_i \to f_j}(x_i)}{\sum_{\alpha_s \in Z} \mu_{x_i \to f_j}(x_i = \alpha_s)}, \forall j \in n(i), \forall x_i \in \mathcal{A}$

        Damping calculation by using equation (19)

    end for

    Decision calculation

B. Gaussian Approximation of Interference (FG-GAI)

In FG-GAI [10], the messages $\mu_{f_j \to x_i}^t(x_i)$ are replaced with Gaussian approximation of the interference. The received signal y_j is given by:

$$y_j = h_{j,i}x_i + \underbrace{\sum_{l \in m(j) \setminus i} h_{j,l}x_l + w_j}_{w_{f_j \to x_i}}, \qquad (21)$$

and the interference term $w_{f_j \to x_i}$ will be modeled as a Gaussian variable with mean $z_{f_j \to x_i}$ and variance $\nu_{f_j \to x_i}$.

As in MPA, the iteration starts with the calculation of the messages from the factor node f_j to the variable nodes x_i. The means $z_{f_j \to x_i}$ and variances $\nu_{f_j \to x_i}$ are calculated as follows:

$$\begin{aligned} z_{f_j \to x_i}^t &= \mathbb{E}(w_{f_j \to x_i}) \\ &= \sum_{l \in m(j) \setminus i} h_{j,l}\mathbb{E}(x_l) \\ &= \sum_{l \in m(j) \setminus i} h_{j,l} \sum_{s=1}^{Z} \mu_{x_l \to f_j}^{t-1}(x_l = \alpha_s)\alpha_s, \end{aligned} \qquad (22)$$

$$\nu_{f_j \to x_i}^t = \sum_{l \in m(j) \setminus i} |h_{j,l}|^2 \sigma^2(x_l) + \sigma_0^2, \qquad (23)$$

where $\alpha_s \in \mathcal{A}$, $h_{j,l}$ is the element of the jth row and l^{th} column of matrix $\mathbf{H_{eff}}$, $\mathbb{E}(x)$ the expectation of x and $\sigma^2(x_l)$ is equal the variance of x_l, defined as:

$$\sigma^2(x_l) = \sum_{s=1}^{Z} \mu_{x_l \to f_j}^{t-1}(x_l)|\alpha_s|^2 - \left| \sum_{s=1}^{Z} \mu_{x_l \to f_j}^{t-1}(x_l)\alpha_s \right|^2. \qquad (24)$$

Then the variables node x_i updates its probability function that is conditioned to the corresponding value of $\mathbf{y}$ vector (y_b) to each x_i that belongs to a valid symbol in constellation (alphabet $\mathcal{A}$) and send to f_j that responds with the mean and variance of the others x_i.

Then the messages from the variables node to factor node are updated. The probabilities for each possible symbols $\mu_{x_i \to f_j}^t(x_i = \alpha_s)$ is calculated from the means and variances that correspond to the factor nodes linked to x_i as follows:

$$\mu_{x_i \to f_j}^t(x_i = \alpha_s) \propto \prod_{b \in n(i) \setminus j} \exp\left(\frac{-|y_b - z_{f_b \to x_i}^t - h_{b,i}\alpha_s|^2}{\nu_{f_b \to x_i}^t} \right). \qquad (25)$$

The marginal distribution $\mu_{x_i}^t(x_i)$ can be calculated taking into account all the incoming messages:

$$\mu_{x_i}^t(x_i) \propto \prod_{b \in n(i)} \exp\left(\frac{-|y_b - z_{f_b \to x_i}^t - h_{b,i}\alpha_s|^2}{\nu_{f_b \to x_i}^t} \right). \qquad (26)$$

Algorithm 2 FG-GAI algorithm

Initialization

Set $\left\{ \mu_{x_i \to f_j}(x_i) = 1/Z, \ z_{f_j \to x_i} = 0, \nu_{f_j \to x_i} = 0 \right\}$

for $t = 1$ to T **do** {Number of iterations}

    for $j = 1$ to O **do** {Computation of messages from FN to VN}

        Compute $z_{f_j \to x_i}^t$ using (22), $\forall i \in m(j)$

        Compute $\nu_{f_j \to x_i}^t$ using (23), $\forall i \in m(j)$

    end for

    for $i = 1$ to O **do** {Computation of messages from VN to FN}

        Compute $\mu_{x_i \to f_j}^t(x_i)$ using (25), $\forall j \in n(i)$

        Damping calculation by (19)

    end for

    for $i = 1$ to O **do** {Normalisation of messages from VN to FN}

        $\mu_{x_i \to f_j}(x_i) = \frac{\mu_{x_i \to f_j}(x_i)}{\sum_{\alpha_s \in Z} \mu_{x_i \to f_j}(x_i = \alpha_s)}, \forall j \in n(i)$

    end for

end for

Computation of LLR

Decision calculation

C. Approximate Message Passing using Gaussian Approximation (AMP-GA)

Another simplification of MPA using Gaussian Approximation is presented by [11]. It is called the Approximation Message Passing Using Gaussian Approximation (AMP-GA). In AMP-GA, basically the messages of mean and variances of the variable nodes are updated from messages of factor nodes by the calculation of a complex Gaussian function.

Let us denote $\mu_{x_i \to f_j}^t(x_i)$ the message sent from the variable node x_i to factor node f_j in the tth iteration, and let us denote $\mu_{f_j \to x_i}^t(x_i)$ the message from the factor node f_j to variable node x_i. Then, the message update rules are given by eq. (17) and eq. (18).

Knowing that symbols belong to a discrete set QAM symbols ($\alpha_s \in \mathcal{A}$), the calculation of the messages requires considerable complexity to marginalize a random vector $\mathbf{x} \setminus x_i$. To deal with such complexity, as in [11] the minimum of

Kullback-Leibler divergence criterion is applied in the AMP-GA to calculate parameters $\hat{x}^t_{x_i \to f_j}$ (mean of projection distribution) and $\hat{\tau}^t_{x_i \to f_j}$ (variance of projection distribution). The message updated from the factor nodes to variable nodes are equal to:

$$\hat{x}^t_{x_i \to f_j} = \sum_{\alpha_s \in A} \alpha_s \mu^t_{x_i \to f_j}(x_i = \alpha_s). \tag{27}$$

$$\hat{\tau}^t_{x_i \to f_j} = \sum_{\alpha_s \in A} |\alpha_s|^2 \mu^t_{x_i \to f_j}(x_i = \alpha_s) - |\hat{x}^t_{x_i \to f_j}|^2. \tag{28}$$

Considering x_i as a continuous random variable and approximating the message into a complex Gaussian function, $\mu^t_{f_j \to x_i}(x_i)$ can be calculated by integration in (17), as follows:

$$\mu^t_{f_j \to x_i}(x_i) = \sum_{\mathbf{x} \backslash x_i} f_j(y_j|\mathbf{x}) \prod_{l \in m(j) \backslash i} \mathcal{N}_{\mathbb{C}}\left(x_l; \hat{x}^t_{x_l \to f_j}, \hat{\tau}^t_{x_l \to f_j}\right)$$
$$\approx \mathcal{N}_{\mathbb{C}}\left(h_{j,i} x_i; z^t_{f_j \to x_i}, \nu^t_{f_j \to x_i}\right), \tag{29}$$

where $\mathcal{N}_{\mathbb{C}}(x; \hat{x}; \hat{\tau}) \triangleq (\pi\hat{\tau})^{-1} \exp(-|x - \hat{x}|^2/\hat{\tau})$ denotes a complex Gaussian function. The parameters $z^t_{f_j \to x_i}$ (mean messages from factor nodes to variable nodes) and $\nu^t_{f_j \to x_i}$ (variance messages from factor nodes to variable nodes) are given by:

$$z^t_{f_j \to x_i} = y_j - \sum_{l \in m(j) \backslash i} h_{j,l} \hat{x}^t_{x_l \to f_j}. \tag{30}$$

$$\nu^t_{f_j \to x_i} = \sigma_n^2 + \sum_{l \in m(j) \backslash i} |h_{j,l}|^2 \hat{\tau}^t_{x_l \to f_j}. \tag{31}$$

Now, substituting $\mu^t_{f_j \to x_i}(x_i) = \mathcal{N}_{\mathbb{C}}\left(h_{j,i} x_i; z^t_{f_j \to x_i}, \nu^t_{f_j \to x_i}\right)$ in (18), the messages $\mu^t_{x_i \to f_j}(x_i)$ can be normalized as follows:

$$\mu^t_{x_i \to f_j}(x_i) = \frac{\mathcal{N}_{\mathbb{C}}\left(x_i; \zeta^{t-1}_{x_i \to f_j}, \gamma^{t-1}_{x_i \to f_j}\right)}{\sum_{x_i \in A} \mathcal{N}_{\mathbb{C}}\left(x_i; \zeta^{t-1}_{x_i \to f_j}, \gamma^{t-1}_{x_i \to f_j}\right)}, \tag{32}$$

where $\gamma^{t-1}_{x_i \to f_j}$ (variance messages from variables nodes to factor nodes) and $\zeta^{t-1}_{x_i \to f_j}$ (means messages from variables nodes to factor node) are given by:

$$\gamma^t_{x_i \to f_j} = \left(\sum_{b \in n(i) \backslash j} \frac{|h_{b,i}|^2}{\nu^t_{f_b \to x_i}}\right)^{-1}. \tag{33}$$

$$\zeta^t_{x_i \to f_j} = \gamma^t_{x_i \to f_j} \sum_{b \in n(i) \backslash j} \frac{h^*_{b,i} z^t_{f_b \to x_i}}{\nu^t_{f_b \to x_i}}. \tag{34}$$

The marginal distribution $\mu^t_{x_i}(x_i)$ can be calculated as follows:

$$\mu^t_{x_i}(x_i = \alpha_s) \propto \exp\left(-\frac{|\alpha_s - \zeta^t_{x_i}|^2}{\gamma^t_{x_i}}\right), \tag{35}$$

where $\gamma^t_{x_i}$ and $\zeta^t_{x_i}$ are the estimated mean and variance of x_i:

$$\gamma^t_{x_i} = \left(\sum_{b \in n(i)} \frac{|h_{b,i}|^2}{\nu^t_{f_b \to x_i}}\right)^{-1}. \tag{36}$$

$$\zeta^t_{x_i} = \gamma^t_{x_i} \sum_{b \in n(i)} \frac{h^*_{b,i} z^t_{f_b \to x_i}}{\nu^t_{f_b \to x_i}}. \tag{37}$$

Algorithm 3 AMP-GA algorithm

Initialization
Set $\left\{\zeta^0_{x_i \to f_j}(x_i) = 0, \ \gamma^0_{x_i \to f_j} = 1000\right\}$
Set $\left\{\hat{x}^0_{x_i \to f_j} = 0, \ \hat{\tau}^0_{x_i \to f_j} = 0, \ z^0_{f_j \to x_i} = 0, \nu^0_{f_j \to x_i} = 0\right\}$
for $t = 1$ to **T do** {Number of iterations}
{Computation of messages from VN to FN}
  for $i = 1$ to O **do**
    Compute $\mu^t_{x_i \to f_j}(x_i)$ using (32), $\forall j \in n(i)$
    Compute $\hat{x}^t_{x_i \to f_j}$ using (27), $\forall j \in n(i)$
    Compute $\hat{\tau}^t_{x_i \to f_j}$ using (28), $\forall j \in n(i)$
  end for{Computation of messages from FN to VN}
  for $j = 1$ to O **do**
    Compute $z^t_{f_j \to x_i}$ using (30), $\forall i \in m(j)$
    Compute $\nu^t_{f_j \to x_i}$ using (31), $\forall i \in m(j)$
  end for
{Computation of messages from VN to FN}
  for $i = 1$ to O **do**
    Compute $\gamma^t_{x_i \to f_j}$ using (33), $\forall j \in n(i)$
    Compute $\zeta^t_{x_i \to f_j}$ using (34), $\forall j \in n(i)$
    Damping calculation by (19)
  end for
end for
Computation of LLR
Decision calculation

D. AMP simplified by Expectation Propagation (AMP-EP)

The AMP-EP proposed by [11] arises as an alternative to reduce the computational complexity of AMP-GA introduced in the calculation of the messages from the variable nodes to factor nodes $\mu^t_{x_i \to f_j}(x_i)$ in (32). These messages are replaced by the so-called symbol belief ($\beta^t(x_i)$) that is approximated by a Gaussian probability density function (PDF) as follows:

$$\beta^t(x_i) \triangleq \frac{\prod_{b \in n(i)} \mu^{t-1}_{f_b \to x_i}(x_i)}{\sum_{x_i \in A} \prod_{b \in n(i)} \mu^{t-1}_{f_b \to x_i}(x_i)}$$

$$\approx \frac{\prod_{b \in n(i)} \mathcal{N}_{\mathbb{C}}\left(h_{b,i} x_i; z^{t-1}_{f_b \to x_i}, \nu^{t-1}_{f_b \to x_i}\right)}{\sum_{x_i \in A} \prod_{b \in n(i)} \mathcal{N}_{\mathbb{C}}\left(h_{b,i} x_i; z^{t-1}_{f_b \to x_i}, \nu^{t-1}_{f_b \to x_i}\right)}. \tag{38}$$

Thus, in this approach, the message $\mu^t_{x_i \to f_j}(x_i)$ is replaced by the symbol belief that is based on a Gaussian PDF, i.e., in this case we have an approximate message $\mu^t_{x_i \to f_j}(x_i)$ calculated from an approximate belief of the symbol $\beta^t(x_i)$. After calculating the symbol belief for each variable node, starts the calculation of parameters $\hat{x}^t_{x_i}$ (means messages) and $\hat{\tau}^t_{x_i}$ (variance messages) and then the exchange of messages of the variable nodes to factor nodes ($\hat{x}^t_{x_i \to f_j}$ and $\hat{\tau}^t_{x_i \to f_j}$). To compute these parameters, firstly the values of $\hat{x}^t_{x_i}$ and $\hat{\tau}^t_{x_i}$ are updated using the calculated belief symbols, as follows:

$$\hat{x}^t_{x_i} = \sum_{\alpha_s \in A} \alpha_s \beta^t(\alpha_s). \tag{39}$$

$$\hat{\tau}^t_{x_i} = \sum_{\alpha_s \in A} |\alpha_s|^2 \beta^t(\alpha_s) - |\hat{x}^t_{x_i}|^2. \tag{40}$$

Finally, $\hat{x}^t_{x_i \to f_j}$ and $\hat{\tau}^t_{x_i \to f_j}$, the mean and variance messages from the variable nodes to the factor nodes are obtained as follows:

$$\hat{\tau}^t_{x_i \to f_j} = \left(\frac{1}{\hat{\tau}^t_{x_i}} - \frac{|h_{j,i}|^2}{\nu^{t-1}_{f_j \to x_i}} \right)^{-1}. \tag{41}$$

$$\hat{x}^t_{x_i \to f_j} = \hat{\tau}^t_{x_i \to f_j} \left(\frac{\hat{x}^t_{x_i}}{\hat{\tau}^t_{x_i}} - \frac{h^*_{j,i} z^{t-1}_{f_j \to x_i}}{\nu^{t-1}_{f_j \to x_i}} \right). \tag{42}$$

Now, the messages from the factor nodes to the variable nodes are updated with the values of $\hat{\tau}^t_{x_i \to f_j}$ and $\hat{x}^t_{x_i \to f_j}$ previously computed. As a result, the messages of variance ($\nu^t_{f_j \to x_i}$) and means ($z^t_{f_j \to x_i}$) will be used as input parameters for the calculation of the Gaussian PDF and thus will update the symbols belief $\beta^t(x_i)$ that will be the basis of calculation of the next iteration. The messages $z^t_{f_j \to x_i}$ and $\nu^t_{f_j \to x_i}$ are computed by equation (30) and (31).

Then, the marginal distribution $\mu^t_{x_i}(x_i)$ is obtained directly from the belief symbol $\beta^t(x_i)$ and the LLR can be obtained as in the MPA.

Algorithm 4 AMP-EP algorithm

Initialization

Set $\left\{ z^0_{f_j \to x_i} = 0, \nu^0_{f_j \to x_i} = 1000 \right\}$

Set $\left\{ \hat{x}^0_{x_i} = 0, \ \hat{\tau}^0_{x_i} = 0, \hat{x}^0_{x_i \to f_j} = 0, \ \hat{\tau}^0_{x_i \to f_j} = 0 \right\}$

for $t = 1$ to T **do** {Number of iterations}
{Computation of messages from FN to VN}
    for $i = 1$ to O **do**
        Compute $\beta^t(x_i)$ using (38), $\forall j \in n(i)$
    end for
    for $j = 1$ to O **do**
        Compute $\hat{x}^t_{x_i}$ using (39), $\forall i \in m(j)$
        Compute $\hat{\tau}^t_{x_i}$ using (40), $\forall i \in m(j)$
        Compute $\hat{x}^t_{x_i \to f_j}$ using (41), $\forall i \in m(j)$
        Compute $\hat{\tau}^t_{x_i \to f_j}$ using (42), $\forall i \in m(j)$
    end for
{Computation of messages from VN to FN}
    for $i = 1$ to O **do**
        Compute $z^t_{f_j \to x_i}$ using (30), $\forall j \in n(i)$
        Compute $\nu^t_{f_j \to x_i}$ using (31), $\forall j \in n(i)$
        Damping calculation by (19)
    end for
end for
Computation of LLR
Decision calculation

E. AMP simplified by First-Order (AMP-FO)

Further simplification AMP-EP, the last alternative of MPA for the reduction of complexity proposed by [11] is the AMP-FO. In this algorithm, the messages are rewritten after recursive updates and the negligible terms are omitted in the large system limit.

To adapt to OTFS decoding, we first rewrite the standard messages in (32) as follows:

$$\mu^t_{x_i}(x_i) = \frac{\mathcal{N}_{\mathbb{C}}\left(x_i; \zeta^{t-1}_{x_i}, \gamma^{t-1}_{x_i}\right)}{\sum_{x_i \in A} \mathcal{N}_{\mathbb{C}}\left(x_i; \zeta^{t-1}_{x_i}, \gamma^{t-1}_{x_i}\right)}, \tag{43}$$

where $\gamma^{t-1}_{x_i}$ (variance messages variables nodes) and $\zeta^{t-1}_{x_i}$ (means messages variables nodes) which are the messages exchanged from variables nodes to factor nodes are given by:

$$\gamma^t_{x_i} = \left(\sum_{b \in n(i)} \frac{|h_{b,i}|^2}{\nu^t_{f_b}} \right)^{-1}. \tag{44}$$

$$\zeta^t_{x_i} = \hat{x}^t_{x_i} + \gamma^t_{x_i} \sum_{b \in n(i)} \frac{h^*_{b,i} z^t_{f_b}}{\nu^t_{f_b}}. \tag{45}$$

with $z^t_{f_b}$ and $\nu^t_{f_b}$ the means and variance messages from factor nodes to variable nodes, respectively. To initiate the exchange of messages from factor nodes to variable nodes, eq. (43) is updated for all variable nodes and then the mean and variance of projection distribution for each symbol of the QAM alphabet is calculated as:

$$\hat{x}^t_{x_i} = \sum_{\alpha_s \in A} \alpha_s \mu^t_{x_i}(x_i = \alpha_s). \tag{46}$$

$$\hat{\tau}_{x_i}^t = \sum_{\alpha_s \in A} |\alpha_s|^2 \mu_{x_i}^t (x_i = \alpha_s) - |\hat{x}_{x_i}^t|^2. \quad (47)$$

Then, all the means and variances of the messages exchanged from factor nodes to variables nodes are calculated as follows:

$$z_{f_j}^t = y_j - \sum_{l \in m(j)} h_{j,l} \ \hat{x}_{x_l}^t + z_{f_j}^{t-1} \frac{\sum_{l' \in m(j)} \hat{\tau}_{x_{l'}}^t |h_{j,l'}|^2}{\nu_{f_j}^{t-1}}. \quad (48)$$

$$\nu_{f_j}^t = \sigma_n^2 + \sum_{l \in m(j)} |h_{j,l}|^2 \ \hat{\tau}_{x_l}^t. \quad (49)$$

Algorithm 5 AMP-FO algorithm

Initialization

Set $\left\{ z_{f_j \to x_i}^0 = 0, \nu_{f_j \to x_i}^0 = 1000 \right\}$

Set $\left\{ \zeta_{x_i}^0 = 0, \gamma_{x_i}^0 = 1000 \right\}$

for $t = 1$ to T **do** {Number of iterations}

{Computation of messages of FN}

   for $i = 1$ to O **do**

      Compute $\mu_{x_i^t (x_i)}$ using (43)

      Compute $\hat{x}_{x_i}^t$ using (46)

      Compute $\hat{\tau}_{x_i}^t$ using (47)

   end for

{Computation of messages of VN}

   for $j = 1$ to O **do**

      Compute $z_{f_j}^t$ using (48)

      Compute $\nu_{f_j}^t$ using (49)

   end for

{Computation of messages of FN}

   for $i = 1$ to O **do**

      Compute $\gamma_{x_i}^t$ using (44)

      Compute $\zeta_{x_i}^t$ using (45)

      Damping calculation by (19)

   end for

end for

Computation of LLR

Decision calculation

IV. COMPLEXITY ANALYSIS

In this section, we analyze the complexity of the considered algorithms by counting their required number of floating-point operations (FLOP). Flop counts are obtained by adding the arithmetic operations associated with the most deeply nested statements in an algorithm [19]. In the previous section we have presented the simplifications introduced by each algorithm in order to reduce the complexity starting from the MPA, followed by FG-GAI, AMP-GA, AMP-EP and AMP-FO (from the most complex to the less complex algorithm).

All the message passing algorithms have a preprocessing step to compute the square norms $|h_{j,l}|^2$ that requires $3PO$ FLOPs.

The complexity of MPA and FG-GAI is mainly due to the message exchange from VN to FN and from FN to VN. However MPA has an additional preprocessing step for calculating

$f_j(y_j|x_i)$ that needs $(14P - 15)ZO$ FLOPs. Processing the $\mu_{f_j \to x_i}^t$ messages requires $(Z^{(P-1)} - 1)(P - 1)ZPO$ FLOPs while the calculation $\mu_{x_i \to f_j}^t$ requires $(P - 2)ZPO$ FLOPs for each iteration. The FG-GAI needs $[6Z^2 - 4Z + (2Z + 2)P - 7]PO$ FLOPs to calculate $z_{f_j \to x_i}^t$ and $\nu_{f_j \to x_i}^t$ and $[(15P - 1)Z - 1]PO$ FLOPs to compute $\mu_{x_i}^t(x_i)$.

The algorithms AMP-GA, AMP-EP and AMP-FO have three steps following the preprocessing step. In the VN messages step, AMP-GA requires $15ZPO$ FLOPs to compute $\{\mu_{x_i \to f_j}^t, \hat{x}_{x_i \to f_j}^t, \hat{\tau}_{x_i \to f_j}^t\}$, AMP-EP requires $(16P + Z - 1)O$ FLOPs to compute $\beta(x_i)^t$ and AMP-FO needs $(15Z + 2)O$ FLOPs to compute $\{\mu_{x_i}(x_i)^t, \hat{x}_{x_i}^t, \hat{\tau}_{x_i}^t\}$. In the FNs step, to calculate $\{z_{f_j \to x_i}^t, \nu_{f_j \to x_i}^t\}$ the AMP-GA needs $(10P - 10)PO$ FLOPs, to compute $\{\hat{x}_{x_i}^t, \hat{\tau}_{x_i}^t, \hat{x}_{x_i \to f_j}^t, \hat{\tau}_{x_i \to f_j}^t\}$ AMP-EP needs $(16P + Z - 1)O$ FLOPs and AMP-FO $(12P + 4)O$ FLOPs to calculate $\{z_{f_j}^t, \nu_{f_j}^t\}$. In the update of messages from VNs AMP-GA needs $(12P - 12)PO$ FLOPs to compute $\{\gamma_{x_i \to f_j}^t, \zeta_{x_i \to f_j}^t\}$, AMP-EP $(10P - 10)PO$ FLOPs to evaluate $\{z_{f_j \to x_i}^t, \nu_{f_j \to x_i}^t\}$ and AMP-FO $(12P + 2)O$ FLOPs to calculate $\{\gamma_{x_i}^t, \zeta_{x_i}^t\}$.

Table I shows the number of total FLOPs per iteration of each algorithms as a function of the size of the used Z-QAM modulation, the number of paths P and the number of VNs and FNs O.

TABLE I
COMPLEXITY BY TOTAL NUMBER OF FLOPs PER ITERATION

Algorithm	FLOPs
MPA	$[(Z^{P-1})(P - 1) + P + 12]ZPO - 15ZO$
FG-GAI	$[6Z^2 + (15P - 5)Z + (2Z + 2)P - 5]PO$
AMP-GA	$(15Z + 22P - 19)PO$
AMP-EP	$(7Z + 16P)O + (10P + 11)PO$
AMP-FO	$(15Z + 24P + 8)O + 3PO$

Based on the expressions of Table I, Figure 4 presents the complexity in terms of FLOPs as a function of the constellation size Z for each algorithm. We have considered the case $P = 4$ paths, 64 subcarriers with 64 symbols ($O = 4096$). We will use the same set of parameters to evaluate the bit error rate performance in this study.
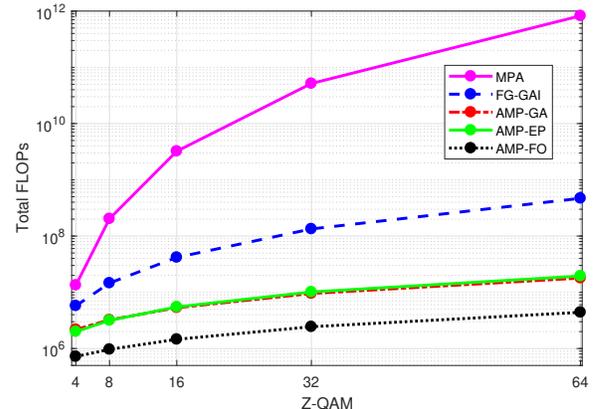
Fig. 4. Complexity Evaluation of the Studied Algorithms

As expected, we can see in Figure 4 that the MPA algorithm is the most complex and its complexity increases considerably when increasing Z, while AMP-FO is the less complex and its complexity increases more slowly when increasing the constellation size. In FG-GAI, replacing $\mu_{f_j x_i}$ with means and variances reduces the complexity by a factor of 75 at $Z = 16$ compared to MPA. By replacing the messages by means and variances, the complexity of the AMP algorithms is further reduced. A complexity reduction factor of 10 is obtained between FG-GAI and AMP-EP. Finally, the AMP-FO algorithm achieves the lowest complexity since its complexity is about 30% of the one of the EP algorithm by computing only the mean and variance information at the VNs and FNs nodes.

V. EXTENSION TO MIMO SYSTEMS

The input-output relation of a SISO-OTFS system given in equation (15) will be replace by MIMO input-output relation taking into account the multiple transmit and receive antennas [14]. Let us define the vectorized received signal at the jth antenna as $\mathbf{y}_j$ and $\mathbf{x}_i$ as the vectorized signal at the ith transmit antenna. We assuming a MIMO system composed of n_t transmit antennas and n_r receive antennas. Then, from equation (15), we have the following set of input-output equations:

$$\begin{aligned}
\mathbf{y}_1 &= \mathbf{H}_{11}\mathbf{x}_1 + \mathbf{H}_{12}\mathbf{x}_2 + ... + \mathbf{H}_{1n_t}\mathbf{x}_{n_t} + \tilde{\mathbf{w}}_1, \\
\mathbf{y}_2 &= \mathbf{H}_{21}\mathbf{x}_1 + \mathbf{H}_{22}\mathbf{x}_2 + ... + \mathbf{H}_{2n_t}\mathbf{x}_{n_t} + \tilde{\mathbf{w}}_2, \\
&\vdots \\
\mathbf{y}_{n_r} &= \mathbf{H}_{n_r 1}\mathbf{x}_1 + \mathbf{H}_{n_r 2}\mathbf{x}_2 + ... + \mathbf{H}_{n_r n_t}\mathbf{x}_{n_t} + \tilde{\mathbf{w}}_{n_r},
\end{aligned} \quad (50)$$

where $\mathbf{H}_{ji}$ is the effective channel vector between the ith transmit antenna and the jth received antenna.

We can rewrite equation (50) in a compact form as follows:

$$\mathbf{y}_{MIMO} = \mathbf{H}_{MIMO}\mathbf{x}_{MIMO} + \tilde{\mathbf{w}}_{MIMO}, \quad (51)$$

where $\mathbf{H}_{MIMO}$ is the MIMO effective channel matrix

$$\mathbf{H}_{MIMO} = \begin{bmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} & \ldots & \mathbf{H}_{1n_t} \\ \mathbf{H}_{21} & \mathbf{H}_{22} & \ddots & \mathbf{H}_{2n_t} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{H}_{n_r 1} & \mathbf{H}_{n_r 2} & \ldots & \mathbf{H}_{n_r n_t,} \end{bmatrix}, \quad (52)$$

and $\mathbf{x}_{MIMO} = [\mathbf{x}_1^T, \mathbf{x}_2^T ... , \mathbf{x}_{n_t}^T]^T$ is the vectorized symbols transmitted vector, $\mathbf{y}_{MIMO} = [\mathbf{y}_1^T, \mathbf{y}_2^T ... , \mathbf{y}_{n_r}^T]^T$ is the vectorized received signal vector and $\tilde{\mathbf{w}}_{MIMO} = [\tilde{\mathbf{w}}_1^T, \tilde{\mathbf{w}}_2^T ... , \tilde{\mathbf{w}}_{n_r}^T]^T$ is the noise vector of MIMO-OTFS systems.

According to this context, extending the study to MIMO-OTFS systems implies a substantial increase in the complexity of the algorithms of the Section III, mainly of the MPA that takes into account each edge between variable node and factor node which depends on the number of paths. Besides that, in a multipath scenario each antenna of the MIMO system configuration will suffer with the fading of each path.

VI. SIMULATION RESULTS

In this section, we will evaluate the Bit Error Rate (BER) performance of the OTFS system considering the different low-complexity algorithms presented in Section III (MPA, FG-GAI, AMP-GA, AMP-EP and AMP-FO) over a delay-Doppler channel model in a multipath scenario. The simulation parameters are given in Table II.

TABLE II
SIMULATION PARAMETERS

Parameter	Value
Carrier frequency (f_c)	4 GHz
Subcarrier spacing (Δf)	15 kHz
Number of subcarriers (M)	64
Number of OTFS symbols (N)	64
Number of paths (P)	4
Modulation scheme	4-QAM
UE speed	60, 120, 180 km/h
Channel estimation	ideal
Doppler shift	234 Hz
Delay shift	1 μs

The channel model is the delay-Doppler Profile multipath model [14]. Based on the parameters given in Table II, we have considered two different scenarios, both with four paths, where each reflector has delay shift multiple of 1 μs and Doppler shifts multiple of 234 Hz. The delay and Doppler shifts for each path in the two considered scenarios are provided in the Table III and Table IV, respectively. In Scenario 1, all reflectors have different Doppler shifts but they are all in the same direction while in Scenario 2 the reflectors are in different directions (two positive and two negative Doppler shifts).

TABLE III
DDP FOR MULTIPATH CHANNEL MODEL - SCENARIO 1

Path index (i)	1	2	3	4
Delay (τ_i)	0 μs	1 μs	2 μs	3 μs
Delay tap (l_i)	0	1	2	3
Doppler (ν_i)	0 Hz	234 Hz	468 Hz	702 Hz
Doppler tap (k_i)	0	1	2	3

TABLE IV
DDP FOR MULTIPATH CHANNEL MODEL - SCENARIO 2

Path index (i)	1	2	3	4
Delay (τ_i)	0 μs	1 μs	2 μs	3 μs
Delay tap (l_i)	0	1	2	3
Doppler (ν_i)	-234 Hz	468 Hz	234 Hz	-234 Hz
Doppler tap (k_i)	-1	2	1	-1

We have first performed a study of the BER performance as a function of number of iterations for each algorithm to establish which value is the more appropriate for the presented system. The results for the scenario 2 and considering $SNR = 12\ dB$ are shown in Figure 5.
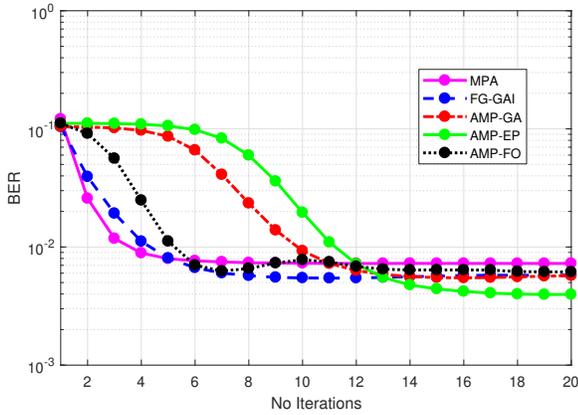
Fig. 5. Analysis of number of iterations for Scenario 2

According to Figure 5, we can infer that, for the considered DDP model, the required number of iterations for FG-GAI algorithm is 10, AMP-GA is 15, AMP-EP is 20 and AMP-FO is 15 iterations. The MPA algorithm converges faster and consequently requires only 5 iterations. Once the required number of iterations has been determined, we have studied the impact of the damping factor for scenario 2 using eq. (19) on the BER performance in the range of 0.45 to 0.75. The influence of the damping factor is given on Figure 6.
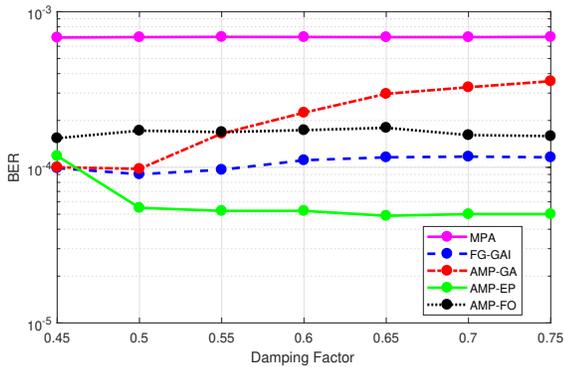

Fig. 6. Analysis of damping factor for Scenario 2

Figure 6 shows that the damping factor does not bring significant gains with regard to BER. The best damping factor for FG-GAI algorithm is 0.55, for AMP-GA is 0.5, AMP-EP is 0.65 and AMP-FO is 0.6. The MPA algorithm showed no variations in BER for the considered range. The same number of iterations and damping factors have been obtained when considering the scenario 1. Table V presents a summary of the damping factors and number of iteration that will be used in the next simulations.

TABLE V
DAMPING FACTOR AND ITERATIONS VALUES

Algorithm	Iterations	damping Factor
MPA	5	0.6
FG-GAI	10	0.55
AMP-GA	15	0.5
AMP-EP	20	0.65
AMP-FO	15	0.6

Using the above parameters, we have simulated the BER performance of the OTFS system for each low-complexity algorithm detector in a SNR range of -13 to 17 dB and both scenarios presented in tables III and IV. Figure 7 and Figure 8 present the BER performance as a function of the SNR for Scenario 1 and Scenario 2 respectively.
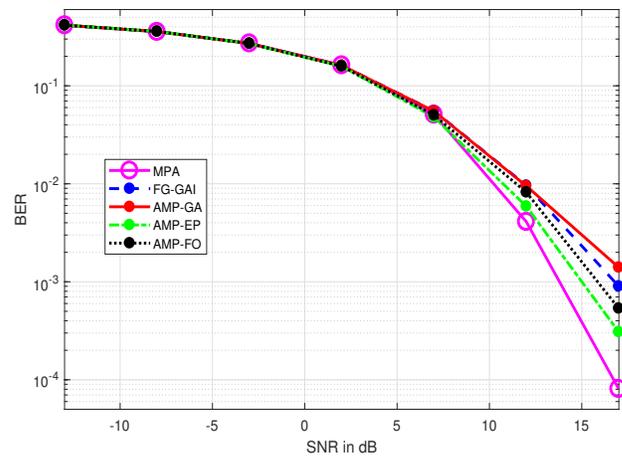

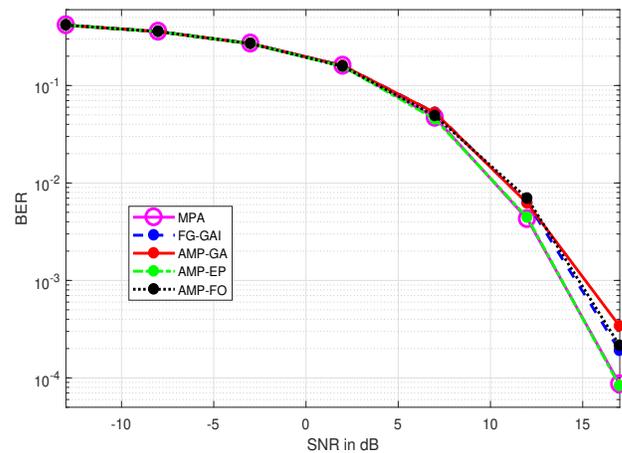Fig. 7. BER Performance of OTFS with Scenario 1

Fig. 8. BER Performance of OTFS with Scenario 2

We used the channel model defined by equation (6), then all graphs are based on the same $\mathbf{H_{eff}}$ matrix.

As shown on Figures 7 and 8, all algorithms have similar performance at low SNR, however we can observe differences from 12 dB. In Scenario 1, the MPA algorithm, which is the

most complex algorithm, achieves the best BER performance. At BER= 10^{-3} the performance losses of AMP-EP and AMP-FO are 1.25 dB and 2.25 dB respectively with respect to MPA. On the other hand, in Scenario 2 the AMP-EP achieves the same performance than the MPA. At BER= 10^{-3} the performance loss of AMP-FO is 0.8 dB with respect to the MPA. In both scenarios, AMP-GA has the worst performance, followed by FG-GAI and AMP-FO.

In both scenarios, the BER performance of the AMP-FO algorithm is relatively closed to the one of the AMP-EP while AMP-FO is significantly less complex. We can conclude that AMP-FO in both studied scenarios provide a good compromise between its complexity and its achieved BER performance.

In relation to the MIMO-OTFS, due to the considerable increase in the complexity of the algorithms, we chose the three least complex algorithms based on AMP: AMP-GA, AMP-EP and AMP-FO to analyze the performance for the second scenario considering a set of two transmit antennas and two received antennas. Thus, according to (52), $\mathbf{H}_{MIMO}$ will have a dimension of $2MN \times 2MN$ (8192×8192) and therefore we will have now 8 non-zero elements in each row and in each column of the factor graph associated to the effective channel matrix since each antenna will suffer from the fading of each of the 4 paths of the considered multipath scenario. The Figure 9 shows the result for the MIMO-OTFS case:
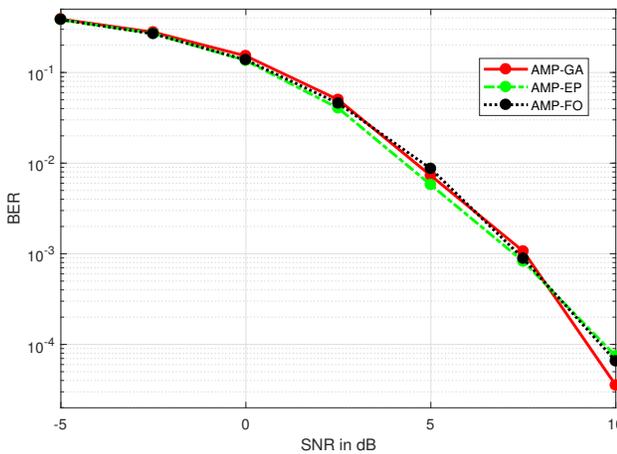

Fig. 9. BER Performance of 2x2 MIMO-OTFS to Scenario 2

As expected, the MIMO-OTFS scheme performs better than the SISO-OTFS, where a gain of 5 dB is observed at $BER = 10^{-1}$. Due to the spatial diversity, the detector algorithms presented a similar performance behavior, diverging a little at 10 dB. In other words, the less complex algorithm AMP-FO also has the best performance-complexity tradeoff for MIMO-OTFS.

VII. CONCLUSIONS AND PERSPECTIVES

In this paper we have studied, evaluated and compared four low-complexity MPA-based detectors for OTFS systems over time-frequency selective channel with high Doppler in term of BER performance and complexity analysis. As expected, the MPA and AMP-EP algorithms achieve the best BER performance. However, the complexity AMP-FO algorithm is significantly lower since it is only about 30% of the complexity of the AMP-EP algorithm. The AMP-FO algorithm is the less complex studied algorithm and it allows a BER performance degradation of less than 1 dB and 2.25 dB at BER $= 10^{-3}$ compared to the AMP-EP and the MPA algorithms respectively in SISO-OTFS. Indeed, the AMP-FO algorithm gives the best performance-complexity tradeoff in both the SISO-OTFS system and the MIMO-OTFS system

As future works, we will study applications of channel estimation techniques in OTFS and consider the extension to massive MIMO systems.

REFERENCES

[1] Y. Carmon, S. Shamai, and T. Weissman, "Comparison of the achievable rates in OFDM and single carrier modulation with i.i.d. inputs," *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 1795–1818, Apr. 2015. [Online] Available: https://arxiv.org/pdf/1306.5781.pdf.

[2] S. Dixit and H. Katiyar, Performance of OFDM in time selective multipath fading channel in 4G systems," *in IEEE 2015 Fifth International Conference on Communication Systems and Network Technologies*, pp. 421–424, Apr. 2015. DOI: 10.1109/CSNT.2015.107.

[3] P. Raviteja, et al., "Interference cancellation and iterative detection for orthogonal time frequency space modulation," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6501-6515, Aug. 2018. DOI: 10.1109/TWC.2018.2860011.

[4] R. Hadani, S. Rakib, M. Tsatsanis, A. Monk, A. J. Goldsmith, A. F. Molisch, and R. Calderbank, "Orthogonal time frequency space modulation," *in Proc. IEEE WCNC*, pp. 1558-2612, Mar. 2017. [Online] Available: https://arxiv.org/pdf/1808.00519.pdf.

[5] F. Wiffen, L. Sayer, M. Z. Bocus, A. Doufexi, and A. Nix, "Comparison of OTFS and OFDM in Ray launched sub-6 GHz and mmWave line-of-sight mobility channels," *in IEEE 29th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sep. 2018. DOI: 10.1109/PIMRC.2018.8580850.

[6] A. R. Reyhani, A. Farhang, M. Ji, R. R. Chen, and B. Farhang-Boroujeny, "Analysis of discrete-time MIMO OFDM-based orthogonal time frequency space modulation," Oct. 2017. [Online] Available: https://arxiv.org/abs/1710.07900v1.pdf.

[7] T. Zemen, M. Hofer, and D. Loeschenbrand, "Low-complexity equalization for orthogonal time and frequency signaling (OTFS)," Oct. 2017. [Online] Available: https://arxiv.org/pdf/1710.09916v1.pdf.

[8] K. R. Murali and A. Chockalingam. "On OTFS modulation for high-doppler fading channels," Oct. 2018. [Online] Available: https://arxiv.org/abs/1802.00929.pdf.

[9] W. Shen, L. Dai, J. An, P. Fan and R. W. Heath, "Channel estimation for orthogonal time frequency space (OTFS) massive MIMO," *IEEE Transactions on Signal Processing*. May 2019. DOI: 10.1109/TSP.2019.2919411.

[10] P. Som, T. Datta, N. Srinidhi, A. Chockalingam, and B. S. Rajan, "Low-complexity detection in large-dimension MIMO-ISI channels using graphical models," *IEEE J. Sel. Topics in Signal Processing*, vol. 5, no. 8, pp. 1497–1511, Dec. 2011. DOI: 10.1109/JSTSP.2011.2166950.

[11] S. Wu, L. Kuang, Z. Ni, J. Lu, D. Huang, and Q. Guo, "Low-complexity detection for large-scale multiuser MIMO-OFDM systems using approximate message passing," *IEEE Journal Sel. Topics in Signal Processing*, vol. 8, no. 5, pp. 902-915, Oct. 2014. DOI: 10.1109/JSTSP.2014.2313766.

[12] R. Hadani and A. Monk, "OTFS: A new generation of modulation addressing the challenges of 5G," Feb. 2018. [Online] Available: https://arxiv.org/ftp/arxiv/papers/1802/1802.02623.pdf.

[13] R1- 1610397, "OTFS PAPR analysis," *in 3GPP TSG RA WG1 Meeting 86*, Oct. 2016.

[14] M. K. Ramachandran and A. Chockalingam, "MIMO-OTFS in high-doppler fading channels: Signal detection and channel estimation," Dec. 2018. DOI: 10.1109/glocom.2018.8647394.

[15] Z. Ding, R. Schober, P. Fan, and H. V. Poor, "OTFS-NOMA: An efficient approach for exploiting heterogenous user mobility profiles," Apr. 2019. [Online] Available: https://arxiv.org/pdf/1904.02783v1.pdf.

[16] P. Raviteja, Y. Hong, E. Viterbo, and E. Biglieri, "Practical pulse-shaping waveforms for reduced-cyclic-prefix OTFS," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 957-961, Oct. 2018. DOI: 10.1109/TVT.2018.2878891.

[17] P. Raviteja, K. T. Phan, Q. Jin, Y. Hong, and E. Viterbo, "Low-complexity iterative detection for orthogonal time frequency space modulation," *in 2018 IEEE Wireless Communications and Networking Conference (WCNC)*, Apr. 2018. DOI: 10.1109/WCNC.2018.8377159.

[18] F. R. Kschischang, B. J. Frey, and Hans-Andrea Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, Vol. 47, no. 2, pp. 498-519, Feb. 2001. DOI: 10.1109/18.910572.

[19] G. H. Golub and C. F. Van Loan, " Matrix computations," London, UK: The Johns Hopkins University Press, 1996.

[20] J. Peter, "Calculation of log-likelihood ratios in a demodulator," *European Patent Application 110150664.0*, Jul. 2011. Patent EP2346223.

Mylene Pischella is an Associate Professor in Telecommunications at Conservatoire National des Arts et Métiers (CNAM) since 2010 since 2010 and has been the deputy director of CEDRIC lab, CNAM, since 2019. She received a Masters degree in engineering in 2002 and a Ph.D in communications and electronics in 2009, both from TELECOM ParisTech. From 2002 to 2009, she was a research engineer at Orange Labs, where she specialized on the optimization of cellular networks and contributed to several European collaborative projects. In 2009-2010, she was an assistant professor at ISEP, Paris, France. In 2019-2020, she was an invited researcher at UCLouvain and at ETIS (UMR 8051). She is an author of more than 70 peer-reviewed papers, three books, one book chapter, and seven patents. Mylene Pischella's research interests are resource allocation in wireless networks, including heterogeneous networks, Device-to-Device communications, NOMA, multi-carrrier modulations, cognitive, cooperative and relaying networks.

Dayse Gonçalves Correia Bandeira is currently a Professor in the area of Telecommunications Systems and Networks at the Federal Institute of Ceara (IFCE), Brazil, and a PhD student on Teleinformatics Engineering at the Federal University of Ceara (UFC) in the Wireless Telecommunications Research Group (GTEL). She is in a sandwich doctorate program by CAPES-COFECUB Projetc at CNAM (Conservatoire National des Arts et Metiers), France, in the CEDRIC laboratory. She was in charge of the coordination of the course of Telematics at IFCE-Taua Campus, Brazil. She received the M.Sc. in Telecommunications Engineering (2015) from IFCE, Brazil, with focus on microwave and integrated optics, at the Photonics Laboratory, IFCE. She is graduated in Telecommunications Engineering (2013) and in Telematics Technology (2007). She also finished the Telecommunications Technician course (2003) at IFCE. Besides the academic experience, she worked as a Technician Specialist in Telecommunications Services (2010-2013) at Embratel in the optical backbone networks area; as a Headend Technician (2003-2009) at NET Fortaleza; and as a Maintenance Technician (2009-2010) at TV Jangadeiro, making adjustments in microwave links for live broadcasts, as well as equalizations in the analog and digital signals of the radio systems and TV, Brazil.

João Cesar Moura Mota is currently a full professor at the UFC. He holds a bachelor's degree in Physics from the Federal University of Ceará (UFC, July 1978), a Master's degree in Telecommunications Engineering from the Pontifical Catholic University of Rio de Janeiro (PUC-Rio, September 1984) and a PhD in Telecommunication Engineering from the State University of Campinas (UNICAMP, August 1992). He was Chief of both Departments: Teleinformatic Engineering and Electrical Engineering, and Deputy Director of Interinstitutional Relations of the Technology Center/UFC. Dr. Mota is currently The Director of Development and Innovation Integrated School of UFC. He was a visiting researcher at TELECOM Sud-Paris (Evry, Aug 1996- Jul 1998), IRCCyN Laboratory (currently LS2N Laboratory, Nantes, Jun-Jul 2006), Laboratory I3S / CNRS (Sophia-Antipolis, Sept 2008- Jul 2009) and Normandy Mechanics Laboratory ((Rouen, Nov 2018-Feb 2019), France. He is a founding member and advisor of the Brazilian Society of Telecommunications, a member of the Brazilian Society of Informatics in Health, former student supervisor of the Institute of Electrical and Electronic Engineers (IEEE) of UFC, a member of the IEEE societies: Signal Processing and Communications. Has experience of research in the area of Telecommunications Engineering, with emphasis on Signal Processing in Communications Systems, working mainly on the following topics: identification and equalization of mobile communication systems, and data surface reconstruction and optimization. Other areas of interest and performance: waveform optimization; multi-linear signal processing; physics, mathematics and scientific methodology for the training of engineers; system modeling; numerical methods and educometry.

Didier Le Ruyet (M'98, SM'11) received his Eng. Degree and his Ph. D. degree from Conservatoire National des Arts et Métiers (CNAM) in 1994 and 2001 respectively. In 2009, he received the "Habilitation à diriger des recherches" from Paris XIII University. From 1988 to 1996 he was a Senior Member of the Technical Staff at SAGEM Defence and Telecommunication, France. He joined Signal and Systems Laboratory, CNAM, Paris as a research assistant in 1996. From 2002 to 2010, he was an assistant professor at CNAM Paris. Since 2010 he is full professor at CNAM in the CEDRIC research laboratory. From 2016 until 2019 he was the deputy director of the CEDRIC.

He has published about 200 papers in refereed journals and conference proceedings and published five books in the area of digital communication.

He has been involved in several National and European projects dealing with multicarrier transmission techniques and multi-antenna transmission. He served as Technical Program Committee member in major IEEE ComSoc and VTS conferences, and as General Chair of the ninth edition of ISWCS'2012 conference and the Co-Chair of the ISWCS 2013 and 2019 edition.

His main research interests lie in the areas of digital communications and signal processing including channel coding, detection and estimation algorithms, filter bank based multi-carrier communication and multi-antenna transmission.