Cooperative game theory as a strategy of resource optimization in PLC networks

JUAN C. VESGA
Escuela de Ciencias Básicas Tecnología e Ingeniería
Universidad Nacional Abierta y a Distancia
Carrera 27 # 40-43 Bucaramanga
COLOMBIA
juan.vesga@unad.edu.co

JAVIER E. SIERRA
Vicerrector de Calidad
Corporación Universitaria del Caribe CECAR
Ctra. Troncal de Occidente - Km 1 - Via Corozal, Sincelejo
COLOMBIA
javier.sierra@cecar.edu.co

GERARDO GRANADOS
Escuela de Ciencias Básicas Tecnología e Ingeniería
Universidad Nacional Abierta y a Distancia
Carrera 27 # 40-43 Bucaramanga
COLOMBIA
gerardo.granados@unad.edu.co

Abstract: - The communications standard for powerline HomePlug AV (HPAV) uses CSMA/CA and TDMA as media access mechanism, where CSMA/CA is designed for transmitting data packets, while TDMA is used for voice and video packets transmission, in order to provide adequate levels of QoS [1]. However, although HPAV can achieve high rates of transmission, it does not have an adequate mechanism for resources allocating, which greatly affects the performance of the network as the number of users increases because to the fact that only one node can transmit at the same time [2]. The objective of this paper is to evaluate the behavior of four techniques: Shapley Value, Nucleolus, Max-min Fairness and MmQoS; as strategy for resources allocating in a channel PLC, when considering the HAN network as a cooperative game of transferable Utility. To make an efficient comparison of the four methods of optimization, it proposes the use of two HAN network scenarios under conditions of saturation. In each scenario it will be applied the four resource optimization methods and by Dunnett's test it will identify what or what methods, or treatments yield the best results during the evaluation process. Upon completion of the analysis of the results, it is concluded that, for the proposed scenarios the value of Shapley was the technique that made a better allocation of BW, to minimize the difference between the BW requested value and BW value assigned to each node, with 95 % confidence.

Key-Words: - Cooperative game theory, Homeplug AV, Local area network, Max-Min Fairness, MmQoS, Nucleolus, Powerline communications, Resources optimization, Shapley Value

1 Introduction

A network can be considered a scenario in which all nodes have the need to permanently transmit, in a way that generates a dispute for logging in the medium and using channel resources according to the needs, which creates situations of inequity and particular satisfaction of each node. In view of the above, the question is: What must be done to distribute equitably the capacity of a PLC channel among all nodes that are part of the network to optimize the resources allocating according to the requirements established for each node, providing adequate levels of QoS and without affecting the performance of another service and even exceed the maximum capacity permitted by the channel? To solve the problem exposed, it is proposed the use of cooperative game theory as solution strategy.

Game Theory is an area of mathematics proposed by John Von Neumann in 1928, and is intended to evaluate the choices an individual can make in a competitive context of gain or loss, against decisions taken by the other competitors. This competitive scenario is called "Game" and the individuals who are part of this scenario are called "Players" [3]. The use of Game Theory establishes three forms for modeling a real scenario: extensive, strategic and coalition. The first two forms are only applicable to non-cooperative games, where the main interest for each player is to obtain the own benefit, regardless of the other players outcome. The third form (Coalition) applies only to games of cooperative type, which corresponds to a game in which two or more players do not compete against each other, but instead, they work together to achieve the same objective and therefore, they win or lose as a group, increasing the probability of obtaining a higher gain compared to the one obtained individually [4]. In a cooperative game it is not necessary to analyze the strategies of the players as occurs in non-cooperative games; it is enough to know the utility it can obtain each coalition and the payments vector associated with the game result [5].

The use of four techniques for equitable resource allocation is proposed in this article: Shapley value, Nucleolus, Max-min Fairness and MmQoS; which form an integral part of the cooperative game theory and taking into account that a PLC network can be represented as a cooperative game with transferable utility.

The paper begins by presenting the conceptual elements of cooperative games with Transferable Utility (TU). Then each optimization method is detailed in the following order: Shapley Value, Nucleolus, Max-min Fairness and MmQoS. After, the analysis of treatments for the proposed methods it is done and the results obtained in each of the established scenarios are presented. Finally the Dunnett's test is applied to select the best method and ANOVA to assess the degree of influence of each of the treatments.

2 Cooperative games with Transferable Utility (TU)

Definition [5]: A cooperative game TU (with transferable utility) is a pair (N, v), where N = {1,2,3 ... n} is the set of players and $v: 2^N \to \mathbb{R}$ is called "*function feature*" of game, with $v(\emptyset) = 0$. Any non-empty subset of N is called "*coalition*". For each $S \subset N$ coalition is assigned a number $v(S)$ which represents the payment that can assure the players that are part of S, regardless of what other players do. The value of a coalition can be considered as the minimum amount that a coalition can get if all the players who are part of it are associated and play as a team.

On several occasions, it is necessary to divide the value of a good or resource into a set of players fairly inside of the cooperative game theory, considering that in many cases the amount to divide is insufficient to meet the demands of each player. Here it is where the problem known as "Bankruptcy" arises.

Definition [6]: A game of bankruptcy is defined as a triple (N, d, E) where $N = \{1,2, \dots, n\}$ is the creditors set, $d = \{d_1, d_2, \dots, d_n\}$ with $d_i \geq 0, \forall i \in N$ is the vector of claims of creditors and E corresponds to the net value to be distributed among the elements of *N*.

For each bankruptcy problem (N, d, E) a cooperative game can be define(N, v). The set of players N is the same set of creditors or demandant in the bankruptcy problem. The value of the S coalition in the game is defined as the property to be divided among the players that was not claimed by the demandants that do not belong to the coalition S. Where $d(S) = \sum_{i \in S} d_i$ is the sum of the claims of all creditors who are part of the coalition S and $d(N \setminus S) = \sum_{i \in N \setminus S} d_i$ the sum of the claims of all creditors who are not part of the S coalition [7].

Definition [5]: A (N, v) cooperative game is a game of bankruptcy if there is a problem of bankruptcy such that (1):

$$v(S) = max\{0, E - d(N \setminus S)\} \; \forall S \subset N \qquad (1)$$

The value of each coalition $v(S)$ is due to a pessimistic assessment of what it can achieve, where after performing a process of sharing between the applicants who are not in the coalition, the balance is assigned to the S coalition. One of the main problems of cooperative game theory TU is how to distribute the total gain $v(N)$ among all players equally and according to the individual participation of each player. Four techniques are proposed in order to solve the optimization problem represented as a set of transferable utility game: Shapley value, nucleolus, Max-min Fairness and MmQoS. Furthermore, a treatment analysis will be performed to identify which of these techniques have better performance not only in optimization, but also in computational complexity, considering that this process is designed to be implemented in embedded devices of low cost.

2.1 The Shapley Value

Lloyd Shapley analyzed for a long time the cooperative games and in 1953 proposed the concept of value of a game (*N*, *v*) for each player $i \in N$ given by expression (2):

$$\varphi_i(v) = \sum_{S \subseteq N : i \in S} \frac{(s-1)!\,(n-s)!}{n!} [v(S) - v(S - \{i\})] \qquad (2)$$

where $n = |N| \; y \; s = |S|$

This value is known as the Shapley value for player i, which is determined exclusively and a priori, by the characteristic function of the game. The Shapley value seeks to establish a series of payments among players so that certain criteria called "axioms" are fulfilled, previously set (efficiency, symmetry, passive player and additivity) generating as a result a single allocation of resources among players [8]. The Shapley value can be interpreted as the expected marginal contribution of i player or as an average of the marginal contributions $[v(S) - v(S - \{i\})]$ of the player to all non-empty coalitions $S \in 2^N$, whereas the coalition of the player is equiprobable in size $(1 \leq s \leq n)$and that all coalitions of size *S*have the same probability [9].

O'Neill [10] demonstrated that the process of distributing in a bankruptcy problem (N, d, E) through the recursive realization method coincides with the Shapley value. This aspect is very important considering that to distribute *E* among its creditors, it is necessary to establish a set of distribution rules that establish any allocation criteria that follows an ethical and professional reasoning, which in this case will be established by using imputations for the game (N, *v*).

An imputation for a game *(N, v)* corresponds to a rational individual payments vector$\varphi(v) \in \mathbb{R}^n$, on which the process of distributing the maximum amount *v (N)* is done among each of the players. To get a suitable solution it is necessary that the payments vector accomplishes the principle of efficiency [3], wherein (3):

$$\sum_{i \in N} \varphi_i(v) = v(N) \qquad (3)$$

In addition, it must accomplish the so-called principle of rational individuality, which requires that the payment to each player *I* is, at least, the amount that the player can get for itself in the game, i.e. (4):

$$\varphi_i(v) \geq (v\{i\}) \forall i \in N \qquad (4)$$

It is possible to suggest that members of each coalition receive a full payment higher or equal to the value of this coalition, which indicates that payments will be coalitional reasonable. A payment vector $\varphi(v) \in \mathbb{R}^n$ is said to be "rational group" if (5):

$$\sum_{i \in S} \varphi_i(v) \geq v(S) \quad \forall S \subseteq N \qquad (5)$$

By time that the imputations must accomplish with the principle of rationality for all coalitions, it is gotten the concept introduced by Guilles which is called *Core* [11]. The *core* $C(v)$ of a game (N, v) is defined as the set of imputations that have group rational property. The expression that defines the *core* of a game is (6):

$$C(v) = \left\{ \varphi(v) \in \mathbb{R}^n \mid \sum_{i \in S} \varphi_i(v) \geq v(S) \quad \forall S \subseteq N, \sum_{i \in N} \varphi_i(v) = v(N) \right\} \qquad (6)$$

Shapley [12] introduced the concept of balanced and balanced game coalitions to establish the conditions that determine if a game has or does not have an empty *Core*. Subsequently it showed that a game (N, v) is balanced if the *core* is not empty $(C(v) \neq \emptyset)$.

The direct calculation of the Shapley value has a temporal complexity $O(n2^n)$ [13], so it is important to seek the possibility of implementing algorithms that present a lower temporary quota. Among the most commonly used algorithms are Hart's and Max-Colell's who proposed an algorithm to calculate the Shapley value which requires a time of $O(n2^n)$ and the algorithm called "Harsanyi dividend" which presents a time complexity of $O(3^n)$ [13].

The routine developed in Matlab for calculating each of the values of transferable utility by the Bankruptcy game is presented in the appendix A.

To establish the resources allocating to the proposed game and in order to calculate the weight vector $\varphi(v)$there will beused the Shapley value. In [14] presents the methodology for calculating each of the values that are part of Shapley Array, according to the number of players, the contribution from each of the coalitions and the probability P (j); values needed to calculate the Shapley value $\varphi_i(v) = \sum_{j \in N} P(j)\, v_{i,j}$ of each player. The routine implemented in Matlab to calculate the Shapley value φ_k (v) ∀ k∈N is presented in the appendix B.

2.2 The Nucleolus

The *Core* is a solution concept that has a major challenge because it can sometimes be a very large set and other times can be an empty set. In view of the above, the concept of "*Nucleolus*" arises, which proposes a solution where the combination of imputations is not empty. The *Nucleolus* is able to overcome the weaknesses present in the *Core*, and delivers as a result a non-empty and unique set. In addition, the nucleolus is part of the Nucleus when it is not empty.

Definition: It is a game (N, v) with a payment distribution $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ efficient among players, i.e. (8):

$$\sum_{i=1}^{n} x_i = v(N) \quad (8)$$

Then, the excess of a coalition *S* with respect to the distribution of payments *x*, is the difference between the value of the coalition *S* and what the coalition receives for the distribution *x*, i.e. (9):

$$e(S, x) = v(S) - x(S) = v(S) - \sum_{i \in S} x_i \quad (9)$$

$e(S, x)$ is the measure of the satisfaction degree of the coalition *S* with the distribution *x*. The higher is $e(S, x)$, the greater is the degree of dissatisfaction. A vector $\theta(x)$ is constructed for each vector of distribution of payments *x* so that excesses are sorted from highest to lowest, in relation to the coalition's order.

Definition: For each component $x \in I(N, v)$, the excesses vector is defined as the vector $\theta(x)$, with2^n components (10):

$$\theta(x) = \left(e(S, x)\right)_{S \in N} = (\theta_1(x), \theta_2(x), \ldots, \theta_{2^n}(x)) \quad (10)$$

Where ,$\theta_k(x) \geq \theta_{k+1}(x) \ \ \forall k = 1,2, \ldots, 2^n - 1$

Given two excesses vectors *x* and *y* which are compared in lexicographical order, item by item in order to identify which of them has a smaller difference or lesser degree of dissatisfaction. The comparison process begins by evaluating the condition of inequality among the first elements of each excesses vector $(\theta_1(x) < \theta_1(y))$. If the condition is met, it can be said that$\theta(x) <_L \theta(y)$, otherwise the process is repeated but with the second element of each vector and so on until any difference is found among the elements. This can be expressed as (11):

$$\max_S\{e(S, x)\} < \max_S\{e(S, y)\} \quad (11)$$

Definition: The *Nucleolus* of a game(N, v), is defined as the set $\mathbb{N}(N, v)$ which can be expressed as follows (12):

$$\mathbb{N}(N, v) = \{x \in I(N, v);\ \theta(x) \leq_L \theta(y), \forall \mathrm{y} \in\ I(N, v)\} \quad (12)$$

Therefore we can say that the *Nucleolus* contains those distributions of payments that are imputations and for which the greatest degrees of dissatisfaction is minimized. Which is a sufficient condition for the existence and uniqueness of the *Nucleolus*, is that (13):

$$\sum_{i=1}^{n} v(\{i\}) \leq v(N) \quad (13)$$

To calculate the *Nucleolus*$x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$of a game *(N, v)*, it is necessary to solve the following linear programming problem (14):

$$\begin{gathered} \min \gamma \\ v(S) - \sum_{i \in S} x_i \leq \gamma \ \ , S \in N, S \neq \emptyset, S \neq N \quad (14) \\ x \in I(N, v) \end{gathered}$$

Where γ is the minimum possible value for the underlying problem, which is reached at a point *x*. Then you can say that *x* is the *Nucleolus*.

The proposed methodology for calculating the *Nucleolus*$x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ proceeds as follows:

Step 1: Set the values of $v(S)$. Taking into account that the total bandwidth demand (BW) exceeds the total bandwidth available; the scenario proposed by the PLC network can be considered as a bankruptcy game. The value for *v (S)* can be calculated by (15):

$$v(S) = max\left\{0, E - \sum_{i \in N-S} d_i\right\} \forall S \subset N \quad (15)$$

Step 2: The Nucleolus $x = (x_1, x_2, x_3, x_4)$ due that it is an imputation; it must meet the following restrictions (16):

$$\begin{gathered} x_i \geq v(\{i\}) \\ \sum_{i=1}^{n} x_i = v(N) \quad (16) \\ i = 1,2,3,4 \end{gathered}$$

Step 3: Finally, it is possible to pose as a *minimax* problem to calculate the *Nucleolus* as it follows (17):

$$\min_{x_1, x_2, x_3, x_4} \{\max\{e_k(S, x)\}\} \quad (17)$$

Hold to:

$$x_i \geq v(\{i\})$$

$$\sum_{i=1}^{n} x_i = v(N)$$

Step 4: The Optimization Toolbox [15] was used to solve the optimization problem, including in Matlab, which allows to make use of various optimization methods. The values for each of the parameters (objective function, the constraints and the starting iteration point of array form) for the proposed problem.

Where,

F: Vector of coefficients of the objective function.
A, b: Correspond to the inequality constraints, where A is the coefficient array and *b*is the vector results for each of the inequalities ($Ax \leq b$).
A_{eq},b_{eq}: Correspond to equality constraints, where$A_{eq}$is the coefficient array and $b_{eq}$is the vector of results for each of the equations ($A_{eq}x = b_{eq}$).
$x0$: Starting point for iteration.

Then the routine developed in Matlab that integrates each of the steps mentioned above is presented, to facilitate the processes of resources allocating using the *Nucleolus*, and can be used in future research (see appendix C).

2.3 MAX-MIN Fairness Algorithm (MMF)

One of the algorithms related to resource allocation in a “fair" way that has been used in various research projects is the algorithm called "*Max Min Fairness*" (MMF), which comes from the theory of cooperative games. In [16][17] there are some of the works that make use of this algorithm as an optimization strategy [18][19][20] are part of the first research works supported in the theory of games that make lexicographic order processes and particularly the work of Schmeidler [21] who introduced the notion of lexicographic order when he defined the *Nucleolus* of a characteristic function of a game, which can be defined as it follows:

Definition [22][23]: In a TU game(N, v), with a characteristic function $v: 2^N \to \mathbb{R}_+$which associates a value $v(S) \geq 0$ for each coalition $S \subseteq N$; It is wanted to find a fair or equitative distribution of the total gain $v(N)$

Among all the players $i = 1,2, \ldots, n$ a payment vector $\varphi \in \mathbb{R}^n$ is defined so that $\varphi_i \geq 0$. For each coalition$S \subseteq N$, is set a $\varphi(S) = \sum_{i \in S} \varphi_i$ y $\sum_{i \in N} \varphi_i = v(N)$. Finally, for any payment vectorφ, there is a vector γ whose components take the values $v(S) - \varphi(S)$, for all $S \subseteq N$. In this way, the vector φ is defined as the *Nucleolus* of the game and γ is the Max-Min Fairness that can be calculated according to (18) and (19):

$$v(S) - \sum_{i \in S} \varphi_i = g_s \leq \gamma_s \quad \forall S \subset \mathrm{N}, \varphi \geq 0 \quad (18)$$

$$g_N = v(N) \quad (19)$$

MMF is an iterative technique that allows to optimally distributing resources among all the elements which are part of the system [24] and can be used in various network scenarios, including a PLC network; where each node can establish various sessions produced by diverse traffic sources and each link can be shared with other existing sessions.

The MMF algorithm is as follows:

a. A vector V is generated, in which the values of BW requested by each node *i* class *r* (BW'_{ir}) according to the service requested ($BW'_{ir} \geq 0$) are registered and the number of traffic sources or number of players (N_j) present in the PLC network are determined.
b. Now the vector V is sorted in ascending order.
c. An initial reference value (BW_{ref}), is calculated by using the expression: $BW_{ref} = \frac{\sum_{k=1}^{N_j} V(k)}{N_j}$. In the first iteration ($i = 1$), if $V(i|_{i=1}) > BW_{ref} \to V(1) = BW_{ref}$,, otherwise the value that presents before making the comparison process is remained, where the value of BW requested by the node would be assigned.
d. A new estimation process of BW_{ref} is performed, taking into account the number of elements that are part of the vector V and which has not been subjected to the comparison process. That is to say:

$$BW_{ref} = \frac{\sum_{k=i}^{N_j} V(k)}{N_j - i} \quad (20)$$

e. In the next iteration ($i = i + 1$), si $V(i) > BW_{ref} \to V(i) = BW_{ref}$, otherwise $V(i)$ keeps the value that represents before the comparison process and the item (d) is repeated. This process is repeated until all the elements of the V vector are valued.

f. The bandwidth is assigned for each node i and class r (BW_{ir}) according to the registered values in the resulting vector V.

2.4 FAIRNESS ALGORITHM MmQoS

MmQoS corresponds to a proposal of the Max-Min Fairness algorithm adaptation to guarantee optimal levels of QoS to each class r in the node i. In [25] it is deeply described the equity algorithm MmQoS.

The algorithm proposed for MmQoS is the next one:

a. It is generated a vector V, in which the values of bandwidth required by each node i and class r (BW'_{ir}) are registered according to the need of the service $|(BW'_{ir} \geq 0)$ and the number of traffic sources or number of players (N_j) presented in the network PLC are determined.

b. To reach the proper levels of QoS in each one of the traffic classes, it is ordered in an ascendant form the vector V taking into account the priority level and the value of BW required of the class r, in a way that the first elements of the vector are attended with a priority resource allocation. Considering the policies of QoS, the packets of voice and video must count with a higher priority than the one of the Data packet, due that they are services that use the protocol UDP and they are so susceptible to retarded upper levels, which affects the performance.

c. A normalization process of the vector V $(V_N(i) = V(i)/\sum_{k=1}^{N_j} V(k)$ is done in order to obtain a better answer when the resource allocation process is going to be done.

d. The initial reference value is calculated $(BW_{ref} \leq 1$ due to the process of normalization) by the use of the expression (21):

$$BW_{ref} = \frac{1}{N_j} \tag{21}$$

e. In the first iteration $(i = 1)$, si $V_N(i|_{i=1}) > BW_{ref} \rightarrow V_N(1) = BW_{ref}$, otherwise $V_N(1)$ keeps the value that represents before doing the process of comparison.

f. A new estimation process of BW_{ref} is done taking into account the quantity of elements that are part of the vector V_N and that the process of comparison have not been done yet. That means:

$$BW_{ref} = \frac{1 - \sum_{k=1}^{i} V_N(k)}{N_j - i} \tag{22}$$

g. In the next iteration $(i = i + 1)$ si $V_N(i) > BW_{ref} \rightarrow V_N(i) = BW_{ref}$, otherwise $V_N(i)$ keeps the value that presents before doing the process of comparison and the item (f) is repeated. This process is repeated until all the elements of the V_N vector are valued.

h. After the process of resource allocation, the conversion process of the vector V_N must be done with the product values of the algorithm MmQoS. For that $V = BW_{TOTAL} * V_N$, with the purpose of establishing the BW allocated for each source of traffic.

i. Finally, the BW is allocated for each node i and class r (BW_{ir}) according to the registered values of Vvector.

2.5 Multiple Comparisons with the Best Treatment (MCB)

The researcher has to use multiple strategies or treatments to solve a problem in several occasions. Due to that, it is important to identify which one or which treatments offers a better result of the problem, and is there where the need of performing processes of comparison emerge among each one of the treatments to select the most accurate, included, when the best method requires an upper computational complexity it is necessary to use heuristic techniques or alternate methods of solution. This methodology is useful when is time to highlight which of these methods offer good results as the optimal method. Moreover it is necessary to evaluate each one of the treatments and compare with the best one.

The method consist in select the treatment or set of treatments that provide the desirable result. The procedure called *"Multiple comparisons with the best (MCB)"* [26] permits the researcher to classify the treatments in a way that the best population is included in a subset with a level of specific fairness. The expression that allows this process is (23):

$$\mu_i - \max_{j \neq i} \mu_j \ \forall i = 1,2, \dots, t \tag{23}$$

Where $\max_{j \neq i} \mu_j$ is the max medium of the treatments without including a μ_i. Si $\mu_i - \max_{j \neq i} \mu_j > 0$, so the treatment i is the best. Otherwise the treatment i is not the best. The simultaneous confidence intervals (ICS) of the MCB for $\mu_i - \max_{j \neq i} \mu_j$ have the restriction of including the Zero. Having in mind that the two treatments won't present identical averages it is established that if $\mu_i - \max_{j\neq i}\mu_j \geq 0$, the treatment i can be considered as the best or one of the best.

To determine which of the treatments is the one that maximizes in a best way the object of study, the following process is applied:

The difference D_i is calculated among each one of the averages medias $\overline{y}_i$(one for treatment), and the highest average media of the rest $\max_{j\neq i}\overline{y}_j$, as it is presented below (24):

$$D_i = \overline{y}_i - \max_{j\neq i}\overline{y}_j \quad \forall i = 1,2,\dots,t \tag{24}$$

And the quantity M,

$$M = d_{\alpha,k,v}\sqrt{\frac{2S^2}{r}} \tag{25}$$

Where $d_{\alpha,k,v}$ is the statistic table corresponding to the multiple comparisons with the best, by the Dunnett's test [27]. $S^2 and\, r$ Correspond to the experimental variance (Mean Square Error MSE) and the number of repetitions respectively.

To calculate the simultaneous intervals of confidence (SIC) with a 100(1- α)% of confidence level, it is important to take into account (26)(27):

$$\text{Lower Limit } L = \begin{cases} D_i - M & Si\ D_i - M < 0 \\ 0 & \text{in another form} \end{cases} \tag{26}$$

$$\text{Upper Limit } U = \begin{cases} D_i + M & Si\ D_i + M > 0 \\ 0 & \text{in another form} \end{cases} \tag{27}$$

If the interest lies in determining which is the best treatment to minimize the variable object of study, the process is the next one:

- The difference D_i among each one of the averages $\overline{y}_i$(one per treatment) is calculated, and the lower media of the rest $\min_{j\neq i}\overline{y}_j$, as it is presented below:

$$D_i = \overline{y}_i - \min_{j\neq i}\overline{y}_j \quad \forall i = 1,2,\dots,t \tag{28}$$

- The quantity M and the *SIC* are calculated with the equations 26 y 27.

The treatment or treatments in which the zero is included between the limits upper and lower, will be considered as the best one or best treatments [26].

2.6 Treatments Analysis for the optimization methods proposed

In order to compare the four optimization methods proposed (Nucleolus, MMF, MmQoS and Shapley) and identify those that present the best result according to the proposed problem, the use of two scenarios that can be present in a Home Area Network (HAN) are proposed. Each one of the resource optimization methods will be applied on these scenarios then, by using the Dunnett's test, it will be identified which one or which are the methods or treatments that provide the best result during the evaluation process. It is important to mention that in each proposed scenario three channel conditions PLC will be established, with the objective of analyzing the behavior of each one of the optimization methods proposed in relation with the maximum capacity of the PLC channel and according to the traffic requirements in each node.

The tool called "Generator of Channel *PLC (GC_PLC)*" is used to calculate the value of BW_T . That tool is developed by the PhD Francisco Javier Cañete, who belongs to the group PLC of the University of Malaga, Spain. GC_PLC permits estimate the behavior of a PLC channel, according to the parameters associated to a network PLC topology, in a typical, residential environment. Additionally, the tool performs an evaluation process of the channel under the 30MHz band, considering the fact that the network PLC adapters under the HPAV standard operate in this frequency band. The necessary information for the use of the tool GC_PLC is presented in [28].

The tool GC_PLC generated the values 159.72 Mbps, 120.65 Mbps y 83.58Mbps; which correspond to the capacity that can be present in a PLC channel, under a channel condition: excellent, regular (typical) and deficient respectively.

The proposed scenarios are the following:

Scenario 1: Saturated channel status under mono-class traffic conditions

A Home Area Network (HAN) compound by 12 PLC nodes that generates only data traffic is proposed in this scenario. In this scenario the node number twelve acts as the main node or coordinator (CCo) and the required bandwidth for each one of the nodes is established in table 1.

Table 1. BW required by each node the scenario 1

Node	BW Requested [Mbps]
1	7,23
2	19,99
3	3,41
4	1,45
5	3,11
6	17,80

7	8,01
8	9,38
9	3,36
10	17,28
11	14,14
12 (Cco)	105,15
Total	**210,31**

In Table 1 it is observed that the total bandwidth required by the HAN network is of 210,31 Mbps, which is superior to the total capacity of the PLC channel in any of the three channel conditions suggested, establishing a estate of channel PLC saturation.

S***cenario 2: Estate of saturated channel under multiclass traffic conditions.***

A network HAN compound by (8) nodes PLC is proposed in this scenario. Each node is composed by an adapter PLC and a traffic source. Each traffic source can generate more than one traffic class *r* simultaneously (voice, data, video and telemetry) the node 8 will be considered as CCo. Traffic classes and the established codec (only applicable to voice or video) for each node *i* and class *r* that are part of the proposed scenario are presented in Table 2.

Table 2. Trafic class for each node *i* and class *r* scenario 2

Node i	Traffic classes Voice (r=1)	Video (r=2)	Control (r=3)	Data (r=4)
1	G.711		X	X
2	G.723.1		X	X
3		MPEG-4		X
4	G.729		X	X
5		MPEG-2	X	
6				X
7	G.711			X
8(Cco)	X	X	X	X

The required BW for each class of traffic for the proposed scenario is presented in Table 3.

Table 3. BW required by each node *i* and class *r* scenario 2

Node i	Traffic classes Voice [kbps]	Video [Mbps]	Control [kbps]	Data [Mbps]
1	81,6		1,5	3,5
2	23,2		2,2	10,2
3		18,4		8,4
4	35,8		2,7	12,7
5		15,38	1,8	
6				16,5
7	102,6			11,6
8(Cco)	243,2	33,78	8,2	62,9

According to the registered values in table 3 the total required BW by the HAN network is of 193.86 Mbps, which is superior to the PLC channel total capacity in any of the three channel conditions suggested before, establishing a channel PLC saturation state.

For the proposed scenarios, it is applied the criteria that the node CCo will be directly connected to the router that allows the exit of the HAN network to internet and on which most of the traffic part will circulate, under the supposed that the most part of the services will depend on internet.

3 Optimization process results in each one of the proposed scenarios

In the proposed scenarios it is wanted to allocate in an equitable way the available BW in the PLC channel$(BW_{Total} = v(N))$ among all of the players $j = 1,2,\dots,n$, by the use of the four methods proposed (Nucleolus, MMF, MmQoS and Shapley). Each player j obeys to a traffic source associated to the pair (node, class) that requires the access to the channel. The case of a bankruptcy game will be considered in the scenarios to establish each one of the imputations of the game.

Tables 4 and 5 show the BW assigned to each one of the players as the result of resources allocating process by the use of the four optimization methods proposed supported in the cooperative game theory and according to each one of the states and conditions of the PLC channel.

Table 4. BW allocated to each player for the scenario 1: Saturated channel-Mono-class

Channel State	Player	BW Requested	Nucleolus	MMF	MmQoS	Shapley
EXCELLENT	**1**	7,23	3,20	7,23	5,491	4,693
	2	19,99	12,82	19,99	15,181	13,033
	3	3,41	2,01	3,41	2,59	2,21
	4	1,45	1,30	1,45	1,101	0,939
	5	3,11	2,03	3,11	2,362	2,016
	6	17,80	10,95	17,8	13,518	11,598

	7	8,01	3,77	8,01	6,083	5,2
	8	9,38	4,57	9,38	7,124	6,093
	9	3,36	2,01	3,36	2,552	2,178
	10	17,28	10,51	17,28	13,123	11,258
	11	14,14	7,96	14,14	10,739	9,204
	12 Cco	105,15	98,58	54,56	79,856	91,298
BW Total		**210,31**	**159,72**	**159,72**	**159,72**	**159,72**
REGULAR	**1**	7,23	3,65	7,23	4,148	3,785
	2	19,99	9,73	17,76	11,468	10,297
	3	3,41	1,73	3,41	1,956	1,79
	4	1,45	0,81	1,45	0,832	0,762
	5	3,11	1,73	3,11	1,784	1,632
	6	17,80	8,70	17,76	10,211	9,202
	7	8,01	3,97	8,01	4,595	4,193
	8	9,38	4,67	9,38	5,381	4,907
	9	3,36	1,72	3,36	1,928	1,763
	10	17,28	8,46	17,28	9,913	8,942
	11	14,14	6,98	14,14	8,112	7,36
	12 Cco	105,15	68,52	17,76	60,322	66,017
Total BW		**210,31**	**120,65**	**120,65**	**120,65**	**120,65**
DEFICIENT	**1**	7,23	3,64	7,23	2,873	3,361
	2	19,99	9,55	9,526	7,944	9,439
	3	3,41	1,86	3,41	1,355	1,577
	4	1,45	1,02	1,45	0,576	0,669
	5	3,11	1,73	3,11	1,236	1,438
	6	17,80	8,50	9,526	7,074	8,371
	7	8,01	3,98	8,01	3,183	3,728
	8	9,38	4,62	9,38	3,728	4,372
	9	3,36	1,84	3,36	1,335	1,554
	10	17,28	8,25	9,526	6,867	8,12
	11	14,14	6,80	9,526	5,619	6,613
	12 Cco	105,15	31,79	9,526	41,788	34,339
BW Total		**210,31**	**83,58**	**83,58**	**83,58**	**83,58**

Table 5. BW allocated to each player for the scenario 2: Saturated channel – Multi-class

Channel State	Player	CoS	BW Requested	Nucleolus	MMF	MmQoS	Shapley
EXCELLENT	**1**	**Voice**	0,0816	2,1190	0,0816	0,0670	0,0580
	2		0,0232	2,0610	0,0232	0,0190	0,0167
	3		0,0358	2,0730	0,0358	0,0290	0,0259
	4		0,1026	2,1400	0,1026	0,0850	0,0741
	5		0,2432	2,2810	0,2432	0,2000	0,2233
	6	**Video**	18,4000	20,437	18,400	15,159	13,735
	7		15,3800	17,417	15,380	12,671	11,16
	8		33,7800	35,817	31,269	27,831	31,015
	9	**Control**	0,0015	2,0390	0,0015	0,0010	0,0011
	10		0,0022	0,0020	0,0022	0,0020	0,0016
	11		0,0027	0,0020	0,0027	0,0020	0,0020
	12		0,0018	0,0010	0,0018	0,0010	0,0013
	13		0,0082	0,0040	0,0082	0,0070	0,0075
	14	**Data**	3,5000	2,1340	3,5000	2,8840	2,4870
	15		10,2000	5,6690	10,200	8,4040	7,3497
	16		8,4000	4,6040	8,4000	6,9210	6,2702
	17		12,7000	7,1440	12,700	10,463	9,1822
	18		16,5000	9,2990	16,500	13,594	11,982
	19		11,6000	6,5010	11,600	9,5570	8,3769
	20		62,9000	37,974	31,269	51,822	57,752
BW Total			**193,8628**	**159,72**	**159,72**	**159,72**	**159,72**
Channel State	**Player**		**BW Solicitado**	**Nucleolus**	**MMF**	**MmQoS**	**Shapley**
REGULAR	**1**	**Voice**	0,0816	0,0620	0,0816	0,0510	0,0449
	2		0,0232	0,0150	0,0232	0,0140	0,0128
	3		0,0358	0,0240	0,0358	0,0220	0,0197
	4		0,1026	0,0790	0,1026	0,0640	0,0566
	5		0,2432	0,1380	0,2432	0,1510	0,1699
	6	**Video**	18,4000	17,464	14,749	11,451	9,8756

	7		15,3800	14,434	14,749	9,5720	8,4300
	8		33,7800	33,037	14,749	21,023	23,597
	9	**Control**	0,0015	0,0001	0,0015	0,0010	0,0008
	10		0,0022	0,0010	0,0022	0,0010	0,0012
	11		0,0027	0,0020	0,0027	0,0020	0,0015
	12		0,0018	0,0010	0,0018	0,0010	0,0010
	13		0,0082	0,0040	0,0082	0,0050	0,0057
	14	**Data**	3,5000	1,6260	3,5000	2,1780	1,9253
	15		10,2000	5,0030	10,200	6,3480	5,6429
	16		8,4000	4,1450	8,4000	5,2280	4,5084
	17		12,7000	6,1960	12,700	7,9040	6,9928
	18		16,5000	8,0490	14,749	10,269	9,0280
	19		11,6000	5,6690	11,600	7,2190	6,3954
	20		62,9000	24,701	14,749	39,146	43,939
BW Total			**193,8628**	**120,65**	**120,65**	**120,65**	**120,65**
Channel State	**Player**		**BW Solicitado**	**Nucleolus**	**MMF**	**MmQoS**	**Shapley**
DEFICIENT	**1**	**Voice**	0,0816	0,0380	0,0816	0,0350	0,0386
	2		0,0232	0,0070	0,0232	0,0100	0,0110
	3		0,0358	0,0070	0,0358	0,0150	0,0170
	4		0,1026	0,0650	0,1026	0,0440	0,0488
	5		0,2432	0,1380	0,2432	0,1050	0,0930
	6	**Video**	18,4000	10,743	8,8972	7,9330	8,9741
	7		15,3800	8,8250	8,8972	6,6310	7,3611
	8		33,7800	25,381	8,8972	14,564	12,913
	9	**Control**	0,0015	0,0001	0,0015	0,0010	0,0007
	10		0,0022	0,0014	0,0022	0,0010	0,0010
	11		0,0027	0,0014	0,0027	0,0010	0,0013
	12		0,0018	0,0016	0,0018	0,0010	0,0009
	13		0,0082	0,0028	0,0082	0,0040	0,0031
	14	**Data**	3,5000	1,6931	3,5000	1,5090	1,6557
	15		10,2000	4,7777	8,8972	4,3980	4,8459
	16		8,4000	4,0782	8,4000	3,6210	4,0969
	17		12,7000	5,6813	8,8972	5,4750	6,0367
	18		16,5000	7,0127	8,8972	7,1140	7,9210
	19		11,6000	5,2912	8,8972	5,0010	5,5132
	20		62,9000	9,8348	8,8972	27,118	24,045
BW Total			**193,8628**	**83,58**	**83,58**	**83,58**	**83,58**

After the optimization process through the four methods proposed the following aspects are highlighted:

- It is observed that the corresponding values of the BW allocated to each one of the players as a result of the proposed methods are very similar. Although, an ANOVA will be raised to identify if there are important differences among the methods or treatments proposed. Moreover, it is observed that some players got a higher allocation of resources than others in each one of the methods used, which could be favorable when analyzing aspects as QoS, especially in those players that required voice or video traffic.
- In the four scenarios, the total sum of the BW allocated to each player is the same as the total available BW in the PLC channel, according to the three channel conditions (Excellent, regular and Deficient) and to the established imputations in each model of optimization.
- Under channel conditions in a saturated state, it is seen that the BW allocated to each player presents a near value to the BW required, which reflects resources allocating in a fair and equitable way with the requirements of each node. Also, suitable levels of QoS are offered to each one of the traffic classes that circulate in the HAN network.

3.1 Result analysis by the procedure MCB or Dunnett's Test

To perform a comparison and evaluation process of the four optimization methods proposed, it was performed a process of random sampling taking as population the results that are registered in Tables 4 and 5. To estimate the size of the sample in a finite population the following expression is used (29):

$$n = \frac{Z_\alpha^2 Npq}{i^2(N-1) + Z_\alpha^2 pq} \tag{29}$$

Where:
n: Size of the sample.
Z_{α}: Value corresponding to the Gauss distribution with an error α. The Z_{α} values for α =0,05 and α =0,01 is of 1,96 and 2,58 respectively.
p: Expected prevalence of parameter evaluator. When this value is unknown it can be considered as 0,5 the value of $q = 1 - p$.
i: Estimated error during the sampling process.

3.2 MCB evaluation in function of the PLC channel conditions

In the first analysis it is evaluated the behavior of each one of the optimization methods proposed for the three established conditions of a PLC channel (excellent, regular and deficient). It was performed a simple random sampling on a finite population according to the registered values of tables 4 and 5.

Tables 6, 7 and 8 show the corresponding parameters to the procedure called *"Multiple comparisons with the best (MCB)"* to perform a process of comparison among the four optimization methods proposed and identify which one or which can be considered as the best one or the best of them and to perform the allocation process of the bandwidth in the inner of a HAN network.

Table 6. MCB about the optimization methods proposed under excellent channel conditions

Method	Media (μ_i)	Max ($\mu_{i\neq j}$)	M	D_i	$D_i - M$ (L)	$D_i + M$ (U)
Nucleolus	12,08	11,66	8,29	0,41	-7,88	8,70
MMF	11,66	12,08	8,29	-0,41	-8,70	7,88
MmQoS	11,45	12,08	8,29	-0,63	-8,92	7,66
Shapley	11,45	12,08	8,29	-0,63	-8,92	7,66

Table 7. Dunnett's test about the optimization methods proposed under regular channel conditions

Method	Media (μ_i)	Max ($\mu_{i\neq j}$)	M	D_i	$D_i - M$ (L)	$D_i + M$ (U)
Nucleolus	9,12	8,73	5,98	0,40	-5,58	6,38
MMF	8,73	9,12	5,98	-0,40	-6,38	5,58
MmQoS	8,65	9,12	5,98	-0,48	-6,46	5,50
Shapley	8,64	9,12	5,98	-0,48	-6,46	5,50

Table 8. Dunnett's test about the optimization methods proposed under deficient channel conditions

Método	Media (μ_i)	Max ($\mu_{i\neq j}$)	M	D_i	$D_i - M$ (L)	$D_i + M$ (U)
Nucleolus	6,33	6,00	3,79	0,34	-3,45	4,13
MMF	5,99	6,33	3,79	-0,34	-4,14	3,45
MmQoS	5,99	6,33	3,79	-0,34	-4,13	3,45
Shapley	6,00	6,33	3,79	-0,34	-4,13	3,45

Considering the registered results in tables 6, 7 and 8, it is said that when the Dunnett's test is performed any of the four methods can be considered as one of the best, due that the simultaneous intervals of confidence including the zero between the upper and the lower limits. Although, when evaluating the difference D_i it is observed that the corresponding optimization method of the *Nucleolus* has the highest average and additionally is the only one of the four methods that meets the $\mu_i - \max_{j\neq i} \mu_j \geq 0$ condition, indicating that the Nucleolus can be regarded as the best of the four optimization methods based on the channel condition.

In order to identify which of the four methods of optimization makes the best fit according to the needs of each one of the players, considering that under a state of saturation the total assigned resource will be lesser than the total required. For this, it was calculated the difference between the required value and the assigned value in each sample in order to identify which of the four methods presents the minimum difference and therefore the best fits. To achieve the objective it is calculated the D_i difference between each of the average $\bar{y}_i$ (one per treatment), and the lower average of the remaining $\min_{j\neq i} \bar{y}_j$ as it presented below (30):

$$D_i = \bar{y}_i - \min_{j\neq i} \bar{y}_j \quad \forall i = 1,2,\dots,t \qquad (30)$$

In this case the values of M and ICS are calculated using equations (26) and (27).

Table 9. Dunnett's Test about optimization methods proposed in a PLC channel under a saturation state

Method	Media (μ_i)	Min ($\mu_{i\neq j}$)	M	D_i	$D_i - M$ (L)	$D_i + M$ (U)
Nucleolus	4,66	4,45	3,28	0,21	-3,07	3,49
MMF	4,61	4,45	3,28	0,15	-3,13	3,43
MmQoS	4,66	4,45	3,28	0,21	-3,07	3,49
Shapley	4,45	4,61	3,28	-0,15	-3,43	3,13

According to the reported results of Table 9, and when applying the Dunnett's test again, any of the four methods can be considered as one of the best, because the simultaneous confidence intervals include zero between the upper and lower limits. However, when assessing the difference D_i it can be seen that the corresponding optimization method to the Shapley value presents the lower average and additionally, it is the only one of the four methods that meet the condition $\mu_i - \max_{j \neq i} \mu_j \leq 0$. Therefore the Shapley value can be regarded as the best of the four methods of optimization on a HAN network under a PLC channel in a state of saturation.

It is very important to mention that based on the results obtained during the treatment process analysis, supported on the MCB process, any of the four suggested methods performs an appropriate resource allocation process. However, one aspect that could make a very clear distinction between a method and another one when it is implemented it would be focused on computational and time complexity of the required algorithms in each optimization method, which can become high or reduced when the number of players increases or decreases respectively. In this situation, the optimization methods such as the Nucleolus and the Shapley value show a considerable increase in the time complexity as the number of players increases due to the combinational process immersed in each algorithm. In view of the above, the optimization methods as MMF and MmQoS could be considered as the solution of the problem proposed, considering that these methods have a much lower computational and temporal complexity than the provided by the Shapley and Nucleolus methods, which fosters their implementation in low-cost embedded systems.

3.3 Treatments analysis using ANOVA

The Analysis of Variance (ANOVA) is a benchmark to assess the degree of influence among a finite number of treatments. From the viewpoint of the experimental design it is suggested that the data that will be evaluated can be organized according to the structure called "Design Array", which is presented in Table 10:

Table 10. Array of block design and treatments for sample space

	Treatments			
Blocks	1	2	⋯	t
1	y_{111}	y_{211}	⋯	y_{t11}
	y_{112}	y_{212}	⋯	y_{t12}
	⋮	⋮	⋯	⋮
	y_{11s}	y_{21s}	⋯	y_{t1s}
Total 1	$y_{11.}$	$y_{21.}$	⋯	$y_{t1.}$
2	y_{121}	y_{221}	⋯	y_{t21}
	y_{122}	y_{222}	⋯	y_{t22}
	⋮	⋮	⋯	⋮
	y_{12s}	y_{22s}	⋯	y_{t2s}
Total 2	$y_{12.}$	$y_{21.}$	⋯	$y_{t2.}$
⋮	⋮	⋮	⋯	⋮
b	y_{1b1}	y_{2b1}	⋯	y_{tb1}
	y_{1b2}	y_{2b2}	⋯	y_{tb2}
	⋮	⋮	⋯	⋮
	y_{1bs}	y_{2bs}	⋯	y_{tbs}
Total b	$y_{1b.}$	$y_{b1.}$	⋯	$y_{tb.}$

Where *t*, *b* and s are the number of treatments, blocks and block samples respectively.Based on the recorded values, it will be calculated each of the parameters that are part of the ANOVA structure (31-35):

$$SCBlock = \sum_{j=1}^{3} \frac{y_{.j.}^2}{ts} - \frac{y_{...}^2}{tbs} \qquad (31)$$

$$SCBlock = \frac{1}{(4*19)}[121.95^2 + 268.95^2 + 406.08^2] - \frac{796.97^2}{4*3*19}$$

$$SCBlock = 531.33$$

$$SCTrat = \sum_{i=1}^{4} \frac{y_{i..}^2}{bs} - \frac{y_{...}^2}{tbs} \qquad (32)$$

$$SCTrat = \frac{1}{(3*19)}[192.5^2 + 208.28^2 + 199.12^2 + 197.07^2] - \frac{796.97^2}{4*3*19}$$

$$SCTrat = 2.31$$

$$SCEM = \sum_{ijk} y_{ijk}^2 - \sum_{ij} \frac{y_{ij.}^2}{s} = 18944.41 - \frac{1}{(19)}[63081.42] \qquad (33)$$

$$SCEM = 15624.33$$

$$SCEE = \sum_{ij} \frac{y_{ij.}^2}{s} - \sum_{i=1}^{t} \frac{y_{i..}^2}{bs} - \sum_{j=1}^{b} \frac{y_{.j.}^2}{ts} + \frac{y_{...}^2}{tbs} \qquad (34)$$

$$SCEE = 3320.07 - 2788.13 - 3317.14 + 2785.81$$

$$SCEE = 0{,}62$$

$$SCTotal = \sum_{ijk} y_{ijk}^2 - \frac{y_{...}^2}{tbs} = 18944.41 - 2785.81 \qquad (35)$$

$$SCTotal = 16158.6$$

Tables 11 and 12 show the structure of an ANOVA for a treatment analysis per blocks and the corresponding values for the proposed scenario.

Table 11. ANOVA Structure analysis for a treatment analysis per blocks

C of V	gl	SC	CM = SC / gl	F
PLC Channel	b – 1	SCBlock	CMBlock	
Treatments	t – 1	SCTreat	CMTreat	CMTreat /CMEE
EE	(b – 1)(t – 1)	SCEE	CMEE	CMEE /CMEM
EM	tb(s – 1)	SCEM	CMEM	
Total	tbs – 1	SCTotal		

The proposed hypotheses for the two relevant factors are:

Factor 1: Treatments
H_o:: No significant differences among treatments
H_A: There are significant differences among treatments

Factor 2: Variance of the experimental error

$$H_o: \sigma_E^2 = 0$$
$$H_A: \sigma_E^2 \neq 0$$

By replacing each of the values, the result of ANOVA is:

Table 12. Results of ANOVA for an analysis of treatments per blocks for the proposed scenario.

C de V	gl	SC	CM = SC / gl	F
PLC Channel	2	531.33	265.66	
Treatments	3	2.31	0.77	7.476
EE	6	0.62	0.103	1.42E-3
EM	216	15624.33	72.335	
Total	227	16158.6	71.18	

Considering the obtained results in the ANOVA, it can be mentioned the following aspects:

- The estimated value F related to the effect among treatments (F_t) reached a value of 7476, that is greater than the critical value of Fisher $F(t-1; (t-1)(b-1); \alpha) = F(3; 6; 0.05) = 4.76$; for that reason the null hypothesis (H_o) is rejected, and significant differences among treatments are evident to calculate a given bandwidth, according to traffic requirements and channel conditions in a state of saturation with 95% of confidence.
- It is noted that the condition of the PLC channel (excellent, regular or deficient) differs significantly, because the average square among blocks has a very high value compared to the experimental error, aspect by which it is possible to conclude that the channel condition can be considered as an important factor when allocating the BW to each node in a PLC network.
- The estimated value of F related to the experimental error variance (F_v) reached a value of 1.42E-3, which is below the critical value of Fisher $F\big((t-1)(b-1); tb(s-1); \alpha\big) = F(6; 216; 0.05) = 2.1$; aspect by which the null hypothesis is accepted, concluding that the error variance is zero, with a confidence level of 95%.

4 Conclusions

In view of the need to perform an equitable distribution of resources according to the service demand among the nodes that are part of a PLC network, it was proposed the use of Cooperative Game Theory with Transferable Utility as optimization strategy and to offer suitable QoS levels in each of the nodes. Four techniques were considered: Shapley Value, Nucleolus, Max-min Fairness and MmQoS. Based on the results it was evident that game theory can be considered as a strategy for optimizing resources in a HAN on Power Line Communications, considering that the four proposed techniques generated appropriate values for resource allocation in each node. However, for the proposed scenarios the Shapley Value was the technique that performed a better allocation of BW, by minimizing the difference between the BW assigned value and the BW requested value to each node with 95% of confidence.

In order to evaluate the four resource optimization methods proposed (Shapley value, Nucleolus, Max-min Fairness and MmQoS) and identify which of them generated the best result (by Dunnett's test), two scenarios were considered under mono-class and multi-class traffic conditions, in a channel state of saturation and non-saturation and with channel capacities of 159.72 Mbps, 120.65 Mbps and 83.58Mbps for excellent, regular (typical) and deficient channel conditions. With the results and after the Dunnett's test in each of the proposed scenarios, any of the four methods can be considered as the best, by including the zero between the upper and the lower limits of the confidence intervals. However, in assessing the $D_i$difference in terms of channel capacity, it was observed that the best of the four optimization methods under channel conditions: excellent, regular or deficient; regardless of the state of saturation of the channel, it was the *Nucleolus*, because it is the only one of the four methods that meet the condition $\mu_i - \max_{j\neq i}\mu_j \geq 0$. In addition, the Shapley value can be regarded as the best of the four optimization methods when the network is in a state of saturation, according to the results obtained and given that this optimization method was the most adjusted to the demands of the service in each node, introducing the smallest average difference between the required value by the service and the assigned value, being the only one of the four methods that meet the $\mu_i - \max_{j\neq i}\mu_j \leq 0$ condition.

A critical aspect when implementing any of the associated algorithms with the Cooperative Game Theory is related to the temporal complexity. In this situation, it was proposed the MmQoS method which could be considered as a solution to the proposed problem, considering that this method not only has a much lower computational and time complexity than the offered by the Shapley and Nucleolus methods, but also the results obtained with the MMF and MmQoS methods compared to the other two methods were quite good, as evidenced by Dunnett's test, which favors its implementation in embedded systems of low cost.

Appendix A

Routine developed in Matlab for calculating each of the values of transferable utility by the Bankruptcy game:

```
%  Important variables that are part of the routine:
%  Nj: Number of Players
%  M_Coalitions: Array of possible coalitions
%  V_Coalition: Value of transferable utility for each coalition

% Routine to calculate all possible combinations
Z = 1: 1: Nj;   % Create a vector with consecutive numbers
                  from 1 to Nj

% Routine to set the number of possible coalitions
n_coal = 0;           % Initial number of coalitions
for  i=1:Nj
     n_coal=n_coal+nchoosek(Nj,i);
            % Number of possible coalitions of Nj
              elements grouped in groups of i elements
end
M_Coalitions=zeros(n_coal,Nj);  % Initializes the
                                  array of coalitions
c=0;
for i=1:Nj              % Calculates the weight of each coalition
   S=nchoosek(Z,i);   % Number of possible coalitions of Z
elements
   nZ=length(S(:,1)); % Number of elements of the vector S
   for j=1:nZ
       c=c+1;          % Increases  counter c
       Suma_d=0;     % Initial value of the sum
       for k=1:i
           Sum_d=Sum_d+V(S(j,k)); % calculates the value of
                                   % coalition and the value
           M_Coalitions(c,k)=S(j,k);  % is recorded in the Array
       end

        %  Procedure to estimate the value of TU considering
             a Bankruptcy game
       Suma_dT=BW_T-(Total_V-Suma_d);
       VAux=[0 Suma_dT];
       V_Coalition(c)=max(VAux);    % TU value for coalition
   end
end
```

Appendix B

Routine implemented in Matlab to calculate the Shapley value φ_k (v) ∀ k∈N:

```
% Important variables that are part of the routine:
% Mk_Coalitions: Coalitions array associated with the player k
% Vk_Coalition: TU value for each coalition associated
    to the player k
% Sub_coal: sub-coalitions array formed by removing
    the player k
% Vk_Sub_coal: TU value for each sub-coalition
% M_Shapley Shapley Array
% B: Coefficient Vector of Shapley (P (j))
% Weight: Vector of values of Shapley for each player k

% Routine to identify Coalitions according to the value of k
for k=1:Nj        % Parameter to consult in the coalitions
    for lg=1:Nj   % Number of elements in the coalition
        nMk=0;
        for i=1:n_coal
            for j=1:Nj
                if M_Coalitions(i,j)==k
                   N_ceros=0;
                   for g=1:Nj
                       if M_Coalitions(i,g)>0
                          N_ceros=N_ceros+1;
                       end
                   end

                  % Identify coalitions that meet the parameters
                     of length lg and player k
                   If  N_ceros==lg
                       nMk=nMk+1;
                       Mk_Coalitions(nMk,:)=M_Coalitions(i,:);
                       Vk_Coalition(nMk)=V_Coalition(i);
                       j=Nj;
                   end
                end
            end
        end
    end

        % Routine to identify sub-coalitions
        Sub_coal=zeros(nMk,Nj);
        for i=1:nMk
             j=0;
             for g=1:Nj
                 if Mk_Coalitions(i,g)>0&Mk_Coalitions(i,g)~=k
```

```
                j=j+1;

                Sub_coal(i,j)=Mk_Coalitions(i,g);
                % Sub-coalitions value
              end
            end
          end

         % Routine to identify the sub-coalition value
         for i=1:nMk
             if lg==1
               Vk_Sub_coal(i)=0;          % A zero is registered
                                            in the sub coalition
             else
                for j=1:n_coal
                    if Sub_coal(i,:)==M_Coaliciones(j,:)
                      Vk_Sub_coal(i)=V_Coalicion(j);
                      % sub-coalition value
                    end
                end
             end
          end
         M_Shapley(k,lg)=sum(Vk_Coalition-Vk_Sub_coal);
         clear Vk_Coalicion;
         clear Vk_Sub_coal;
         clear Mk_Coaliciones;

         clear Sub_coal;
    end
end

% Routine to calculate the Shapley coefficients
for S=1:Nj
  B(S)=factorial(S-1)*factorial(Nj-S)/factorial(Nj);
end

%Routine to calculate the weight of each player
for  i=1:Nj
     Z=M_Shapley(i,:).*B; %Calculates the weight
                           corresponding to the Shapley value
     Weight(i)=sum(Z);  % for each player and they are stored
                          in the Weight vector
end
```

Appendix C

Routine developed in Matlab to facilitate the processes of resources allocating using the *Nucleolus*

```
% Routine for resource allocation processes supported in
   the nucleolus.
% Important variables that are part of the routine:
% Nj: number of players.
% M_Coalitions: Array of possible coalitions
% V_Coalition: Value of transferable utility (TU) for
   each coalition
% n_coal: coal: Number of possible coalitions
% V: Vector of demands

Total_V=sum(V);  % Calculates the total traffic demand,
                 % according to the demands  vector
                 % (V) for a number of players Nj

n_coal=0;   % Number of  initial coalitions
for i=1:Nj
     n_coal=n_coal+nchoosek(Nj,i);
      % Number of possible coalitions of  Nj
        elements in size groups i.
end
M_Coalitions=zeros(n_coal,Nj);
      % Initializes the Array of coalitions
c=0;
for i=1:Nj   % Calculates the weight of each coalition.
     S=nchoosek(Z,i); % Number of possible coalitions
                      % of elements Z
     nZ=length(S(:,1)); % Number of elements of vector S
     for j=1:nZ
         c=c+1;       % It increments counter c
         Suma_d=0;  % Initial value of the sum
         for k=1:i
             Suma_d=Suma_d+V(S(j,k));
               % The value is calculated for the coalition
             M_Coalitions(c,k)=S(j,k);
               % and the value of the coalition is recorded
         end

          % Procedure to estimate the value of a transferable
          % utility considering a Bankruptcy game.
         Suma_dT=BW_T-(Total_V-Suma_d);
         VAux=[0 Suma_dT];
         V_Coalition(c)=max(VAux);
              % TU value for the coalition
     end
end
M_Coalitions;   % Array of coalitions
V_Coalition;     % Transferable utility value by coalition

% Adequacy of parameters to be optimized by toolbox

% Procedure to establish array A of  Optimization ToolBox
MMZ=zeros(n_coal,Nj);
fori=1:n_coal
      for j=1:Nj
         if  M_Coalitions(i,j)>0
           MMZ(i,M_Coalitions(i,j))=1;
                % Setting values in the A array
         end
      end
end

MM2=ones(n_coal,1);  % Creates a column of ones
MM3=[MMZ MM2];     % and it is concatenated as the
                        last column to MM1
MM3(n_coal,:)=[];  % Deletes the last row of MM3 array
MM4=eye(Nj);        % Creates an identity array according to Nj
MM5=zeros(Nj,1)    % Insert a column of zeroes in the
                        identity array
MM6=[MM4 MM5];
A=[MM6;MM3];    % Concatenates the two resulting array A of
                        the ToolBox
A=-1*A;

% Procedure for establishing the vector b of the
   Optimization ToolBox
Mb=V_Coalition';
Mb(n_coal)=[];
 MM5=zeros(Nj,1);  % Inserts column of zeroes in the identity
array
b=[MM5;Mb];        % Concatenates the two resulting Array A
b=-1*b;
```

```
% Procedure for establishing the values of Aeq, beq, x0 and
   F required by the optimization toolbox.
Aeq=ones(1,Nj);    % Establish the equality constraints
Aeq=[Aeq 0];
beq=V_Coalition(n_coal);
x0=zeros(1,Nj+1);  % Initial conditions
F=zeros(1,Nj);       % Objective function
 F=[F 1];

 [x,fval] = linprog(F,A,b,Aeq,beq,x0)
                %  x it contains the BW allocated to each player as a
                   result to the optimization problem.
```

References:

[1] N. Anatory, J. & Theethayi, *Broadband Power-Line Communication Systems: Theory and Applications.*, 1a ed. Southampton, England: WIT Press, 2010.

[2] H. Latchman, K. Srinivas, L. Yonge, and S. Gavette, *Homeplug AV and IEEE 1901: A Handbook for PLC Designers and Users*, 1a ed. New Jersey, USA: Wiley-IEEE Press, 2013.

[3] J. Pérez, J. Jimeno, and E. Tena, *Teoría de juegos*, 1 ed. Madrid, España: Pearson-Prentice Hall, 2003.

[4] D. Ramírez R., “Cooperación en la cadena de suministro de la energía eléctrica en Colombia,” Universidad del Norte, 2008.

[5] B. Peleg and P. Sudhölter, *Introduction to the theory of cooperative games*. Springer, 2007.

[6] M. Rodríguez, *Contribuciones a la teoría del valor en juegos en forma estratégica y en problemas de bancarrota*. Universidad Santiago de Compostela, 2005.

[7] J. García, “Complejidad y algoritmos en juegos cooperativos. Thesis PhD,” Universidad de Sevilla, 2000.

[8] A. Magaña, “Formación de coaliciones en los juegos cooperativos y juegos con múltiples alternativas. Thesis PhD,” Universidad Politécnica de Cataluña, 1996.

[9] 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.

[10] B. O’Neill, “A problem of rights arbitration from the Talmud,” *Math. Soc. Sci.*, vol. 2, no. 4, pp. 345–371, 1982.

[11] D. Gillies, “Some theorems on n-person games,” Princeton University, 1953.

[12] L. Shapley, “On balanced sets and cores,” *Nav. Res. Logist. Q.*, vol. 14, no. 4, pp. 453–460, 1967.

[13] J. R. Fernández G., “complejidad y algoritmos en Juegos Cooperativos,” Universidad de Sevilla, 2000.

[14] J. C. Vesga, G. Granados, and J. E. Sierra, “El valor de shapley como estrategia de optimización de recursos sobre Power Line Communication (PLC).,” *Ing. y Cienc.*, vol. 11, no. 22, pp. 189–211, 2015.

[15] P. Berens, “CircStat: a MATLAB toolbox for circular statistics,” *J. Stat. Softw.*, vol. 31, no. 10, pp. 1–21, 2009.

[16] X. Huang and B. Bensaou, “On max-min fairness and scheduling in wireless ad-hoc networks: analytical framework and implementation,” *Proc. 2nd ACM Int. Symp. Mob. ad hoc Netw. Comput.*, 2001.

[17] Y. Sheng and M. H. MacGregor, “Dynamic Resource Allocation Based on Weighted Max-Min Fairness,” in *Communication Networks and Services Research Conference (CNSR), 2011 Ninth Annual*, 2011, pp. 247–256.

[18] E. L. Hahne, “Round-robin scheduling for max-min fairness in data networks,” *IEEE J. Sel. Areas Commun.*, vol. 9, no. 7, pp. 1024–1039, 1991.

[19] J. Chen, I. Cidon, and Y. Ofek, “A local fairness algorithm for the MetaRing, and its performance study,” in *[Conference Record] GLOBECOM ’92 - Communications for Global Users: IEEE*, 1992, pp. 1635–1641.

[20] J. S.-C. Chen, I. Cidon, and Y. Ofek, “A local fairness algorithm for gigabit

LAN’s/MAN's with spatial reuse,” *IEEE J. Sel. Areas Commun.*, vol. 11, no. 8, pp. 1183–1192, 1993.

[21] D. Schmeidler, “The Nucleolus of a Characteristic Function Game,” *SIAM J. Appl. Math.*, vol. 17, no. 6, pp. 1163–1170, Nov. 1969.

[22] I. Curiel, *Cooperative game theory and applications: cooperative games arising from combinatorial optimization problems*. Dordrecht: Kluwer Academic Publishers, 1997.

[23] E. Kohlberg, “On the nucleolus of a characteristic function game,” *SIAM J. Appl. Math.*, vol. 20, no. 1, pp. 62–66, 1971.

[24] D. Nace, M. Pioro, and L. Doan, “A tutorial on max-min fairness and its applications to routing, load-balancing and network design,” *4th IEEE Int. Conf. Comput. Sci. Res. Innov. Vis. Futur.*, p. 30, 2006.

[25] G. Granados A. and J. C. Vesga F., “Optimización de una red multiservicio sobre un canal PLC bajo MmQoS,” *Ing. y Desarro. Univ. del Norte*, vol. 33, no. 2, pp. 260–280, 2015.

[26] J. Hsu, “Constrained simultaneous confidence intervals for multiple comparisons with the best,” *Ann. Stat.*, 1984.

[27] R. Kuelh, *Diseño de experimentos: principios estadísticos para el análisis y diseño de investigaciones*, 2nd. ed. Thompson, 2001.

[28] F. Canete, “User guide for PLC channel generator v. 2,” Universidad de Málaga, 2011.