



USE OF THE DEEGAN-PACKEL POWER INDEX TO OPTIMIZE MEDIA ACCES IN LAN NETWORKS WHIT BLP TECHNOLOGY*

Juan Carlos Vesga Ferreira**

Martha Fabiola Contreras Higuera***

Jose Antonio Vesga Barrera****

Recibido: 13/04/2021 • Aceptado: 15/12/2021

<https://doi.org/10.22395/rium.v21n40a7>

Abstract

A LAN network is considered as a scenario in which the nodes regularly have the need to transmit, which generates a constant dispute to access the medium, and BPL (Broadband Power Line) networks are no exception. BPL technology is characterized by using the electrical network as a means of transmission and for this it makes use of several standards, IEEE 1901 among them, due to its wide acceptance. Based on the growth in the demand for new services and to offer acceptable levels of QoS (Quality of Service), IEEE 1901 operates a hybrid medium access mechanism, supported by CSMA/CA and TDMA. However, this strategy has shown little efficiency when increasing the number of nodes in the network, because only one of the nodes can transmit at a time. This article proposes a medium access mechanism on OFDMA (Orthogonal Frequency-Division Multiple Access), based on weighted voting game theory and supported by the Deegan-Packel power index, in order to optimize access. in the time and frequency domain, improving network performance and allowing multiple nodes to transmit simultaneously.

Keywords: Media Access; IEEE 1901; Deegan-Packel Index; Weighted Voting Games; OFDMA; BPL Technology.

* Finished research article presented within the framework of the research project Bandwidth optimization for networks supported on Broadband Powerline under the use of cooperative game theory lasting two years and financed by the Universidad Nacional Abierta y a Distancia (UNAD).

** Doctor of Engineering. Associate professor. School of Basic Sciences. Technology and Engineering. Universidad Nacional Abierta y a Distancia. Colombia. E-mail: juan.vesga@unad.edu.co. Orcid: <http://orcid.org/0000-0003-3764-7265>

*** MSc in Electronic Engineering. Occasional Professor. School of Basic Sciences. Technology and Engineering. Universidad Nacional Abierta y a Distancia. Colombia. E-mail: martha.contreras@unad.edu.co. Orcid: <https://orcid.org/0000-0002-4394-2589>

**** MSc in Engineering. Research Professor. Corporación Universitaria de Ciencia y Desarrollo. Colombia. E-mail: jose.vesga@unicenciabga.edu.co. Orcid: <http://orcid.org/0000-0003-2403-4646>

Uso del índice de poder Deegan-Packel para la optimización de acceso al medio en redes LAN con tecnologías BPL

Resumen

Una red LAN es considerada como un escenario en el cual los nodos tienen la necesidad recurrente de transmitir, lo cual genera una disputa constante para acceder al medio, y las redes BPL (Broadband Power Line) no son una excepción. La tecnología BPL se caracteriza por usar las redes energéticas como medios de transmisión y para esto usa varios estándares, estando el IEEE 1901 entre ellos, debido a su amplia aceptación. Basados en el crecimiento de la demanda de estos nuevos servicios y para ofrecer niveles aceptables de QoS (Calidad del Servicio en sus siglas en inglés), el IEEE 1901 opera un mecanismo híbrido de acceso al medio soportado por CSMA/CA y TDMA. Sin embargo, esta estrategia ha mostrado muy poca eficiencia al incrementarse el número de nodos en la red, ya que solo puede haber un nodo transmitiendo al mismo tiempo. Este artículo propone un mecanismo de acceso al medio en OFDMA (Orthogonal Frequency-Division Multiple Access), basado en juegos de votación ponderada y soportado por el índice de poder Deegan-Packel con el fin de optimizar el acceso en el tiempo y frecuencia del dominio, mejorando así el desempeño de la red y permitiendo que múltiples nodos transmitan simultáneamente.

Palabras clave: acceso a medios; IEEE 1901; índice Deegan-Packel; juegos de votación ponderada, OFDMA, tecnología BPL.

INTRODUCTION

BPL Technology uses the electrical network as the physical means of transmission, and as the transmission standard IEEE 1901 (Home Plug AV) [1]. It also uses OFDM as a multiplexing technique and a hybrid medium access mechanism supported on CSMA/CA and TDMA where CSMA/CA is intended for the transmission of data packets, while TDMA is used for the transmission of voice and video packets, in order to offer optimal levels of QoS [2]. Although IEEE 1901 can achieve transmission rates close to 500 Mbps, the standard lacks policies and a suitable medium access mechanism for optimal allocation of resources in the frequency and time domain [3]. This notoriously affects the provision of services over IP and the performance of the network, as the number of users and the dynamic conditions of the channel increase [4]. Therefore, the following question arises: What should be done to optimize access to a BPL channel, according to the channel conditions and the requirements established by each node, in order to offer adequate QoS levels and without affecting the performance of another service? Given what was proposed, the use of weighted voting game theory, supported by the Deegan-Packel power index, is proposed as a strategy to optimize access to the BPL channel.

A BPL LAN network is a scenario in which the nodes have permanent needs to transmit, which generates disputes to access the medium, causing situations of inequity and waste of resources [5]. However, it is possible to establish an access order to the BPL channel by using Democratic systems supported on weighted voting game theory. In addition, by considering the nodes of a network as players that can work cooperatively, the probability of making a profit increases higher than that obtained from acting individually [6].

The concept of power is a very useful criterion for decision-making in various systems, and game theory makes an in-depth study of this topic. Each player can have different levels of power over the voting process, depending on the value or number of votes associated with it. However, there is not a linear dependence with the number of votes because there is a possibility that each player generates coalitions with other players [7]. Considering the above, a methodology is proposed to establish a *scheduler* per IEEE 1901 period (33.33ms), supported in the Deegan-Packel power index, to optimize the allocation of the spectrum in each node and class of service (in the time and frequency domains) estimating the allocation of the bitrate (BW) and the requirements of QoS. In addition, the use of OFDMA is proposed (*Orthogonal Frequency-Division Multiple Access*) [8] instead of OFDM (*Orthogonal Frequency Division Multiplexing*), which allows more than one node to transmit simultaneously through the BPL channel [9, 10].

The generation of the scheduler is a process managed and socialized by the node that fulfills the function of Central Coordinator (CCo) in accordance with the provisions of the IEEE 1901 standard. The Scheduler is considered as a strategy that seeks to maximize throughput and minimize collisions that occur in a multi-user scenario under a common collision domain, such as occurs in a BPL network [11].

1. METHODOLOGY

1.1 Weighted voting games

Definition: A weighted voting game is defined as a triple (N, w, q) where $N = \{1, 2, \dots, n\}$ is the set of players, $w = (w_1, w_2, \dots, w_n)$ with $w_i \geq 0$ for $i = 1, 2, \dots, n$ is a distribution of weights and $q \in \mathbb{R}^+$ is the quota or majority required to win a voting process. Generally, $q > \frac{T}{2}$, where T corresponds to the total sum of the weights $T = w_1 + w_2 + \dots + w_n$. A weighted voting game is represented as expressed in equation (1) [12]:

$$[q : w_1, w_2, \dots, w_n] \quad (1)$$

When in a voting game none of the voters has enough votes to equal the established quota, it is necessary to carry out a process of cooperation between several of them to unite their votes and establish alliances, called “coalitions”, which allow them to exceed the established quota and thereby win the voting process.

The power index is a number that is assigned to each player in order to indicate the level of power that he exercises over the game, and can be interpreted as a percentage of power for each group within the system [13]. The power indices quantify the possibilities of cooperation that each group has within the winning coalitions. The more weight, the more power. However, there are situations in which, despite the fact that the groups have different weights, they can present equal power indices [14]. In this article, the power index of Deegan-Packel will be used as a strategy for generating a Scheduler for channel access in a BPL network.

1.2 Deegan-packel power index

The power index for a given participant in a weighted voting game is established as follows

1. Find the total number of minimal winning coalitions, that is, coalitions in which all participants are critical.
2. Find among all the g minimal coalitions, those $x(P)$ coalitions that contain the participant P and calculate in them the inverse of the total number of voters $n(x(P))$ that compose it.

- The index is obtained by dividing the sum of these inverses by the total number of minimal coalitions.

If there are g minimal coalitions and in them there are m in which the participant P appears, the power index can be calculated through the equation (2):

$$I_{DP}(P) = \frac{\sum_{j=1}^m \frac{1}{n_j(x(P))}}{g} \tag{2}$$

1.3 Algorithms for weighted voting games.

The calculation of power indices in weighted voting games is a problem that presents high levels of computational complexity when the game contains more than 20 players, due to the use of permutations within the classical algorithms [15]. Fortunately, there is a way to accomplish this process without having to inspect all possible coalitions, by using generating functions, which offer an indirect method for calculating the number of configurations of certain types. Let $\{a_j\} \forall j > 0$ be a sequence of real numbers, which can be represented by equation (3).

$$f(x) = \sum_{j=0}^{\infty} a_j x^j \tag{3}$$

The function $f(x)$ is called the "ordinary generating function" of the sequence $\{a_j\} \forall j > 0$. Ordinary generating functions provide a method of counting the number of elements $c(k)$ in a finite set [16]. It is possible to represent an ordinary generating function using equation (4).

$$f(x) = \sum_{k>0} c(k) x^k \tag{4}$$

The number of subsets of k elements of $N = \{1, 2, \dots, n\}$ can be determined by the equation (5) which allows to calculate the binomial coefficients.

$$\binom{n}{k} = \frac{n(n-1)\dots(n-k-1)}{k!} \tag{5}$$

Therefore, the ordinary generating function that approximates the binomial coefficients is given by equation (6).

$$\sum_{k>0} \binom{n}{k} x^k = (1+x)^n \tag{6}$$

in Ribnikov [17] information is found that describes in greater depth the aspects related to the use and properties of the generating functions. Next, the generating functions are used as a strategy to calculate the Deegan-Packel power index in a weighted voting game (N, ν) , defined by $\nu = [q; w_1, w_2, \dots, w_n]$.

1.4 Estimation of the deegan-packel power index using generating functions

To calculate the Deegan-Packel power index in a simple game (N, ν) , a generating function must be used that allows calculating the cardinal of all possible coalitions, from which the winning coalitions must be chosen and finally the minimal ones (N, ν) [18]. For this, it is defined $m_s^i(\nu)$ which corresponds to the number of minimal winning coalitions S to which the player belongs $i \in N$, formed by s players. Therefore, the Deegan-Packel power index of i in a weighted voting game can be computed by equation (7).

$$I_{dp} = \frac{1}{N(\nu)} \sum_{s=1}^n \frac{m_s^i(\nu)}{s} \quad (7)$$

The methodology for calculating the Deegan-Packel power index is as follows:

Step 1: The generating function $S(x, Z_1, Z_2, \dots, Z_n, t)$ must be calculated by equation (8).

$$S(x, Z_1, Z_2, \dots, Z_n, t) = \prod_{j=1}^n (1 + x^{w_j} z_j t) \quad (8)$$

Step 2: From the generating function, those monomials in which the power of the variable x is less than q must be eliminated, in order to determine the winning coalitions.

Step 3: Eliminate those monomials that can be divisible by some other monomial of those that are established in the group of winning coalitions. The number of monomials of the resulting function corresponds to the number of game-winning minimal coalitions $N(\nu)$.

Step 4: To obtain the Deegan-Packel value of a player $i \in N$, simply select those terms of the resulting function in which the variable Z_i appears and take the coefficient of said monomial ($m_s^i(\nu)$). The exponent of the variable t indicates the number of players that are part of said minimal coalition (s). Taking these two elements into account, we proceed to calculate the Deegan-Packel power index for player i using equation (9).

$$I_{dp} = \frac{1}{N(\nu)} \sum_{s=1}^n \frac{m_s^i(\nu)}{s} \quad (9)$$

Example: Consider the weighted voting game (N, ν) given from the shape which is presented in the equation (10).

$$\nu = [q; w_1, w_2, w_3, w_4, w_5, w_6] = [q; 28, 165, 4, 3, 3] \quad (10)$$

Step 1: We proceed to calculate the value of q .

$$q = \frac{28+16+5+4+3+3}{2} = 29,5 \approx 30$$

Step 2: We proceed to calculate the generating function by using the equation (11).

$$S(x, Z_1, Z_2, \dots, Z_n, t) = \prod_{j=1}^n (1 + x^{w_j} z_j t) \tag{11}$$

The result is as follows:

$$\begin{aligned} S(x, Z_1, Z_2, Z_3, Z_4, Z_5, Z_6, t) &= \prod_{j=1}^6 (1 + x^{w_j} z_j t) \\ &= (1+x^{28} z_1 t)(1+x^{16} z_2 t)(1+x^5 z_3 t)(1+x^4 z_4 t)(1+x^3 z_5 t)(1+x^3 z_6 t) \end{aligned}$$

Step 3: From the generating function, those monomials in which the power of the variable x is less than q must be eliminated, in order to determine the winning coalitions. In view of the above, the resulting function is the following:

$$\begin{aligned} &Z_1 Z_2 Z_3 Z_4 Z_5 Z_6 t^6 x^{59} + Z_1 Z_2 Z_3 Z_4 Z_5 t^5 x^{56} + Z_1 Z_2 Z_4 Z_5 Z_6 t^5 x^{54} + Z_1 Z_2 Z_3 Z_4 Z_6 t^5 x^{56} + \\ &Z_2 Z_3 Z_4 Z_5 Z_6 t^5 x^{31} + Z_1 Z_3 Z_4 Z_5 Z_6 t^5 x^{43} + Z_1 Z_2 Z_3 Z_5 Z_6 t^5 x^{55} + Z_1 Z_2 Z_4 Z_5 t^4 x^{51} + Z_1 \\ &Z_2 Z_3 Z_4 t^4 x^{53} + Z_1 Z_2 Z_4 Z_6 t^4 x^{51} + Z_1 Z_3 Z_4 Z_5 t^4 x^{40} + Z_1 Z_2 Z_3 Z_5 t^4 x^{52} + Z_1 Z_4 Z_5 Z_6 t^4 \\ &x^{38} + Z_1 Z_2 Z_5 Z_6 t^4 x^{50} + Z_1 Z_3 Z_4 Z_6 t^4 x^{40} + Z_1 Z_2 Z_3 Z_6 t^4 x^{52} + Z_1 Z_3 Z_5 Z_6 t^4 x^{39} + Z_1 Z_2 \\ &Z_4 t^3 x^{48} + Z_1 Z_4 Z_5 t^3 x^{35} + Z_1 Z_3 Z_4 t^3 x^{37} + Z_1 Z_2 Z_3 t^3 x^{49} + Z_1 Z_4 Z_6 t^3 x^{35} + Z_1 Z_2 Z_6 t^3 x^{47} \\ &+ Z_1 Z_2 Z_5 t^3 x^{47} + Z_1 Z_3 Z_5 t^3 x^{36} + Z_1 Z_5 Z_6 t^3 x^{34} + Z_1 Z_3 Z_6 t^3 x^{36} + Z_1 Z_4 t^2 x^{32} + Z_1 Z_5 t^2 \\ &x^{31} + Z_1 Z_2 t^2 x^{44} + Z_1 Z_3 t^2 x^{33} + Z_1 Z_6 t^2 x^{31} \end{aligned}$$

Step 4: We proceed to identify the monomials that represent the minimal coalitions. For this, those monomials that can be divisible by some other monomial of those that are established in the group of winning coalitions must be eliminated. The result of this process is the following:

$$Z_2 Z_3 Z_4 Z_5 Z_6 t^5 x^{31} + Z_1 Z_4 t^2 x^{32} + Z_1 Z_5 t^2 x^{31} + Z_1 Z_2 t^2 x^{44} + Z_1 Z_3 t^2 x^{33} + Z_1 Z_6 t^2 x^{31}$$

The value of $N(v)=6$, considering that there are 6 monomials that make up the resulting function of minimal coalitions.

Step 5: We proceed to calculate the Deegan-Packel value for each player $i \in N$. For the case of player $i = 1$, the following procedure is carried out:

Those terms of the resulting function in which appears are selected.

$$Z_1 Z_4 t^2 x^{32} + Z_1 Z_5 t^2 x^{31} + Z_1 Z_2 t^2 x^{44} + Z_1 Z_3 t^2 x^{33} + Z_1 Z_6 t^2 x^{31}$$

The exponent of the variable t indicates the number of players that are part of said minimal coalition (s). Taking these two elements into account, we proceed to perform the calculation corresponding to the Deegan-Packel power index for player $i=l$ by using the following expression:

$$I_{dp}(1) = \frac{\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2}}{6} = \frac{5}{12} = 0,4167$$

Repeating the same process for the other players, the result is as follows:

$$I_{dp}(2) = I_{dp}(3) = I_{dp}(4) = I_{dp}(5) = I_{dp}(6) = \frac{7}{60} = 0,1167$$

In summary, the Deegan-Packel value for the proposed game is as follows:

$$I_{dp}(i) = \left(\frac{5}{12}, \frac{7}{60}, \frac{7}{60}, \frac{7}{60}, \frac{7}{60}, \frac{7}{60} \right)$$

To facilitate future research processes related to the subject of power indices, a routine was developed in Matlab, which allows the calculation of the Deegan-Packel Power Index. The pseudocode corresponding to the routine is as follows:

\\ Routine to calculate the Deegan-Packel power index.

\\ The variables used inside the routine are:

\\ V: Vector of weights for each player

\\ q: Average value to consider the winning coalition

\\ Nj: Number of players

\\ VDeegan: Vector corresponding to the power indices for each player

Beginning

v←[28; 16; 5; 4; 3; 3]; \\ The vector of weights for each player is defined

Q←ceil(sum(V)/2); \\ The mean value is calculated

nj←6;

\\ The Nj symbolic variables of Zj are created

Z←linspace(1,Nj,Nj); \\ Create a vector Z with Nj elements

Z←sym('Z',[N j 1]); \\ Convert vector Z to symbolic mode

PZ← one;

For i from one until nj

PZ ← PZ*Z(i); \\ PZ corresponds to the product of the elements of Z

End_To

VDeegan← zeros(Nj,1); \\ Initialization of the vector corresponding to the Deegan-Packel value

[or] ← RandSimRepeticao(0,1E4,Nj); **\| 10000 random numbers are generated without repeating between 0 and Nj**

symsx you Y

P ← zeros(Nj-1,1);

yes ← $y^{*(1+(x^{\wedge}V(1)*Z(1)*t*y))^{*(1+(x^{\wedge}V(2)*Z(2)*t*y^{\wedge}2))}$;

PotY ← 4;

For he from 3 until nj \| Routine to establish the generating functions Bi(x)

s = $s^{*(1+(x^{\wedge}V(l)*Z(l)*t*y^{\wedge}PotY))}$;

PotY = 4^l ;

End_To

s1 ← expand(s); **\| Perform symbolic multiplication**

[cxz cy] ← coeffs(s1,y); **\| Extract the coefficients of a polynomial cxz take the values of \| xz and c and the values of y (auxiliary variable)**

c x ← cxz/(PZ*t^Nj); **\| Extract the factor X of each monomial**

[num, den] ← numden(cx);

c x ← num;

PotX ← subs(log10(cx),x,10); **\| Evaluate the polynomial and the value of x=10 and record the total weight of each coalition**

PotX ← simplify(PotX); **\| Simplify values by extracting the exponents of x**

TO ← $x.^{PotX}$;

CT ← cxz./(PZ*a); **\| Extract the factor t from each monomial**

[num, den] ← numden(ct);

CT ← num;

PotT ← subs(log10(ct),t,10); **\| Evaluate the polynomial and the value of t=10 and record the total weight of each coalition**

PotT ← simplify(PotT); **\| Simplify values by extracting the exponents of t**

M ← find(PotX >= q); **\| The monomials that meet the condition that the power of X is >= q are selected.**

PotXWin ← PotX(m); **\| Winning powers of X**

PotTWin ← PotT(m); **\| Winning powers of T**

Winners ← cxz(m);

nWin ← length(PotTWin);

For ifrom one until (nWin-1) \| Routine to sort the PotWin vector ascending

For jfrom (i+1) until (nWin) \| and the array of winners

Yes PotTWin(j) > PotTWin(i)

auxiliary ← PotTWin(j);

```

PotTWin(j) ← PotTWin(i);
PotTWin(i) ← aux;
Aux1 ← MWinners(j);
MWinners(j) ← MWinners(i);
MWinners(i) ← aux1;
End yes
End_To
End_To

```

```

\\ Routine to extract monomials that cannot be divided by other terms
R1← MGwinners; \\ A copy of the winning monomials is extracted
t1← PotTWin;
n1← length(R1); \\ The number of winning coalitions is identified
Nminimals←one;
While (n1>1)
  CMinimals(Nminimals) ← R1(n1); \\ Record the first minimal monomial
  PTMinimals(Nminimals) ← t1(n1); \\ Record the power of t of the minimal monomial
  R ← R1/R1(n1) \\ The vector R1 is divided over the last monomial in order to identify
  which are not divisible
  [num, den] ← numden(R); \\ The numerator and denominator of the previous process
  are extracted in each monomial
  M ← find(den ≠ one); \\ Those in which the denominator is different from 1 are identified
  R1← R1(m); \\ Those monomials that were not divisible during the process are
  generated.
  t1←t1(m);
  n1←length(R1);
  Nminimals←Nminimals+1; \\ Increase the number of minimal coalitions
End_While
n1←length(R1);
Yesn1>0
  CMinimals(Nminimals) ←R1(1);
  PTMinimals(Nminimals) ←t1(n1);
End yes

```

```

\\ Calculation of the Deegan-Packel Power Index
mw←length(CMinimals); \\ Mw corresponds to the total of minimal monomials
For i from one until nj
  R ←CMinimals/Z(i); \\ Divide the vector R1 over the last monomial in order to
  \\ identify which are not divisible
  [num, den] ← numden(R); \\ The numerator and denominator of the previous process
  are extracted in each monomial

```

```

M ← find(den ≠ one); \ Those whose denominator is different from 1 are identified
R1 ← CMinimals(m); \ Extract those monomials that were not divisible during the process.
t1 ← PTMinimals(m);
n1 ← length(R1);
sum ← 0;
For j from one until n1 \ Procedure that calculates the Deegan power indices
Sum ← sum + (1/t1(j));
End_To
VDeegan(i) = sum/Mv;
End_To

Print VDeegan \ Prints the vector corresponding to the power indices for each player

```

2. RESULTS

2.1 Proposed scenario

As a practical scenario, a LAN network made up of ten (10) BPL nodes is proposed. Each node has a BPL adapter and a traffic source. Each traffic source can generate more than one traffic class r simultaneously. For this particular case, only Data and Video traffic were considered. The 10th node will be considered as the main node or Coordinator (CCo). To estimate the channel conditions, the tool called “PLC Channel Generator (GC_PLC)” was used, written in MATLAB and developed by the PLC Group of the University of Malaga. In [19] information on the use of the GC_PLC tool is presented.

For the proposed scenario, the GC_PLC tool generated the values of 110.2 Mbps, considering typical conditions of a BPL channel. Table 1 displays that the total BW required by the nodes is 175.36 Mbps, which is higher than the total BW available in the BPL channel, thereby establishing a network insaturation state. Further, the values corresponding to the BW assigned for each node and class of traffic are presented using Nucleolus as a strategy for the optimization of resources in the BPL network. In [20] The methodology used to estimate the assigned BW for each player is found.

Table 1. BW requested and assigned for each node of the proposed scenario

Player or Node i	Class of Service	Priority Factor	BW Requested	Assigned BW (Nucleolus)
1	Data	1	11.69	7.19
	Video	3	9.28	5.10
2	Data	1	5.44	2.15
	Video	3	13.36	8.58
3	Data	1	5.22	3.37

Player or Node i	Class of Service	Priority Factor	BW Requested	Assigned BW (Nucleolus)
4	Data	1	13.93	9.06
	Video	3	18.57	13.09
5	Data	1	11.79	7.24
6	Data	1	14	9.12
7	Data	1	16.5	11.28
	Video	3	16.21	11.03
8	Data	1	11.88	7.32
	Video	3	13.45	8.65
9	Video	3	8.66	4.56
	Data	1	5.38	2.48
Total BW			175.36	110.20

Source: own elaboration.

2.2 Estimation of weights for each node

Considering that there is no evidence of a mechanism to calculate the weight of a player, in a simple weighted voting game, the following criteria were adopted to establish each of the required weights. Since each player can present more than one class of service, it is necessary to calculate a weight for each class r and player i . The value of r will be considered as the priority factor of the class of service to establish QoS mechanisms within the network. The total weight for player i can be calculated by using the equation (12) [14].

$$W_i = \sum_{r=1}^4 W_{i,r} \quad (12)$$

On the other hand, the weight for each class r can be calculated by using the equation (13).

$$W_{i,r} = FP_{i,r} \left(\frac{BW_{i,r}}{\min_{j \in N, k=r} BW_{j,k}} \right) \quad (13)$$

Where $\min_{j \in N, k=r} BW_{j,k}$ corresponds to the minimum value of BW assigned among all the players whose class of service is $k = r$, $FP_{i,r}$ is the priority factor of the class r player i and $[\ast]$ represents the operator that returns the closest largest integer of the recorded operation. Subsequently, we proceed to calculate the weights for each node according to the proposed criteria. The result of this process is recorded in Table 2.

Table 2. Estimated weights for each node and class of service

Player or Node <i>i</i>	Class of Service	Priority Factor	Assigned BW (Nucleolus)	Total BW for Node	Weight	Overall Weight
1	Data	1	7.19	12.29	4	10
	Video	3	5.10		6	
2	Data	1	2.15	10.72	1	7
	Video	3	8.58		6	
3	Data	1	3.37	3.37	2	2
4	Data	1	9.06	22.16	5	14
	Video	3	13.09		9	
5	Data	1	7.24	7.24	4	4
6	Data	1	9.12	9.12	5	5
7	Data	1	11.28	11.28	6	6
8	Video	3	11.03	18.34	9	13
	Data	1	7.32		4	
9	Video	3	8.65	8.65	6	6
10	Video	3	4.56	7.04	3	5
	Data	1	2.48		2	
Total BW			110.20	110.2		

Source: own elaboration.

2.3 Scheduler generation

The last objective of this article is to establish the access schedule to the medium (scheduler), by assigning the domain of time and frequency to each node *i* based on the power index of Deegan-Packel. This process is managed by the CCo and broadcast to each of the nodes, reducing the probability of collision and optimizing the use of the resource.

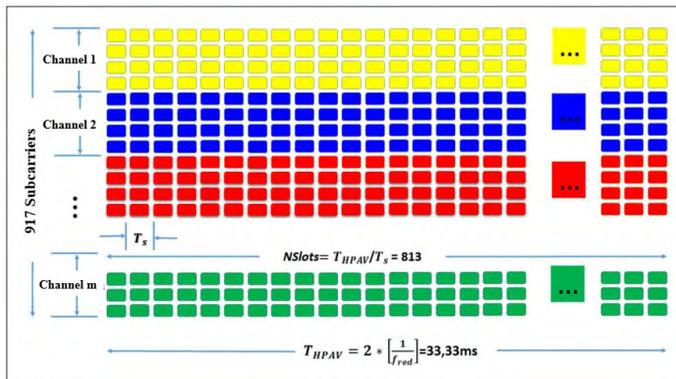


Figure 1. Proposed map for time and frequency distribution of a BPL channel.

Source: own elaboration.

Figure 1 shows the proposed map corresponding to the time and frequency distribution of a BPL channel, supported in OFDMA. The frequency domain is made up of the 917 subcarriers, and the time domain by 813 time slots. On this frequency-time map, the allocation of resources will be made for each node, according to the assigned BW and power index.

The proposed methodology for generating the scheduler is as follows:

Step 1: Establish the BW value assigned to each node that is part of the network, through the Nucleolus as a strategy for bandwidth optimization [21].

Step 2: Determine the total weight for each node and based on this information proceed to calculate the Deegan-Packel power index.

Table 3. Consolidated information for generating the Scheduler

Node i	Class of Serv.	Priority Factor	BW Requested [Mbps]	Assigned BW (Nucleolus) [Mbps]	BW i [Mbps]	Weight ($W_{i,j}$)	Overall Weight	Deegan-Packel Power Indices	Media Access Shift
1	Data	1	11.69	7.19	12.29	4	10	0.1095	3
	Video	3	9.28	5.10		6			
2	Data	1	5.44	2.15	10.72	1	7	0.0966	8
	Video	3	13.36	8.58		6			
3	Data	1	5.22	3.37	3.37	2	2	0.0613	10
4	Data	1	13.93	9.06	22.16	5	14	0.1177	2
	Video	3	18.57	13.09		9			
5	Data	1	11.79	7.24	7.24	4	4	0.0912	9
6	Data	1	14	9.12	9.12	5	5	0.1007	6
7	Data	1	16.5	11.28	11.28	6	6	0.1017	4
8	Video	3	16.21	11.03	18.34	9	13	0.1189	1
	Data	1	11.88	7.32		4			
9	Video	3	13.45	8.65	8.65	6	6	0.1017	5
10	Video	3	8.66	4.56	7.04	3	5	0.1007	7
	Data	1	5.38	2.48		2			
Total BW			175.36	110.20	110.20				

Source: own elaboration.

In Table 3, Steps 1 and 2 of the proposed methodology are consolidated, establishing the BW, the power index for each node, by using the Nucleolus and the Deegan-Packel Power Index, respectively.

Step 3: Using Engset's formula, estimate the recommended number of channels on which the entire BPL channel will be divided. Engset's formula is given by equation (14).

$$P_{bl}(m, \rho, N) = \frac{\rho^m \binom{N}{m}}{\sum_{i=1}^m \rho^i \binom{N}{i}} \quad (14)$$

where,

P_{bl} : Block Chance

ρ : ITraffic intensity generated by each node

m : Estimated number of channels

N : Number of nodes that make up the BPL network

For the proposed scenario the input parameters are:

NNodes=10;

PI=0.01;

Ro =1.59 (BW_Requested/BW_Assigned =175.36Mbps/110.2Mbps)

[server, realPI, realPb]=find_server(NNodes, Ro, PI)

The use of the function corresponding to the Engset formula gave a result of 10 servers, which is equivalent to the suggestion to divide the BPL channel into 10 channels with an estimated blocking probability (realPb) of 0.0208.

Step 4: Estimate the priority level of access to each of the m channels in each node, ordered from highest to lowest priority, considering the conditions of the BPL channel and the tone map present in each node. The result of this process is a matrix where the channels with the greatest intention of use by each node when requesting access to the medium are recorded. For practical purposes, numbers from 1 to 10 were randomly generated, which represent the priority order of each node to access the channel according to the possible response of the channel and the tone map generated. The result of this process is found in Table 4, which follows a matrix where the channels with the greatest intention of use by each node when requesting access to the medium are recorded.

Table 4. Channel priority per node for the proposed scenario

Node	Option 1	Option 2	Option 3	Option 4	Option 5	Option 6	Option 7	Option 8	Option 9	Option 10
1	10	2	8	3	7	6	1	9	3	7
2	6	7	5	10	7	3	9	1	2	4
3	8	4	8	10	2	5	3	7	5	1
4	3	7	6	5	2	1	4	10	8	9
5	9	4	3	8	7	1	5	2	10	6
6	5	1	9	3	6	8	9	10	2	4
7	4	5	6	9	10	1	8	3	2	7
8	1	8	4	9	2	5	6	7	3	10
9	1	3	2	4	5	7	6	8	9	10
10	8	9	6	2	1	4	5	7	10	3

Source: own elaboration.

Step 5: Form a Resulting Matrix (MR) in which all the parameters mentioned above are consolidated by node, and which must be ordered in descending order based on the shift assigned to access the medium. From this resulting matrix, the number of time slots equivalent to the assigned BW and the channel or channels suggested by the node are established, in an orderly manner and in accordance with the number of slots available per channel. It is important to keep in mind that the maximum number of time slots per channel is 813 and that the resource allocation can be distributed over more than one channel. The process is repeated until all nodes are assigned. Table 5 presents the resulting matrix MR, which consolidates all the information required for the construction of the scheduler.

Table 5. Resulting matrix for the proposed scenario

Node	BW [Mbps]	Turn	Op.1	Op.2	Op.3	Op.4	Op.5	Op.6	Op.7	Op.8	Op.9	Op.10
1	12.29	3	10	2	8	3	7	6	1	9	3	7
2	10.72	8	6	7	5	10	7	3	9	1	2	4
3	3.37	10	8	4	8	10	2	5	3	7	5	1
4	22.16	2	3	7	6	5	2	1	4	10	8	9
5	7.24	9	9	4	3	8	7	1	5	2	10	6
6	9.12	6	5	1	9	3	6	8	9	10	2	4
7	11.28	4	4	5	6	9	10	1	8	3	2	7
8	18.34	1	1	8	4	9	2	5	6	7	3	10
9	8.65	5	1	3	2	4	5	7	6	8	9	10
10	7.04	7	8	9	6	2	1	4	5	7	10	3

Source: own elaboration.

Next, the pseudocode of the routine that was elaborated in Matlab to carry out the generation of the scheduler based on the guidelines established in the MR matrix is presented.

\\ Routine to establish the Scheduler of a BPL network

\\ The MR Matrix is ordered in ascending order according to the turn of access to the medium.

```
Kmax ← NChannels+3;
For i from one until (Nj-1)
  For j from (i+1) until nj
    Yes MR(j,3) < MR(i,3)
    For k from one until Kmax
      auxiliary ← MR(i,k);
      MR(i,k) ← MR(j,k);
      MR(j,k) ← aux;
    End_To
  End yes
End_To
End_To
```

```
Total_Slots ← NSymbols*NChannels;
```

```
For i from one until nj \\ A Channel is the Matrix that manages channel availability
  AChannel(i,1) ← i; \\ A Channel(i,1) --> Node
  AChannel(i,2) ← 0; \\ A Channel(i,2) --> Initial Slot
End_To
K ← 0; \\ Schedule Record 1
For l from one until nj
  K ← k+1;
\\ Estimation of the number of slots required proportional to the assigned BW.
  Num_Slots ← ceil(Total_Slots*MR(i,2)/sum(MR(:,2)));
  NPri ← 4; \\ Priority channel position
  CPri ← MR(i, NPri); \\ Channel Priority
  Okay ← 0;
  While(ok = 0)
    Schedule(k,1) ← MR(i,1); \\ Node Number
    Schedule(k,2) ← MR(i,2); \\ Assigned Bandwidth
    Schedule(k,3) ← MR(i,3); \\ Media access shift
```

Yes(NSymbols >= (ACanal(CPri,2)+Num_Slots))\| Si Slots required < slots available
in the CPri channel

Schedule(k,4)←CPri;\| Assigned Channel

Schedule(k,5)←AChannel(CPri,2)+1;\| Start Slot

Schedule(k,6)←AChannel(CPri,2)+Num_Slots; \| Final Slot

AChannel(CPri,2)←AChannel(CPri,2)+Num_Slots;\| New Initial Slot value

Okay← one;

Otherwise\| On the contrary

Schedule(k,4)←CPri;\| Assigned Channel

auxiliary ←NSymbols-AChannel(CPri,2);

Schedule(k,5)←AChannel(CPri,2)+1;\| Initial Slot Part 1

Schedule(k,6)←AChannel(CPri,2)+Aux;\| Final Slot Part 1

AChannel(CPri,2)←AChannel(CPri,2)+Aux; \| New Initial Slot value

K←k+1;\| New Schedule record

NPri←NPri+1;\| Next Priority channel position

YesNPri<(NChannels+4)

CPri←MR(i,NPri);\| New Channel Priority

Num_Slots←Num_Slots-Aux;

Otherwise

ok=1;

End yes

End yes

End_While

End_To

PrintSchedule \| Print the generated Scheduler

NNodes←10;\| Number of nodes that are part of the LAN network

nj←NNodes; \| Number of players = Number of nodes

\| Estimation of HPAV eigenparameters for 60Hz

three←1/60;\| Period of an electrical network signal at 60Hz

Tsymbol←40.96e-6;\| Time of an OFDM symbol for HPAV

NSymbols←floor(2*Tred/Tsymbol);\| Number of symbols per Beacon duration

\| HPAV (2 network cycles)→813 symbols

\| Using the Engset formula to Calculate the number of Channels according to the
Traffic Intensity

\| and the available bandwidth.

p1← 0.01; \| Block chance 1% suggested.

Load_Erlangs ← 1.59; \\ Load.

server, realPI, realPb] ← find_server(NNodes, Load_Erlangs, PI);

NChannels ← server; \\ Estimated number of channels

The result of the process corresponding to the scheduler generated for the proposed scenario is as follows:

Table 6. Scheduler of the proposed scenario

Node	BW [Mbps]	Turn of access to the medium	Channel	Initial Slot	end slot
8	18.34	1	1	1	813
8	18.34	1	8	1	541
4	22.16	2	3	1	813
4	22.16	2	6	1	9
4	22.16	2	7	1	813
1	12.29	3	2	1	94
1	12.29	3	10	1	813
7	11.28	4	4	1	813
7	11.28	4	5	1	20
9	8.65	5	2	95	733
6	9.12	6	5	21	693
10	7.04	7	8	542	813
10	7.04	7	9	1	248
2	10.72	8	6	10	800
5	7.24	9	9	249	783
3	3.37	10	2	734	813
3	3.37	10	5	694	813

Source: own elaboration.

Table 6 shows the schedule established for the proposed scenario (*scheduler*), where the node, the assigned BW, the access shift to the medium, the channel or channels assigned to each node to carry out the transmission process and the time of access to the medium by means of the definition of the Slots are indicated. Start and end time for each i node.

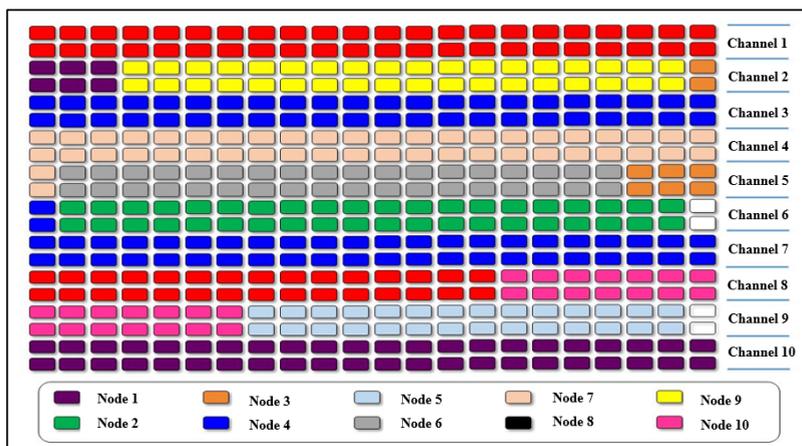


Figure 2. Map generated for the proposed scenario.

Source: own elaboration.

According to the results obtained, the map of Figure 2 is presented, where it is observed that node 8 was assigned entirely to channel 1 and from slot 1 to 541 in channel 8. Node 4 was assigned entirely to channels 3 and 7 and from slot 1 to 9 on channel 6. And so on, you can verify the assignment of frequency and time assigned to the remaining nodes. Additionally, it is observed that the allocation of resources was carried out properly and without wasting slots that are part of the frequency-time spectrum.

2.4 Comparative analysis between access mechanisms to the environment represented by the MS-HPAV VS. MH-HPAV models

In order to carry out a comparative analysis between technologies, the Hybrid Model (MH-HPAV) was taken as a reference, which makes use of the CSMA/CA and TDMA media access mechanisms in a hybrid way, currently used in PLC technology (Power Line Communications) under the HomePlugAV (HPAV) standard and on the other hand, the Smart Model (MS-HPAV), which proposes OFDMA as a medium access mechanism [16] which significantly affects the network performance when the number of users increases. In light of the foregoing, it is proposed the use of the Smart Model (SM and on which the use of the Deegan-Packel power index was articulated.

During the analysis process, the Throughput, Delay and Efficiency parameters were evaluated in each of the models, based on the scenario proposed in section 2.1 and the information recorded in Table 3. Tables 7, 8 and 9 show the results obtained for each of the models, in accordance with the proposed scenario. It is important to mention that within the analysis process only the CSMA/CA, TDMA and OFDMA access mechanisms were considered due to the particularity of BPL technology.

Table 7. Results of Throughput, Delay and Efficiency for MS-HPAV(proposed) vs.MH-HPAV(current)

Node	Class of Serv.	BW Requested	Assigned BW (Nucleolus)	Smart HPAV model (proposed) OFDMA + Deegan Packel			HPAV Hybrid Model (Current) CSMA/CA+TDMA		
		[Mbps]	[Mbps]	Throughput [Mbps]	delay [ms]	Efficiency %	Throughput [Mbps]	delay [ms]	Efficiency %
1	Data	11.69	7.19	5.99	49.17	51.24	3.09	43.72	26.45
	Video	9.28	5.1	4.25	49.69	45.80	3.21	54.04	34.62
2	Data	5.44	2.15	1.79	50.42	32.90	0.92	37.98	16.99
	Video	13.36	8.58	7.15	48.82	53.52	5.41	59.83	40.46
3	Data	5.22	3.37	2.81	50.11	53.83	1.45	39.37	27.76
4	Data	13.93	9.06	7.55	48.70	54.20	5.71	50.63	40.97
	Video	18.57	13.09	10.91	47.71	58.75	5.63	60.42	30.31
5	Data	11.79	7.24	6.03	49.16	51.15	4.56	47.60	38.69
6	Data	14	9.12	7.60	48.69	54.29	3.92	45.91	28.01
7	Data	16.5	11.28	9.40	48.15	56.97	7.11	54.33	43.07
8	Video	16.21	11.03	9.19	48.22	56.69	4.74	58.08	29.26
	Data	11.88	7.32	6.10	49.14	51.35	4.61	47.73	38.82
9	Video	13.45	8.65	7.21	48.81	53.61	3.72	55.38	27.65
10	Video	8.66	4.56	3.80	49.82	43.88	2.87	53.14	33.17
	Data	5.38	2.48	2.07	50.33	38.48	1.07	38.36	19.82
Total BW		175.36	110.2						

Source: own elaboration.

In order to assess whether the Smart Model yields better results than the Hybrid Model, the following hypotheses are proposed:

Throughput Evaluation

$$H_o: \mu_{Thr_MH} \leq \mu_{Thr_MS}$$

$$H_a: \mu_{Thr_MH} > \mu_{Thr_MS}$$

Where μ_{Thr_MH} and μ_{Thr_MS} correspond to the Throughput value obtained when using the Hybrid and Smart Models, respectively. The H_o hypothesis states that the Smart Model yields better results than the Hybrid Model, as it presents an average Throughput value higher than the current method and the H_a establishes the opposite condition.

Delay Evaluation

$$H_o: \mu_{R_MS} \leq \mu_{R_MH}$$

$$H_a: \mu_{R_MS} > \mu_{R_MH}$$

Where μ_{R_MH} and μ_{R_MS} correspond to the average delay obtained when using the Hybrid and Smart Models, respectively. The H_o hypothesis states that the Smart Model yields better results than the Hybrid Model, as it presents a lower average delay than the current method and the H_a establishes the opposite condition.

Efficiency Evaluation

$$H_o: \mu_{\eta_MH} \leq \mu_{\eta_MS}$$

$$H_a: \mu_{\eta_MH} > \mu_{\eta_MS}$$

Where μ_{η_MH} and μ_{η_MS} correspond to the average efficiency when using the Hybrid and Smart Models, respectively. The H_o hypothesis states that the Smart Model yields better results than the Hybrid Model, as it presents an average efficiency higher than the current method and the H_a establishes the opposite condition.

To accept or reject each of the hypotheses proposed, it is necessary to test hypotheses on the difference in means with paired sampling, using the so-called paired t-test. Making use of the miniTap statistical software, and the values recorded in Table 7; the result obtained in each of the tests is consolidated in Table 8:

Table 8. Paired t-test results for each of the established hypotheses

Variable	H_o	Degrees of freedom	Statistical	Acceptance range $\{t: t < T_{(\alpha; n-1)}\}$	It is accepted
Throughput	$\mu_{Thr_MH} \leq \mu_{Thr_MS}$	14	-6,496	$(-\infty, 1.7613)$	Yes
Time delay	$\mu_{R_MS} \leq \mu_{R_MH}$	14	-0.312	$(-\infty, 1.7613)$	Yes
Efficiency	$\mu_{\eta_MH} \leq \mu_{\eta_MS}$	14	-10,898	$(-\infty, 1.7613)$	Yes

Source: own elaboration.

Table 8 shows that the value of the statistic for each of the cases is within the acceptance interval, an aspect for which the H_o is accepted in all three cases. In view of the above, it is concluded that with the Smart Model, which makes use of OFDMA as a mechanism to access the medium supported by the use of the Deegan-Packel power index, a better performance was obtained than the Hybrid Model (supported in CSMA/CA and TDMA), corresponding to the current state of PLC technology under the HomePlug AV standard, taking into account that for the proposed scenario, the MS-HPAV allows to achieve better levels of Throughput, delay and efficiency compared to the MH- HPAV, with 95 % confidence.

Table 9. Comparative chart MS-HPAV vs MH-HPAV.

Variable	Model	Half	% Of difference
Throughput	MS-HPAV	6.12	58.3
	MH-HPAV	3.87	
Time delay	MS-HPAV	49.13	-1.28
	MH-HPAV	49.77	
Efficiency	MS-HPAV	50.44	58.93
	MH-HPAV	31.74	

Source: own elaboration.

On the other hand, Table 9 presents a comparative table of the Throughput, Delay and Efficiency variables for the MS-HPAV and MH-HPAV models, based on the average value and the percentage of difference. V_{MS} and V_{MH} correspond to the value obtained with the Smart Model and the Hybrid Model, respectively. According to the results obtained, the following aspects can be stated:

- The Throughput value achieved by MS-HPAV is 58.3 % higher than MH-HPAV, an aspect that reflects a significant improvement in the transmission processes under the proposed media access strategy.
- Delay levels are reduced by 1.28 % in MS-HPAV versus MH-HPAV, which is quite favorable for transmission of services such as voice and video.
- The MS-HPAV indicates an increase in system efficiency of 58.93 % compared to the current media access mechanism, optimizing resource allocation and offering adequate QoS levels to each of the nodes that are part of the network.

3. CONCLUSIONS

The use of power indices when defining the Scheduler per transmission period in a BPL network can be considered a very useful strategy. The power index represents the degree of influence that the node exerts on the network and according to the assigned value, the channel access order is established. Based on the results obtained, it was observed that the value corresponding to the power index can be estimated based on the weight reflected by the node, which depends on various factors such as: assigned BW, priority level and traffic class. At the end of the process, the CCo orders the power indices in descending order, in order to optimize the allocation of BPL channel resources, in the time and frequency domain for each node. If two nodes have equal power indices, the weight will be considered as a tiebreaker factor.

The process of calculating the power indices for each of the nodes that are part of a BPL network can be considered a problem as the number of nodes increases, because the classic algorithms make use of permutations, raising the levels of computational complexity required. However, a routine was developed in Matlab, which allows estimating the Deegan-Packel power index through the use of generating functions, offering reduced computational and temporal complexity, to be implemented in the future in low-cost embedded systems. The development of this routine in Matlab can be considered a great contribution for future research, since it can be used in other simulation scenarios that require it.

REFERENCES

- [1] J. C. Vesga, J. E. Sierra, and J. A. Barrera, *Indian journal of science and technology IndJST*, vol. 11, no. 8. 2018.
- [2] J. C. Vesga, J. E. Sierra, and J. A. Barrera, "Performance evaluation under an AFR scheme CSMA/CA for HomePlug AV supported in Bianchi's Model," *Indian J. Sci. Technol.*, vol. 11, no. 8, pp. 1–14, Feb. 2018, doi: 10.17485/ijst/2018/v11i8/117511.
- [3] H. A. Latchman, S. Katar, L. Yonge, and S. Gavette, "The HomePlug AV Network Architecture," in *Homeplug AV and IEEE 1901*, Hoboken, NJ, USA: John Wiley & Sons, Inc., 2013, pp. 12–17.
- [4] R. Alvarez, J. Araque, and J. E. Sierra, "A Novel Smart Home Energy Management System: Architecture and Optimization Model," *Indian J. Sci. Technol.*, vol. 10, no. 26, pp. 1–8, Jun. 2017, doi: 10.17485/ijst/2017/v10i26/110684.
- [5] J. C. Vesga Ferreira, G. Granados Acuna, and J. A. Vesga Barrera, "Bandwidth optimization in BPL networks using nucleolus and max-min fairness techniques," *Rev. Ing. Univ. Medellin*, vol. 18, no. 34, pp. 165–180, Jun. 2019, doi: 10.22395/rium.v18n34a10.
- [6] J. C. Vesga, J. E. Sierra, and G. Granados, "Cooperative game theory as a strategy of resource optimization in PLC networks," Accessed: Jul. 06, 2018. [Online]. Available: <http://www.wseas.org/multimedia/journals/communications/2016/a605804-981.pdf>.
- [7] J. C. Vesga, G. Granados, and J. A. Barrera, "Allocation of Medium Access Order Over Power Line Communications (PLC) Supported on Weighted Voting Games," *Indian J. Sci. Technol.*, vol. 10, no. 24, pp. 1–14, Jun. 2017, doi: 10.17485/ijst/2017/v10i24/102667.
- [8] S. Ezzine, F. Abdelkefi, J. P. Cances, V. Meghdadi, and A. Bouallegue, "Joint network coding and OFDMA based MAC-layer in PLC networks," in *18th IEEE International Symposium on Power Line Communications and Its Applications*, Mar. 2014, pp. 311–315, doi: 10.1109/ISPLC.2014.6812363.

- [9] S. Bannister and P. Beckett, "Enhancing powerline communications in the 'Smart Grid' using OFDMA," in *Power Engineering Conference, 2009. AUPEC 2009. Australasian Universities*, Sep. 2009, pp. 1–5.
- [10] D. Zhang, Y. Wang, and J. Lu, "QoS aware relay selection and subcarrier allocation in cooperative OFDMA systems," *IEEE Common. Lett.*, vol. 14, no. 4, pp. 294–296, Apr. 2010, doi: 10.1109/LCOMM.2010.04.092315.
- [11] L. T. Berger, A. Schwager, and J. J. Escudero-Garz as, "Power Line Communications for Smart Grid Applications," *J. Electr. Comput. Eng.*, vol. 2013, pp. 1–16, 2013, doi: 10.1155/2013/712376.
- [12] A. Maga a, "Coalitions formation in cooperative games and games with multiple alternatives. PhD Thesis," Polytechnic University of Catalonia, 1996.
- [13] M. E. Febles, "Power Sharing System in Local Elections," *Numbers*, vol. 39, pp. 13–20, 1999.
- [14] J. C. Vesga, J. E. Sierra, and G. Granados, "Media access over PLC technology using Shapley-Shubik power index," *WSEAS Trans. Math.*, vol. 15, pp. 511–522, 2016.
- [15] J. Garc a, "Complexity and algorithms in cooperative games. PhD Thesis," University of Seville, 2000.
- [16] J. C. Vesga, J. E. Sierra, and G. Granados, "Evaluation of Hybrid (MH) and Smart (MS) models based on game theory for resources allocation in HomeplugAV networks," Accessed: Jul. 06, 2018. [Online]. Available: <http://www.wseas.org/multimedia/journals/communications/2016/a585804-982.pdf>.
- [17] K. Ribnikov, *Combinatorial Analysis*. Moscow: Mir Publisher, 1988.
- [18] L.S. Shapley, "'A value for n-persons games' in Contributions to the Theory of Games II," *Ann. Math. Stud.*, no. 28, pp. 307–317, 1953.
- [19] F. Canete, "User guide for PLC channel generator v. 2," University of Malaga, Malaga-Spain, 2011.
- [20] S. Sancha, F.J. Canete, L. Diez, and JT Entrambasaguas, "A Channel Simulator for Indoor Power-line Communications," presented at *2007 IEEE International Symposium on Power Line Communications and Its Applications*, 2007, pp. 104–109, doi: 10.1109/ISPLC.2007.371106.
- [21] J. C. Vesga, G. Granados, and J. E. Sierra, "The Nucleolus as a Strategy for Resources Optimization in LANs on Power Line Communications," *Indian J. Sci. Technol.*, vol. 9, no. 28, pp. 1–14, 2016, doi: 10.17485/ijst/2016/v9i28/89332.