EUDFC - Enhanced Unequal Distributed Type-2 Fuzzy Clustering Algorithm

Antonio Jesús Yuste-Delgado, Juan Carlos Cuevas-Martinez, and Alicia Triviño-Cabrera

***Abstract*— Wireless sensor networks are composed of nodes that monitor the environment and send their measurements to a base station. Due to the reduced computational and energetical resources of the sensor nodes, network organization must take into account those constraints to reduce energy consumption and, in turn, to prolong the lifetime of the network. Thus, collaborative management techniques, like clustering, are usually implemented. This paper proposes a novel unequal distributed clustering algorithm for wireless sensor networks that employs a new set of input variables. Compared with previous works, these variables help for a more accurate estimation of the convenience of a node to be a cluster head. The developed algorithm relies on local information of the sensors to diminish the data exchange, reduce the interference and the consumption of the transmission processes. Because of the underlying uncertainty of the local data, a type-2 fuzzy logic system constitutes the basis of the proposed clustering algorithm whose knowledge base is sampled to allow its feasible implementation in a node. In addition, it is used an unequal scheme for the cluster sizes because it has been demonstrated that those methods are more convenient in terms of energy consumption. The simulation results demonstrate the effectiveness of the algorithm to extend the network lifetime.**

***Index Terms*— Wireless sensor network, fuzzy system, distributed system, unequal clustering.**

I. Introduction

THE applications of Wireless Sensor Networks (WSNs) are increasing day by day and are expected to be widespread according to the paradigm of the Internet-of-Things [1] or smart cities [2]. In the environmental and industrial monitoring, these networks are composed of several hundreds of nodes which measure some magnitudes and transmit them to the Base Station (BS) for further processing [3]. The BS is a particular element in the network. Firstly, it is equipped with higher computational resources than other nodes do. Secondly, its functionality is global as it recollects the data from the whole network to take the corresponding decisions (e.g. activation of actuators) [4].

Due to the significant amount of components in WSNs, the replacement of the nodes' batteries is advised against. Consequently, the use of techniques to reduce the energy consumption while being operative is strongly recommended. Clustering techniques are designed towards this goal [5]. Based on this strategy, nodes in the network are grouped into clusters. The data transmission is sent locally to one particular node in the cluster referred to as the Cluster Head (CH). The CH recollects the data from all the nodes in its group and then it decides whether the aggregation is suitable or not for the consequent transmission of the information to the BS. Following this scheme, the power consumed for the data transmission from each node to the BS is reduced even when there are two messages involved (one from each node to the CH and from each CH to the BS). This is due to the fact that the power required for the transmission of a message is at least proportional to the square of the distance separating the transmitter and the receiver. Therefore, if CHs are placed closer to their contributing nodes than the BS is, the energy saved in those transmissions allows for longer lifetimes in WSN. Thus, the benefits of the clustering techniques have prompted the proposal of several algorithms.

Some of the first approaches were pure stochastic algorithms as in [6] and [7]. The main idea of those methods was that the nodes internally generate a random number and compare it with a threshold in order to decide whether they should become a CH or not. Usually, those CHs could not be elected again after several rounds. This kind of clustering algorithm was extended to use other type of information available in nodes (e.g. the residual energy) when defining the threshold [5], [8]–[10]. In order to minimize the message exchange, the nodes infer some information about other nodes from the data they can gather by themselves. To deal with the uncertainty of these estimations, other methods like those based on Computational Intelligence (CI) were employed [11], [12]. The CI-based approaches are considered more appropriate for the WSN as they are able to adequately manage the uncertainty associated to the estimations done by the own sensors. Current CI-based clustering algorithms use CI to infer a probability for each node or for a group of nodes to become potential CHs. Then, through a random process, some of them are selected as CHs. This random selection could be accomplished by each node individually in distributed solutions or by the BS in a centralized approach. Distributed approaches are more efficient in terms of the energy consumed. Moreover, distributed methods are based on simpler devices but more complex algorithms.

Concerning the parameters employed by the clustering methods for WSNs, after a deep review of bibliography it

Manuscript received February 2, 2019; accepted February 12, 2019. Date of publication February 19, 2019; date of current version May 16, 2019. This work was supported by the Department of Telecommunication, Universidad de Jaén, for the 2019 Program of Teaching and Research grants in Telematics Engineering. The associate editor coordinating the review of this paper and approving it for publication was Dr. Amitava Chatterjee. *(Corresponding author: Antonio Jesús Yuste-Delgado.)*

A. J. Yuste-Delgado and J. C. Cuevas-Martinez are with the Universidad de Jaén, EPS de Linares, 23700 Linares, Spain. (e-mail: ajyuste@ujaen.es; jccuevas@ujaen.es).

A. Triviño-Cabrera is with the Universidad de Málaga, Escuela de Ingenierías Industriales, 29071 Málaga, Spain.

Digital Object Identifier 10.1109/JSEN.2019.2900094

can be concluded that the two most common parameters used are the residual energy of a node and its distance to the base station. Clustering methods also rely in parameters that give information about the density of nodes around any sensor. This is the case of MOFCA [13], its evolution named FUCA [14] and those proposed in [15] and [16]. In some approaches, density is also measured in terms of centrality [17].

The previous analysis shows that the current approaches tend to use three variables both for stochastic methods and for CI methods, as in [18]. Nevertheless, as we presented in [19], four variables can accomplish better solutions with an affordable increase of complexity. Particularly, the present fully-distributed approach continues using the residual energy and the relative distance to the BS as in [19]. However, we have substituted the contribution of the nodes and the efficiency by the relative distance of the nodes in a cluster to its cluster head (*Rch*) and the number of rounds that the node has not been acting as CHs (*Ctr*).

As was previously stated, the increment of the number of inputs could imply a higher complexity of the algorithm, which could increase the computational costs of the nodes. Those costs have been avoided in the present approach by sampling the whole inputs and output matrix of the fuzzy system leading to a numeric resolution of the system. In this way, the size of the nodes memory is affordable while we get a valid accuracy of the system. One of the main novelties of [19] that is also employed in our present work is the use of a type-2 fuzzy logic system. This system is able to cope with some uncertainties of the inputs, like the estimation of the location of a node or the current status of the rest of the network. These estimations are based on information that nodes can gather by themselves, so that each node can save energy and improve its battery lifetime because they send less control messages and can avoid complex devices like GPS chips. In this way, the WSN will satisfy the requirements of applications that need of fast deployment of unstructured wireless sensor networks formed by expendable devices. This is the case of disaster monitoring in the wilderness or detecting survivors in rubble after earthquakes.

Another feature of the present approach is the unequal formation of the clusters. Thus, the size of clusters is not fixed so that those clusters that are placed close to the BS tend to be smaller than those which are far from it. According to [14], unequal clustering reports a fair energy consumption among the nodes and, in turn, it extends the network lifetime.

Finally, the proposed clustering algorithm makes use of turns. Each turn is composed of two rounds. In the first round, the CHs are elected. The selection of those CHs has a pre-candidate phase in order to choose only those with better probability. The second round only readjusts the functionality of those nodes affected by dead CHs without executing any CI-based process. In this way, the energy consumption in the nodes is reduced taking advantage from the low dynamics of WSNs.

Consequently, the main contribution of the paper is a turn-based fully-distributed unequal clustering algorithm for WSN that extends the work in [19] with a new set of input variables that are able to estimate the convenience of a node to be a CH in a more accurate way. We also provide practical information about how to implement the type-2 fuzzy algorithm in resource constrained devices. It also includes details about the communication protocol, which are not always detailed in other works, as we stated in [20]. Moreover, the proposed unequal clustering algorithm has been tested in a wide configuration of scenarios. Those simulation results demonstrate that the proposed clustering algorithm is able to prolong the operation of the network when compared with other proposals such as [19].

The rest of the paper is structured as follows. The next section presents the related work while the third section describes the proposed approach in detail. Section IV shows the results of the simulation experiments and their explanation. Finally, the paper includes a section about the conclusions.

II. Related Work

Clustering is the technique by which nodes are divided into groups or clusters, being one of the node in the cluster the Cluster Head (CH). In distributed approaches, every node decides about its convenience to be a CH and acts accordingly. To decide about becoming or not a CH, the node executes an algorithm. These algorithms can be purely stochastic or based on CI. Among the CI-based methods, there are some algorithms based on type-1 fuzzy logic and type-2 fuzzy logic. The most relevant clustering algorithms are described next.

LEACH (Low-Energy Adaptive Clustering Hierarchy) [6], [7] is a completely distributed stochastic method based on an incremental probability for a node to be CH. Once a CH selects itself as a node, it broadcasts an advice message and the closest non-CH nodes join its group. The probability depends on parameter p (usually 0.05) and a node can be only once a CH for a fixed number of rounds, which depends on the desired number of CHs. Due to its simplicity and its minimum processing cost, its results are still considered as a reference even in current research works. LEACH has been extended to include other parameters, like residual energy, to help to judiciously select the CHs [5], [8]–[10]. Other stochastic approaches, such as R-LEACH [21], combine this information in a mathematical function to generate a random number, which is compared with a threshold in order to decide whether to become or not CH.

Conversely, CI employs other techniques to generate the random number to decide about the node to be CHs [11], [12]. Hybrid CI infer a probability for each node, or for a group of nodes that are potential CHs. In particular, they assume that the estimations used for the decision are prone to errors so that the mechanisms to cope with this condition are activated. As a result, some of nodes are selected as CHs. This random selection could be accomplished by each node individually in distributed solutions [14], [15], [17], [22] or by the BS in centralized solutions [23], [24]. Some authors establish that the advantages of centralized methods rely on the use of an updated network knowledge, which must be accomplished with additional hardware and more control information [23]. Furthermore, this implies extra energy costs for the nodes as they have to send more information to the central unit. In the other hand, more complex algorithms allow for the reduction of the data exchange. However it forces nodes to make estimations about the status of other devices in the network. Due to the uncertainty involved in any decision with just local data, fuzzy systems (type 1 and type 2) are identified as suitable for distributed clustering algorithms.

An example of the application of fuzzy systems to clustering in WSNs is the work by Gupta et al. (Cluster-head Election using Fuzzy Logic for Wireless Sensor Networks) [23] which can be considered the first clustering method for WSNs that used fuzzy logic. It is a centralized method that relies on an expert system held in the BS. The expert system uses the residual energy, the number of neighbors and centrality to infer the probability of being CH for each separate node. For this method, the BS requires precise node placement information and continuous feedback about nodes energy. Another type-1 fuzzy logic is FUCA (Fuzzy based Unequal Clustering Algorithm) [14], which can be considered an improved version of MOFCA (MultiObjective Fuzzy Clustering Algorithm) [13]. This algorithm selects the CHs in a distributed method, being the cluster sizes different. Each node has a T probability of being selected as a tentative CH. Those tentative CH runs a fuzzy system based on their residual energy, distance to the base station and node density to obtain a competition radius and a rank. The rank value is broadcast at a distance equal to competition radius. The candidate with the highest rank will be final CH. The problem arises if no normal nodes receive the rank message, due to value of the competition radius. In this case, a node without a CH would send its data directly to the BS. FBUC (Fuzzy logic Based Unequal Clustering for wireless sensor networks) [17] is also a distributed method with unequal clustering. It has been developed as an improvement of fuzzy Energy-Aware Unequal Clustering algorithm (EAUCF) [25]. FBUC sets the candidate CHs based on the individual probability introduced by LEACH. After that, those candidates infer a competition radius through a fuzzy system whose inputs are the distance to BS, the residual energy and tentative cluster head node degree. Later, all those candidate CHs broadcast their current energy at the radius distance. Only those CHs with the highest residual energy within the range are final CHs. The exact computation of the input variable node degree it is not described in the paper, but it can be supposed as the number of desired nodes within the competition radius divided by the total number of nodes.

Alternatively, CRT2FLACO (Clustering Routing protocol for WSN based on type-2 Fuzzy Logic and Ant Colony Optimization) [24] is a centralized method that chooses the CH based on the output of a type-2 Mamdani fuzzy logic system that runs on the base station. This method needs the instantaneous residual energy every round, the number of neighbor nodes density and the distance to the base station. The outputs are the probability of being CH and a competition radius to avoid CH concentration. The selected CHs also organize themselves in a transmission chain through an Ant Colony Optimization (ACO) method. This method needs an exact localization due to the requirements of density, however, it lacks of a precise description about how to obtain it. This shortcoming is overcome by CHEETAH (Cluster Head Enhanced Election Type-2 Fuzzy Algorithm for wireless sensor networks) [19]. This is a fully distributed method based in a type-2 fuzzy system that uses four variables: the residual energy of the node, the relative distance to the base station, the average number of nodes for each time a node was CH and the efficiency. This last parameter stands for the estimated power saved by a group of nodes when they relay their data to the CH instead of sending them directly to the BS. The type-2 fuzzy system infers an interval whose upper limit is used as the probability of being a CH when a node has not been a CH in the previous round, whereas the lower limit of the interval is used if the node was a CH in the previous round.

In the next section it is detailed the proposed distributed unequal clustering algorithm. This method is based on the selection of the best candidate CHs regarding the probability that every node infers by its embedded type-2 fuzzy system. The turn-based scheduling helps to improve network lifetime due to the inertia of the system.

III. Proposed Distributed Unequal Clustering Method

A. Method Overview

As it was previously stated, the design of the proposed distributed unequal clustering method (EUDFC) for WSNs is based on CHEETAH [19]. Nevertheless, we confirmed the convenience of modifying some aspects of it. Thus, after a deep study of different parameters and their effects in the clustering algorithm, the proposed approach presents four main changes from CHEETAH:

- A new definition of turn-based scheduling to set the instants at which the clusters heads are selected or their status are verified.
- Incorporation of a selection algorithm that determines the best CHs among all the candidate-CHs to avoid an excessive number of CH that could result from a stochastic election. The maximum number of CHs per turn is the total $n \times p$, where n is the total number of nodes in the network and p is the optimum CH factor.
- Suppression of the input variable efficiency in the fuzzy system and inclusion of a new variable that takes into account the history of the node, that is, the number of rounds that the nodes has not contributed as a CH. In this way, we incorporate the recommendation that the role of the CHs should be taken by different nodes during the WSN operation in order to make a fair consumption among all the nodes.
- Suppression of the input variable that measures the mean value of nodes to which a CH is serving in its cluster and the inclusion of a new variable that measures the proximity of the contributing nodes. With this variable, we incorporate the idea that a CH should serve close nodes as the transmission of the messages requires an energy inversely proportional to the square of the distance separating the two nodes. Due to this fact, it is important to know the relative distance between the CHs and their contributing nodes.

Therefore, in order to understand EUDFC, some of its main features are outlined as follows while the overall process of the approach is depicted in Fig. 1.

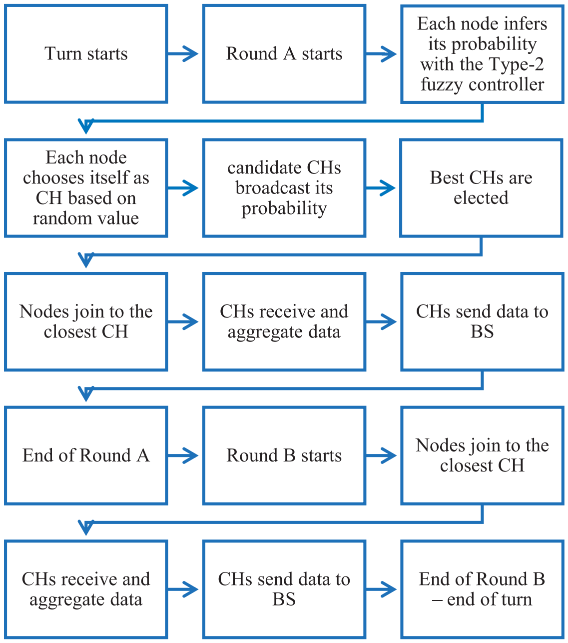

Fig. 1. System process overview.

EUDFC is scheduled in time slots called turns. Each turn is also divided into two separate stages or rounds named A and B. In both rounds data is gathered by normal nodes, collected and aggregated by CHs, and eventually sent to the base station. However, CHs are only elected in round A, whereas in round B only data communication takes place because the CHs are the same as in round A. Although the CHs are the same in round A and B, nodes may opt to join a different CH in round B (see Section III-E for details). This feature tries to take advantage from the inherent steadiness of these kinds of networks where nodes are static. It is also an advantage for the nodes that usually switch between working and idle phases.

In addition, the election of a node as a CH is based on the probability that each node infers internally using an embedded type-2 fuzzy controller (see Section III-D). This probability is compared with the one generated by other nodes and sent in specific messages. Consequently, nodes whose inferred probability is lower than the top $n \times p$ received probabilities keep themselves as normal nodes while the top $n \times p$ become CHs.

The network model that was employed to develop EUDFC is detailed in the next section. The proposed approach is also completed with the main guidelines concerning the implementation of the communication protocol (see Section III-C).

B. Network Model

The network model describes the main features and the main conditions that rule a WSN in order to design and test any clustering algorithm. Thus, its specifications are fundamental to effectively develop any clustering method for WSNs. Accordingly, the proposed model relies on the following common assumptions:

- There are n nodes in the network, which are randomly deployed in a fixed size square area, called the deployment field. The surface of the deployment field is $d_x \times d_y$ m^2.
- The base station is located in a predefined location. Besides, it is supposed to be connected to the electrical grid so an infinite power source is assumed in it.
- The transmission power of any sensor node is high enough to cover all the deployment field and the BS.
- The sensor nodes and the base station are completely stationary after deployment.
- Sensor nodes are homogeneous, isolated, battery powered and have the same initial energy when the network starts to operate.
- Communication channel is symmetric and without either interference or losses, such as it is assumed in other clustering studies as the ones presented in [14], [23], and [13].
- All the sensors have the same monitoring and sensing tasks based on the WSN application.

C. Communication Protocol

The network model described above have to be complemented with a communication protocol to define how the data are exchanged among nodes. The communication protocol assumed in this work represents a conventional mechanism to exchange information among the nodes over the network layer. It fits into the common testing parameters used for clustering of WSNs while it is designed to maximize the knowledge a node can gather from its neighbors without incurring in additional control packets. Besides, it must be noted that the design of the communication protocol directly affects to the power consumed by the node in every information interchange. For this reason, the proposed protocol provides a minimal set of options to carry out with the clustering method and to save as much energy as possible. Consequently, all the packets (control or data messages) sent by a node (even for the start-up message of the base station) must convey a header followed by the payload with application data. Header fields are described next:

- Message type (8 bits). The type field is composed of 8 flags:
  - Start-up (00h): before the WSNs starts its duty, the BS broadcasts this unique start-up message that is used by every node to estimate its own distance to it.
  - Cluster advice message or ADV (01h): control message broadcast by candidate CHs to communicate their inferred probability to other candidates. This message includes in the payload the value of the probability obtained by the embedded type-2 fuzzy controller. It is a fixed point value between 0 and 1 that does not need more than 16 bits.
  - Node join message (02h): control message sent by nodes to their closest CH. This message will also be sent by a candidate CH whose probability value is lower than the best candidate. This message must also contain in the payload a field with the identification of the chosen CH.
  - Node data (04h): this message is sent by normal nodes to a CH with the information gathered of their surroundings coded in the payload field.

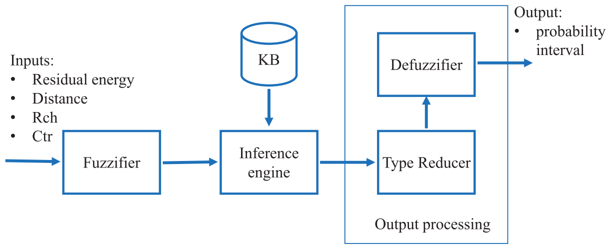

Fig. 2. eT2FC interval type-2 fuzzy controller.

- Aggregated data message (08h): data message sent by CHs to the base station with the aggregated data of all the contributing nodes in the payload.

- Node id (16 bits). A unsigned integer field with the identifier of the node.
- Tx power (16 bits). The transmission power in dBm units in a fixed point value.

As can be noticed, the additional information sent for each message does not imply a significant overhead. Moreover, to make feasible the comparison with other clustering method the length of control and data messages was set to a maximum of 200 and 2000 bits respectively.

D. Embedded Type-2 Fuzzy System

As it has been commented previously, each node embeds on an interval type-2 fuzzy logic controller (FLC) [26] or eT2FC whose main purpose is to obtain the probability interval of being a cluster head.

The fuzzy logic controller eT2FC which follows a Mamdani structure [27], links inputs and output in a staged process as can be seen in Fig. 2. The main difference between type-1 and type-2 fuzzy controllers is the "output processing" block. The other blocks are as in type-1 fuzzy controllers: (i) a fuzzifier to normalize crisp input values, which in type-2 fuzzy logic implies a more complex processing; (ii) a knowledge base (KB), which defines the relationships between inputs and output with IF THEN clauses; (iii) the inference engine that infers a solution based on particular input values and the knowledge base; (iv) the output processing block which produces the output interval for type-2 FLCs (in this last case, a new block, "type reducer", is used to obtain the output) and (v) the defuzzifier, which gives crisp values through denormalization.

In contrast to type-1 systems, an interval type-2 FLC allows for an easier mechanism to design fuzzy rules from expert knowledge and from natural language. This is due to the fact that we do not need to specify a crisp number as in type-1 systems. Conversely, the membership grade of an type-2 fuzzy system is an interval, which also increases the robustness of the system. Moreover, interval type-2 FLCs are more adaptive than type-1 fuzzy systems due to the possibility of defining complex relationships between their inputs and outputs. This is of particular interest when the inputs are not precise measurements, which may occur in distributed clustering algorithms where nodes exclusively employ local information to estimate their convenience of being CH as EUDFC does.

Therefore, in a WSNs running EUDFC each node infers its own output interval which is defined as (o_l, o_r). This interval is then used by the CH election algorithm in order to choose the correct probability of being CH according to its status which will be discussed in the next subsection.

To carry out the special features of distributed clustering the eT2FC has been designed to make use of four input variables: residual energy (R), relative distance to the base station (D), proximity of the contributing nodes to the CH (Ctr) and a factor depending on the number of rounds a CH has not been elected since the last time it was a CH (Rch). As CHEETAH [19], EUDFC employs the same two variables, R and D, but the other two has been changed, as it was commented previously, to improve the performance when the base station is outside the deployment field and to delay the instant when the first node dies. Thus, the input variables are defined as follows:

- Residual energy (R) of the node: R is the division of the current available energy of the node by its initial energy E_0, both in Joules. Each node updates the value of R every round.
- Relative distance (D) to the base station (BS) is the division of the distance of the node to the BS by the distance from the furthest corner of the deployment field to the BS. Each node estimates the distance to the BS from the Received Signal Strength Indicator (RSSI) of the signal of the start-up message that the BS sends only once. The value of variable D remains constant during the lifetime of a node because nodes and BS are static.
- Variable Ctr is the relationship between the average distances of the contributing m nodes (those that are not CH but generate data messages) to their associated CH. Those distances are estimated by the RSSI of the join messages. It is compared with the radius (r), which is defined as the radius of the area of a circle with the same area than the deployment field but divided by p, the optimum number of clusters per round according to Heinzelman *et al.* [7]. This variable is updated every time a node is elected as CH and it has contributing nodes. Ctr gives information about the proximity of the contributing nodes to a node elected as CH. Therefore, values close to 0 inform that the CH is probably in a dense populated area, whereas values close to 1 or equal to 1, remark that most of the contributing nodes are relatively far from the CH. The exact calculus of Ctr is detailed in Eq. 1.

$$Ctr = min\left(\frac{\sum_{i=1}^{m} d_i}{r}, 1\right) \tag{1}$$

$$r = \sqrt{\frac{d_x \cdot d_y}{n \cdot p \cdot \pi}} \tag{2}$$

where:

- m is the total number of nodes that join the CH,
- d_i is the estimated distance of the contributing node i to the CH,
- d_x and d_y are the dimensions of the deployment field in meters,

  - p is the optimum CH factor in LEACH whose value is set to 0.05
- Variable *Rch* measures the number of rounds a node has not been selected itself as a CH divided by a threshold. This variable has been added to the system in order to balance the CH election and distribute this role in a more uniform way among all the nodes in the WSN. The main purpose of this variable is to increase the probability of being elected as CH in those nodes with low values in *R*, *D* and *Ctr* variables. Then, nodes recently selected as CHs will have low values of *Rch* to discouraged re-election as CH whereas nodes not selected or elected as CH during many previous rounds will have high values, or even 1. The value of variable *Rch* in node i, Rch_i, can be obtained as follows:

$$Rch_i = min\left(\frac{r_i}{ch_t}, 1\right) \quad (3)$$

where:
  - r_i is the number of rounds from the last time the node i elected itself as a CH,
  - ch_t is the cluster head threshold defined as the maximum number of rounds a CH it is supposed to wait for being a CH again. This value is a fixed parameter of the method and is empirically set to 5.

Thus, if r_i is greater than ch_t, Rch_i will be 1.

As can be noticed, all the input variables have crisp values from 0 to 1 to avoid any additional normalization to be fuzzyfied.

These four parameters are evaluated in each node to get the input variables of eT2FC, which infers the output variable *chance* that is represented as the interval (o_l, o_r). This interval represents a base probability for a node ($chance_i$) of being elected as a candidate CH. However, this decision not only relies on those values. The final number of CHs depends on the CH election algorithm, explained in further detail in the next section. In summary, that algorithm establishes that each node has to calculate the output interval (o_l, o_r) to become a candidate CH. Afterwards, it chooses o_l as its base probability if it already was a CH in the previous turn, or it elects o_r in another case (equation 4). Then, the node calculates a random number between 0 and 1, if this number is lower than the selected probability it becomes a candidate CH.

$$chance_i = \begin{cases} o_l & ch_i = 1 \\ o_r & ch_i = 0 \end{cases} \quad (4)$$

where ch_i equals 1 if the node was a CH in the previous turn and 0 otherwise.

To complete the definition of eT2FC as any other fuzzy system, it is necessary to describe its knowledge base. First, each input variable is defined by three fuzzy sets(also referred to as the membership functions). The input variables are normalized so that their values range from 0 to 1, as represented in the x-axis of Fig. 3. The variables share the same layout with three membership functions: low, medium and high, as depicted in Fig. 3. The definition of three membership functions help to distinguish among three performance regions. Secondly, the output variable *chance* is defined by five fuzzy sets: very low, low, medium, high and very high, as shown in Fig. 4. Fuzzy set layouts have been adapted from those in [28] taking into account the special characteristics of the input variables and expert knowledge from previous work. A similar approach is followed in [19], [29], and [30]. In a type-2 fuzzy logic, the interval of the input variables in the x-axis defines the uncertainty associate to its value. In this way, the error due to estimations can be considered. Additionally, the interval of the y-axis in a type-2 fuzzy logic set is incorporated to reduce the effect of fluctuation in output values due to the high amount of uncertainty in those magnitudes. The triangular and trapezoidal shapes for the membership functions are selected to ease the implementation of the mathematical functions in resource-constrained devices [31], [32]. Finally, the rule base is compound by 81 (3^4) IF-THEN rules. The rule base is shown in Table I. The rule base has been developed from the expert knowledge of the research in WSN systems in a similar way to previous research papers like CHEETAH [19] or [24]. To allow the implementation of eT2FC in nodes with constrained resources, the whole solution space has been sampled. Hence complex calculations can be replaced by only memory access.

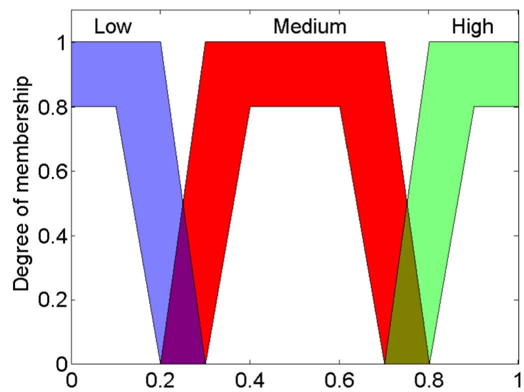

Fig. 3. Fuzzy sets for input variables R, D, Ctr and Rch.

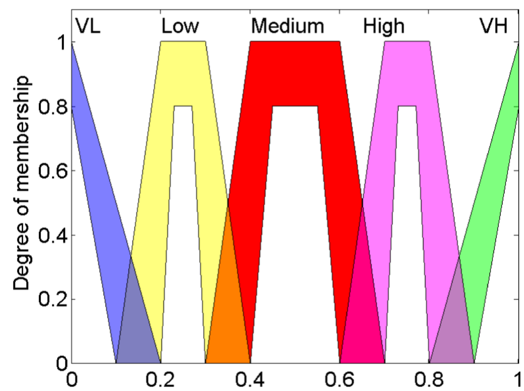

Fig. 4. Fuzzy set layout for output variable *chance*.

E. Cluster Head Election Algorithm

The proposed algorithm rules the election of the CHs in the WSN through a turn-based scheduling to maximize the lifetime of the WSN avoiding control messages in some stages. It has been designed taking into account common and realistic WSN featuressuch as applications, switching between

TABLE I
RULE BASE FOR eT2FC INTERVAL TYPE-2 FUZZY CONTROLLER

R	*D*	*Ctr*	*Rch*	*chance*
L	L	L	L	L
L	L	L	M	L
L	L	L	H	M
L	L	M	L	M
L	L	M	M	M
L	L	M	H	L
L	L	H	L	L
L	L	H	M	M
L	L	H	H	M
L	M	L	L	VL
L	M	L	M	VL
L	M	L	H	L
L	M	M	L	VL
L	M	M	M	L
L	M	M	H	L
L	M	H	L	VL
L	M	H	M	L
L	M	H	H	L
L	H	L	L	VL
L	H	L	M	VL
L	H	L	H	L
L	H	M	L	L
L	H	M	M	L
L	H	M	H	L
L	H	H	L	L
L	H	H	M	L
L	H	H	H	M
M	L	L	L	L
M	L	L	M	M
M	L	L	H	H
M	L	M	L	M
M	L	M	M	M
M	L	M	H	H
M	L	H	L	M
M	L	H	M	H
M	L	H	H	VH
M	M	L	L	L
M	M	L	M	M
M	M	L	H	M
M	M	M	L	M
M	M	M	M	M
M	M	M	H	H
M	M	H	L	M
M	M	H	M	H
M	M	H	H	H
M	H	L	L	L
M	H	L	M	L
M	H	L	H	L
M	H	M	L	VL
M	H	M	M	L
M	H	M	H	M
M	H	H	L	M
M	H	H	M	H
M	H	H	H	VH
H	L	L	L	VL
H	L	L	M	VL
H	L	L	H	L
H	L	M	L	VL
H	L	M	M	VL
H	L	M	H	L
H	L	H	L	VL
H	L	H	M	VL
H	L	H	H	L
H	M	L	L	M
H	M	L	M	M
H	M	L	H	H
H	M	M	L	M
H	M	M	M	M
H	M	M	H	H
H	M	H	L	M
H	M	H	M	H
H	M	H	H	VH
H	H	L	L	VL
H	H	L	M	L
H	H	L	H	L
H	H	M	L	L
H	H	M	M	L
H	H	M	H	M
H	H	H	L	M
H	H	H	M	M
H	H	H	H	H

idle and duty cycles, the available resources and processing requirements in nodes and common MAC protocols like Time Division Multiple-Access (TDMA) widely employed in WSNs [7], [33], [34]. This algorithm will control the WSN until the last node depletes its battery because it is a fully distributed method where nodes do not receive any control information from the BS (except the start-up message) to manage their performance. Therefore, the WSN starts its duty when the BS sends the start-up message, that is also used by every node to estimate its own distance to it. Afterwards, the turn-based scheduling begins and rules the whole lifetime of the WSNs. As it was shown in Fig.1, each turn is divided into two rounds: (i) CH election for the turn, or round A and (ii) CH acknowledgment, or round B, in which the same alive CHs of the previous rounds are self-selected. A flow chart of turn and rounds scheduling is presented in Fig. 5.

As is shown in Fig. 5, both rounds are also divided into several phases. Besides both rounds end with the same phase called data transferring. Thus, round A whose duty is to select the set of CHs for the whole turn has three phases: (i) the pre-candidate CH advertisement, (ii) final CH election and (iii) data transferring. In contrast, round B is divided into only

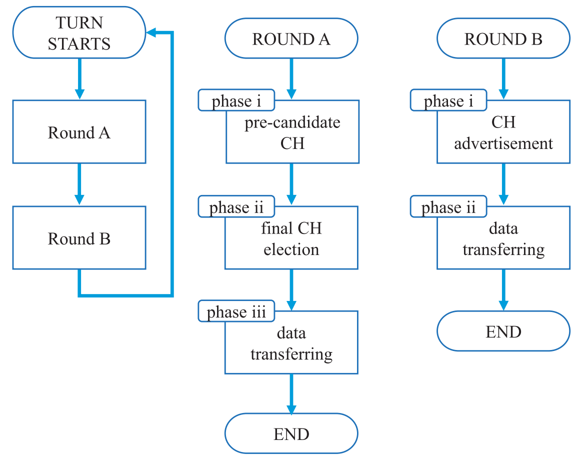

Fig. 5. Turn and rounds scheduling.

two phases: (i) CH advertisement and (ii) data transferring; because in round B no CH is newly selected.

From the overall process, the pre-candidate CH advertisement, round A-phase i), is the main phase of the method because the best potential CHs are selected here. After this stage, the nodes can have two different roles: candidate-CH or normal node (no CH). Candidate-CH are those nodes that selected themselves as candidate to be CH because they randomly generated a value less or equal than their inferred $chance_i$ provided by eT2FC. Alternatively, normal nodes are those whose random values were over their $chance_i$. After this decision, all the nodes start listening to the channel and each candidate-CH waits for a random number of seconds to broadcast its ADV message at maximum power. That means that the message is generated with enough transmission power to reach any point of the deployment field. After receiving all the ADV messages from candidates, those candidate-CHs whose $chance_i$ value is lower than the $(n \times p)$-th higher values among the received advertisements become normal nodes. Due to the complexity of this phase, the algorithm that rules the election of the candidate-CHs is detailed as follows:

```
/* Round A  pre-candidate CH adv.*/
update(R, Ctr and Rch)
(ol,or) = eT2FC(R, D, Ctr, Rch)
if ch=1~then
    chance = ol/3
else
    chance = or/2
end if
if rand()<=chance
    type=$CH_candidate$
else
    type = normal
end if
if(type == $CH_candidate$)
    sleep(rand()*1000)
    send($ADV_message$)
end if
i = 0
on receiving ($ADV_message$,i)
```

```
        chance[i]=$ADV_message.chance$;
        i++;
    end receiving
    chances[] =sort(chances[],descend)
    if (type == $CH_{candidate}$
        && $chance<chances[0:(np-1)])$
    // own chance is lower than the
    // top n x p chances listened.
       type=normal
    end if
    /*End Round A pre-candidate CH adv.*/
```

Once a candidate-CH receives all ADV messages and switches to CH or normal node, it starts phase ii) of round A, referred to as the final CH election. In this phase, normal nodes (including those converted in normal ones after comparing their chances) send the join message to the closest CH by estimating the distance from the RSSI of the ADV message. In order to get advantage of the position of each contributing node regarding their selected CH, they adjust their transmission power according the previously estimated distance. As a result, all the nodes have chosen their CH and no more messages are needed because all the ADV messages were sent at maximum transmission power and reached every node, so any node knows the best chances and they can select a final CH accordingly.

Finally, once each normal node has elected its CH, round A-phase iii) starts. In this stage all normal nodes send a data message which conveys their measured values to their CH. After receiving all the data, each CH aggregates the received values and sends a data message to the BS, finishing round A.

In round B-phase i), all the alive CHs from round A send again an ADV message, but in this case, their chance are not evaluated. Normal nodes join the same CH than before. Afterwards, data transferring phase takes place as it was in round A. Nevertheless, if a CH that was chosen in round A is dead, its contributing nodes will send data packets directly to the BS.

As can be observed in Fig. 5, the pre-candidate CH phase is omitted in Round B due to the high degree of inertia of those kinds of networks, which implies low fluctuations in the input variables within a short period of time. Although variable *Rch* is used to balance the probability of being CH, our studies have detected that similar chance values are obtained for a round and its following one. Therefore, in round B, join messages are also omitted to extend network lifetime.

To check the performance of the above distributed clustering method, based in a type-2 fuzzy system for CH selection with turn-based scheduling, it has been implemented and simulated in MATLAB for several different scenarios and compared with other typical clustering algorithms. The promising results are detailed in the next section.

IV. Evaluation

The proposed clustering algorithm EUDFC, as it was commented above, has been developed from our previous method CHEETAH [19] to improve its behavior in networks with the BS located far from the center of the deployment field. It was also considered that some of the input variables should be modified. Firstly, it was necessary to achieve a more feasible way of calculate them from the information that a node can gather by itself. Secondly, due to the experimental results of CHEETAH, it was also necessary to delay the first dead of a node in the network. Moreover, those improvements and the design of a turn-based scheduling for the cluster formation was aimed to eventually extend significantly the lifetime of the WSN.

Therefore, in order to check the suitability of the proposed clustering method, EUDFC, we have compared it with CHEETAH and with other three recent works in clustering methods: CRT2FLACO [24], FUCA [14] and FBUC [17]. Moreover, other two well-known methods are chosen to act as reference for the performance of the newer ones. They are LEACH [7] and the one proposed by Gupta *et al.* [23]. All these methods were previously explained in Section II.

Each tested method has been implemented in Matlab to make the comparison possible under the same conditions. Those conditions are the network model (detailed in Section III-B and the energy model for data communications (see Section IV-A). Additionally, all the algorithms have been compared with the results published in each research paper to ensure the reliability of the experiments.

A. Energy Model for Radio Communications

The energy model for communications used for the experiments is the one introduced by LEACH [7]. This first order radio model is the most widely used model for clustering in WSN. It assumes symmetric propagation channels and it also states that the energy spent in communications for an l-bit message can be estimated through two equations: one for the transmission (E_{Tx}) and another for the reception (E_{Rx}). The exact calculation is detailed as follows:

$$E_{Tx}(l,d) = f(x) = \begin{cases} l \cdot (E_{elec} + E_{fs} \cdot d^2), & d \le d_0 \\ l \cdot (E_{elec} + E_{mp} \cdot d^4), & d > d_0 \end{cases} \tag{5}$$

$$d_0 = \sqrt{\frac{E_{fs}}{E_{mp}}} \tag{6}$$

$$E_{Rx}(l) = E_{elec} \cdot l \tag{7}$$

where l is the number of bits of the message, E_{elec} the energy per bit spent in the transmitter or receiver circuitry, d is the distance that the message must reach, E_{fs} is the energy spent by the amplifier to obtain an acceptable bit error rate in free space model ($d \le d_0$) and E_{mp} is the energy spent by the amplifier to obtain an acceptable bit error rate in multipath (mp) model ($d \le d_0$).

For reception and data aggregation, the energy in the reception process of l-bits is defined in equation 8:

$$E_{Rx-DA} = (E_{elec} + E_{DA}) \cdot l \tag{8}$$

where E_{DA} is the energy spent by the processing unit of a CH in aggregating the data received from a contributing node.

The exact values uses in all the experiments for the parameters in equations 5-8 are detailed in Table II.

TABLE II

PARAMETERS FOR THE FIRST ORDER RADIO MODEL AND DATA AGGREGATION

Parameter	Value
E_{elec}	50 nJ/bit
E_{fs}	10 pJ/bit/m2
E_{mp}	0.0013 pJ/bit/m4
E_{DA}	5 nJ/bit

TABLE III

EXPERIMENT SETUP VALUES

Parameter	Value
Nodes deployed	100
Initial node energy	0.5 J
Control message length	200 bits
Data message length	2000 bits

TABLE IV

BASE STATION LOCATION FOR EACH EXPERIMENT

Experiment	Location	Coordinates
Scenario 1	center	(50,50) m
Scenario 2	corner	(100,0) m
Scenario 3	outside	(200,50) m

B. Experiment Setup

To compare EDUFC with the other six clustering methods, three different experiments, or scenarios, have been driven under the same propagation parameters (see Table II). For the three experiments, 100 nodes, with the same initial energy (0.5 J), are randomly deployed within an square area of 100x100 m^2. An uniform probability distribution is used to determine the position of the nodes in the area. Control messages and data messages are set to 200 bits and 2000 bits respectively, which are common values used in the bibliography. These parameters are summarized in Table III. Each experiment differs in the location of the BS to represent as many deployments as possible in real networks layouts. The location of the base station for each experiment is detailed in Table IV.

To allow a better comparison of the proposed clustering algorithm EUDFC and the effects of the FCL configuration (type1 or type-2), two sets of experiments for the three same scenarios were carried out. For the first set or SET 1, methods CRT2FLACO, FUCA, Gupta et al., FBUC, CHEETAH and EUDFC were implemented as type-1 FLCs, whereas for the second set or SET 2, those methods were implemented as type-2 FLCs. For both sets, the implementation of LEACH remains the same because it is a stochastic method not supported by any FLC. This experiment setup is summarized in Table V, where we can identify which methods correspond to the original proposal and which methods have been adapted.

For both sets, each scenario and each method (LEACH, Gupta et al., FUCA, FBUC, CRT2FLACO, CHEETAH and EUDFC) have been simulated in Matlab under 30 different network deployments. All the simulations model the behavior of the network in a turn by turn basis following the algorithm detailed in each contribution. This means that there is a selection phase to choose the CHs, which will imply a variable number of exchanged control messages. It is followed by a data communication phase that involves sending the measured data from normal nodes to their selected CH and from those CHs to the BS afterwards. All the communications costs are estimated employing the energy model detailed in Subsection IV-A. The energy of any communication is subtracted from the nodes and the remaining energy is checked any time a node needs to do a new operation. At the beginning of every turn nodes with no energy, or dead nodes, are removed from the following iterations of the algorithm. Moreover, to maintain a stable benchmark, all the simulations are stopped when only 10% of the nodes are alive because we have observed in previous empirical results that there are no significantly changes over 90% of dead nodes. From each simulation three typical metrics related with the lifetime of WSNs are obtained: (i) the round when at least one node has depleted its battery, or First Node Dies (FND) round, (ii) the round when at least half of the nodes have died or Half of the Nodes are Dead round (HND) and (iii) the round when only 10% of the nodes remain alive or Last Node Dead (LND) round. Specifically, FND shows the goodness of the method in balancing the energy consumption of the method, mainly in the beginning, when nodes have less information about their behavior and/or the network status whereas HND shows how the clustering method deals with the problem of high number of dead nodes. In addition, The interpretation of these metrics is related to their relative value. For instance, if HND is too close to FND compared with LND, it means that the algorithm has a high dependence on the first dead, whereas if it is too close to LND it shows a bad resilience to node death.

TABLE V

METHOD VERSION (TYPE-1 OR TYPE2) FOR EXPERIMENT SETS

Method	Set 1 Type-1 FLC	Set 2 Type-2 FLC
LEACH	-	-
CRT2FLACO	adapted	original
FUCA	original	adapted
Gupta et al.	original	adapted
FBUC	original	adapted
CHEETAH	adapted	original
EUDFC	adapted	original

Finally, in order to compare the methods under study, we calculated the average value for each metric from the thirty simulations corresponding to the different spatial distribution that the nodes could have. The exact values obtained in all the experiments are detailed in the next section.

C. Experiment Results

The results of the experiments are shown in two sets of three graphs. Thus, Figures 6 - 8 show the results for LEACH and the other clustering method (Gupta et al., FUCA, FBUC, CRT2FLACO, CHEETAH and EUDFC) with type-1 FLCs or Set 1, whereas Figures 9 - 11, or set 2, show the results for LEACH, the proposed clustering method EUDFC and the other methods employing type-2 FLCs. Each graph is divided into three groups corresponding to the metrics

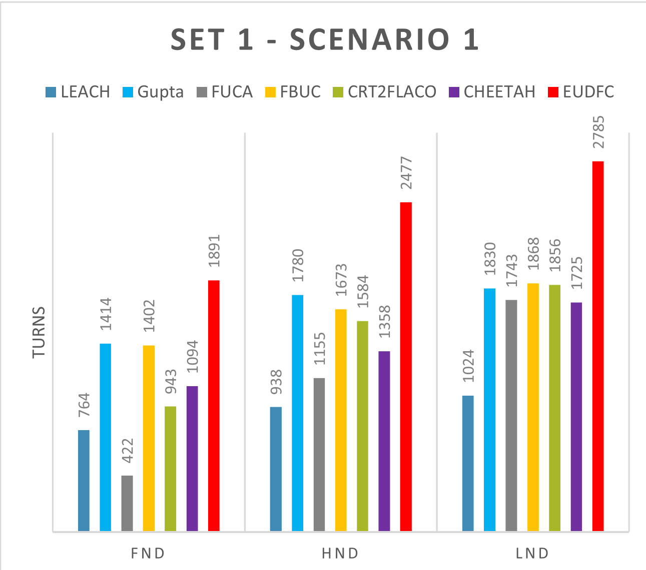

Fig. 6. Results for scenario 1 using type-1 FLCs, with BS at (50,50) m.

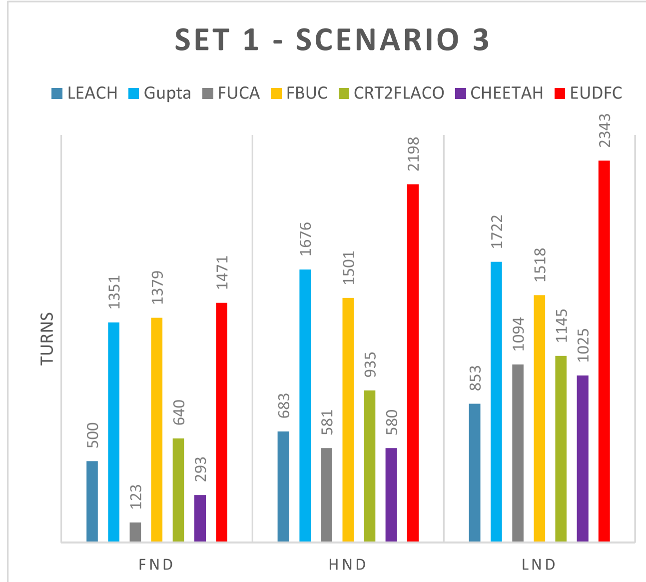

Fig. 8. Results for scenario 3 using type-1 FLCs, with BS at (200,50) m.

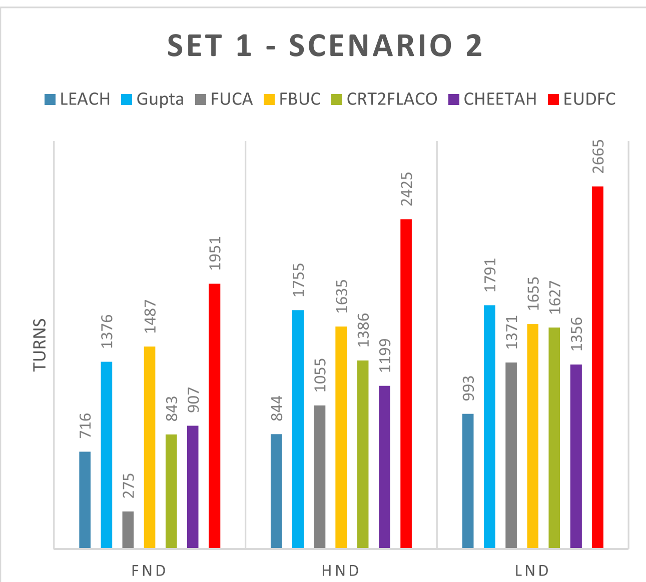

Fig. 7. Results for scenario 2 using type-1 FLCs, with BS at (100,0) m.

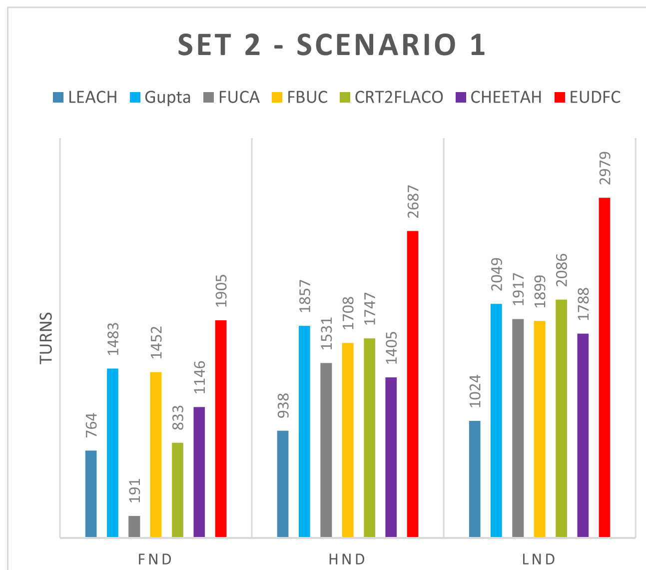

Fig. 9. Results for scenario 1 using type-1 FLCs, with BS at (50,50) m.

described previously: FND, HND and LND. As was previously commented, the number of each bar represents the average value of turns obtained in the 30 simulations for the metric in that scenario.

As can be noticed, for both sets of experiments the results depicted in Figures 6-11 (red bar) show a significantly better behavior of EUDFC in terms of FND, HND and LND for the three scenarios. In terms of FND, EUDFC can delay the first dead of a node even after almost all the nodes in the network are dead compared with all the algorithms in scenarios 1 and 2 and for all but FBUC and Gupta for scenario 3. This is accomplished thanks to the effects of *Ctr* and *Rch* variables that allow for a better balance in CH selection due to their evolution compared to residual energy or distance. Additionally, the inclusion of variables *Ctr* and *Rch* has clearly improved the results of CHEETAH, which also obtained better results than LEACH, CRT2FLACO and FUCA for some of the parameters and scenarios.

Taking into account the differences between type-1 and type-2 FLCs, in general terms we can state that the results obtained with type-2 FLCs, even without a fine tuning of the fuzzy systems, outperform those derived from type-1 FLCs in terms of the network lifetime. Those result are justified by the better adaptability of the type-2 FLCs to uncertainty. The estimations carried out by the nodes to infer other nodes' status are more prone to errors when less nodes are alive in the network as less messages can be analyzed to infer their behavior. This explains why type-2 FCL significantly increases

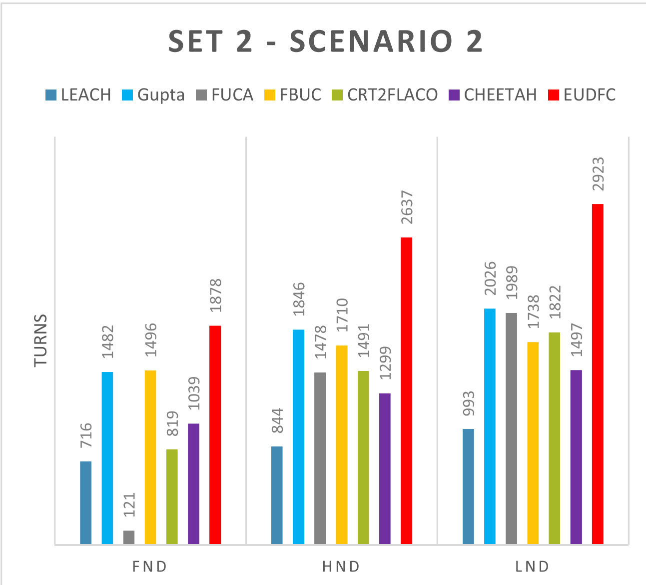

Fig. 10. Results for scenario 2 using type-1 FLCs, with BS at (100,0) m.

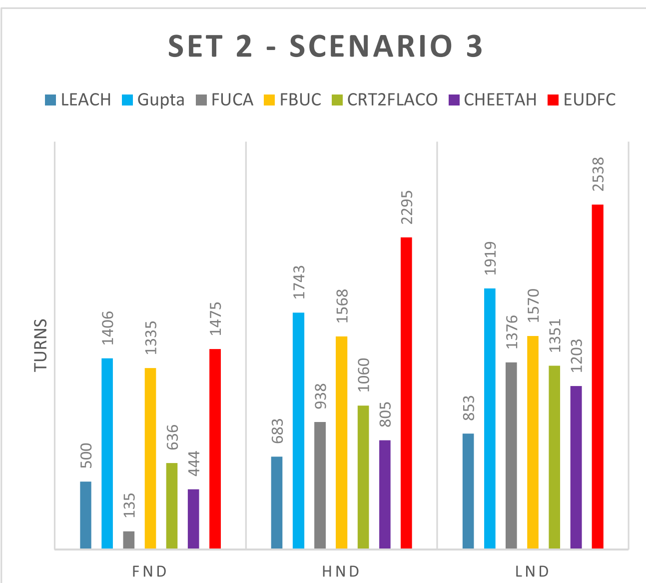

Fig. 11. Results for scenario 3 using type-1 FLCs, with BS at (200,50) m.

HND and LND while FND is not greatly impacted by the type of implementation of the FCL. Nevertheless, FUCA is not sensible to the type of FLC implementation.

We also remark that EUDFC improves network lifetime about a 50% compared with the best other approaches. Specifically, for scenario 3 where BS is located far from the deployment field, the value of LND obtained by EUDFC is 47,38% higher than the original method by Gupta et al. These values show clearly that EUDFC has achieved its goals extending network lifetime considerably even for the most unfavorable scenarios.

It is important to analyze which features of the EUDFC algorithm are triggering the advantages observed in the experimental results. For this analysis, we incrementally add features

TABLE VI

IMPACT OF THE ENHANCEMENTS TO THE METHOD EMPLOYED IN CHEETAH

Enhancement	FND	HND	LND
Inclusion of Rch	106.0%	102.4%	95.7%
Inclusion of Ctr	110.9%	107.0%	97.6%
Best candidate CHs	114.2%	113.6%	98.2%
Type I fuzzy - Turn scheduling	228.0%	212.0%	176.2%
EUDFC	226.4%	226.4%	190.9%

to original version of CHEETAH until the complete EUDFC is implemented. Consequently, the three same experiments were also driven for each stage of the evolution from CHEETAH to EUDFC. whose results are shown in Table VI. The values presented in Table VI measure the percentage of variation compared with the average results obtained with CHEETAH.

As can be observed in the table, the incremental effects of variables *Rch* and *Ctr* both delayed significantly the first dead of a node and HND although LND decreased. The third row details the results obtained after the modification in the selection algorithm to elect only the best candidate CHs. This enhancement achieved an important improvement in the results for FND and HND compared with CHEETAH whereas LND decreased slightly. Table VI also shows the results of the proposed algorithm employing a type-1 FLC, which does not achieve better values for HND or LND compared with the proposed method with a type-2 FLC. The last row shows the increment obtained by the proposed approach EUDFC, which includes the turn-based scheduling of the clustering algorithm.

V. CONCLUSION

In this paper, we have presented a fully unequal distributed method that can obtain a substantially better performance than other current methods, even when compared with those with a centralized approach. This improvement is due to its communication protocol and the designed clustering algorithm, which is based on an interval type-2 fuzzy system and a novel turn-based scheduling. These properties allow a more efficient use of energy on each node, without losing synchronization thanks to the CH advertisement message for each round of the turn. The input set of the fuzzy controller has been modified from the previous proposal named CHEETAH. Specifically, two new variables were introduced. The first one, *Rch* allows a better balance in CH selection than the one obtained by the efficiency variable presented in CHEETAH. The other variable used *Ctr*, which gives information about the proximity of the nodes that join a CH, has demonstrated that centrality or node density variables are really useful, mainly for the first stages. Furthermore, the selection of the best CHs from a set of candidates, which is a common solution in current work, has also improved FND and HND. Finally, as it has been commented, turn-based scheduling adds more complexity to the overall process but it improves significantly network lifetime.

We have to remark that we have introduced the precise calculation of every variable for the input variables of the eT2FC to allow future comparisons and encourage discussion about clustering in WSN. For future work, the effects of rounds with no CH, particularly when the network lifetime is nearly over will be studied. The adjustment of the clustering mechanism for heterogeneous nodes will also be addressed.

REFERENCES

[1] L. Xu, R. Collier, and G. M. P. O'Hare, "A survey of clustering techniques in WSNs and consideration of the challenges of applying such to 5G IoT scenarios," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1229–1249, Oct. 2017.

[2] M. Baniata and J. Hong, "Energy-efficient unequal chain length clustering for wireless sensor networks in smart cities," *Wireless Commun. Mobile Comput.*, vol. 2017, Sep. 2017, Art. no. 5790161.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[4] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Comput. Netw.*, vol. 52, no. 12, pp. 2292–2330, Aug. 2008.

[5] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Comput. Commun.*, vol. 30, nos. 14–15, pp. 2826–2841, Oct. 2007.

[6] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. 33rd Annu. Hawaii Int. Conf. Syst. Sci.*, Jan. 2000, p. 10.

[7] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

[8] X. Liu, "A survey on clustering routing protocols in wireless sensor networks," *Sensors*, vol. 12, no. 8, pp. 11113–11153, Aug. 2012.

[9] S. P. Singh and S. C. Sharma, "A survey on cluster based routing protocols in wireless sensor networks," *Procedia Comput. Sci.*, vol. 45, pp. 687–695, Jan. 2015.

[10] G. Devika and A. G. Karegowda, "A pragmatic study of LEACH and its descendant routing protocols in WSN," *Int. J. Comput. Intell. Inform.*, vol. 4, no. 4, pp. 300–307, Mar. 2015.

[11] R. Asorey-Cacheda, A.-J. García-Sánchez, F. García-Sánchez, and J. García-Haro, "A survey on non-linear optimization problems in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 82, pp. 1–20, Mar. 2017.

[12] R. V. Kulkarni, A. Forster, and G. K. Venayagamoorthy, "Computational intelligence in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 1, pp. 68–96, 1st Quart., 2011.

[13] S. A. Sert, H. Bagci, and A. Yazici, "MOFCA: Multi-objective fuzzy clustering algorithm for wireless sensor networks," *Appl. Soft Comput.*, vol. 30, pp. 151–165, May 2015.

[14] D. Agrawal and S. Pandey, "FUCA: Fuzzy-based unequal clustering algorithm to prolong the lifetime of wireless sensor networks," *Int. J. Commun. Syst.*, vol. 31, no. 2, p. e3448, Jan. 2018.

[15] B. Baranidharan and B. Santhi, "DUCF: Distributed load balancing unequal clustering in wireless sensor networks using fuzzy approach," *Appl. Soft Comput.*, vol. 40, pp. 495–506, Mar. 2016.

[16] P. Neamatollahi and M. Naghibzadeh, "Distributed unequal clustering algorithm in large-scale wireless sensor networks using fuzzy logic," *The J. Supercomput.*, vol. 74, no. 6, pp. 2329–2352, Jun. 2018.

[17] R. Logambigai and A. Kannan, "Fuzzy logic based unequal clustering for wireless sensor networks," *Wireless Netw.*, vol. 22, no. 3, pp. 945–957, Apr. 2016.

[18] P. Nayak and A. Devulapalli, "A fuzzy logic-based clustering algorithm for WSN to extend the network lifetime," *IEEE Sensors J.*, vol. 16, no. 1, pp. 137–144, Jan. 2016.

[19] J. Cuevas-Martinez, A. J. Yuste-Delgado, and A. Triviño-Cabrera, "Cluster head enhanced election type-2 fuzzy algorithm for wireless sensor networks," *IEEE Commun. Lett.*, vol. 21, no. 9, pp. 2069–2072, Sep. 2017.

[20] J. C. Cuevas-Martinez, M. A. Gadeo-Martos, J. A. Fernandez-Prieto, J. Canada-Bago, and A. J. Yuste-Delgado, "Wireless intelligent sensors management application protocol-WISMAP," *Sensors*, vol. 10, no. 10, pp. 8827–8849, Sep. 2010.

[21] N. Sah, "Performance evaluation of energy efficient routing in wireless sensor networks," in *Proc. Int. Conf. Signal Process., Commun., Power Embedded Syst. (SCOPES)*, Oct. 2016, pp. 1048–1053.

[22] J.-M. Kim, S.-H. Park, Y.-J. Han, and T.-M. Chung, "Chef: Cluster head election mechanism using fuzzy logic in wireless sensor networks," in *Proc. 10th Int. Conf. Adv. Commun. Technol.*, vol. 1, Feb. 2008, pp. 654–659.

[23] I. Gupta, D. Riordan, and S. Sampalli, "Cluster-head election using fuzzy logic for wireless sensor networks," in *Proc. 3rd Annu. Commun. Netw. Services Res. Conf. (CNSR)*, May 2005, pp. 255–260.

[24] W.-X. Xie, Q.-Y. Zhang, Z.-M. Sun, and F. Zhang, "A clustering routing protocol for WSN based on type-2 fuzzy logic and ant colony optimization," *Wireless Pers. Commun.*, vol. 84, no. 2, pp. 1165–1196, Sep. 2015.

[25] H. Bagci and A. Yazici, "An energy aware fuzzy approach to unequal clustering in wireless sensor networks," *Appl. Soft Comput.*, vol. 13, no. 4, pp. 1741–1749, Apr. 2013.

[26] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning —I," *Inf. Sci.*, vol. 8, no. 3, pp. 199–249, 1975.

[27] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Jan./Feb. 1985.

[28] D. Wu and J. M. Mendel, "Uncertainty measures for interval type-2 fuzzy sets," *Inf. Sci.*, vol. 177, no. 23, pp. 5378–5393, 2007.

[29] F. Zhang, Q.-Y. Zhang, and Z.-M. Sun, "ICT2TSK: An improved clustering algorithm for WSN using a type-2 Takagi-Sugeno-Kang fuzzy logic system," in *Proc. IEEE Symp. Wireless Technol. Appl. (ISWTA)*, Sep. 2013, pp. 153–158.

[30] V. Rajaram, S. Srividhya, and N. Kumaratharan, "Impact of fuzzy inference system for improving the network lifetime in wireless sensor networks—A survey," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, Apr. 2018, pp. 933–937.

[31] A. Barua, L. S. Mudunuri, and O. Kosheleva, "Why trapezoidal and triangular membership functions work so well: Towards a theoretical explanation," *J. Uncertain Syst.*, vol. 8, no. 3, pp. 1–6, 2014.

[32] O. A. M. Ali, A. Y. Ali, and B. S. Sumait, "Comparison between the effects of different types of membership functions on fuzzy logic controller performance," *Int. J.*, vol. 76, pp. 76–83, Mar. 2015.

[33] I. Demirkol, C. Ersoy, and F. Alagoz, "MAC protocols for wireless sensor networks: A survey," *IEEE Commun. Mag.*, vol. 44, no. 4, pp. 115–121, Apr. 2006.

[34] P. K. Pal and P. Chatterjee, "A survey on TDMA-based MAC protocols for wireless sensor network," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 4, no. 6, pp. 219–230, Jun. 2014.

Antonio Jesús Yuste-Delgado received the Ph.D. degree in telecommunication engineering from the Universidad de Málaga in 2012. He is currently an Associate Professor with the Department of Telecommunication Engineering, Universidad de Jaén, Spain. His main research interests include wireless sensor networks and applications of fuzzy system in computer networks. Nowadays, he is a member of the research group Telematics System Engineering Group and the Center of Advanced Studies in Information and Communication Technologies, Universidad de Jaén.

Juan Carlos Cuevas-Martinez received the M.S. degree in telecommunication engineering from the Universidad de Málaga in 2002 and the Ph.D. degree in telecommunication engineering from the Universidad de Jaén, Spain, in 2014. He is currently an Assistant Professor with the Department of Telecommunication Engineering, Universidad de Jaén. His main research interests include wireless sensor networks and networking and soft computing. Nowadays, he is a member of the research group Multimedia and Multimodal Processing Group, Universidad de Jaén.

Alicia Triviño-Cabrera was a Telecommunication Engineer and a Computer Science Engineer with the University of Málaga, Spain, in 2002 and 2008, respectively. Her thesis, which was defended in 2007, focused on wireless networking. In 2010, she started teaching at the University of Málaga, where she is currently an Associate Professor. Part of her research work has been conducted in prestigious centers, such as the Samsung Advanced Institute of Technology, South Korea, in 2004, Laboratory of Communications, University of Coímbra, in 2007, Orange Laboratories Research and Development, France, in 2009, and Aston University, U.K., in 2017.