

# A Multilayer Perceptron Artificial Neural Network combining the best features of the traditional IFDMA and LFDMA schemes

Mohammed Abdased<sup>1</sup>, Jabril Ramdan<sup>2</sup>

<sup>1</sup>Faculty of Engineering Gharian University, Libya

<sup>2</sup>Faculty of Science and natural resources, Aljafara University, Almamora, Libya.

Received 30 October 2023; revised 14 November 2023; accepted 14 November 2023

## ABSTRACT

The main objective this study is to propose development of a new subcarrier-mapping scheme using GIFDMA for the LTE/LTE-A system. This new scheme addresses the user and frequency diversity by allocating the subcarrier and combining the best features of the traditional IFDMA and LFDMA schemes. The Interleaved FDMA (IFDMA) is a special case of GIFDMA that uses non-contiguous subcarrier in contrast to LFDMA that utilizes adjacent subcarrier. Both LFDMA and IFDMA subcarrier mappings significantly reduce PAPR at the expense of high time complexity, in order to reduce the time complexity and heuristic correlation between interleave level and roll-off factors for fast PAPR estimates, a Multilayer Perceptron Artificial Neural Network (MLP-ANN) and Random Forest (RF) algorithms are introduced. RF is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

Keywords: GIFDMA, LFDMA, IFDMA, LTE / LTE-A, PAPR, MLP-ANN, RF

## المخلص

يعالج هذا LTE / LTE-A لنظام GIFDMA الهدف الرئيسي من هذه الدراسة هو اقتراح تطوير مخطط جديد للموجات الحاملة الفرعية باستخدام التقليدية LFDMA و IFDMA المخطط الجديد المستخدم وتنوع التردد من خلال تخصيص الموجة الحاملة الفرعية والجمع بين أفضل ميزات الذي يستخدم موجة حاملة فرعية مجاورة LFDMA التي تستخدم حاملة فرعية غير متصلة على عكس GIFDMA هي حالة خاصة من (IFDMA) على حساب التعقيد الزمني العالي، من أجل تقليل التعقيد الزمني PAPR تقلل بشكل كبير IFDMA و LFDMA تقلل كل من تعيينات الناقل الفرعي (Perceptron (MLP-ANN) السريعة، شبكة متعددة الطبقات PAPR والارتباط التجريبي بين مستوى التشخير وعوامل الانقلاب لتقديرات هي خوارزمية تستخدم على نطاق واسع في مشاكل التصنيف والانحدار. يبني القرار على عينات مختلفة Random Forest (RF). RF ويأخذ تصويت الأغلبية للتصنيف والمتوسط في حالة الانحدار.

## 1. Introduction

This study presents the development of a new subcarrier-mapping scheme using GIFDMA for the LTE/LTE-A system. This new scheme addresses the user and frequency diversity by allocating the subcarrier and combining the best features of the traditional IFDMA and LFDMA schemes. The methodology starts with the formulation of the mathematical model by introducing an interleaving level comprising the IFDMA and LFDMA features (Pandey et al 2015) [8], (Rahman, M.

M. & Manir, S. B. 2012) [9]. Moreover, the study emphasizes on the general methodology used in this research, and provides the details of the proposed GIFDMA scheme, including a new allocation scheme mathematical model, partition and assignment procedure. Furthermore, it presents the details of the GIFDMA simulation model and validation, including the impact of the interleaving level on the signal retrieval and iteration between all symbols, extends this research using A multilayer perceptron artificial neural network (MLP-ANN) to train the input data for minimum MLP error.

## 2. Conceptual Model

In this study, the interleave level played a vital role as it helped in controlling the mapping schemes. The parameter was noticed to be essential as it assisted in the seamless transition of the IFDMA to the LFDMA scheme. In LFDMA, every user symbol represents the allocated neighbouring frequencies present in the bandwidth which led to low-frequency diversity. However, in IFDMA the user symbols were spread over the whole bandwidth, thus it provided an advantage of the frequency diversity over the LFDMA. In the proposed GIFDMA scheme, the interleave level controlled and partitioned the user symbols in different blocks within the whole bandwidth. Figures 1, 2 and 3 describe the mapping scheme allocations for the IFDMA, LFDMA, and GIFDMA schemes, respectively.

USERS											
$U_3$	$U_1$	$U_2$	$U_3$	$U_1$	$U_2$	$U_3$	$U_1$	$U_2$	$U_3$		
3	4	5	6	7	8	9	10	11	12		First group, $G_1$
$U_3$	$U_1$	$U_2$	$U_3$	$U_1$	$U_2$	$U_3$	$U_1$	$U_2$	$U_3$		
15	16	17	18	19	20	21	22	23	24		Second group, $G_2$
$U_3$	$U_1$	$U_2$	$U_3$	$U_1$	$U_2$	$U_3$	$U_1$	$U_2$	$U_3$		
27	28	29	30	31	32	33	34	35	36		Third group, $G_3$
$U_3$	$U_1$	$U_2$	$U_3$	$U_1$	$U_2$	$U_3$	$U_1$	$U_2$	$U_3$		
39	40	41	42	43	44	45	46	47	48		Fourth group, $G_4$

Fig. 1 :IFDMA Mapping Scheme Allocations

USERS													
$U_1$	$U_1$	$U_1$	$U_1$	$U_2$	$U_2$	$U_2$	$U_2$	$U_3$	$U_3$	$U_3$	$U_3$		
1	2	3	4	5	6	7	8	9	10	11	12		$G_1$
$U_1$	$U_1$	$U_1$	$U_1$	$U_2$	$U_2$	$U_2$	$U_2$	$U_3$	$U_3$	$U_3$	$U_3$		
13	14	15	16	17	18	19	20	21	22	23	24		$G_2$
$U_1$	$U_1$	$U_1$	$U_1$	$U_2$	$U_2$	$U_2$	$U_2$	$U_3$	$U_3$	$U_3$	$U_3$		
25	26	27	28	29	30	31	32	33	34	35	36		$G_3$
$U_1$	$U_1$	$U_1$	$U_1$	$U_2$	$U_2$			$U_3$	$U_3$	$U_3$	$U_3$		
				$U_2$		$U_2$							
37	38	39	40	41	42	43	44	45	46	47	48		$G_4$

Fig. 2: LFDMA Mapping Scheme Allocations

USERS													
$U_1$	$U_1$	$U_2$	$U_2$	$U_3$	$U_3$	$U_1$	$U_1$	$U_2$	$U_2$	$U_3$	$U_3$		
1	2	3	4	5	6	7	8	9	10	11	12		$G_1$
$U_1$	$U_1$	$U_2$	$U_2$	$U_3$	$U_3$	$U_1$	$U_1$	$U_2$	$U_2$	$U_3$	$U_3$		
13	14	15	16	17	18	19	20	21	22	23	24		$G_2$
$U_1$	$U_1$	$U_2$	$U_2$	$U_3$	$U_3$	$U_1$	$U_1$	$U_2$	$U_2$	$U_3$	$U_3$		
25	26	27	28	29	30	31	32	33	34	35	36		$G_3$
$U_1$	$U_1$	$U_2$	$U_2$	$U_3$	$U_3$	$U_1$	$U_1$	$U_2$	$U_2$	$U_3$	$U_3$		
37	38	39	40	41	42	43	44	45	46	47	48		$G_4$

Fig. 3: GIFDMA Mapping Scheme Allocations

For IFDMA as shown in the Figure 1, the bandwidth is divided into four groups ( $G_1$  to  $G_4$ ) with 48 subcarriers (1 to 48) allocations. Every group having 12 subcarrier assignments ( $G_1 = \{1$  to

12},  $G_2 = \{13 \text{ to } 24\}$ ,  $G_3 = \{25 \text{ to } 36\}$ , and  $G_4 = \{37 \text{ to } 48\}$ ) and three users ( $U_1$ ,  $U_2$ , and  $U_3$ ), hence every user has four subcarriers allocation (MUG) in each group (as an example each user is represented by a particular colour). Assume that the transmitter acknowledges using the IFDMA scheme, then at the receiver, the first user would utilise the allocation subcarriers  $M_{11} = \{1, 4, 7, 10\}$  for the first terminal ( $U_1$ ) in the first group, while the second group would take allocations  $M_{12} = \{13, 16, 19, 22\}$ . The third group would exploit allocations  $M_{13} = \{25, 28, 31, 34\}$  whereas the fourth group would use  $M_{14} = \{37, 40, 43, 46\}$  as shown in Figure 3 The distribution and subcarrier allocation for the second and third users ( $U_2$  and  $U_3$ ) would follow similar pattern as the first user.

For LFDMA, as shown in Figure 2 the subcarriers distribution are arranged as four contiguous subcarrier allocation for each user in each group. As an example, in the first group, the first user will take assignments  $M_{11} = \{1, 2, 3, 4\}$ , the second user would take  $M_{21} = \{5, 6, 7, 8\}$  and the third user will use  $M_{31} = \{9, 10, 11, 12\}$ . The allocation distributions for the other three groups follow the same approach.

For GIFDMA as shown in Figure 3, the allocation subcarriers in the bandwidth would distribute each two subsequent assignments for each user in all the groups. As an example for the first user, the subcarrier assignment will be as follows:  $M_{11} = \{1, 2, 7, 8\}$ ,  $M_{12} = \{13, 14, 19, 20\}$ ,  $M_{13} = \{25, 26, 31, 32\}$ , and  $M_{14} = \{37, 38, 43, 44\}$  for the first until fourth group respectively. For the other users, the distribution can be distributed following the same manner.

The above example considered the bandwidth,  $N = 12$  and the number of subcarrier,  $M = 4$ , hence the spreading factor,  $S = N/M = 3$ , and so forth the number of users. The recovered signal was calculated for the various subcarrier mapping techniques. The schemes distribute outputs over the entire band of  $N$  subcarriers, with zeros filled (or assigned with bit '0') in unused subcarriers to overcome bit overlapping.

### 3. Mathematical Formulation

The mathematical representation for the IFDMA and LFDMA, and the derivation of GIFDMA schemes are as follow.

#### 3.1 IFDMA

Let  $S$  be the number of users,  $N$  is the total number of subcarriers,  $M$  is the total number of subcarriers per user, then for  $n$ -th user, in the  $g$ -th group, the subcarrier allocation can be

represented by Equation (1) and (2) (Cho et al. 2010)[2], it can be derived by substitute the parameters  $IL=1$ , and  $m_1=m=0,1,\dots,M-1$  into Equation (2) as following:

$$n = (S - 1)m_1 + m \quad (1)$$

$$n = Sm_1 - m_1 + m \quad (2)$$

Therefore, the IFDMA recovered signal as following:

$$n = Sm_1 \quad (3)$$

$n$  is the retrieve signal of the IFFT sequence  $\mathbf{x}[n]$ ,  $S$  is the spreading factor,  $s$  is the indexing of the spreading factor,  $m$  is the subcarrier indexing of  $M$ , and  $m_1$  is the number of blocks. By relating to Equation (3) and assigning the indices as demonstrated in Table 1, the result will be as following:

**Table 1:** Partitions and assignment of IFDMA

	$X_0$	0	0	$X_1$	0	0	$X_2$	0	0	$X_3$	0	0
$m$	0	0	0	1	0	0	2	0	0	3	0	0
$m_1$	0	0	0	1	0	0	2	0	0	3	0	0
$S$	3	0	0	3	0	0	3	0	0	3	0	0
$n$	0	0	0	3	0	0	6	0	0	9	0	0

### 3.2 LFDMA

In converse, for the LFDMA scheme by assuming and applying the indices  $IL=4$ ,  $m_1=0$  and  $m=0, 1, 2$  and  $3$  into Equation (2) yields

$$n = (S - 1)Mm_1 + m + sM \quad (4)$$

$$n = SMm_1 - Mm_1 + m + sM \quad (5)$$

Since  $m_1 = 0$ ;

Herein, the signal of LFDMA is as follows:

$$n = sM + m \quad (6)$$

Since there is no spreading factor in the LFDMA ( $s=0$ ) therefore the distribution of subcarriers are adjacent symbols (Singh & Patidar 2015)[10] and the retrieved signal and subcarrier index are equal. Table 2 describes the partitions and the assignment of the values for the LFDMA scheme after using Equations 6.

**Table 2:** Partitions and assignment of LFDMA

	$X_0$	$X_1$	$X_2$	$X_3$	0	0	0	0	0	0	0	0
$m$	0	1	2	3	0	0	0	0	0	0	0	0
$m_1$	0	0	0	0	0	0	0	0	0	0	0	0
$s$	0	0	0	0	0	0	0	0	0	0	0	0

S	3	3	3	3	0	0	0	0	0	0	0	0
n	0	1	2	3	0	0	0	0	0	0	0	0

### 3.3 GIFDMA

In the GIFDMA structure, the complete domain was sectioned into different blocks with subcarrier length,  $I_L$ . The subcarrier indexing,  $i$ , was controlled by the number of blocks,  $m_1$ , thus by taking into consideration the subcarrier index presents in every block, the recovered signal index,  $n$ , can be expressed as:

$$n = [(S - 1)m_1]I_L + m \tag{7}$$

$$n = (S - 1)I_L m_1 + m \tag{8}$$

The defining  $I_L$  characterises the GIFDMA case in the event of the example described in Table 3 in the following manner the value of the parameters as the following;

$$I_L = 2, S = 3, m_1 = 0, \text{ and } 1 \text{ and } m = 0, 1, \dots, (M - 1)$$

By substituting the parameters in Equation (8) and assigning the indices mentioned above yields, the allocation for every user illustrates the partitions and the assignment of the proposed GIFDMA scheme. The outputs of the input data are apportioned over the entire bandwidth with zeros possessing in unused subcarriers. Table 3 illustrates the partitions and assignment of GIFDMA.

**Table 3:** Partitions and assignment of GIFDMA

	X	X <sub>1</sub>	0	0	0	0	X <sub>2</sub>	X <sub>3</sub>	0	0	0	0
m	0	1	0	0	0	0	2	3	0	0	0	0
m	0	0	0	0	0	0	1	1	0	0	0	0
l												
S	3	3	0	0	0	0	3	3	0	0	0	0
n	0	1	0	0	0	0	6	7	0	0	0	0

The interleave level,  $I_L$ , in the subcarrier allocation was seen to change the sequence of the user symbols for the receiver. Hence, for investigating the effect of  $I_L$  on the retrieved signal using a mathematical model, the  $I_L$  could be defined as below:

$$I_L = 2^x, \tag{9}$$

where  $x = 0, 1, \dots, \log_2 (M)$ .

Hence, the GIFDMA scheme was considered as a general scheme, which contained a variable interleave level. Its changeability led to a seamless transition from the IFDMA to the LFDMA, which was controlled by the  $I_L$ . Thus, it was seen that the IFDMA and the LFDMA were special cases of the GIFDMA scheme. The introduction of the interleave level made the transition from

IFDMA to LFDMA schemes feasible. Wherein, IL refers to the interleave level,  $x$  was the interleave index,  $M$  refers to the number of the subcarrier allocations for every user. Based on the IL defined in Equation (9), it was observed that the scheme allowed for a variable interleaving of the user symbols for the bandwidth. For instance, if  $M = 4$ , the interleave index  $x$  would display the values ranging between  $\{0, 1, 2\}$  and, thus, the value of the IL, to range between 1 and 4 which match the values of IFDMA, GIFDMA and LFDMA schemes, as described in Figure 4.

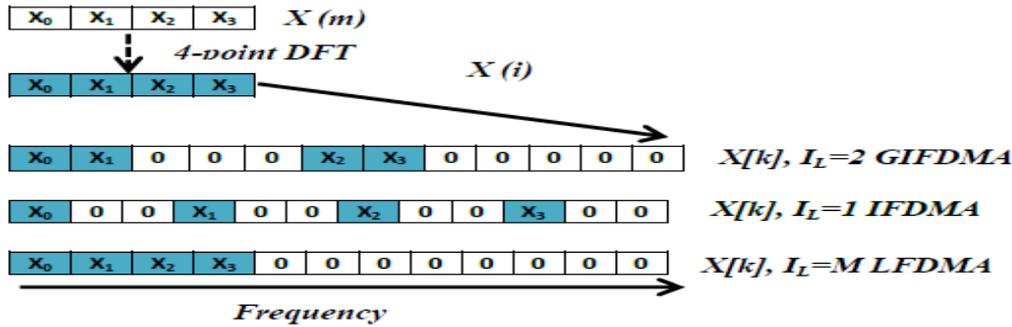


Fig. 4: Partitions and assignment of multiple mobile terminal users

The GIFDMA concept along with its variable allocation level, IL and the impact of changing the allocations for every user was explained using the mathematical derivation of the output signal. The retrieved signal was expressed in the following manner:

$$\hat{x}[k] = \begin{cases} \hat{x}\left[\frac{k-l}{SI_L}\right] & k = m_1 SI_L + l, m_1 = 0, 1, \dots, \frac{M}{I_L} - 1 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

The IFFT output sequence  $\hat{x}[n]$  with

$$\hat{x}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \hat{x}[k] e^{j2\pi \frac{n}{N} k} \quad (11)$$

$$n = M \cdot s + m \quad (12)$$

where  $s$  is the index of the spreading factor,  $M$  is the number of subcarriers, IL is interleaved levels  $s, m, l$  are indices for  $S, M$  and IL

$$s = 0, 1, \dots, S-1, \quad m = 0, 1, \dots, M-1, \quad l = 0, 1, \dots, I_L-1$$

$$\hat{x}[n] = \frac{1}{N} \sum_{m_1=0}^{N-1} \hat{x}[m_1] e^{j2\pi \frac{n}{N} k} \quad (13)$$

The overall system bandwidth is expressed as:

$$N = SM \quad (14)$$

$$\hat{x}[n] = \frac{1}{S} \cdot \frac{1}{M} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi \frac{M \cdot s + m}{SM} k} \quad (15)$$

Substituting the index k in Equation (10) into Equation (15), obtained:

$$\hat{x}[n] = \frac{1}{S} \cdot \frac{1}{M} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi \left( \frac{M \cdot s + m}{SM} \right) (m_1 S I_L + l)} \quad (16)$$

$$\hat{x}[n] = \frac{1}{S} \left( \frac{1}{M} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi s \cdot m_1 I_L + \frac{m}{M} m_1 I_L + \frac{M \cdot s + m}{SM} l} \right) \quad (17)$$

To simplify Equation (17) yields (See Appendix A for the proof)

$$\hat{x}[n] = \frac{1}{S} \left( \frac{1}{M} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi s \cdot m_1 I_L + \frac{m}{M} (m_1 I_L + \frac{s}{S} l + \frac{l}{S})} \right) \quad (18)$$

The final equation is as follow:

$$\hat{x}[n] = \frac{1}{SM} \sum_{m_1=\frac{-l}{SI_L}}^{M-1-l} \hat{x}[m_1] e^{j2\pi \frac{m I_L}{M} m_1 e^{j2\pi \frac{n}{N} l}} \quad (19)$$

The DFT-spreading scheme for IFDMA, the IFFT input signal  $\hat{x}[n]$  at the transmitter can be expressed as:

$$I_L = 1, \quad l = 0,$$

$$\hat{x}[n] = \frac{1}{S} \left( \frac{1}{M} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi s \cdot m_1 + e^{j2\pi \frac{m}{M} m_1}} \right) \quad (20)$$

The resultant time symbols  $\hat{x}[n]$  are a repetition of the original input symbols  $x[m]$  in the time domain. Thus, the PAPR of IFDMA signal is the same as in the case of the conventional single carrier signal (Myung et al. 2006a)[4].

$$\hat{x}[n] = \frac{1}{S} \cdot x[m] \quad (21)$$

The DFT-spreading scheme for LFDMA, the IFFT input signal  $\hat{x}[n]$  at the transmitter can be expressed as:

$$I_L = M, \quad m_{1l} = 0, \quad l = 0, \dots, M-1,$$

$$m_1 = \frac{k-l}{SI_L} \quad (22)$$

Since  $m_1=0$ , it can be substituted in Equation (3.21) then

$$k-l=0 \text{ then } k=l \quad (23)$$

Hence output sequence of  $\hat{x}[n]$

$$\hat{x}[n] = \frac{1}{N} \sum_{l=0}^{N-1} \hat{x}[l] e^{j2\pi \frac{n}{N} l} \quad (24)$$

$$\hat{x}[n] = \frac{1}{S} \cdot \frac{1}{M} \sum_{l=0}^{N-1} \hat{x}[l] e^{j2\pi \frac{M \cdot s + m}{SM} l} \quad (25)$$

For  $s \neq 0$

$$\hat{x}[n] = \frac{1}{S} \cdot \frac{1}{M} \sum_{l=0}^{N-1} \hat{x}[l] e^{j2\pi \left[ \frac{s}{S} + \frac{m}{SM} \right] l} \quad (26)$$

For  $s=0$

$$\hat{x}[n] = \frac{1}{S} \cdot x[m] \quad (27)$$

For GIFDMA

$$1 < I_L < M \quad m_1 = 0, 1, \dots, \frac{M}{I_L} - 1 \quad l = 0, 1, \dots, I_L - 1 \quad (28)$$

Hence the output sequence of  $\hat{x}[n]$

$$\hat{x}[n] = \frac{1}{SM} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi \frac{m I_L}{M} m_1} e^{j2\pi \frac{n}{N} l} \quad (29)$$

For  $s \neq 0$

$$\hat{x}[n] = \frac{1}{SM} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi \frac{m I_L}{M} m_1} e^{j2\pi \frac{M s + m}{SM} l} \quad (30)$$

For  $s=0$

$$\hat{x}[n] = \frac{1}{SM} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi \frac{m I_L}{M} m_1} e^{j2\pi \frac{m}{SM} l} \quad (31)$$

$$\hat{x}[n] = \frac{1}{S} e^{j2\pi \frac{m}{N} l} \frac{1}{M} \sum_{m_1=0}^{M-1} \hat{x}[m_1] e^{j2\pi \frac{m I_L}{M} m_1} \quad (32)$$

$$\hat{x}[n] = \frac{1}{I_L S} e^{j2\pi \frac{n}{N} l} x[m] \quad (33)$$

#### 4. Simulation Model

The goal of mapping scheme is to distribute the subcarriers in an optimal assignment to reduce the difficulty of the mapping. In this study a new subcarrier mapping was generated by introducing Interleave Level ( $I_L$ ). The GIFDMA was considered as general scheme and the IFDMA and LFDMA are special case by GIFDMA. It consisted of a variable interleave level. Its changeability of the  $I_L$  degree is subject to the receiver acknowledged it can be IFDMA, LFDMA and GIFDMA. As shown the simulation in Figure 5, the scheme was controlled by the degree level of the  $I_L$ . The IFDMA and the LFDMA will be executed when  $I_L=1$  and  $I_L=M$ , respectively. For example, if  $I_L=M=4$ , then GIFDMA is accomplished when  $I_L=2$  and 3, the value of  $I_L$  and  $M$  are subject to change according to mapping scheme for instance if  $I_L=M=7$ , then GIFDMA is accomplished when  $I_L=2, 3, 4,$  and 6. The IFDMA and the LFDMA will be performed when  $I_L=1$  and  $I_L=M$ . and so forth. The simulation model for the subcarrier mapping is shown in Figure 5.

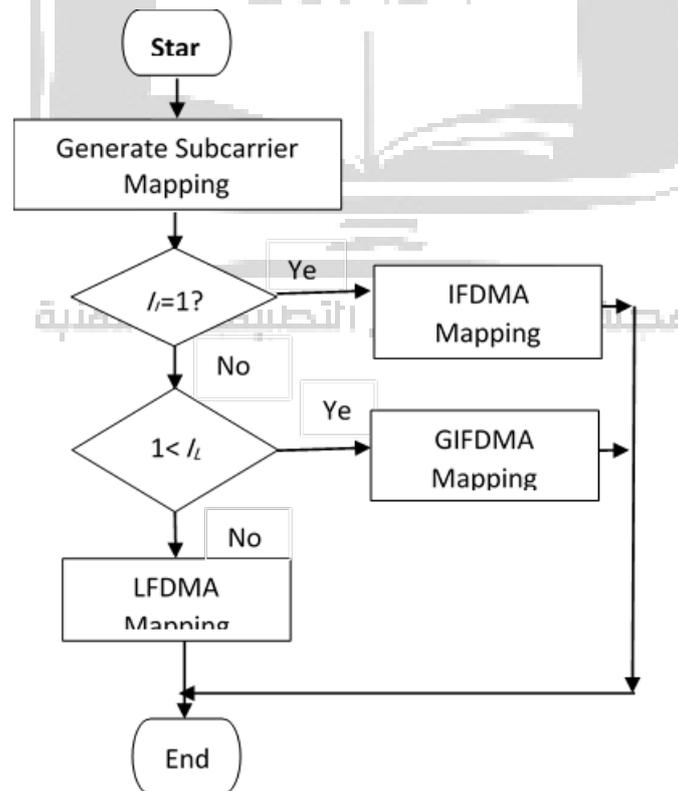


Fig. 5: Flow chart of GIFDMA Mapping

#### 3.1 Implementation of A Multilayer Perceptron (MLP)

A general correlation between  $\log_2(I_L)$  and PAPR (dB) cannot be achieved by means of a simple set of mathematical expression; rather, single experimental points are obtained by time consuming algorithms, as explained earlier in Simulation model . However, the problem of extending the set

of points for any arbitrary value of  $\log_2$  (IL) and any alpha would be intractable with commonly available computing power. On the other hand, an empirical model and a simple algorithm could accelerate the calculation, bearing in mind that any output of such a model would have to be trusted without – or with minimal – further checks through experimental data. Such a model would be required to repeat some available experimental data during its creation. Artificial Neural Networks (ANN) is particularly useful under the aspect of predicting further data when trained to memorize and generalize a much smaller subset of available data. In particular, Multilayer Perceptron (MLP) network were developed with this task in mind. Ding et.al (2013) [3]. MLP networks belong to the class of supervised training ANN, in a way they need some experimental data to make them capable of modeling fairly complex and non linear relations between input and output values. With this premise, a predicting model between  $\log_2$  (IL), alpha and PAPR (dB) can be obtained with a MLP, provided a small set of experimental data are produced beforehand.

After the MLP is successfully trained, it can be used simply inserting a suitable input, such as a set  $[\log_2$  (IL), alpha] and the output PAPR is generated in matter of milliseconds. Whether the output is correct or not, the verification can only be done with lengthy and time-consuming calculations detailed in this Chapter. This process would be a part of the MLP post-processing, the validation step.

As per any MLP, the training data is formed by set of [input, target] values, in which the target is the desired result when the corresponding input is fed into the network. In this research, the MLP training data is consequently provided by the following experimental points, which have been previously obtained:

The input is the list of vectors  $x_i = (\log_2(I_L)_i, \alpha_i)$  which are disposed as rows in the input matrix x; The target is the list of values PAPR<sub>i</sub>, which form the column vector tar. the flow chart of MLP is shown in Figure 6.

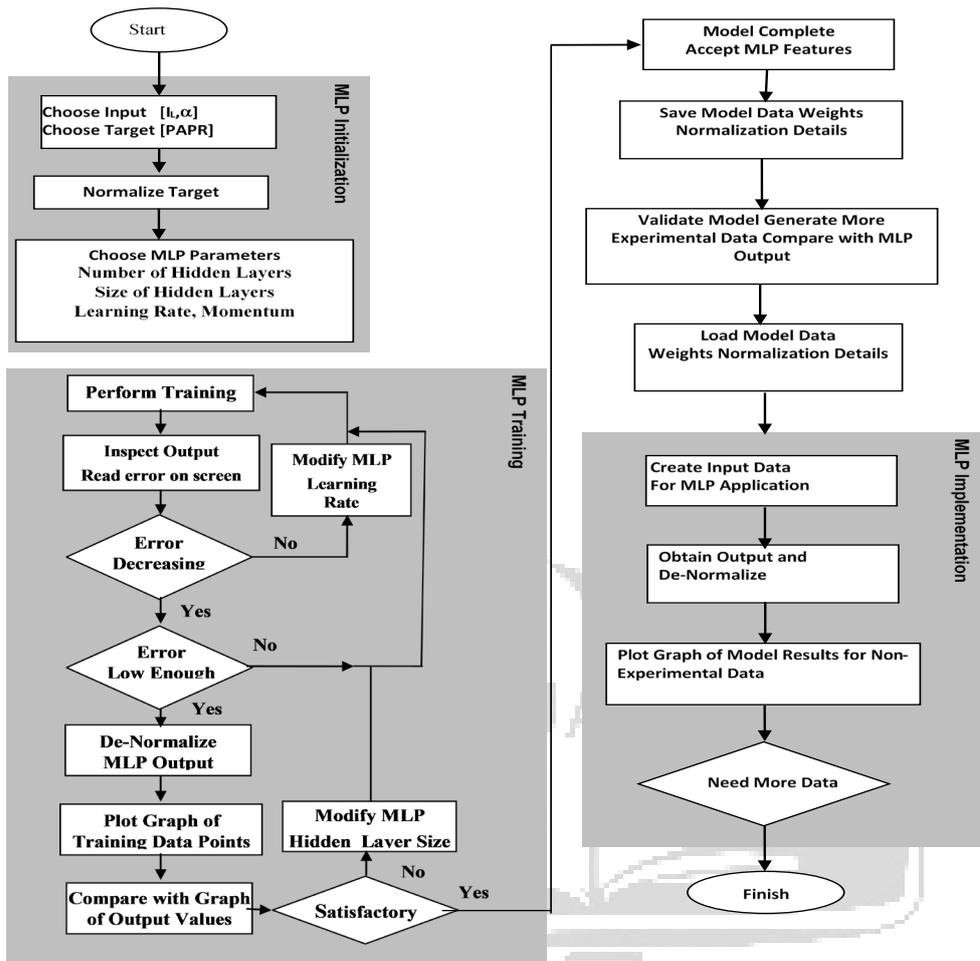


Fig. 6: Flow chart of the Multilayer Perceptron generation, training and implementation

The MLP processes the input in a set of stages, which repeat as many times as the number of layers in the MLP. The simplest of MLP contains one hidden layer before the output layer, thus requiring two weight matrixes and two filtering processes. The following description applies for a one hidden layer MLP, which was selected as generalization tool in this research. Fath, et al. (2020)[4].

The training process can be performed by feeding the input vectors one by one (on-line training) or all together as a matrix (batch training). The latter is preferable as it provides faster reduction of the training error and it is well suited to be implemented in Matlab. Its early development issue of being storage demanding is no more a problem with modern computers.

The computational sequence performed in MLP training requires two phases. First, the input matrix is fed to the MLP, in order to calculate the hidden layer matrix  $h$  and the output (feed forward process). Each row of the vector  $x$  is multiplied by a suitable weight  $W_{ij}$ , then the weighted components are summed together and the sum is filtered through a sigmoid activation

function,  $af$ , to obtain a new vector, which forms a row of the hidden layer,  $h$ . The process is repeated to obtain more hidden layers or directly the MLP output (Haykin 2009)[6] (Grippio 2015)[5]. The feed forward expressions in batch training are:

$$h = af(x \cdot W_1) \quad (34)$$

$$out = af(h \cdot W_2) \quad (35)$$

The training consists in finding set of weight matrixes  $W_1, W_2$  for which the output  $out$  differs as little as possible from the target tar. For this purpose, the error value is first calculated:

$$err = \frac{1}{2} \|tar - out\|^2 \quad (36)$$

This error value, with respect of the above equations, is a function of the input matrix – which is constant during training – and the weights. The process of weight correction is performed during the back propagation phase. Back propagation consists of calculating the gradient of the error function in the error expression with respect of each weight component. The correction of the weight components is applied in proportion to the gradient, according to the principle of maximum descent.

The whole process, feed forward and back propagation, is repeated even thousands of times as necessary while the error value decreases. This, in its numerical expression, is used a general and universal indication of the efficiency of the training process, as its real-time inspection provides insight on the quality of weight correction. Typically, a threshold value of the error is set, below which the training is deemed complete and the error value inspection tells if the set threshold is being quickly reached.

The error value inspection can show that training can become slow and starts taking lots of computing time. The training speed is tuned by appropriate choices of the learning rate, momentum and initial weight values. The fine tuning is done by inspection of the initial error descent but normally takes a few tries only. The final error threshold is chosen by inspection of the results. A too low value might mean overtraining in which the result may become useless. In the end, the MLP is supposed to reproduce the training data, as the training error is evaluated on the difference between the MLP output and the target values themselves. Graphical inspection is nonetheless necessary to spot the general behavior of the MLP. In this particular case, overtraining – with the error threshold set too low – would cause the constant alpha lines very wobbly and probably not representative of the real experimental behavior. A good MLP must be capable to accept small errors on the training data while favoring generalization over

memorization. The same behavior should be accepted in the validation phase, where the MLP should offer a good but not exact match between its output and new experimental data, which were not used for training purposes.

The weight matrixes  $W1W2$  store all the information of the model and no further data needs to be saved for any future application of the MLP. Bearing in mind the purpose for which the model was developed, its implementation requires the application of the feed forward phase of the MLP. The only requirement would be having the set of the two values  $[\log_2 (IL), \alpha]$ , feed to the MLP and get the best prediction of the corresponding PAPR as output, in matter of microseconds. The architecture of the MLP implementation in matrix form allows as well the use of many input vectors stacked as a matrix. In summary, the MLP purpose and architecture will be as follows.  
Purpose: provide a heuristic link between  $\log_2 (IL)$ ,  $\alpha$  and PAPR (dB) MLP feature: one hidden layer Input:  $\log_2 (IL)$  and  $\alpha$  Target: PAPR (dB) Stop Criterion: MLP standard error (Eq. ...) decreases.

Validation: use new data – or data not used for training Usage: apply the feed forward phase only to estimate PAPR from input  $[\log_2 (IL), \alpha]$ .

#### 4.2 Implementation of random forest (RF)

In this research random forest based on the training experiments, same as MLP. The evaluation of results of this research used Weka software; these are Parameters for classifiers with features extractions used for datasets. These parameters as shown in table 4 are selected depending on improved of accuracy results. Random Forest (RF) is mainly practical under the feature of predicting further data when trained to memorize and simplify a much smaller subset of presented data. RF belong to the group of managed training ANN, in a way they require some experimental data to make them able of modeling fairly complex and nonlinear relations between input and output values same as MLP. With this hypothesis, a predicting model between  $\log_2 (IL)$ ,  $\alpha$  and PAPR (dB) can be achieved with a RF, provided a small set of experimental data are created in advance. After the FR is successfully trained, it be able to be use basically inserting a suitable input, such as a set  $[\log_2 (IL), \alpha]$  and the output PAPR is produced (Amrehn et al 2018) [9]. RF networks as same as MLP belong to the class of managed training ANN, in a manner they need some experimental data to generate them able of modeling fairly complex and nonlinear relations between input and output values. With this idea, a predicting model between  $\log_2 (IL)$ ,  $\alpha$  and PAPR (dB) can be obtained with a RF, provided a small set of experimental data are made in advance. The 21 seeds express 7 levels of  $\log_2 (IL)$  and three levels of  $\alpha$ , which have been experienced with GIFDMA so as to execute faster PAPR output.

**Table 4:** Illustrates the RF Classifier Parameter for GIFDMA

Classifiers Type	Parameter Types	GIFDMA
RF	Instances	21 (3 level of alpha and 7 level of IL)
	Attributes	3 (alpha= 0 – 0.5 -1 )
	Test mode	Training set
	Seeds	21
	Training Time/sec	0.66

### 4.3 Performance Evaluation Criteria

The performance of the proposed subcarrier allocation system will be evaluated mainly according to the PAPR and Bit Error Rate (BER) performances under various setting and constraints such as roll-off factors, FFT/IFFT size, number of subcarrier, and number of bits per QAM. However, for PAPR performance measure, the Complementary Cumulative Distribution Function (CCDF) is commonly used instead of Cumulative Distribution Function (CDF). The CCDF of the PAPR denotes the probability that the PAPR of a data block exceed a given threshold.

## 5. PAPR Time Analysis

Time analysis is uses to measure the time (average) taken by a set of codes or algorithms to process data or be kept running, as an element of the measure of information. Moreover, the time taken reflects how fast a program function takes to process a given input. Every process will be evaluated and monitored in the MLP process, be it the training phase and the implementation by the input of new data and new points and finally calculate the necessary time.

In the case of the MLP approach, the overwhelming part of the time consumption is taken by the generation of the training data in the form of a set of (IL,  $\alpha$ ) couples, as well as of the actual MLP algorithm training. In this research, the training data is taken as the output of the simulation algorithm. No more of a couple dozen points are needed and their generation has to be performed a single time. The actual training of the MLP algorithm has to be carried on a single time as well, until a set of ANN weights are satisfactorily generated. The actual implementation of the trained MLP requires the run of a single command line, thus generating millions of predicted values of the PAPR in matter of seconds.

In this study, a new subcarrier allocation scheme is introduced. The proposed subcarrier allocation scheme involves an integration of IFDMA and LFDMA. This integration is achieved by offering an interleave levels index that can be used to move from the IFDMA scheme towards the LFDMA scheme. Therefore, the proposed subcarrier allocation scheme is a generalized form of IFDMA with a variable interleaves level. The proposed scheme is called generalized Interleaved

Frequency Division Multiple Access (GIFDMA) throughout this paper. This study depends on the DFT-spreading technique.

### 5.1 GIFDMA PAPR and BER Evaluation

In this section, the proposed GIFDMA scheme is compared with (Myung & Goodman 2008)[7]; (Yadav & Bera 2015)[11]. The PAPR performance for GIFDMA has been analysed for the LTE frame structure. The modulation process (16-QAM, 64-QAM), and a Complementary Cumulative Distribution Function (CCDF) were employed in the PAPR computation. The comparison with (Myung & Goodman 2008)[7] and (Yadav & Bera 2015) [11] took in account the same parameters were used in their researches. Both of them contain the plots of CCDF of PAPR for total bandwidths of  $N = 512$  and  $256$ , while the number of subcarriers are  $M = 128$  and  $64$ .

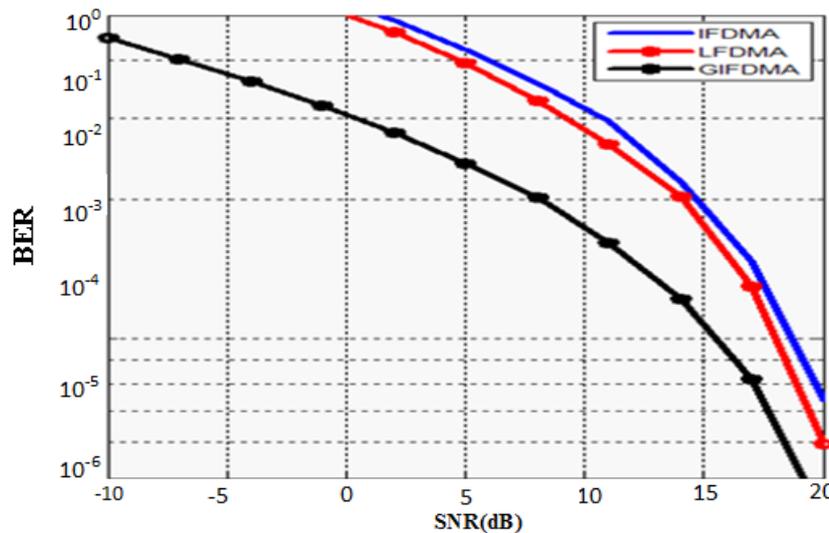


Fig. 7: BER versus SNR

As shown in Figure 7 compares the performance of the proposed scheme (GIFDMA) against standard schemes (IFDMA) and (LFDMA) in terms of bit error rate (BER) versus average received Signal to Noise Ratio (SNR) curves. It may be seen that the values of BER for the GIFDMA between  $10^{-0.5}$  to  $10^{-6}$  and SNR. Ranges from  $-10$  dB to  $20$  dB, whereas the values of the IFDMA and LFDMA schemes between around  $10^1$  and one and  $10^{-5.5}$

### 5.2 GIFDMA using MLP technique

Figure 8 demonstrates the effect of the IL on the PAPR performance; when  $IL = 1$ , points to the IFDMA scheme; whereas when  $IL = 2$  to  $32$ , it indicates the plan of GIFDMA. When  $IL = M$

(maximum) or 64, the LFDMA scheme is selected. The PAPR performance with the various IL values improves with a higher  $\alpha$  value and is more controlled by using the different IL values.

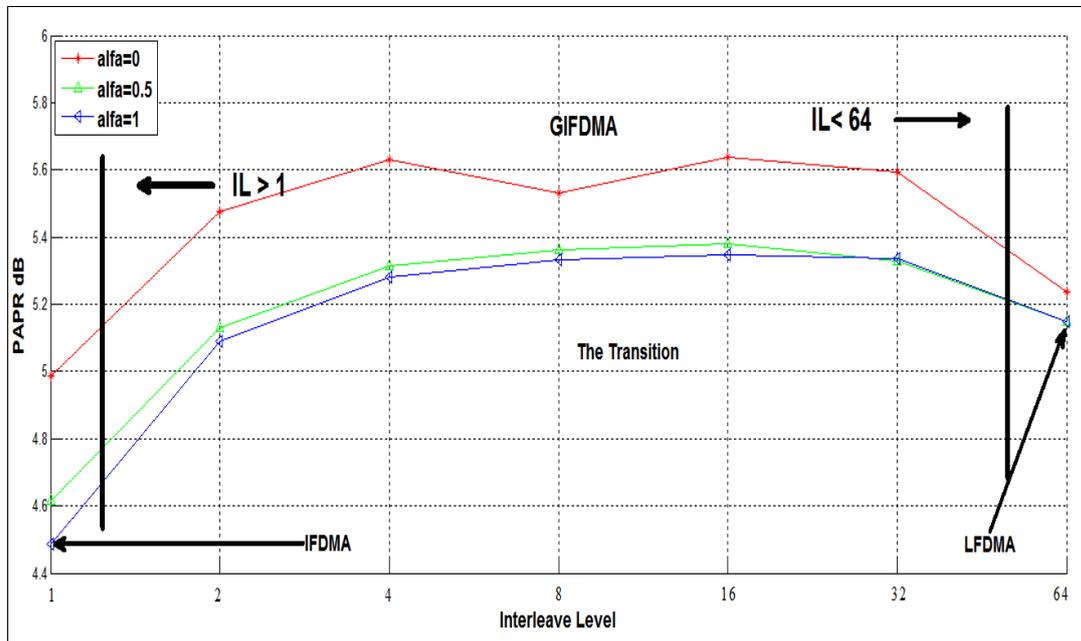


Fig. 8: Distributions and Transitions between IFDMA and LFDMA using the Interleave Level

The shift takes place according to the mapping allocations scheme for the IFDMA, the GIFDMA, and the LFDMA. The bandwidth is divided into groups' symbols. As an example, each group has 12 symbols (subcarriers), and according to interleaving level, it is possible which mapping is available to occupy the allocations. In this proposed work, the IFDMA method for each user uses four subcarriers are divided in equidistance as previously assigned. In the mean time for LFDMA, every four subsequent subcarriers were dedicated for each user and finally for GIFDMA every two subsequent subcarriers being set up for each user.

This alteration can overcome the user and frequency diversity problem. Also, the PAPR significantly decreases when the value of the roll-off factor ( $\alpha$ ) increases from 0 to 1. The PAPR signal value for the new scheme varies between the PAPR values of the IFDMA and LFDMA schemes.

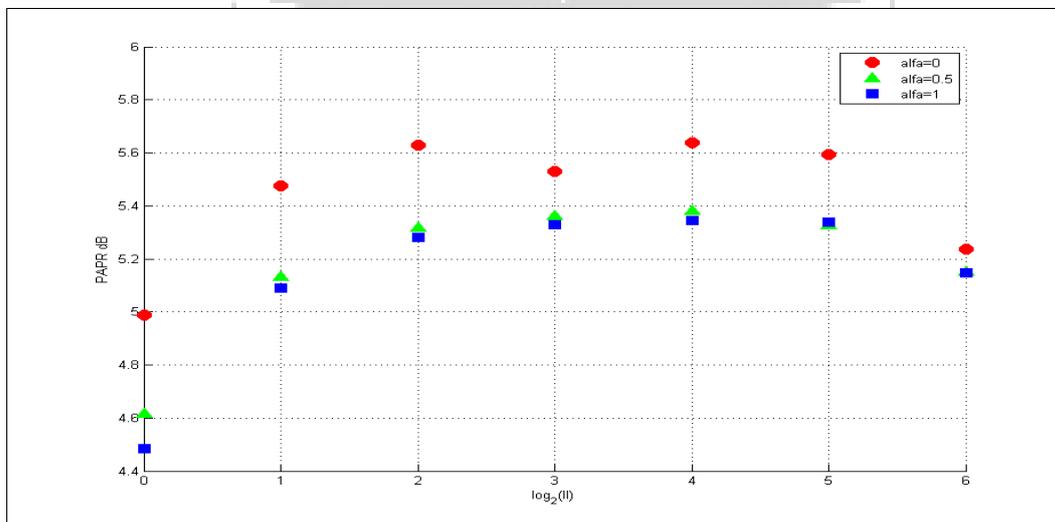
The experimental points, shown in the Figure 9, were used to train the MLP. There are 21 points forming three dotted lines for  $\alpha$  equal to 0, 0.5, and 1 respectively. The choice of the training data is critical for an adequate performance of the MLP. The training data are selected from the experimental data and are supposed to cover the whole experimental domain, with a reasonable distribution all along. In this research, the values of

$\alpha=0$  and  $\alpha=1$ , together with  $\alpha=0.5$ , are considered sufficient. However, only the inspection of the MLP output after training is capable of confirming the adequacy of the choice of training data. The MLP was designed with the following parameters as shown in Table 5.

**Table 5** Multilayer Perceptron parameters

	Population of 21 vectors $[x, \alpha]$ where $x$ is the abscissa of each point, 0 to 6.
Target	The 21 ordinates of the points, normalized to have their values belonging to the interval $[0.1-0.9]$ to match the MLP output.
Hidden units	A lower number helps learning over memorizing; it was chosen 5 after a few quick trials.
Stop criterion	Error in the MLP output equal to 0.0004

The concept of training is for MLP to give a good approximation of training data (the lines for all  $x$ ,  $\alpha = 0, 0.5$ , and  $1$  used for training should almost go through the points). For other  $x$  values and  $\alpha$  will not being used in the data training, thus the MLP outputs a value of PAPR. This PAPR value is considered of good choice, but the output still can be validated and compared with extra experimental data. In the Figure 9, some output lines for all  $x$  and some selected  $\alpha$  were plotted. Interestingly, line for  $\alpha = 0.75$  is below the line for  $\alpha = 1$  as observed in the Figure.



**Fig. :9**The 21 points used to train the MLP Input

This graph represents a typical behavior of the MLP error as calculated in Equation 36 as a function on the number of iterations. The final number of 221400 iterations represents the number needed to achieve the targeted error at which the MLP is deemed to be fully trained. Twenty one points were needed to have a sufficient landscape of the PAPR behavior as a function of  $\alpha$  and  $\log(IL)$ . After which, all other PAPR values can be achieved by ANN.

"The 21 points in Figure 9 used to train the MLP. The input is the  $(\log_2(I), \alpha)$  values, while the target is the  $y$  values. It took 1 minute to train the MLP. The MLP output for the experimental alpha value 0, 0.5, and 1, to evaluate the MLP performance. The MLP prediction for alpha equal to 0.1 to 0.4 dotted line: prediction for alpha equal to 0.75.

It is to be noted that the line is below alpha equal to 1 line. It can be observed that the difference when alpha is larger than 0.5 are minimal. The result also confirms the adequacy of training data. Figure 10 illustrates the output lines for MLP.

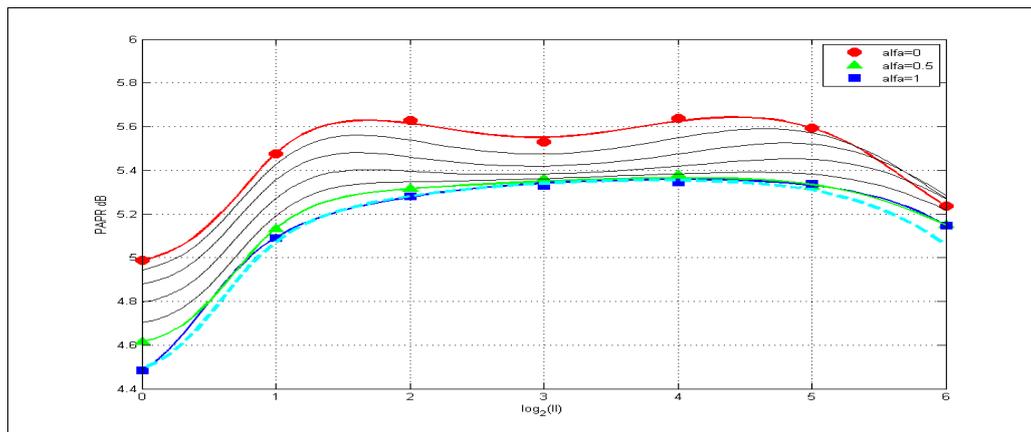


Fig. 10: The output of the MLP after training

In Figure 10 the lines are the output of the MLP after training. Lines for the alphas used for training roughly get through the points. For other  $(x, \alpha)$  values, the estimated PAPR is given by the MLP output after training is finished.

The MLP error value, provided by the Equation 36, is normally inspected during training by displaying its value in real time. Its inspection is important as the gradient descent can become slow and training can take lots of computing time. The training speed is tuned by appropriate choices of the learning rate, momentum and initial weight values.

The fine tuning is done by inspection of the error in the initial iterations of the descent. There is the possibility of a stop and restart, but this normally takes a few tries only. The final value of the error is chosen by inspection of the results. A too low value might mean overtraining in which the result may become useless and consequently a previously saved set of weights will have to be considered as better training result.

A typical training run might take hundreds of thousand loops of feed forward – back propagation calculation. The behavior of the error descent strongly depends on the complexity of the MLP – number of weights and of hidden layers, and on the initial values of the weight matrixes. The following graph represents the error behavior of a typical run. In this case the training required

221,400 iterations and lasted 14.2 seconds. The final error value, calculated according to Equation 3.36 was set at 0.0004. For the sake of time analysis of the training process, the training run was repeated several times. It was observed that in 2 to 3 cases the run had to be stopped in matter of seconds when the inspected error would not start decreasing below  $10^{-4}$  as shown in Figure 11. In all other cases the target error values was achieved in around 12 to 30 seconds.

While it took the time of 14.2 seconds to train the neural network and achieve the above results, this MLP constitutes a model of the relation between log (IL) and PAPR. It is important to stress that the MLP training needs to be performed only once, which will dramatically improve the computing time for new predictions of the PAPR value. Further values of PAPR can be easily obtained by running the MLP in feed forward mode, whose output can be obtained each time in matter of milliseconds when provided the appropriate  $(x, \alpha)$  input. Figure 11 shows the MLP error behavior for a typical run, which needed 221,400 iterations and 14.2 seconds of computing time.

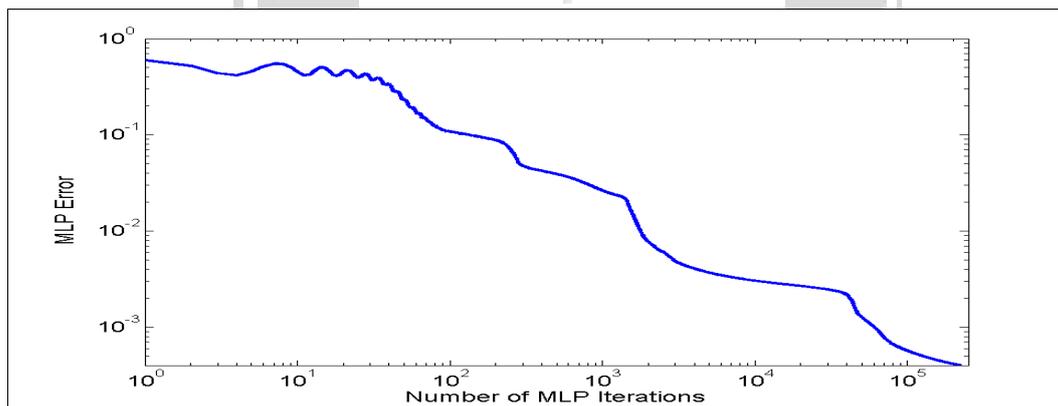


Fig. 11: The MLP error behavior for a typical run

### 5.3 GIFDMA Analysis Based on Rf

The experimental, shown in the Figure 12, were utilized the Random Forest (RF) algorithm, this figure test the three level of  $\alpha$  (0, 0.5, and 1). The selection of the data is basic for a satisfactory performance of the RF. The data are chosen from experimental data and are accepted to cover the entire experimental domain, with a reasonable distribution from the beginning. In this research, the values of  $\alpha = 0$  and  $\alpha = 1$ , together with  $\alpha = 0.5$ , are viewed as adequate. Nonetheless, just the examination of the RF output after test is capable of confirming the adequacy of the choice of data. The idea for RF is to give a good estimation of data and can compare with extra experimental data. . From the result the RF is recommended for fast calculation of the PAPR value estimation.

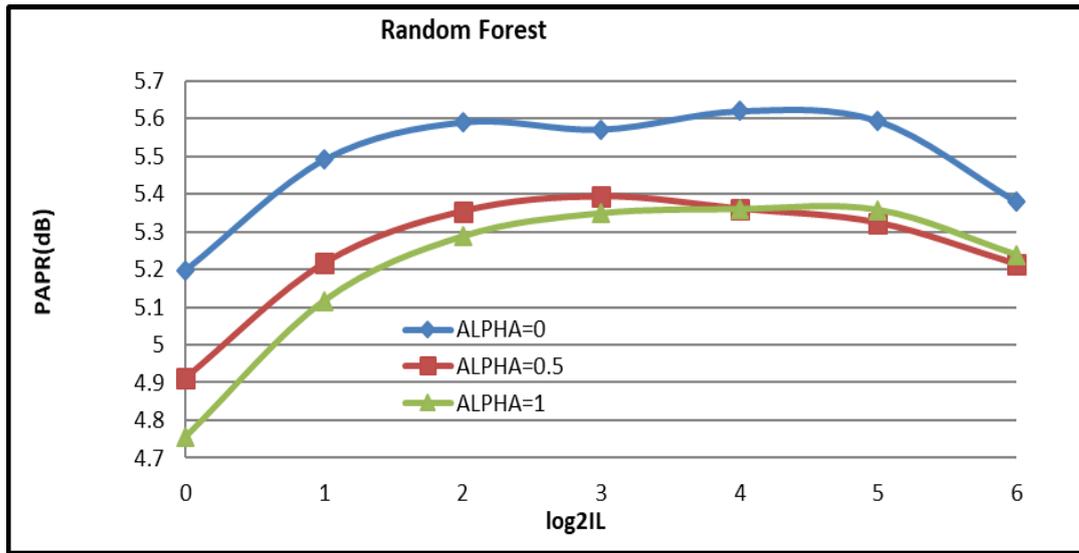


Fig. 12: The output of the RF after training

Figure 12 demonstrates the PAPR performance of GIFDMA using RF algorithm. The PAPR performance with the various IL values get better with a higher alpha value and is more controlled by using the different IL values. The result is depending on the predicting of data already trained and simplified a much smaller subset of presented data.

## 6. Time complexity analysis

In order to test execution time of the PAPR signal of GIFDMA the single experimental points are obtained by time consuming algorithms. The time has been taken with GIFDMA, GIFDMA with MLP and GIFDMA with RF as shown in table 5. The execution time for the GIFDMA scheme compared to the GIFDMA with MLP and GIFDMA with RF are presented in Table 6. The FFT size = 512, 16-QAM and alpha = {0, 0.5, and 1}.

It is obvious from result shown in Table 6 that the execution time of PAPR with MLP is faster than original GIFDMA, whereas the GIFDMA with RF is slightly faster the GIFDMA with MLP. This result leads to produce a quick and proficient count of the PAPR esteem estimation. The FFT size (N) value been not altered because it hasn't significant effect on the PAPR value according to previous results.

Table 6: Execution time GIFDMA with MLP-ANN and RF

<i>Execution Time (second)</i>					
$\alpha$	N	QAM	GIFDMA	GIFDMA with MLP	GIFDMA with RF
0	512	16	27.5	0.15	0.02
0.5	512	16	22.7	0.032	0.03
1	512	16	12.5	0.018	0.01

MLP networks belong to the class of supervised training ANN and RF is classified. In this study, it is noticed that the MLP uses hidden lines; therefore any value of alpha can be used, which make it more accuracy than RF. On the contrary, RF uses certain values of  $\alpha$  as noticed in weka program which make it a bit different against MLP on the other hand RF is slightly faster than MLP in terms of PAPR performance.

## 7. References

- [1] Amrehn, M., Mualla, F., Angelopoulou, E., Steidl, S., & Maier, A. (2018). The random forest classifier in WEKA: Discussion and new developments for imbalanced data. arXiv preprint arXiv:1812.08102.
- [2] Cho, Y. S., Kim, J., Yang, W. Y. & Kang, C. G. 2010. MIMO-OFDM Wireless Communications with Matlab. John Wiley and Sons.
- [3] Ding, S., Li, H., Su, C., Yu, J., & Jin, F. (2013). Evolutionary artificial neural networks: a review. Artificial Intelligence Review, 39(3), 251-260.
- [4] Fath, A. H., Madanifar, F., & Abbasi, M. (2020). Implementation of multilayer perceptron (MLP) and radial basis function (RBF) neural networks to predict solution gas-oil ratio of crude oil systems. Petroleum, 6(1), 80-91.
- [5] Grippo, L., Manno, A., & Sciandrone, M. (2015). Decomposition techniques for multilayer perceptron training. IEEE transactions on neural networks and learning systems, 27(11), 2146-2159.
- [6] Haykin, S. S., Haykin, S. S., Haykin, S. S. & Haykin, S. S. 2009. Neural Networks and Learning Machines. 3. Pearson Upper Saddle River, NJ, USA:.
- [7] Myung, H. G. & Goodman, D. J. 2008. Single Carrier Fdma: A New Air Interface for Long Term Evolution. John Wiley and Sons.
- [8] Pandey, N., Paulus, R. & Jaiswal, A. 2015. Performance Improvement of IFDMA and LFDMA Using NCT Technique. International Journal of Computer Applications 122(10): 13-16.
- [9] Rahman, M. M. & Manir, S. B. 2012. Performance Analysis of SC-FDMA and OFDMA in LTE Frame Structure. International Journal of Computer Applications 45(23): 31-38.
- [10] Singh, S. K. & Patidar M. K. 2015. PAPR and SNR Performance Analysis of IFDMA and LFDMA Technique in a Single Carrier Frequency Division Multiple Access System. International Journal Of Engineering And Computer Science 4(5): 11889-11894.
- [11] Yadav, S. P. & Bera, S. C. 2015. Single Carrier Fdma Technique for Wireless Communication System. India Conference (INDICON), 2015 Annual IEEE, hlm. 1-6.