HAL
open science

A Computationally Efficient Discrete Bit-Loading Algorithm for OFDM Systems Subject to Spectral-Compatibility Limits

Thanh Nhân Vo, Karine Amis Cavalec, Thierry Chonavel, Pierre Siohan

To cite this version:

Thanh Nhân Vo, Karine Amis Cavalec, Thierry Chonavel, Pierre Siohan. A Computationally Efficient Discrete Bit-Loading Algorithm for OFDM Systems Subject to Spectral-Compatibility Limits. IEEE Transactions on Communications, 2015, 63 (6), pp.2261 - 2272. 10.1109/TCOMM.2015.2424890 . hal-01197396

HAL Id: hal-01197396

https://hal.science/hal-01197396v1

Submitted on 11 Sep 2015

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.

A Computationally Efficient Discrete Bit-Loading Algorithm for OFDM Systems Subject to Spectral-Compatibility Limits

Thanh Nhan VO[1], Karine AMIS[1], Thierry CHONAVEL[1], Pierre SIOHAN[2]
1: Telecom Bretagne / UMR CNRS 6285 Lab-STICC; 2: Orange Labs

***Abstract*—This paper considers bit-loading algorithms to maximize throughput under total power and spectral mask constraints in interference-free OFDM systems. The contribution is twofold. First, we propose a simple criterion to switch between two well-known algorithms of the literature: the conventional Greedy and Greedy-based bit-removing (with maximum allowable bit loading initialization) algorithms. Second, we present a new low-complexity loading algorithm that exploits the bit vector obtained by rounding the water-filling algorithm solution to the associated continuous-input rate maximization problem as an efficient initial bit vector of the Greedy algorithm. We theoretically prove that this bit vector has two interesting properties. The first one states that it is an efficient bit vector, i.e., there is no movement of a bit from one subcarrier to another that reduces the total used power. The second one states that the optimized throughput, starting from this initial bit vector, is achieved by adding or removing bits on each subcarrier at most once. Simulation results show the efficiency of the proposed algorithm, i.e., the achievable throughput is maximized with significant reduction of computation cost as compared to many algorithms in the literature.**

***Index Terms*—Bit-loading, OFDM, Greedy Algorithm, Rate-adaptive, Low-complexity algorithm.**

I. Introduction

OFDM has been adopted in many wireless communication systems such as IEEE 802.11a/g (WLAN), IEEE 802.16 WiMax and recent long-term evolution (LTE) standard [1], [2], [3]. It is also exploited in wired systems such as asymmetric digital subscriber line (ADSL) or IEEE P1901 power line communication (PLC) [4]. In OFDM systems, with channel state information available at the transmitter, a loading algorithm can be used to allocate power and bits to the subcarriers under given constraints.

In the wired communication systems such as ADSL and PLC, a spectral mask constraint (i.e., peak-power constraint) must be taken into account to ensure the compatibility with other radio systems [4], [5]. Several optimal discrete bit-loading algorithms have been proposed in the literature e.g. [6], [7], [8], [9], [10], [11]. Many other works such as [12], [13], [14] proposed a sub-optimal solution by *rounding off* the continuous solution of an optimization problem and claimed complexity advantages over the conventional Greedy algorithm [15] with minimum performance difference with respect to the optimal discrete solution. While all these algorithms provide different performance to complexity trade-off possibilities for discrete bit allocation, the problem of peak-power constraint has not been exclusively addressed. Major contributions related to discrete bit allocation under the peak-power constraint were done by Baccarelli *et al.* in [16] for the continuous bit-loading and in [15] for the discrete bit-loading and by Papandreou *et al.* in [11]. In [16], a solution for the discrete rate maximization is given by compensating the solution of the continuous rate maximization. It introduces a variable α so that the total power use corresponding to the integer bit-loading after compensation is as close as possible to the total allowable power. However, in [16], the algorithm optimality was not proved and the final bit allocation depends on the number of iterations used to fix α. In [15], it is demonstrated that the conventional Greedy algorithm yields the global optimum for the discrete bit-loading problem. Unfortunately, its complexity is a non-decreasing function of the total allowable power. In [11], it is claimed that the bit-removing algorithm should be used to solve the discrete rate maximization under the total power and peak-power constraints. Its performance in terms of computation cost is a non-increasing function of the total allowable power. The first contribution of this paper is a criterion to switch between the conventional Greedy algorithm and bit-removing algorithm to reduce the global computation cost.

Our second contribution is a new low-complexity bit loading algorithm for the throughput maximization problem under total power and peak-power constraints. To this end, instead of using a zero bit loading initialization in the conventional Greedy algorithm or the maximum allowable bit loading initialization in the bit-removing algorithm in [11], we propose a novel initial bit vector resulting from the rounding of the Water-filling (WF) solution of the continuous bit loading problem. This approach has been used in [17] for the rate maximization in OFDM systems with the presence of interference resulting from an insufficient cyclic prefix and its efficiency has been shown through simulation results. However, for the problem in [17], the optimality of this approach has not been demonstrated. In this work, for the discrete bit-loading problem in interference-free OFDM systems, we prove that the use of the proposed initial bit vector in the Greedy procedure can achieve the global optimum solution. In addition, starting from this proposed initial bit vector, we also prove that to obtain the global optimum bit-loading, the number of bits per subcarrier needs to be increased or removed at most once. The computation cost as well as the run-time are theoretically analyzed and compared through simulations for the proposed algorithm,

the conventional Greedy algorithm [15], the Greedy-based bit-removing algorithm with maximum allowable bit loading initialization [11] and the sub-optimal algorithm in [16].

The paper is organized as follows. The throughput maximization problem under the total power and peak-power constraints and some well-known algorithms of the literature to solve it are given in Section II. Section III analyzes the hybrid approach between the conventional Greedy algorithm and bit-removing algorithm. The new low-complexity loading algorithm is given in Section IV. Simulation results are reported in Section V. Finally, Section VI is dedicated to conclusions and perspectives.

II. Discrete bit-loading problem and existing algorithms

A. Discrete bit-loading problem

The discrete bit-loading problem in interference-free OFDM systems, i.e., zero-Doppler (no ICI) and cyclic prefix longer than the channel response (no ISI), under the total power and peak-power constraints is given in (1),

$$\begin{aligned} \text{maximize} \quad & \sum_n b_n \\ \text{s.t.} \quad & b_n \leq A_{max} \mid b_n \in \mathbb{N} \\ & b_n \leq c_n = \log_2\left(1 + \frac{g_n P_n}{\Gamma}\right) \\ & \sum_n P_n \leq P_{tot} \\ & P_n \leq P_n^{max} \end{aligned} \tag{1}$$

where b_n and P_n are the number of bits and the power allocated to the subcarrier n; A_{max} is the maximum number of bits defined by the maximum order constellation; g_n is the channel gain to noise ratio; $\Gamma \geq 1$ is the "SNR gap" that effectively estimates the gap (in terms of signal to noise ratio) between subcarrier capacity and actual rate conveyable (bits/symbol). It depends on the desired target error probability P_E, coding gain γ_C and required margin γ_M [18], [19]: $\Gamma = \frac{1}{3}\left[Q^{-1}(\frac{P_E}{4K})\right]^2 \frac{\gamma_M}{\gamma_C}$, where $Q^{-1}(x)$ is the inverse tail probability of the standard normal distribution and K is an edge-effect correction factor fast approaching unity for medium and large-size QAM constellations. This SNR gap has been used in many bit loading algorithms to calculate the number of bits allocated on a subcarrier [11], [12], [15], [16], [20], [21]. Interested readers can find more details about this "SNR gap" in [18], [22].

Let us denote by

$$\begin{aligned} & b_{max}(n) = \lfloor \log_2\left(1 + \frac{g_n P_n^{max}}{\Gamma}\right) \rfloor, \\ & b_{max}^r(n) = \min(A_{max}, b_{max}(n)), \\ & P_{max}^r(n) = \frac{(2^{b_{max}^r(n)} - 1)\Gamma}{g_n} \leq P_n^{max} \end{aligned} \tag{2}$$

(where $\lfloor \ \rfloor$ is the *floor* function) the maximal number of bits limited by the peak-power constraint on subcarrier n, the maximal effective number of bits and the maximal effective power allocated on subcarrier n.

Problem (1) can be rewritten as

$$\begin{aligned} \text{maximize} \quad & \sum_n b_n \\ \text{s.t.} \quad & b_n \leq c_n = \log_2\left(1 + \frac{g_n P_n}{\Gamma}\right), \quad b_n \in \mathbb{N} \\ & \sum_n P_n \leq P_{tot} \\ & P_n \leq P_{max}^r(n) \end{aligned} \tag{3}$$

B. Existing algorithms in the literature

1) Conventional Greedy or Greedy-based bit-adding with zero bit loading initialization (Z-GBA) algorithm: A well-known optimal solution of problem (1) is obtained from the Z-GBA algorithm. The bit vector is initialized to the null vector. At every iteration, a subcarrier with minimum required incremental power is allocated an additional bit if the power constraints remain fulfilled. In [15], it is proved that this algorithm yields the global optimum solution of (1). Its performance in terms of computation cost depends on the total allowable power: it requires a higher computation cost when the total allowable power is high and vice versa.

2) Greedy-based bit-removing with maximum allowable bit loading initialization (M-GBR) algorithm: This algorithm has been used in [11] to solve the problem (1). The initial number of bits allocated on all subcarriers is set to their maximal allowable number of bits, i.e., $\mathbf{b}_{max}^r$. Then, at every iteration, one bit is removed on a given subcarrier if its power gain is the maximum one. The iterative procedure is stopped when the total power constraint is fulfilled. Its performance in terms of computation cost depends on the total allowable power: it is lower when the total allowable power is high.

3) Sub-optimal algorithm (BFB): This algorithm was proposed by E. Baccarelli, A. Fasano and M. Biagi in [16]. Hence, in the remainder of the paper, we will refer to it as BFB algorithm. Its principle consists in two steps. Firstly, it solves the associated continuous optimization problem (3) as follows

$$\begin{aligned} \text{maximize} \quad & \sum_n c_n \\ \text{s.t.} \quad & \sum_n P_n \leq P_{tot} \\ & P_n \leq P_{max}^r(n) \end{aligned} \tag{4}$$

The solution of (4) is found by solving the following equation

$$f(S_1) = \sum_n \left[S_1 - \frac{\Gamma}{g_n}\right]_0^{P_{max}^r(n)} - P_{tot} = 0 \tag{5}$$

with

$$[x]_0^p = \begin{cases} p & x \geq p \\ x & 0 < x < p \\ 0 & x \leq 0 \end{cases} \tag{6}$$

In [16], two methods have been proposed to solve (5). First, an "Iterative Water-filling" (IWF) is proposed to find the exact solution with a complexity that is in the order of the square of the number of subcarriers. Second, a secant-based loading algorithm is used to find a reliable root of (5)

with a complexity that grows only linearly with the number of subcarriers.

The capacity corresponding to subcarrier n is

$$c_n = \log_2\left(1 + \left[S_1 - \frac{\Gamma}{g_n}\right]_0^{P^r_{max}(n)}\right) \tag{7}$$

In the second step, the bi-section method is applied to find the maximum value of α defined hereinafter in (8) so that the total power constraint is fulfilled. The number of bits allocated to subcarrier n is given by

$$b_n = [\lfloor c_n + \alpha \rfloor]_0^{b^r_{max}(n)} \tag{8}$$

III. Hybrid approach between the Z-GBA and M-GBR algorithms

In Section II, we have reminded that the complexity of the Z-GBA algorithm and of the M-GBR algorithm are a non-decreasing function and a non-increasing function of the total allowable power, respectively. Thus, in the region of high values of P_{tot}, the M-GBR algorithm should be used instead of the Z-GBA algorithm and vice versa. In this section, we propose a simple threshold to switch between the two algorithms. In fact, the complexity of both algorithms is dominated by a product of the number of iterations to obtain the optimum bit and power allocation vectors ($\mathbf{b}^{op}$ and $\mathbf{P}^{op}$) and the complexity per iteration. Moreover, the complexity per iteration of both algorithms is almost the same, i.e., we find the subcarrier that requires minimum power to add one bit or find the subcarrier for which the power gain when removing one bit is maximal and then adjust the number of bits on this subcarrier. Let us denote by ℓ_{BA} and ℓ_{BR} the number of iterations to obtain $\mathbf{b}^{op}$ in the Z-GBA and the M-GBR algorithms and N denotes the number of used subcarriers. In Table I, the total number of operations for the Z-GBA and for the M-GBR algorithms are provided. They are dominated by $\ell_{BA}N$ and $\ell_{BR}N$. Note that in these algorithms, only one bit is added or removed at each iteration. Then we have $\ell_{BA} = \|\mathbf{b}^{op}\|_1$, $\ell_{BR} = \|\mathbf{b}^r_{max}\|_1 - \|\mathbf{b}^{op}\|_1$.

Let us define ΔP_{BR} and ΔP_{BA} by

$$\Delta P_{BR} = \|\mathbf{P}^r_{max}\|_1 - \|\mathbf{P}^{op}\|_1, \tag{9}$$

$$\Delta P_{BA} = \|\mathbf{P}^{op}\|_1. \tag{10}$$

The proposed criterion relies on the following theorems.

Theorem 1. *There exist $\nu > \mu > 1$ so that if $\frac{\Delta P_{BR}}{\Delta P_{BA}} \leq \mu$ then $\ell_{BR} < \ell_{BA}$ and if $\frac{\Delta P_{BR}}{\Delta P_{BA}} \geq \nu$ then $\ell_{BR} > \ell_{BA}$.*

Proof: See Appendix A. ■

This theorem gives us a way to determine the regions where the M-GBR or the Z-GBA algorithm should be preferred. However, due to the ignorance of $\mathbf{P}^{op}$, we cannot calculate ΔP_{BR} and ΔP_{BA} directly.

Theorem 2. *If $\|\mathbf{P}^r_{max}\|_1 > P_{tot}$ and $P_{tot} >> P^r_{max}(n)$, $\forall n$, we have $\|\mathbf{P}^{op}\|_1 \approx P_{tot}$.*

Proof: See Appendix B. ■

In practice, the condition of $P_{tot} >> P^r_{max}(n)$, $\forall n$ is generally fulfilled since the number of active subcarriers is high. Then, ΔP_{BR} and ΔP_{BA} can be approximated as

$$\Delta P_{BR} \approx \|\mathbf{P}^r_{max}\|_1 - P_{tot}\ ; \Delta P_{BA} \approx P_{tot}. \tag{11}$$

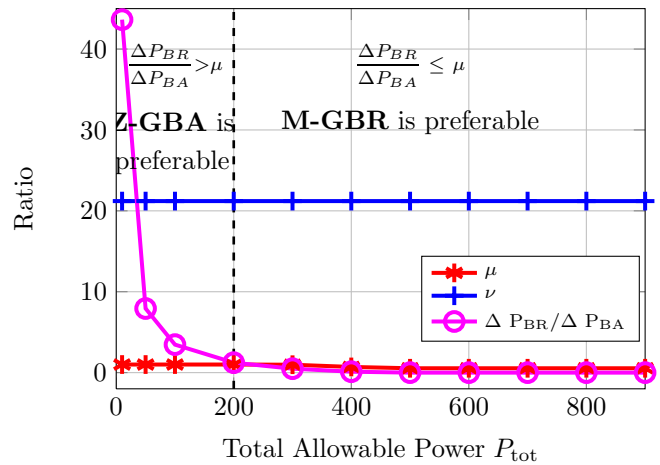

Fig. 1: μ, ν values and switch between Z-GBA and M-GBR algorithms.

An illustration of the switch between both algorithms is illustrated in Fig. 1. We checked that the values of μ and ν vary little with respect to P_{tot} for a given channel realization. However, we also checked that the value of ν highly depends on the channel realization. In contrast, μ can be considered constant ($\mu \approx 1$) w.r.t. channel realization. To reduce the complexity, we only take into account the criterion $\frac{\Delta P_{BR}}{\Delta P_{BA}} \leq \mu$, where $\mu = 1$, to determine the switch between the Z-GBA and the M-GBR algorithms. We refer to it as hybrid algorithm. This approach is based on the criterion $\frac{\Delta P_{BR}}{\Delta P_{BA}} \leq \mu$. If the criterion is fulfilled, the M-GBR algorithm is used since $\ell_{BR} < \ell_{BA}$. Otherwise, the Z-GBA algorithm is used. Obviously, this simple criterion can only yield a sub-optimal switch between the two algorithms as we will see in the simulation section.

IV. A new low-complexity loading algorithm: Theoretical analysis and implementation

A. Theoretical analysis

The WF algorithm provides an optimal loading solution to problem (4), denoted by $\{c_n^{WF}\}$. We define the rounding of the WF solution as $b_n^{WFR} = \text{round}(c_n^{WF})$, where

$$\text{round}(x) = n \quad \Leftrightarrow \quad -\frac{1}{2} \leq x - n < \frac{1}{2},\ n \in \mathbb{N} \tag{12}$$

We denote by $\mathbf{b}^{WFR}$ the bit vector resulting from the rounding of the WF solution and by $\mathbf{P}^{WFR}$ the corresponding power allocation calculated by

$$P_n^{WFR} = \frac{(2^{b_n^{WFR}} - 1)\Gamma}{g_n} \tag{13}$$

In [23], Campello has defined an efficient bit vector for the bit/power loading. A bit vector is said efficient if there is no pair of subcarriers so that the power gain obtained by removing one bit from one subcarrier can be used to add one bit to another subcarrier. We recall that in the optimum Z-GBA algorithm, at every step in the bit-adding procedure, the bit vector is always efficient [15].

Theorem 3. *$\boldsymbol{b}^{WFR}$ is an efficient bit vector.*

Proof: See Appendix C. ■

In the following, we prove that the power use corresponding to $\mathbf{b}^{WFR}$ is the most efficient.

Theorem 4. *For any bit vector $\boldsymbol{b} \neq \boldsymbol{b}^{WFR}$ if $\|\boldsymbol{b}\|_1 = \|\boldsymbol{b}^{WFR}\|_1$, then $\|\boldsymbol{P}^{WFR}\|_1 < \|\boldsymbol{P}\|_1$ with $\boldsymbol{P}$ the power vector associated with $\boldsymbol{b}$ and $\|.\|_1$ the l_1 vector norm.*

Proof: See Appendix D. ■

B. Proposed WFR-GBL algorithm

Based on the analysis above, we propose the following simple and optimal algorithm for the discrete bit-loading problem in (1). Its principle consists in exploiting the bit vector obtained by rounding the water-filling algorithm solution of problem (4) as an efficient initial bit vector of the Greedy algorithm. Then, we calculate the current power use $P_{use} = \sum_n P_n^{WFR}$. If $P_{use} \leq P_{tot}$, the Greedy-based bit-adding is used. Else, the Greedy-based bit-removing is applied to yield the final bit allocation from $\mathbf{b}^{WFR}$. We named it the Water-filling rounding Greedy-based bit loading (WFR-GBL) algorithm.

Pseudocode of the WFR-GBL algorithm

1: Calculate $\mathbf{b}_{max}$, $\mathbf{b}^r_{max}$, $\mathbf{P}^r_{max}$ as in (2).
2: Calculate $P^r_{tot} = \sum_n P^r_{max}(n)$.
3: **if** $P^r_{tot} \leq P_{tot}$ **then**
4: $\quad b(n) = b^r_{max}(n)$
5: $\quad P(n) = P^r_{max}(n)$
6: **else**
7: $\quad$ Solve Eq. (5) to find S_1 by using Iterative Water-Filling or secant-based loading algorithm in [16].
8: $\quad$ Calculate $b_n^{WFR} = \text{round}\Big(\Big[\big(\log_2(g_n) + S_2\big)\Big]_0^{b^r_{max}(n)}\Big)$, $P_n^{WFR} = \dfrac{(2^{b_n^{WFR}} - 1)\Gamma}{g_n}$ and $P_{use} = \sum_n P_n^{WFR}$.
9: $\quad$ **if** $P_{use} \leq P_{tot}$ **then**
10: $\quad\quad$ Use Greedy-based bit-adding as in [15] to add the number of bits on the subcarriers that have not reached yet their effective maximal number of bits $\mathbf{b}^r_{max}$.
11: $\quad$ **else**
12: $\quad\quad$ Use Greedy-based bit-removing as in [16] to remove the number of bits on the subcarriers that have not reached yet 0.
13: $\quad$ **end if**
14: **end if**

To prove that the WFR-GBL algorithm converges to the global optimum solution, we first derived the following result.

Theorem 5. *Let $\boldsymbol{b}^e$ and $\boldsymbol{b}^f$ be two efficient bit vectors and $\boldsymbol{P}^e$ and $\boldsymbol{P}^f$ be the corresponding power allocation vectors, respectively. Then,*

$$\|\boldsymbol{P}^e\|_1 \geq \|\boldsymbol{P}^f\|_1 \Leftrightarrow \boldsymbol{b}^e \geq \boldsymbol{b}^f \text{ (component-wise)} \tag{14}$$

Proof: See Appendix E. ■

Pseudocode of the modified secant-based algorithm

1: Set ϵ to a desired tolerance value.
2: Set $count = 0$, $S_1^{old} = 0$.
3: Set $x_0 = \min_n \dfrac{1}{g_n}$, $x_1 = \max_n\{P^r_{max}(n) + \dfrac{1}{g_n}\}$.
4: Set $f_0 = -P_{tot}$, $x_1 = \sum_n P^r_{max}(n) - P_{tot}$.
5: **while** $count < 5$ **do**
6: $\quad S_1 = \dfrac{\frac{1}{2}f_1x_1 - f_0x_0}{\frac{1}{2}f_1 - f_0}$
7: $\quad f = \sum_n \Big[S_1 - \dfrac{\Gamma}{g_n}\Big]_0^{P^r_{max}(n)} - P_{tot}$
8: $\quad$ **if** $f > 0$ **then**
9: $\quad\quad x_1 = S_1$
10: $\quad\quad f_1 = f$
11: $\quad$ **else**
12: $\quad\quad x_0 = S_1$
13: $\quad\quad f_0 = f$
14: $\quad$ **end if**
15: $\quad$ **if** $\frac{S_1 - S_1^{old}}{S_1} < \epsilon$ **then**
16: $\quad\quad count = count + 1$
17: $\quad$ **end if**
18: $\quad S_1^{old} = S_1$
19: **end while**
20: $P_n = \Big[S_1 - \dfrac{\Gamma}{g_n}\Big]_0^{P^r_{max}(n)}$

Theorem 6. *The WFR-GBL algorithm converges to the globally optimal bit allocation.*

Proof: See Appendix F. ■

Note that in Step 7 of the pseudocode implementation of the WFR-GBL algorithm, the secant-based loading should be used to find S_1 for two reasons. First, note that the "efficiency" of $\mathbf{b}^{WFR}$ is independent of S_1, so that we can use an approximate version of the root of Eq. (5) instead of the exact one. Second, its complexity grows only linearly with the number of subcarriers. However, its main drawback is that in many cases, the same end-point is retained twice in a row. To avoid it, we use a modified secant-based (also called Illinois) algorithm [24] whose pseudocode implementation is given above. In the following, starting from $\mathbf{b}^{WFR}$, we prove that the optimized bit-loading is achieved by adding or removing the number of bits per subcarrier at most once.

Theorem 7. *To achieve the optimized throughput from $\boldsymbol{b}^{WFR}$, the number of bits on every subcarrier needs to be increased or removed at most once.*

Proof: See Appendix G. ■

C. Complexity analysis

Let us denote by L_s the number of iterations to find S_1 in the secant-based loading procedure and L_r the number of iterations to find α in the BFB algorithm. In [15], it is shown that L_s and L_r are independent of N. To find the optimum (i.e., maximum or minimum) of an array of size N, it requires about N operations. The calculations of $\mathbf{b}_{max}$ (and of $\mathbf{b}^{WFR}$),

TABLE I: Number of operations for the algorithms.

Z-GBA	$(7+\ell_{BA})N+3\ell_{BA}$
M-GBR	$(11+\ell_{BR})N+3\ell_{BR}$
BFB	$(2L_s+7L_r+17)N$
WFR-GBL	$(2L_s+\ell_{WFR}+22)N+3\ell_{WFR}$

$\mathbf{b}^r_{max}$ from $\mathbf{b}_{max}$ (and of P_{use}) and $\mathbf{P}^r_{max}$ from $\mathbf{b}^r_{max}$ (and of $\mathbf{P}^{WFR}$) require $5N$, N and $4N$ operations.

Let us denote by ℓ_{WFR} the number of iterations to obtain $\mathbf{b}^{op}$ in the WFR-GBL algorithm. Note that in WFR-GBL algorithm, only one bit is added or removed at each iteration. Then we have

$$\ell_{WFR} = \left| \|\mathbf{b}^{op}\|_1 - \|\mathbf{b}^{WFR}\|_1 \right| \tag{15}$$

In the following, we summarize the main steps for each algorithm.

- Z-GBA algorithm: calculate $\mathbf{b}_{max}$, $\mathbf{b}^r_{max}$ and required powers to add one bit to zero for every subcarrier + Greedy-based iteration to add the number of bits on subcarriers. In [15], it is shown that every iteration requires $N+3$ operations.
- M-GBR algorithm: calculate $\mathbf{b}_{max}$, $\mathbf{b}^r_{max}$, $\mathbf{P}^r_{max}$ and required power to remove one bit from $b^r_{max}(n)$ for every subcarrier + Greedy-based iteration to remove the number of bits on subcarriers.
- BFB algorithm: calculate $\mathbf{b}_{max}$, $\mathbf{b}^r_{max}$, $\mathbf{P}^r_{max}$ + Secant-based algorithm (Illinois algorithm) + 'alpha' compensation procedure. In [15], the number of iterations of the secant-based loading procedure to find S_1 and the bi-section method to find α is about $(2L_s+1)N$ and $(7L_r+6)N$, respectively.
- WFR-GBL algorithm: calculate $\mathbf{b}_{max}$, $\mathbf{b}^r_{max}$, $\mathbf{P}^r_{max}$ + Secant-based algorithm (Illinois algorithm) + calculate $\mathbf{b}^{WFR}$, $\mathbf{P}^{WFR}$, P_{use} and the power required (power gain) for every subcarrier to add (remove) one bit from b^{WFR}_n + either Greedy-based bit-removing or bit-adding.

The total number of operations required for each algorithm is summarized in Table I.

V. Simulation Results

To validate the proposed algorithm, we use the multi-path PLC channels whose transfer function can be modeled as [25]:

$$H(f) = A\sum_{n=1}^{N_p}(w_n + z_n f^{K_2})e^{-(a_0+a_1 f^{K_1})l_n}e^{-j2\pi f l_n/\nu_c} \tag{16}$$

where ν_c is the speed of electromagnetic waves in the copper medium, N_p is the number of propagation paths, and l_n is the length of the n-th path. Parameters A and w_n relate to the path amplitude, while parameters a_0, a_1, K_1, K_2 and z_n govern the frequency dependence of the channel transfer function. The values of the parameters for each of the nine classes can be found in [25].

For the noise, we only take into account the background noise, which can be modeled as a colored Gaussian noise with power spectral density [26] defined as

$$A_N(f) = b_0 + b_1|f|^{b_2}. \tag{17}$$

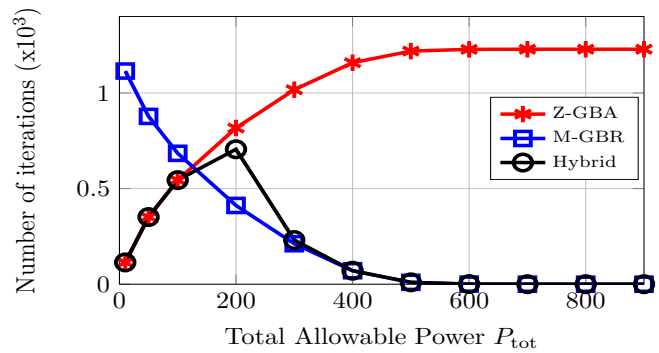

Fig. 2: Number of iterations per subcarrier comparison.

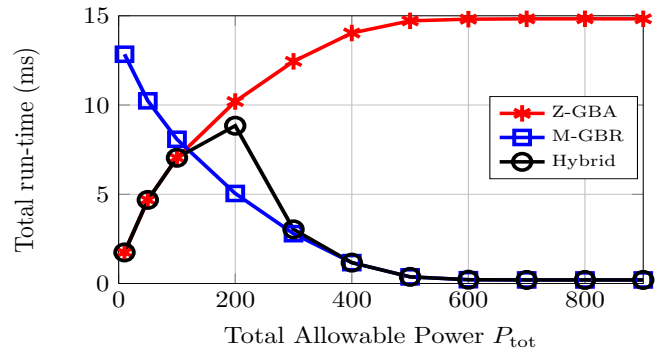

Fig. 3: Total run-time comparison.

We also suppose that there is zero-Doppler and the guard interval is chosen so that there is neither inter-carrier interference nor inter-symbol interference. Simulation results are obtained with the following parameters:

- Number of used subcarriers $N = 917$.
- Allowable set of number of bits on a subcarrier $\mathcal{A} = \{0, 1, 2, 3, 4, 5..., 11, 12\}$.
- $P^{max}_n = 1$, $\forall n \in \{1, 2, ..., N\}$ (normalized to $P_0\Delta f$ where $P_0 = -55$ dBm (1 Hz) is the spectral mask value defined by the IEEE P1901 standard and Δf is the subcarrier spacing between two consecutive subcarriers).
- P_{tot} (normalized to $P_0\Delta f$) varies from 10 to 900.
- $\Gamma = 7$, for a target SER value of 10^{-5} [27].
- $\epsilon = 0.01$ and $L_r = 10$.
- Number of channel realizations: 1000.

A. Hybrid approach between the Z-GBA and M-GBR algorithms

The number of iterations and the run-time of the Z-GBA, M-GBR and hybrid algorithms are illustrated in Figs. 2 and 3 (with PLC class 2 channels). We can observe that the simple criterion exploited in the hybrid algorithm gives us a judicious switch between the two algorithms. Its performance is the same as the M-GBR algorithm in the region of high P_{tot} and is the same as the Z-GBA algorithm in the region of small P_{tot}.

B. WFR-GBL algorithm performance

We test the following algorithms: the WFR-GBL, the Z-GBA, the M-GBR and the BFB algorithms.

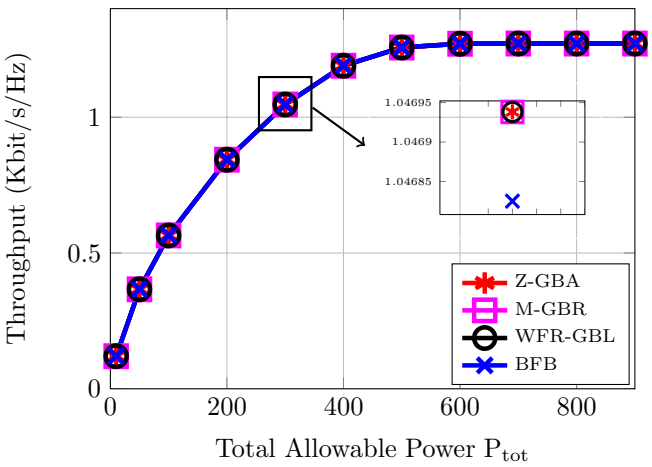

Fig. 4: Achieved throughput comparison.

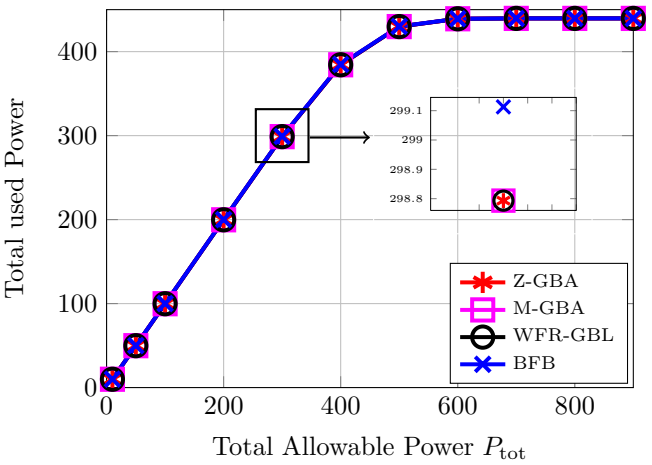

Fig. 5: Total used power comparison.

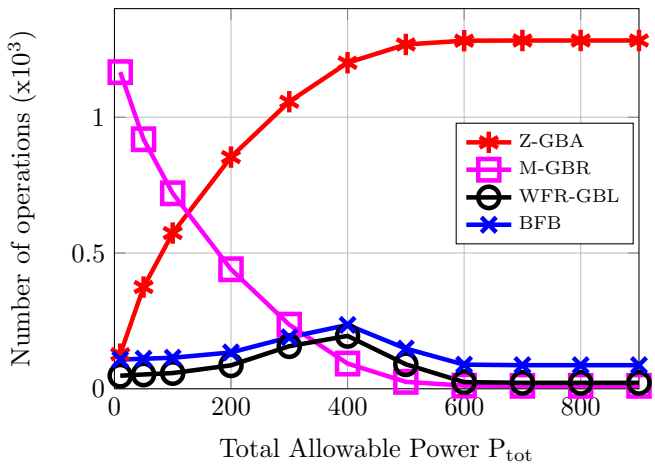

Fig. 6: Number of operations per subcarrier comparison.

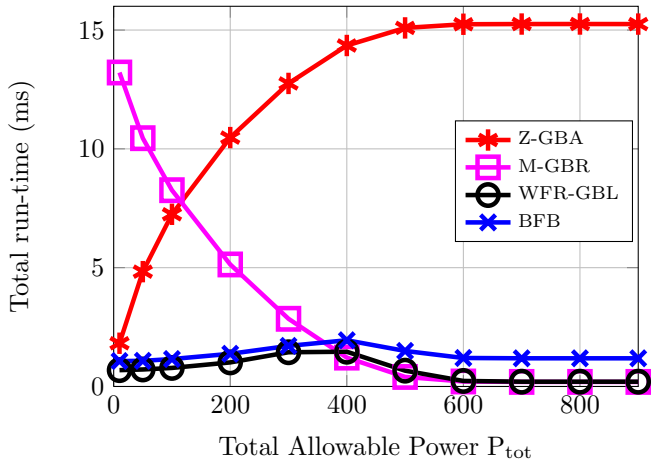

Fig. 7: Total run-time comparison.

Figs. 4 and 5 show the achieved throughput and the total power use obtained with the four algorithms. We can check that the throughput achieved with the WFR-GBL algorithm is the same as the optimal one (obtained with the Z-GBA and the M-GBR algorithms). In addition, we have also checked that both bit/power allocations obtained by the Z-GBA, the WFR-GBL and the M-GBR algorithms are always the same. This confirms the optimality of the WFR-GBL algorithm. The throughput achieved with the BFB algorithm is slightly degraded but it seems to converge to the optimal one as L_r increases.

The total required number of operations per subcarrier and the total run-time comparisons are illustrated in Figs. 6 and 7. We can see that both performance indicators of the Z-GBA algorithm and the M-GBR algorithm are non-decreasing and non-increasing function of P_{tot}, respectively. Moreover, the total number of operations as well as the run-time of the BFB algorithm and the WFR-GBL algorithm little vary w.r.t. P_{tot}. It is also shown that the complexity of the WFR-GBL algorithm is less than the one of the BFB algorithm and strongly reduced as compared to the Z-GBA algorithm or the M-GBR algorithm. The number of operations per subcarrier, calculated by dividing the total number of iterations given in Table I by N, and the total run-time are shown in Table II. Both of them are results averaged over the range of P_{tot}.

TABLE II: Average number of operations per subcarrier and average total run-time comparisons between the algorithms over various $P_{tot} \in [10, 900]$.

	Number of operations	Total run-time (ms)	Total run-time (relative)
Z-GBA	962.25	11.60	16.8
M-GBR	331.37	3.85	5.6
BFB	126.35	1.32	1.9
WFR-GBL	70.76	0.69	1

We have also tested the WFR-GBL algorithm for all 9 classes of PLC channels. We recall that every class has a particular average channel attenuation [25]. In all cases, we have checked that the WFR-GBL algorithm can always achieve the same bit/power allocation as the ones obtained by the Z-GBA and M-GBR algorithms, i.e., the optimum one. The relative total run-time (in average over P_{tot}) w.r.t. the channel class is shown in Fig. 8. We can observe that the relative run-time of the Z-GBA algorithm as well as the run-time of the M-GBR algorithm is an increasing function of the channel class. This can be explained by the fact that the higher the channel class, the less variable and attenuated the channel

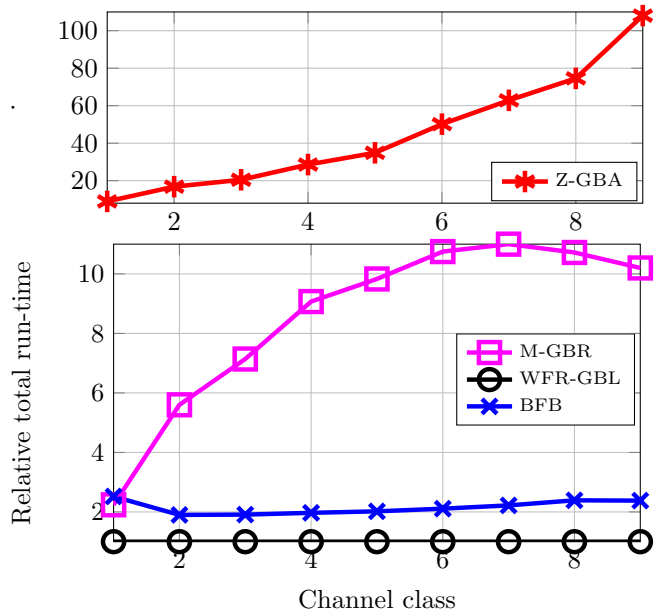

Fig. 8: Relative total run-time (averaged over various $P_{tot} \in [10, 900]$) vs channel class.

frequency response [25]. Thus, $\|\mathbf{b}^{op}\|_1$ and $\|\mathbf{b}^r_{max}\|_1$ increase with the index of the channel class. This explains an increasing behavior of the relative run-time of the Z-GBA algorithm as well as the M-GBR algorithm. However, for channel classes 7, 8 and 9, we observe a decrease of the relative run-time of the M-GBR algorithm. This is explained by the fact that for these classes of channel, $\|\mathbf{b}^{op}\|_1$ increases with the index of the channel class while most entries of $\mathbf{b}^r_{max}$ are equal to A_{max} and thus do not depend on the channel frequency response. Hence, the number of iterations used in the M-GBR algorithm, that is equal to $\|\mathbf{b}^r_{max}\|_1$ - $\|\mathbf{b}^{op}\|_1$, is a decreasing function of the channel class (for classes 7, 8 and 9).

The WFR-GBL algorithm strongly reduces the run-time as compared to the Z-GBA or the M-GBR algorithms. In addition, its run-time is about half of the BFB algorithm run-time. Note that while the WFR-GBL algorithm yields the global optimum solution for problem (1), the BFB algorithm is only sub-optimal.

Figs. 9 and 10 illustrate the total number of operations and the run-time of the algorithms when we change the number of subcarrier, i.e., N = 256, 512 and 917, with P_{tot} fixed to 100 (normalized to $P_0\Delta f$). We observe that the WFR-GBL algorithm always outperforms the Z-GBA, the M-GBR and the BFB algorithms. In addition, the higher the number of active subcarriers, the higher the complexity reduction.

VI. Conclusion

In this paper, we have firstly introduced a simple criterion to switch between two well-known algorithms to solve the discrete bit-loading in interference-free OFDM systems: the conventional Greedy (Z-GBA) and the Greedy-based bit-removing with maximum allowable bit loading initialization (M-GBR) algorithms. Secondly, we have proposed a novel low-complexity optimal WFR-GBL algorithm. Its optimality

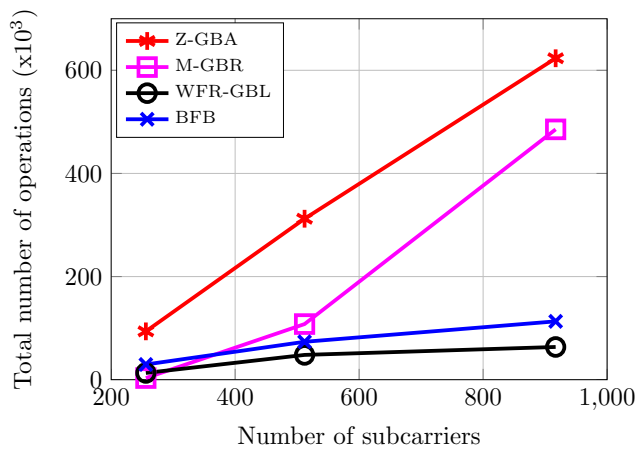

Fig. 9: Total number of operations vs number of subcarriers.

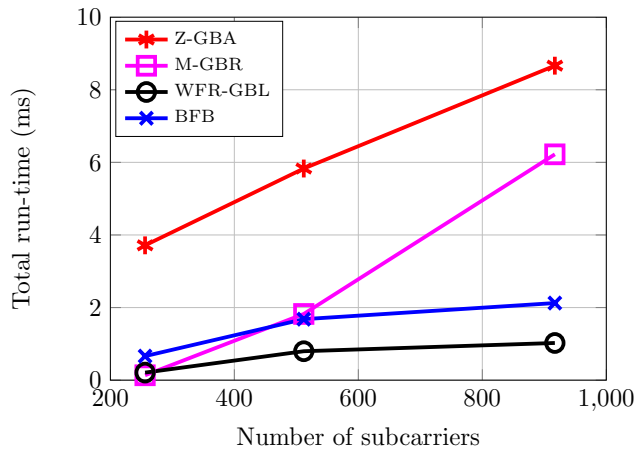

Fig. 10: Total run-time vs number of subcarriers.

has been theoretically proved. Its principle consists in first exploiting the bit vector obtained by rounding the Water-filling solution to the associated continuous bit allocation problem as an initial bit-vector in the Greedy algorithm and secondly to load up or to remove bits on the subcarriers to be loaded up or be removed at most once. We have compared the proposed WFR-GBL algorithm with the Z-GBA, the M-GBR and the BFB algorithms. The advantage in terms of computation cost has been theoretically analyzed. Simulation results have shown the efficiency of the proposed WFR-GBL algorithm in terms of achieved throughput and run-time with different configurations, such as different numbers of subcarriers, different PLC channel classes and different total power constraints. In all cases, the proposed algorithm outperforms the reference algorithms.

Appendix A
Proof of Theorem 1

Let us denote by ε^f_{BR} and ε^l_{BR} the power obtained when removing one bit in the first and last iterations of the M-GBR algorithm to obtain $\mathbf{b}^{op}$ from $\mathbf{b}^r_{max}$; ε^f_{BA} and ε^l_{BA} the power required to load up one bit in the first and last iterations of the Z-GBA algorithm to obtain $\mathbf{b}^{op}$ from a null bit vector.

Note that at every iteration in the M-GBR algorithm, one bit is removed so that the power gain is maximum and in the Z-GBA algorithm, one bit is loaded up so that the power required is minimum. Thus, $\varepsilon^f_{BR} > \varepsilon^l_{BR}$, $\varepsilon^f_{BA} < \varepsilon^l_{BA}$ and we have

$$\begin{cases} \ell_{BR}\ \varepsilon^l_{BR} < \Delta P_{BR} < \ell_{BR}\ \varepsilon^f_{BR} \\ \ell_{BA}\ \varepsilon^f_{BA} < \Delta P_{BA} < \ell_{BA}\ \varepsilon^l_{BA} \end{cases} \tag{18}$$

$$\Rightarrow \frac{\ell_{BR}\ \varepsilon^l_{BR}}{\ell_{BA}\ \varepsilon^l_{BA}} < \frac{\Delta P_{BR}}{\Delta P_{BA}} < \frac{\ell_{BR}\ \varepsilon^f_{BR}}{\ell_{BA}\ \varepsilon^f_{BA}} \tag{19}$$

Let us note $\nu = \frac{\varepsilon^f_{BR}}{\varepsilon^f_{BA}}$ and $\mu = \frac{\varepsilon^l_{BR}}{\varepsilon^l_{BA}}$. Clearly, $\nu > \mu$ and from (19),

$$\frac{\Delta P_{BR}}{\Delta P_{BA}}\frac{1}{\nu} < \frac{\ell_{BR}}{\ell_{BA}} < \frac{\Delta P_{BR}}{\Delta P_{BA}}\frac{1}{\mu} \tag{20}$$

Moreover, ε^l_{BR} and ε^l_{BA} are also the power required to load up one bit and the power obtained when removing one bit from $\mathbf{b}^{op}$, respectively. Because $\mathbf{b}^{op}$ is an efficient bit vector then $\varepsilon^l_{BR} > \varepsilon^l_{BA}$ thus $\mu > 1$ (see Appendix C).

From (20), if $\Delta P_{BR}/\Delta P_{BA} \geq \nu$, then $\ell_{BR}/\ell_{BA} > 1$ and the number of iterations used in the M-GBR algorithm is higher than that used in the Z-GBA algorithm and the Z-GBA algorithm should be chosen. On the other side, if $\Delta P_{BR}/\Delta P_{BA} \leq \mu$, then $\ell_{BR}/\ell_{BA} < 1$ and the M-GBR algorithm should be chosen.

Appendix B
Proof of Theorem 2

When $\|\mathbf{P}^r_{max}\|_1 > P_{tot}$, there is always an allowable subcarrier that cannot be allocated one additional bit due to the total power constraint at the end of the Z-GBA algorithm. Let us denote by n_0 such an allowable subcarrier. Thus, $\Delta P_{n_0} \uparrow = \frac{2^{b^{op}_{n_0}}\Gamma}{g_{n_0}}$ is the required power to add one bit on this subcarrier ($b^{op}_{n_0} \to b^{op}_{n_0} + 1$). Then, we have

$$\|\mathbf{P}^{op}\|_1 \leq P_{tot} < \|\mathbf{P}^{op}\|_1 + \Delta P_{n_0} \uparrow \tag{21}$$

$$\Rightarrow 1 - \frac{\Delta P_{n_0} \uparrow}{P_{tot}} < \frac{\|\mathbf{P}^{op}\|_1}{P_{tot}} \leq 1 \tag{22}$$

Because n_0 is an allowable subcarrier, we have

$$b^r_{max}(n_0) = \log_2\left(1 + \frac{P^r_{max}(n_0) g_{n_0}}{\Gamma}\right) \geq 1 \tag{23}$$

$$\Rightarrow \frac{\Gamma}{g_{n_0}} \leq P^r_{max}(n_0) \tag{24}$$

and $b^{op}(n_0) < b^r_{max}(n_0)$

$$\Rightarrow P^{op}(n_0) = \frac{(2^{b^{op}(n_0)-1})\Gamma}{g_{n_0}} < P^r_{max}(n_0) \tag{25}$$

Let us assume that $P^r_{max}(n) << P_{tot}$, $\forall n$. Let us consider two cases: $b^{op}_{n_0} \neq 0$ and $b^{op}_{n_0} = 0$. In the first case, we have:

$$\Delta P_{n_0} \uparrow = \frac{(2^{b^{op}_{n_0}} - 1)\Gamma}{g_{n_0}} + \frac{\Gamma}{g_{n_0}} < 2P^r_{max}(n_0) << P_{tot} \tag{26}$$

In the second case, we obtain

$$\Delta P_{n_0} \uparrow = \frac{\Gamma}{g_{n_0}} \leq P^r_{max}(n_0) << P_{tot} \tag{27}$$

In both cases, $\Delta P_{n_0} \uparrow << P_{tot}$.

Let us denote $\epsilon = \frac{\Delta P_{n_0} \uparrow}{P_{tot}} << 1$. Using (22), we have

$$1 - \epsilon < \frac{\|\mathbf{P}^{op}\|_1}{P_{tot}} \leq 1 \Rightarrow \frac{\|\mathbf{P}^{op}\|_1}{P_{tot}} \approx 1 \tag{28}$$

Appendix C
Proof of Theorem 3

We prove that $\mathbf{b}^{WFR} = \text{round}(\mathbf{c}^{WF})$, is an efficient bit vector. The allocated power and continuous capacity for subcarrier n after using WF algorithm for problem (4) are given by [16]

$$\begin{cases} P^{WF}_n &= \left[\left(S_1 - \frac{\Gamma}{g_n}\right)\right]_0^{P^r_{max}(n)} \\ c^{WF}_n &= \log_2\left(1 + \frac{P_n g_n}{\Gamma}\right) = \left[\left(\log_2(g_n) + S_2\right)\right]_0^{b^r_{max}(n)} \end{cases} \tag{29}$$

where S_1 is the root of (5) and $S_2 = \log_2(S_1/\Gamma)$.

The number of bits after the rounding and their corresponding power are

$$\begin{cases} b^{WFR}_n &= \text{round}(c^{WF}_n) \\ P^{WFR}_n &= \frac{(2^{b^{WFR}_n} - 1)\Gamma}{g_n} \end{cases} \tag{30}$$

Let us denote by b^{WFR}_1, b^{WFR}_2 the number of bits obtained by WF + rounding on any pair of distinct subcarriers. Without loss of generality, we suppose that $\Delta b = b^{WFR}_1 - b^{WFR}_2 \geq 0$. Then,

Case 1: $0 < b^{WFR}_k \leq b^r_{max}(k) - 1$, $k = 1, 2 \Rightarrow c^{WF}_k < b^r_{max}(k)$. By using $m = \text{round}(x) \Leftrightarrow m - 1/2 \leq x < m + 1/2$, then

$$\Rightarrow b^{WFR}_1 - 1/2 - (b^{WFR}_2 + 1/2) < c^{WF}_1 - c^{WF}_2 < b^{WFR}_1 + 1/2 - (b^{WFR}_2 - 1/2) \tag{31}$$

$$\Rightarrow \Delta b - 1 < \log_2(g_1) + S_2 - (\log_2(g_2) + S_2) < \Delta b + 1 \tag{32}$$

$$\Rightarrow 2^{\Delta b - 1} < \frac{g_1}{g_2} < 2^{\Delta b + 1} \tag{33}$$

- The required power to add one bit to subcarrier 1 and the power gain by removing one bit from subcarrier 2 are $\Delta P^{WFR}_1 \uparrow = \frac{2^{b^{WFR}_1}\Gamma}{g_1}$ and $\Delta P^{WFR}_2 \downarrow = \frac{2^{b^{WFR}_2 - 1}\Gamma}{g_2}$. Using (33), $\frac{g_1}{g_2} < 2^{\Delta b + 1} \Rightarrow \Delta P^{WFR}_1 \uparrow > \Delta P^{WFR}_2 \downarrow$.
- By following the same reasoning and using $\frac{g_1}{g_2} > 2^{\Delta b - 1}$ given by (33), we also have $\Delta P^{WFR}_2 \uparrow > \Delta P^{WFR}_1 \downarrow$.

Case 2: $b^{WFR}_1 = b^r_{max}(1)$, $0 < b^{WFR}_2 < b^r_{max}(2)$. In this case $\log_2(g_1) + S_2 \geq b^r_{max}(1) - 1/2$, then

$$\log_2(g_1) + S_2 - (\log_2(g_2) + S_2) > b^r_{max}(1) - b^{WFR}_2 - 1 \tag{34}$$

$$\Rightarrow \log_2(g_1) - \log_2(g_2) > \Delta b - 1 \Rightarrow \frac{g_1}{g_2} > 2^{\Delta b - 1} \tag{35}$$

Then, $\Delta P^{WFR}_2 \uparrow > \Delta P^{WFR}_1 \downarrow$.

Case 3: $0 < b^{WFR}_1 \leq b^r_{max}(1)$, $b^{WFR}_2 = 0$. In this case, $\log_2(g_2) + S_2 < 1/2$.

$$\Rightarrow \quad \log_2(g_1) + S_2 - \log_2(g_2) - S_2 > b^{WFR}_1 - b^{WFR}_2 - 1 \tag{36}$$

$$\Rightarrow \quad \log_2(g_1) - \log_2(g_2) > \Delta b - 1 \tag{37}$$

$$\Rightarrow \quad \frac{g_1}{g_2} > 2^{\Delta b - 1} \tag{38}$$

Then, as in case 2, $\Rightarrow \Delta P^{WFR}_2 \uparrow > \Delta P^{WFR}_1 \downarrow$.

In all cases, there is no movement of a bit from one subcarrier to another that reduces the total required power. Thus, $\mathbf{b}^{WFR}$ is an efficient bit vector.

Appendix D
Proof of Theorem 4

Let us denote by $\Delta P_n^{WFR}\uparrow$ and $\Delta P_n^{WFR}\downarrow$ the required power to add one bit to subcarrier n and the power gain by removing one bit from subcarrier n. Then,

$$\Delta P_n^{WFR}\uparrow=\frac{(2^{b_n^{WFR}})\Gamma}{g_n};\quad \Delta P_n^{WFR}\downarrow=\frac{(2^{b_n^{WFR}-1})\Gamma}{g_n} \tag{39}$$

Let I stand for the subset of subcarriers such that $b_n^{WFR} < b_n$ and J the subset of subcarriers such that $b_m^{WFR} > b_m$. As $\|\mathbf{b}^{WFR}\|_1 = \|\mathbf{b}\|_1$, we have $\sum_{n\in I}(b_n - b_n^{WFR}) = \sum_{m\in J}(b_m^{WFR} - b_m)$.

Let us consider $\mathbf{b}^{WFR}$. We denote by $\Delta P\uparrow$ the total increase of power required to load up each subcarrier n of I such that it bears b_n bits and $\Delta P\downarrow$ the total power gain from removing bits from each subcarrier m of J such that it bears b_m bits. The required total power associated with this new bit loading is equal to

$$\|\mathbf{P}\|_1 = \|\mathbf{P}^{WFR}\|_1 + \Delta P\uparrow - \Delta P\downarrow \tag{40}$$

$$\begin{aligned}\Delta P\uparrow &= \sum_{n\in I}(2^{b_n} - 2^{b_n^{WFR}})\frac{\Gamma}{g_n} = \sum_{n\in I}(2^{b_n - b_n^{WFR}} - 1)\ \Delta P_n^{WFR}\uparrow \\ &\geq \sum_{n\in I}(b_n - b_n^{WFR})\ \Delta P_n^{WFR}\uparrow > \varepsilon_{min}^I \sum_{n\in I}(b_n - b_n^{WFR})\end{aligned} \tag{41}$$

where $\varepsilon_{min}^I = \min_{n\in I}\ \Delta P_n^{WFR}\uparrow$ and since $2^q - 1 \geq q, \forall q \in \mathbb{N}^+$.

$$\begin{aligned}\Delta P\downarrow &= \sum_{m\in J}(2^{b_m^{WFR}} - 2^{b_m})\frac{\Gamma}{g_m} \\ &= \sum_{m\in J} 2(1 - 2^{b_m - b_m^{WFR}})\ \Delta P_m^{WFR}\downarrow \\ &\leq \sum_{m\in J}(b_m^{WFR} - b_m)\ \Delta P_m^{WFR}\downarrow < \varepsilon_{max}^J \sum_{m\in J}(b_m^{WFR} - b_m)\end{aligned} \tag{42}$$

where $\varepsilon_{max}^J = \max_{m\in J}\ \Delta P_m^{WFR}\downarrow$ and $1 - 2^{-q} \leq q/2, \forall q \in \mathbb{N}^+$.

In the proof of the efficiency of $\mathbf{b}^{WFR}$, we have proved that for any pair of subcarriers i and j, $\Delta P_i^{WFR}\uparrow > \Delta P_j^{WFR}\downarrow$. Thus, $\varepsilon_{min}^I > \varepsilon_{max}^J$ and then $\Delta P\uparrow > \Delta P\downarrow$. Finally,

$$\|\mathbf{P}\|_1 = \|\mathbf{P}^{WFR}\|_1 + \Delta P\uparrow - \Delta P\downarrow > \|\mathbf{P}^{WFR}\|_1 \tag{43}$$

The equality $\|\mathbf{P}\|_1 = \|\mathbf{P}^{WFR}\|_1$ holds only when $\mathbf{b} = \mathbf{b}^{WFR}$.

Appendix E
Proof of Theorem 5

The sufficient condition, i.e., $\mathbf{b}^e \geq \mathbf{b}^f \Rightarrow \|\mathbf{P}^e\|_1 \geq \|\mathbf{P}^f\|_1$, is easily demonstrated by the fact that the power allocated on a subcarrier is an increasing function of allocated number of bits, i.e., $\mathbf{b}^e \geq \mathbf{b}^f$ (component-wise) $\Rightarrow \mathbf{P}^e \geq \mathbf{P}^f$ (component-wise) and thus $\|\mathbf{P}^e\|_1 \geq \|\mathbf{P}^f\|_1$.

To demonstrate the necessary condition, i.e., $\mathbf{b}^e$ and $\mathbf{b}^f$ are two efficient bit vectors, if $\|\mathbf{P}^e\|_1 \geq \|\mathbf{P}^f\|_1$ then $\mathbf{b}^e \geq \mathbf{b}^f$, we use the counter-evidence approach.

Suppose that $\|\mathbf{P}^e\|_1 \geq \|\mathbf{P}^f\|_1$ and $\exists n_0 :\ b_{n_0}^e < b_{n_0}^f$. If $\forall n \neq n_0$, $b_n^e \leq b_n^f$, then $\|\mathbf{P}^e\|_1 < \|\mathbf{P}^f\|_1$, which contradicts the hypothesis. Thus, $\exists n_1$: $b_{n_1}^e > b_{n_1}^f$. Then, we have

$$b_{n_0}^e < b_{n_0}^f \Rightarrow b_{n_0}^e \leq b_{n_0}^f - 1 \tag{44}$$

$$\Rightarrow \Delta P^{\uparrow}_{b_{n_0}^e \to b_{n_0}^e+1} = \frac{2^{b_{n_0}^e}\Gamma}{g_{n_0}} \leq \frac{2^{b_{n_0}^f - 1}\Gamma}{g_{n_0}} = \Delta P^{\downarrow}_{b_{n_0}^f \to b_{n_0}^f - 1} \tag{45}$$

and

$$b_{n_1}^e > b_{n_1}^f \Rightarrow b_{n_1}^e \geq b_{n_1}^f + 1 \tag{46}$$

$$\Rightarrow \Delta P^{\downarrow}_{b_{n_1}^e \to b_{n_1}^e - 1} = \frac{2^{b_{n_1}^e - 1}\Gamma}{g_{n_1}} \geq \frac{2^{b_{n_1}^f}\Gamma}{g_{n_1}} = \Delta P^{\uparrow}_{b_{n_1}^f \to b_{n_1}^f + 1} \tag{47}$$

Since $\mathbf{b}^f$ is an efficient bit vector, $\Delta P^{\uparrow}_{b_{n_1}^f+1 \to b_{n_1}^f} > \Delta P^{\downarrow}_{b_{n_0}^f \to b_{n_0}^f - 1}$. Then using (45) and (47), we deduced that $\Delta P^{\uparrow}_{b_{n_0}^e \to b_{n_0}^e+1} < \Delta P^{\downarrow}_{b_{n_1}^e \to b_{n_1}^e - 1}$ which contradicts the hypothesis "$\mathbf{b}^e$ is an efficient bit vector".

Thus, $b_n^e \geq b_n^f$, $\forall n$, i.e., $\mathbf{b}^e \geq \mathbf{b}^f$ (component-wise).

Appendix F
Proof of Theorem 6

We consider two cases: a) $P_{tot}^r \leq P_{tot}$ and b) $P_{tot}^r > P_{tot}$.

In the first case, the total power constraint is always fulfilled. Thus, $P_n = P_{max}^r(n)$ and $b_n = b_{max}^r(n)$ is the globally optimal allocation.

If $P_{tot}^r > P_{tot}$, we have to consider two cases: i) $\|\mathbf{P}^{WFR}\|_1 \leq P_{tot}$ and ii) $\|\mathbf{P}^{WFR}\|_1 > P_{tot}$.

In the case i), the Greedy-based bit-adding algorithm is applied and its solution $\mathbf{b}^{op}$ satisfies $\mathbf{0} < \mathbf{b}^{WFR} \leq \mathbf{b}^{op}$. In the following, we prove that $\mathbf{b}^{WFR}$ must be an intermediate bit vector in the Z-GBA algorithm and thus $\mathbf{b}^{op}$ is the globally optimal solution.

Let us denote by $\mathbf{b}_{BA}^I$ the intermediate bit vector obtained in the Z-GBA algorithm such that $\|\mathbf{b}_{BA}^I\|_1 = \|\mathbf{b}^{WFR}\|_1$. Let $\mathbf{P}_{BA}^I$ stand for the corresponding power allocation. According to [15], the bit vector at any step always yields the minimum total power use. So, $\|\mathbf{P}_{BA}^I\|_1 \leq \|\mathbf{P}_{WFR}\|_1$. On the other hand, by applying Theorem 2, we have $\|\mathbf{P}_{BA}^I\|_1 \geq \|\mathbf{P}_{WFR}\|_1$. We thus deduce that $\|\mathbf{P}_{BA}^I\|_1 = \|\mathbf{P}_{WFR}\|_1$ and $\mathbf{b}_{BA}^I = \mathbf{b}^{WFR}$.

Consequently, the WFR-GBL algorithm in case i) converges to the same solution as the Z-GBA algorithm, i.e., to the global optimum one.

In the case ii), $\|\mathbf{P}_{max}^r\|_1 \geq \|\mathbf{P}^{WFR}\|_1 \geq P_{tot} \geq \|\mathbf{P}^{op}\|_1$ and according to Theorem 6, we have $\mathbf{b}_{max}^r \geq \mathbf{b}^{WFR} \geq \mathbf{b}^{op}$. Note that the M-GBR algorithm yields at every step an efficient bit vector. Using the same reasoning as before, we deduce that $\mathbf{b}^{WFR}$ is an intermediate bit vector of the M-GBR algorithm.

Consequently, the WFR-GBL algorithm in case ii) conveys to the same solution as the M-GBR algorithm, i.e., to the global optimum one.

We have proved that in all cases, the WFR-GBL algorithm yields the global optimum solution for problem (1).

APPENDIX G
PROOF OF THEOREM 7

In this appendix, we aim to prove that, after the initialization step, a given subcarrier state (allocated bit number and power) is modified at most once. To this end, we consider two cases: a) $P^r_{tot} \leq P_{tot}$ and b) $P^r_{tot} > P_{tot}$.

In the first case, $P^r_{tot} \leq P_{tot}$ and the algorithm consists of a single iteration. All subcarriers are allocated their maximum number of bits with maximum power level, i.e., subcarrier n will be allocated $b^r_{max}(n)$ bits and a power equal to $P^r_{max}(n)$. This allocation obviously achieves the global optimum from the initialization state.

In the second case, $P^r_{tot} > P_{tot}$ and we have to distinguish between the bit-adding and the bit-removing procedures depending on whether $P_{use} = \|\mathbf{P}^{WFR}\|_1$ after initialization be less or greater than P_{tot}. Our reasoning relies on two lemmas introduced hereinafter.

The first lemma states that the subcarriers selected for update within first successive iterations of the algorithm WFR-GBL are necessarily different. Then, in the second lemma, we prove that the number of bits added or removed by the WFR-GBL algorithm is upper-bounded by Card($\mathcal{U}$), where $\mathcal{U}$ denotes the set of subcarriers that can be added (case $\|\mathbf{P}^{WFR}\|_1 \leq P_{tot}$) or removed (case $\|\mathbf{P}^{WFR}\|_1 > P_{tot}$) at least one bit in the Greedy-based procedure and Card($\mathcal{U}$) is the cardinality of $\mathcal{U}$.

In the following, we denote by j the current iteration index and by i_j the index of the subcarrier selected for update.

Lemma 7.1. *Given $k \leq$ Card($\mathcal{U}$), k successive iterations of the WFR-GBL algorithm necessarily update k different subcarriers, i.e., $i_j \neq i_\ell$ for all $1 \leq \ell < j \leq k$.*

Proof: We prove the lemma only for the case $\|\mathbf{P}^{WFR}\|_1 \leq P_{tot}$ as the same reasoning holds for the another case. Then, the algorithm applies the bit-adding procedure with the initialization state $(\mathbf{b}^{WFR}, \mathbf{P}^{WFR})$.

We first prove that $i_2 \neq i_1$. The required powers to add one bit on both subcarriers at second iteration are:

$$\Delta P^{\uparrow}_{b_{i_1} \to b_{i_1}+1} = \frac{2^{b_{i_1}}\Gamma}{g_{i_1}} \tag{48}$$

$$\Delta P^{\uparrow}_{b_{i_2} \to b_{i_2}+1} = \frac{2^{b_{i_2}}\Gamma}{g_{i_2}} \tag{49}$$

where $b_{i_1} = b^{WFR}_{i_1} + 1$ and $b_{i_2} = b^{WFR}_{i_2}$.

Since $\mathbf{b}^{WFR}$ is an efficient bit vector, we have $\frac{2^{b^{WFR}_{i_1}}\Gamma}{g_{i_1}} > \frac{2^{b^{WFR}_{i_2}-1}\Gamma}{g_{i_2}}$, which implies that

$$\Delta P^{\uparrow}_{b_{i_1} \to b_{i_1}+1} > \Delta P^{\uparrow}_{b_{i_2} \to b_{i_2}+1} \tag{50}$$

This means that i_2 is necessarily different from i_1. We can easily generalize the result and deduce that $i_k \neq i_\ell$ for $1 \leq \ell < k \leq$ Card($\mathcal{U}$), which states the first lemma. An equivalent reasoning can be applied to prove the lemma when $\|\mathbf{P}^{WFR}\|_1 > P_{tot}$.

■

Lemma 7.2. $\ell_{WFR} = \left| \|\boldsymbol{b}^{op}\|_1 - \|\boldsymbol{b}^{WFR}\|_1 \right| \leq$ *Card*($\mathcal{U}$).

Proof: Let us assume that $\|\mathbf{P}^{WFR}\|_1 \leq P_{tot}$. Then, the final bits vector $\mathbf{b}^{op}$ is componentwise higher than the initial one $\mathbf{b}^{WFR}$.

The Greedy-based bit-adding procedure is applied and $\mathcal{U} = \{n | c^{WF}_n < b^r_{max}(n)\}$. Let us denote by $\mathcal{U}^C$ the complementary set of $\mathcal{U}$. Note that $c^{WF}_n \leq b^r_{max}(n)$, $\forall n$. Thus $\mathcal{U}^C = \{n | c^{WF}_n = b^{WFR}_n = b^r_{max}(n)\}$. Obviously, the optimized throughput is always less than the sum capacity obtained by the Water-filling. Then, we have

$$b^{WFR}_n = \text{round}(c^{WF}_n) > c^{WF}_n - 1/2, \ \forall n \tag{51}$$

$$\sum_n b^{op}_n < \sum_n c^{WF}_n \tag{52}$$

Thus,

$$\ell_{WFR} = \sum_n (b^{op}_n - b^{WFR}_n) \leq \sum_n \left(c^{WF}_n - b^{WFR}_n\right) \tag{53}$$

$$= \sum_{n \in \mathcal{U}} \underbrace{\left(c^{WF}_n - b^{WFR}_n\right)}_{<1/2} + \underbrace{\sum_{n \in \mathcal{U}^C} \left(c^{WF}_n - b^{WFR}_n\right)}_{0} \tag{54}$$

$$< \frac{\text{Card}(\mathcal{U})}{2} < \text{Card}(\mathcal{U}) \tag{55}$$

Let us now assume that $\|\mathbf{P}^{WFR}\|_1 > P_{tot}$. Then, the final bits vector $\mathbf{b}^{op}$ is componentwise less than the initial one $\mathbf{b}^{WFR}$: $\mathbf{b}^{op} \leq \mathbf{b}^{WFR} \leq \mathbf{b}^r_{max}$. The Greedy-based bit-removing procedure is applied, $\mathcal{U} = \{n | b^{WFR}_n \leq 1\}$ and $\mathcal{U}^C = \{n | b^{WFR}_n = b^{op}_n = 0\}$. Let us define $b^{WFD}_n = \lfloor c^{WF}_n \rfloor$, we have

$$b^{WFR}_n = \text{round}(c^{WF}_n) \leq b^{WFD}_n + 1, \ \forall n \tag{56}$$

$$b^{op}_n \geq b^{WFD}_n, \ \forall n \tag{57}$$

The second inequality results from the fact that $\mathbf{b}^{WFD}$ is also an efficient bit vector and the total power use corresponding to $\mathbf{b}^{WFD}$ is less than P_{tot} (and thus less than or equal to $\|\mathbf{P}^{op}\|_1$ since $\|\mathbf{P}^{op}\|_1 \approx P_{tot}$). Using Theorem 5, we obtain $\mathbf{b}^{op} \geq \mathbf{b}^{WFD}$. The proof of the efficiency of $\mathbf{b}^{WFD}$ relies on an equivalent reasoning as used in Appendix C and uses $\lfloor x \rfloor = m \Rightarrow m \leq x < m+1$.

Then, we have

$$\ell_{WFR} = \sum_n (b^{WFR}_n - b^{op}_n) \tag{58}$$

$$= \sum_{n \in \mathcal{U}} \left(b^{WFR}_n - b^{op}_n\right) + \underbrace{\sum_{n \in \mathcal{U}^C} \left(b^{WFR}_n - b^{op}_n\right)}_{0} \tag{59}$$

$$= \sum_{n \in \mathcal{U}} \left(b^{WFR}_n - b^{op}_n\right) \leq \sum_{n \in \mathcal{U}} \left(b^{WFD}_n + 1 - b^{WFD}_n\right) \tag{60}$$

$$\leq \text{Card}(\mathcal{U}) \tag{61}$$

Finally, we have proved for both cases,

$$\ell_{WFR} = \left| \|\mathbf{b}^{op}\|_1 - \|\mathbf{b}^{WFR}\|_1 \right| \leq \text{Card}(\mathcal{U}). \tag{62}$$

■

On one hand, Lemma 7.1. states that the first k iterations, where k is upper-bounded by the cardinality of $\mathcal{U}$, update k different subcarriers. On the other hand, Lemma 7.2. states that the variation of the number of bits between initialization and optimum convergence states is upper-bounded by the cardinality of $\mathcal{U}$. From both lemmas, we deduce that to obtain $\mathbf{b}^{op}$, starting from $\mathbf{b}^{WFR}$, the number of bits allocated to a given subcarrier will be increased or decreased by at most one bit.

REFERENCES

[1] R. Prasad, *OFDM for wireless communication*. Norwood, MA: Artech House Publisher, 2004.

[2] A. Ghosh, J. Zhang, G. Andrews, and R. Muhamed, *Fundamental of LTE*. Prentice Hall, 2010.

[3] L. Hanzo, W. Webb, and T. Keller, *Single and multi-carrier quadrature amplitude modulation-principles and application for personal communications, WLANs and broadcasting*. New York: Wiley, 2000.

[4] IEEE Std 1901-2012, "IEEE Standard for Broadband over Power Line Networks: Medium Access Control and Physical Layer Specification."

[5] ANSI Std. T1.413-1998, "Asymmetric digital subscriber line (ADSL) metallic interface."

[6] Stanford University, "Multi-channel modulation," http://www.stanford.edu/class/ee379c/readerfiles/chap4.pdf, [Online], pp. 311-322.

[7] A. Leke and J. M. Cioffi, "A maximum rate loading algorithm for discrete multitone modulation systems," *Proc. IEEE Globecom*, vol. 3, pp. 1514–1518, 1997.

[8] J. Campello, "Practical bit loading for DMT," *Proc. IEEE ICC*, vol. 2, pp. 801–805, 1999.

[9] B. S. Krongold, K. Ramchandran, and D. L. Jones, "Computationally efficient optimal power allocation for multicarrier communication systems." *IEEE Trans. Commun.*, vol. 48(1), pp. 23–27, 2000.

[10] R. V. Sonalkar and R. R. Shively, "Efficient bit-loading algorithm for DMT applications." *IEEE Commun. Lett.*, vol. 4(3), pp. 80–82, 2000.

[11] N. Papandreou and T. Antonakopoulos, "Bit and power allocation in constrained multicarrier systems: The single-user case," *EURASIP Journal on Advances in Signal Processing*, 2008.

[12] P. S. Chow, J. M. Cioffi, and J. A. C. Bingham, "A practical discrete multitone transceiver loading algorithm for data transmission over spectrally shaped channels," *IEEE Trans. Commun*, vol. 43(2), pp. 773–775, 1995.

[13] A. Czylwik, "Adaptive OFDM for wideband radio channels," *Proc. IEEE Globecom*, pp. 713–718, 1996.

[14] R. F. H. Fischer and J. B. Huber, "A new loading algorithm for discrete multi-tone transmission," *Proc. IEEE Globecom*, pp. 724–728, 1996.

[15] E. Baccarelli and M. Biagi, "Optimal integer bit-loading for multicarrier ADSL systems subject to spectral-compatibility limits," *Signal Processing (EURASIP)*, vol. 84(4), pp. 729–741, 2004.

[16] E. Baccarelli, A. Fasano, and M. Biagi, "Novel efficient bit-loading algorithms for peak-energy-limited ADSL-Type multicarrier systems," *IEEE Trans. Signal Process.*, vol. 50(5), pp. 1237–1247, 2002.

[17] T. N. Vo, K. Amis, T. Chonavel, and P. Siohan, "Achievable Throughput Optimization in OFDM Systems in the Presence of Interference and its Application to Power Line Networks," *IEEE Trans. Commun*, vol. 62(5), pp. 1704–1715, 2014.

[18] T. Starr, J. Cioffi, and P. Silvermann, *Understanding Digital Subscriber Line Technology*. New York: Prentice-Hall, 1999.

[19] J. Bingham, *ADSL, VDSL and Multicarrier Modulation*. New York: Wiley, 2002.

[20] W. Rhee and J. M. Cioffi, "Increase in capacity of multiuser OFDM system using dynamic subchannel allocation," *Proc. IEEE VTC*, vol. 2, pp. 1085–1089, 2000.

[21] A. G. Armada, "A simple multiuser bit loading algorithm for multicarrier WLAN," *Proc. IEEE ICC*, vol. 4, pp. 1168–1171, 2001.

[22] I. Kalet, "The multitone channel," *IEEE Trans. Comm.*, vol. 37, pp. 119–124, 1998.

[23] J. Campello, "Optimal discrete bit loading for multicarrier modulation systems," *Proc. IEEE ISIT*, p. 193, 1998.

[24] M. Dowell and P. Jarratt, "A modified regula falsi method for computing the root of an equation," *BIT Numerical Mathematics*, vol. 11, pp. 168–174, 1971.

[25] M. Tonello, F. Versolatto, B. Bejar, and S. Zazo, "A fitting algorithm for random modeling the PLC channel." *IEEE Trans. Pow. Del.*, vol. 27(3), pp. 1477–1484, 2012.

[26] R. Hashmat, P. Pagani, and T. Chonavel, "Analysis and modeling of background noise for inhome MIMO-PLC channels," *Proc. IEEE ISPLC 2012*, pp. 316–321, 2012.

[27] J. M. Cioffi, G. D. Dudevoir, M. V. Eyubouglu, and G. D. Forney, "MMSE decision-feedback equalizers and coding - part II: Coding results," *IEEE Trans. Commun*, vol. 43(10), pp. 2595–2604, 1995.