International Journal of Engineering & Technology, 7 (4) (2018) 5926-5933

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.22843
Research paper

Modified firefly with fuzzy based clustering algorithm for cluster-based networks

V. Hema [1] *, K. Kungumaraj [2]

[1] Research Scholar, Mother Teresa Women's University, Kodaikanal, Tamil Nadu, India
[2] Assistant Professor, Dept., of Computer Science, Mother Teresa Women's University, Kodaikanal
*Corresponding author E-mail: hema.rengamani78@gmail.com

Abstract

Cluster-based network comprises a group of clusters with each cluster contains a collection of nodes. The control structures offer effective utilization of resources while managing large dynamic networks. Cluster based network is mainly used for effective load balancing. Different kinds of cluster-based architectures presented in the study for different usage. In cluster based networks, the choice of cluster heads (CH) in cluster based network is a challenging task. Presently, meta-heuristic algorithms become very popular and employed to select CHs effectively. In this paper, we introduce a modified firefly with fuzzy based clustering algorithm and it operates in two levels: modified firefly algorithm (MFA) for candidate CH selection and fuzzy logic for final CH selection. The traditional FF algorithm in modified by the inclusion of tumbling effect to develop MFA and it uses processing capability as a measure to identify the candidate CHs. Next, five input parameters named as residual energy, neighboring node distance, distance to main server, node centrality and node degree to select the final CHs from candidate CHs. An extensive experimentation takes place to verify the goodness of the MFFCA in terms of different performance measures and the results depicted that the MFFCA outperforms the compared clustering techniques.

Keywords: *Fiber Optic Communication; Dictionary Based Coding; Data Compression; Textual Dataset*.

1. Introduction

A communication network is a unit with dynamic state changes in an irregular way. The network should have the ability to retain with the modifications in intrinsic characteristics like connectivity, capacity and offered load to satisfy the performance goals prescribed. The control functions govern the network's performance with the following competitive goals: faster and precise response when adapt with the network nature to the present network state, less resource utilization. The networks which continuously modify the connectivity (e.g., mobile networks) or many configuration parameters are best served through self-organized controlled structure [1]. They are constructed and undergo maintenance by the nodes, moreover offer maximum network availability, quicker response to state modifications, low probability of configuration errors. The proper network control structure and techniques to use the kind of structures are selected based over the control functions are performed, network size and the anticipated frequency and magnitude of alterations in the state of the network [2].

Cluster-based control structure [3] offer effective resource utilization during the control of networks which are highly dynamic. In the cluster based control, the physical network is converted to a virtual network of interlinked node clusters [14]. There could be more than one controllers are present in each cluster and it has the characteristic to take decisions for nodes present in the cluster, develop and deal out representation of cluster state for exterior uses [15].

Cluster-based control networks enhances the effectiveness of resource utilization with developing contexts [4] for: handling wireless communication between many nodes for reducing channel contention, creating routing backbones to minimize the radius of the network, abstract the network state data to minimize its variability and quantity. An architecture of cluster based network is depicted in Figure 1. The network contains a number of sensor nodes and some of the nodes are grouped together to form a cluster. Once the data is available at the CH, upon request or automatically, the CHs will forward the data to BS or the intended user based on the application requirement.

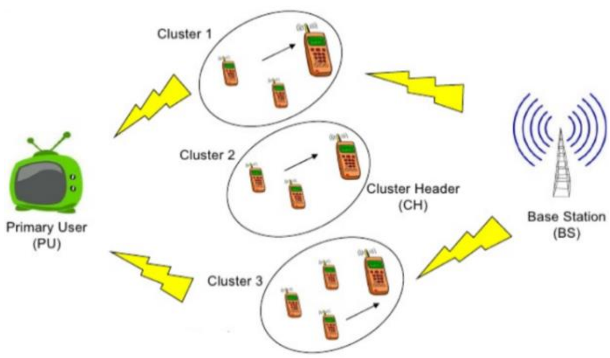


Fig. 1: A Cluster-Based Network Architecture.

Clustering is a very important task in cluster based networks [5]. Various clustering methodologies have been employed to cluster the network and choose the proper CHs for each cluster [6, 7]. Computational intelligence approaches such as reinforcement learning [9], bio inspired algorithms [16-19], neural networks [8] and fuzzy logic can be employed to cluster the network. The nodes in the network are inter-dependent to one other and hence it leads to the development of inter-related measures for the identification of proper CHs in the network. Fuzzy logic finds helpful in different cases because of higher level of uncertainty [10], [11]. Fuzzy logic is extensively employed in decision making process and it is advantageous to traditional methods in terms of complexity, implementation cost, flexibility, faster and low requirements. Here, fuzzy logic is used to select CHs [12]. In general, fuzzy logic utilizes various input variables like residual energy, processing capability, work load, node degree and on, to identify proper CHs. Presently, different works has been done based on bio-inspired algorithms available in the literature. In this paper, we devise modified firefly with fuzzy based clustering algorithm (MFFCA) and it operates in two levels: modified firefly algorithm (MFA) for candidate CH selection and fuzzy logic for final CH selection. The traditional FF algorithm in modified by the inclusion of tumbling effect to develop MFA and it uses processing capability as a measure to identify the candidate CHs. Next, five input parameters named as neighboring node distance, distance to main server, residual energy, node degree and node centrality to select the final CHs from candidate CHs. An extensive experimentation takes place to verify the goodness of the MFFCA in terms of different performance measures and the results depicted that the MFFCA outperforms the compared clustering techniques. In short, the contribution of the paper is listed as follows:

- Devise a MFFCA to identify the CHs and construct clusters in the cluster-based network
- Present MFA with tumbling effect to select a list of candidate CHs which has the capability to become final CHs
- Present fuzzy logic with five input parameters to select the final CHs from the tentative CH selection
- Validate the results of MFFCA using different performance measures

The paper organization is defined as follows: section 2 discusses the different clustering architectures with necessary explanation and diagrams. Section 3 introduces the proposed MFFCA in detail and is validated using different measures in section 4. At the end, section 5 defines the conclusion.

2. Cluster based network architectures

To attain particular purposes, the cluster based control structures are used in cluster based networks and are demonstrated below.

2.1. Link cluster architecture

In multiple-access broadcast environments, this kind of architecture is employed. For diminishing interference, unique cluster nodes are produced in a way the transmissions are managed in a contention-free approach. Every node in the network separately arranges itself into interconnected clusters. Fig. 2 depicts that every cluster comprises a CH, zero or more actual nodes, one or more gateways. Within a cluster, the CH allocates resources and transmissions. The adjacent clusters are connected by gateways. To form an association to a member of the other, a gateway might connect directly two clusters as a member mutually; otherwise connect indirectly two clusters as a member of one. Hence, it contains disjoint and overlapping clusters. Over a physical network, the nodes set up a link-clustered structure which is shown below.

For CH election, there are two algorithms projected in the literature, named as connectivity-based clustering and identifier-based clustering. The implementation may be either centralized or distributed. The highest count of neighbors (connectivity based) or the highest numbered identifier (identifier based) is selected as a CH for the cluster comprising which node as well as its one-hop neighbors with the centralized version.

- Through broadcasting a set of neighbors with bi-directional connectivity which they are capable to hear and through receiving broadcasts from neighbors.
- Forming clusters by electing CHs.
- Between clusters agreeing over gateways.

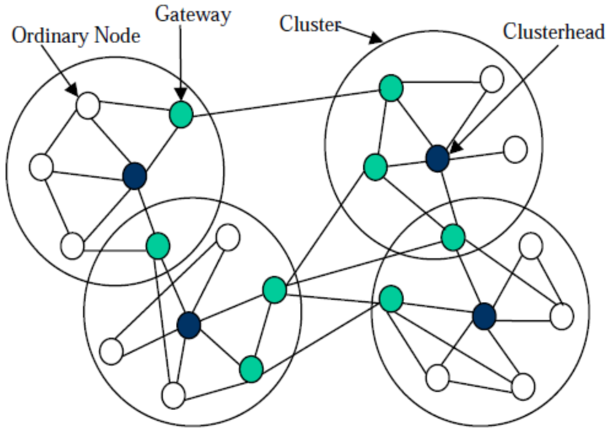


Fig. 2: Link-Clustered Architecture.

A node chooses itself if it comprise the highest numbered identifier or lowest within the neighbor with the clustering in distributed version of identifier. While in connectivity-based clustering, a node is eligible to be a CH if it has association with all the neighbouring nodes which are not selected as CHs. For the two algorithms discussed above, the clustering variants of algorithms which forms disjoint clusters are demonstrated. In such situations, CHs are employed as support to cluster formation and not chosen as CHs.

2.2. Near-term digital radio (NTDR) network

In huge tactical networks, Near-Term Digital Radio (NTDR) networking is modeled and deployed. This is employed as a cluster methodology for backbone formation. Fig. 3 depicts NTDR forms a group of cluster, every one comprising of a CH, that are associated together forms a routing backbone. As similar to the link-clustered architecture, it contains single level of clusters with nodes with one hop of a CH. For direct inter-cluster communication, CHs function serves as gateways. Excluding the neighbor node each and every intra-cluster communication has to traverse by the CH as one hop to every other.

Owing to the outage of CHs otherwise repeated movement of the node all the nodes can become CHs when there is a need quickly out of the node interconnection modifies. Through receiving and broadcasting beacons occasionally, every node keep the way of the neighbors with that it comprise a bi-directional connectivity. NTDR does not get beacon out of any CH otherwise if it gets beacons advertising two contrast partition identifiers, it selects itself as CH. The NTDR method attempts to become CH by limiting the count of nodes concurrently by the following methods: If a node that has any of the above constraints for becoming CH, it waits for an interval of time and test the constraints again. After the waiting period if it exists, the node consider as the CH role. By proclaiming its condition immediately, every new CH issue beacon in rapid succession.

A node seeking cluster affiliation prefer clusters such that

- The node and the CH belong to the similar organization.
- The signal from the CH is transmitted at low power however received at high strength.
- The resulting cluster size is comparatively small.
- A cluster member remains affiliated with its selected CH awaiting one of the following occurs:
- The CH relinquish its role.
- The CH's beacons no longer list the member.
- The received signal strength out of the CH is inappropriately low.

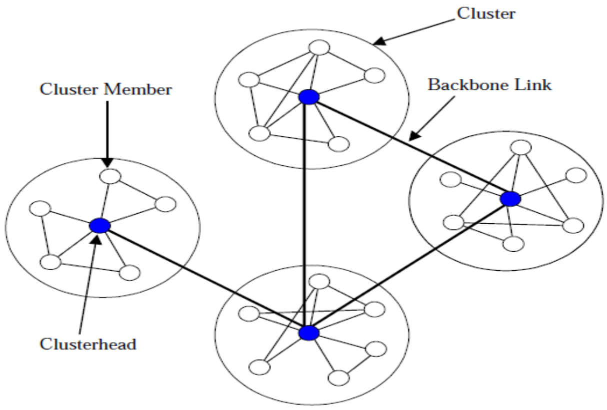


Fig. 3: NTDR Network Architecture.

2.3. Virtual subnet architecture

A collection of many disjoint routing backbones to offer load balancing and connectivity in fault-tolerant in a multihop mobile wireless network is virtual subnet architecture. The network is segmented into a collection of disjoint physical subnets cluster based on the node locality. To organize a virtual subnet, member of various physical subnet is grouped together, in which it span all of the physical subnets. To diminish interferences while communicating, all the virtual subnets and neighboring physical subnet are assigned with various frequencies. Together the maximum number Q of the virtual subnet and the maximization number P of the physical subnets are predefined in network. As depicted in Fig 4, every node is an exact member of more virtual subnets or one physical. A node may belong to numerous virtual subnets if it frequently communicates with the subnets' members. The virtual and physical subnet affiliations are identified by the node address with the cluster-based control organization through suffixing the address demonstrating the virtual subnet and by prefixing the address demonstrating the physical subnet. A node may contain different address suffix and also multiple addresses if it is a member of various virtual subnets. For structuring and employing numerous overlaid routing backbones in ad hoc network, the virtual subnet architecture presents a structure. Hence, prior to recognizing a original ad hoc network, a full collection of network methods named as distribution of address and routing information, subnet clustering, computing of routes, frequency assignment, and packet forwarding are well suited with this organization.

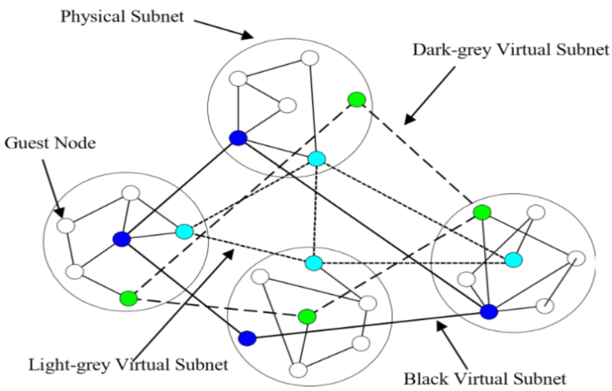


Fig. 4: Virtual Subnet Architecture.

2.4. Recently proposed text compression algorithms

To change the conventional coding methodologies like arithmetic coding and Huffman coding, [13] projected a neural network (NN) based compression method. The projected technique depends on the procedure of "predictive or model based coding". It is used to compress few newspaper articles and achieves enhanced CR when compared to LZ method. However, when compared to the conventional techniques, it shows slow performance. The two issues present in the compression of dynamic databases are addressed by [14]. The issues are managing the document insertions and memory need at the decoding side. Compression with ant dictionaries (DCA) algorithm is built by [15] which uses "negative information" about the text by means of ant dictionaries. The advantages of these methods are faster compression rate for predefined sources, synchronization property results to effective parallel compression and faster decompression rate. To improve the efficacy of the given compression methods, few pre-processing methods are applied before compression. A pre-processing method was projected by [16] for text compression. It needs no external dictionaries, because it is language independent. It works under algorithm that operates in a series manner named as word replacement, capital letter conversion, phrase replacement, end of line (EOL) coding, recording of alphabets. But, the pre-processing cost seems to be high. A new compression method is presented in [17] for small text files. It is modeled specially for very tiny text minimization. Other compression method known as b64 pack for short messages is projected in [18]. It works under two steps and it is a lightweight and efficient method. In the initial step, it transfers the actual text to a compressed format. In the subsequent step, it includes the transformation which reduces the message size through a built-in fraction of the actual size. The pre-processing algorithm projected in [19] for BWT coding, pre-processing algorithm for Huffman coding, versions of LZ coding, arithmetic coding. With a reduced file size, a genetic reversible transformation is commenced to transfer a text file to another format. For static text compression through the relaxation of prefix property of the dictionary, [20] commenced a greedy method for static text compression. To achieve increased speed in distributed systems, a finite state machine (FSM) implementation of greedy based compression with an arbitrary dictionary is projected. To compress the sentiment sentences in aspect based sentiment analysis, a novel compression method known as sent comp is projected in [21]. In the domain of text compression, [22] uses data mining tools. By combining the frequent itemset mining (FIM), Huffman coding is enhanced. For frequently occurring patterns, shorter codewords has been assigned. A graph based technique is used for searching series of characters employed in compression procedure presented in [23]. In one pass of the graph, this scheme develops a graph in one pass of the text and mines each and every pattern that is significant for compression. For textual data, [24] projected a new FIM based Huffman coding technique employing hash table (FPH2). Character based method is employed in the conventional Huffman coding methods whereas optimized (pruned) set patterns are implicated in the coding process.

3. The proposed algorithm

MFFCA operates in two levels: modified firefly algorithm (MFA) for candidate CH selection and fuzzy logic for final CH selection. The traditional FF algorithm in modified by the inclusion of tumbling effect to develop MFA and it uses processing capability as a measure to identify the candidate CHs. Next, five input parameters namely, residual energy, neighboring node distance, distance to main server, node degree and node centrality to select the final CHs from candidate CHs.

3.1. MFA based candidate CH selection

Depending on the brightness of the flashes, Xin-She commenced FF algorithm between the year 2007 and 2008 [13]. The above features constraints the visual distance of FFs. Over many hundred meters, FF is capable to communicate during night. To create a new optimization algorithm, the FF flashes can be formulated in such as way that it may be incorporated and the objective function to be enhanced. The FF technique follows three rules:

- FFs are unisex
- Attractiveness depends on the brightness of the flashlights; the lesser bright FF will be attracted towards the brighter FF. Hence, the FF attractiveness is a monotonically decreasing

function of the distance $r_{ij} = d(x_j, x_i)$ to the selected FF, e.g. the exponential function.

$$r_{ij} = \| x_i - x_j \| \tag{1}$$

$$\beta = \beta_0 e^{-\gamma r_{ij}} \tag{2}$$

Where β_0 is the attractiveness at $r_{ij} = 0$ and γ is the light absorption coefficient at the source.

- The movement of a FF i is concerned to other FF j and is estimated as

$$x_{i,k} \leftarrow (1-\beta)x_{i,k} + \beta x_{j,k} + u_{i,k} \tag{3}$$

$$u_{i,k} = \propto \left(rand\ 1 - \frac{1}{2}\right) \tag{4}$$

The specific FF x_i with maximum fitness will moves in an arbitrary manner based on the following equation.

$$x_{i,k} \leftarrow (1-\beta) \tag{5}$$

$$x_{i^{max},k} \leftarrow x^{+}_{i^{max},k} u_{i^{max},k} \tag{6}$$

$$u_{i^{max},k} = \propto \left(rand\ 2 - \frac{1}{2}\right) \tag{7}$$

Where $rand\ 1 \approx U(0,1) rand\ 2 \approx (0,1)$ are random numbers obtained from uniform distribution? Brightness of a FF is subjective or estimated by the landscape of the objective function. The Cartesian distance between two FFs i and j $x_i$and x_j can be computed as

$$r_{ij} = \| x_i - x_j \| = \sqrt{\sum_{k=1}^{d} \left(x_{i,k} - x_{j,k}\right)^2} \frac{1}{\Gamma^m} \tag{8}$$

Where $x_{i,k}$ is the k^{th} component of the spatial coordinate x_i of the i^{th} FF. The r_{ij} in 2D space is estimated in Eq. (9).

$$r_{ij} = \sqrt{(x_i - x_j)^2 - (y_i - y_j)^2} \tag{9}$$

The movement of a FF i is mesmerized to a brighter FF and can be expressed as

$$x_i = x_i + \beta_0 e^{-\gamma r_{i,j}^2}(x_j - x_i) + \alpha \in_i \tag{10}$$

where 2nd term denotes attraction and 3rd term indicate the randomization. Here, α is the randomized parameter and $\in_i$ is a vector of random numbers derived from Gaussian or uniform dispersion. As each FF works in an autonomous way, it can be applicable for parallel implementation.

By employing FF algorithm, the candidate nodes may be chosen that depends on actual processing capability of the nodes. A firefly containing enhanced light intensity, i.e. node with high computation power would be attracted by other firefly and it would be chosen as candidate CH nodes. Depending on the processing capability measure, the FF algorithm ranks the nodes as candidate CH nodes. But, in some cases, when none of the brighter FF is present, then the FF will starts to move randomly. So, the FF algorithm is modified with the tumbling effect to improve the exploitation of search space.

The movement of bacteria in the human intestine during the search process of locating rich nutrients apart out of risky places is carried out using the locomotory organelles called flagella by chemotactic movement in either means [14], i.e., swimming (in the identical direction as the preceding step) or tumbling (in an reverse direction from the preceding one). The chemotactic movement of bacterium is defined in Eq. (11), which computes the output defined in Eq. (12)

$$H_t = H_{t-1} + H_t \times \frac{\Delta H_t}{\sqrt{\Delta^T}} \tag{11}$$

$$O_t = \sigma_o(W_o H_t + B_o) \tag{12}$$

where Δ^T is the random number produced in the range of [-1, 1]. Using the tumbling effect, the FF algorithm can easily explore the search space in all the directions and its efficiency will be significantly increased. Subsequently, to select final CHs out of the candidate CHs, fuzzy logic would be used. Algorithm 1 demonstrates the pseudo code of the MFA algorithm.

Algorithm 1: Calculate_distance_Firefly(K.dNN)

Begin Algorithm
Step 1: Initialize, *f(dNN)*←Objective Function, dNN_i← Generate Initial Population (i=1,2,3…n), δ←Light Intensity, *I*← Formulate Light Intensity *(I α f(x))*, *Ɣ*←Absorption Coefficient.
Step 2:Repeat this **until** *T<Max_Generation*
For *i = 1*to *n* do
    For *j = 1* to *n* do
        If *(I_j>I_i)*
Vary attractiveness with distance r via exp(*-Ɣr)*
Tumbling effect
Shift Firefly *i* approaching towards *j*
Assess current solution and update *δ*
End If
End For
End For
Step 3: Find the Best Cost and Rank Fireflies
Step 4: Return n nearest Best Cost NDN nodes
End Algorithm

3.2. Fuzzy based final CH selection

In cluster-based network, fuzzy logic is widely used in the present times. When uncertainties are high and network is unstable, several protocols employ fuzzy logic in such cases. A fuzzy logic is less complex, flexible, needs reduced computational resources, fault tolerant, many fuzzy logic based protocols that are built to deal the clustering issue.

In cluster construction, there are two levels named as cluster formation and CH selection. In the construction step, to create balanced clusters, BS executes fuzzy logic. By equally distributing the CHs in the whole network, the balanced clusters are attained. Within a cluster based network, the correct selection of CH is a highly important task. Fuzzy logic with five input parameters is employed for CH selection named residual energy, neighboring node distance, distance to main server, node degree and node centrality as input parameters and final CH as the output variable. Residual energy points out the available energy level to maintain operation in the network. The distance measure among the neighboring and node is the neighboring node distance. The count of neighbor nodes denotes to the node degree in the vicinity. The enhanced rate of node degree denotes the higher count of neighbors and intra-cluster communication would be high. To communicate with other nodes, node centrality is also significant where it describes how far the sensor nodes are centrally positioned. There is a high chance to become CH, when nodes with a higher rate of centrality, hence it diminishes the intra-cluster communication distance. To choose CHs effectively, the five input variables is highly significant. Usually, during the clustering process, these parameters would be employed.

3.2.1. Fuzzification of input variables

Residual energy, neighboring node distance, distance to main server, node degree and node centrality are the input parameters of MFFCA. Firstly, the crisp input maps it to an exact linguistic parameter of the fuzzy parameters. Table 1 depicts the linguistic parameters of the input and output variables.

Table 1: Parameters and Their Possible Values

	Parameters	Linguistic Variables
Input Parameters	Residual Energy (RE)	Low, Medium , High
	Distance to main	Close, Medium, Far

	server (D)	
	Neighboring Node Distance (NND)	Close, Medium, Far
	Node Centrality (NC)	Low, Medium, High
	Node Degree (ND)	Low, Medium, Many
Output Parameters	Final CH (FCH)	Very low, Low, Rather low, Medium, Rather High, High, Very high

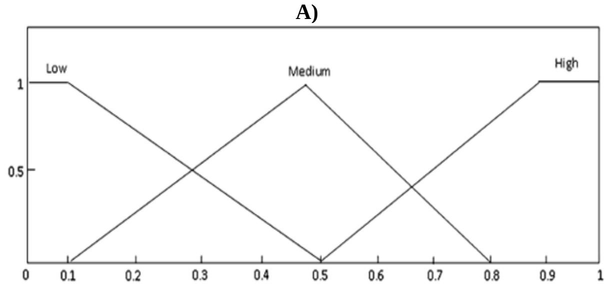
3.2.2. Membership functions

Figs 5(a)-(g) depicts the membership function of the output and input parameters. The trapezoidal membership function is employed for boundary parameters (very high, close, far, high, very low, low, very small and very large) and triangular membership is employed for the middle variables. The triangular and trapezoidal membership functions are expressed in Eq. (13) and Eq. (14).

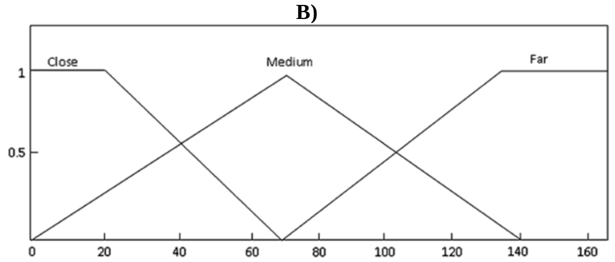
$$\mu_{A1}(x) = \begin{cases} 0 & x \le a1 \\ \dfrac{x - a1}{b1 - a1} & a1 \le x \le b1 \\ \dfrac{c1 - x}{c1 - b1} & b1 \le x \le c1 \\ 0 & c1 \le x \end{cases} \qquad \text{Eq. (13)}$$

$$\mu_{A1}(x) = \begin{cases} 0 & x \le a1 \\ \dfrac{x - a2}{b2 - a2} & a2 \le x \le b2 \\ 1 & b2 \le x \le c2 \\ \dfrac{d2 - x}{d2 - c2} & c2 \le x \le d2 \\ 0 & d2 \le x \end{cases} \qquad \text{Eq. (14)}$$

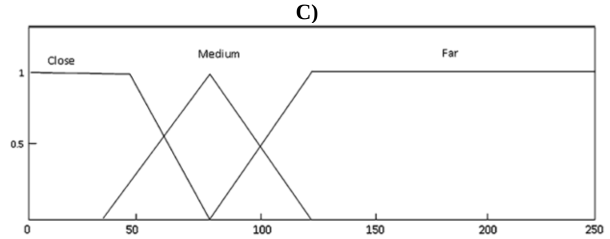
A)



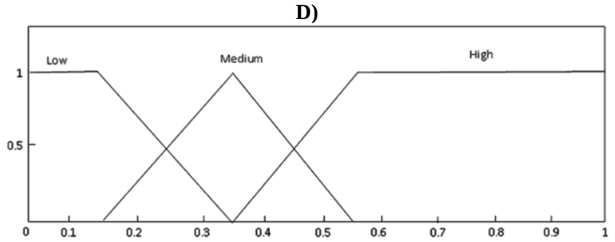
B)



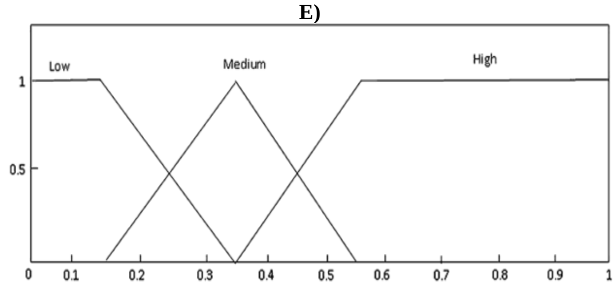
C)



D)



E)



F)

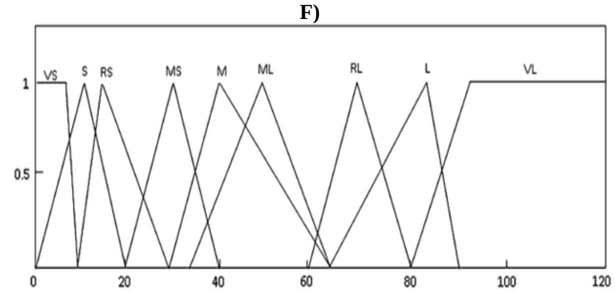


Fig. 5: Membership Functions A) RE, B) D, C) DNN, D) ND, E) NC, F) FCH.

3.2.3. Fuzzy decision blocks/ rule base

By means of linguistic parameters the rule base table comprises a count of input and output variables. By employing a collection of if then-rules, the input in addition to output parameters are combined. Inference engine is the one to estimate the rules. Table 2 depicts the fuzzy rule base. For example, in Eq. (15), a sample given if-then condition. While the five inputs be A_1^iANDA_2^iAND A_3^iAND A_4^i AND A_5^i, then the output will be B_1^i and B_2^i.

$$\mathrm{Rule(i)IF} x_1 \mathrm{is} A_1^i \mathrm{AND} x_2 \mathrm{is} A_2^i \mathrm{AND} x_3 \mathrm{is} A_3^i \tag{15}$$

$$\mathrm{AND} x_4 \mathrm{is} A_4^i \mathrm{AND} x_5 \mathrm{is} A_5^i \mathrm{THEN} y_1 \mathrm{is} B_1^i \ \mathrm{AND} y_2 \mathrm{is} B_2^i$$

Where 'i' is the ith rule in the rule base table, A_1, A_2,...., A_5 is the fuzzy set of x_1, x_2, .., x_5. Since 5 input parameters are employed, the count of rules would be 243 in total. They are made simple and efficient Mamdami inference system.

3.2.4. Defuzzification

For defuzzification procedure, the Centroid of Area (COA) method is applied and is expressed in Eq. (16). This procedure converts the fuzzified output parameters to a crisp rate representing the node probability to become a CH and a cluster size.

Each individual node sends a CH_CANDITATE_MSG to further nodes in its communication radius when PBCH is estimated. The node ID and rate of PBCH are enclosed in the message.

$$\mathrm{COA} = \frac{\int \mu_A(x).xdx}{\int \mu_A(x).dx} \tag{16}$$

Table 2: Fuzzy Rule Base Table

Rule	Input variable					Output variable
	RE	D	DNN	ND	NC	FCH
1	L	C	F	L	L	L
2	L	C	F	L	L	Very L
3	L	C	F	L	L	Very L
.						
20	L	C	C	L	L	L
.						
28	L	M	F	L	L	Rather L
57	L	F	F	L	L	L
61	L	F	F	L	L	Rather L
.						
63	L	F	F	L	L	Very L
.						
82	M	C	F	M	M	M

.						
109	M	M	F	M	M	Rather H
.						
127	M	M	C	M	M	Rather L
.						
136	M	F	F	M	M	Rather H
.						
190	H	M	F	H	H	H
.						
217	H	M	F	H	H	Very H
218	H	F	F	H	H	Rather H
219	H	F	F	H	H	M
223	H	M	F	H	H	Rather H
224	H	F	F	H	H	M
225	H	F	F	H	H	Rather L
226	H	M	M	H	H	Very H
227	H	F	M	H	H	H
228	H	F	M	H	H	Rather H
232	H	M	M	H	H	M
233	H	F	M	H	H	M
234	H	F	M	H	H	M
.						
238	H	M	C	H	H	H
239	H	F	C	H	H	Rather H
240	H	F	C	H	H	M
241	H	M	C	H	H	Rather H
242	H	F	C	H	H	M
243	H	F	C	H	H	M

L: Low, M-Medium; H-High, C-Close; F-Far

The nodes comprising higher probability are selected as a CH and send CH_WON to the closer nodes. Out of its nearby nodes, the node may multiply CH_WON. In such cases, it send CH_JOIN message and connect to the closer CH. Prior to accept new members, the closest CH ensure the present cluster size while receiving CH_JOIN message. It accept the fresh cluster member through replying CM_ACCEPT message when the entire count of present cluster members is not higher than the estimated cluster size otherwise, it will send CH_REJECT message.

It retransmits a CM_JOIN message to the next closer CH except the freshly rejected CH in addition this process continues until it joins to a new CH, once node receives a CM_REJECT message. It chooses itself as CH, in some cases while a node fails to join to any other CH inside its coverage region 'R'. Hence, no nodes are inaccessible in cluster-based network and each node belong to a cluster. After many rounds, the CHs rotation takes place to preserve the CHs from premature death. It takes place while the residual energy of the CH comes under a rate (15% of initial value). New CH would be chosen over the probability of becoming CH, once the residual energy of a CH crosses the threshold rate. Hence, this removes the premature death of CH and lead to enhance the network lifespan.

4. Performance evaluation

To assess the highlights of the MFFCA, it is tested in a network setup and different metrics are used to investigate the results of the applied protocol. The MFFCA is implemented in MATLAB (R2013a). The simulation parameter is given in Table 3. A network of 200 nodes deployed in the area of 100x100m^{2}as shown in Fig. 6 and the cluster construction is shown in Fig. 7.

Table 3: Simulation Parameters

Parameters	Value
Area	100x100
Number of nodes	200
E_0	0.5J
E_{elec}	50nJ/bit
ε_{fs}	100pJ/bit/m^2
ε_{fs}	100pJ/bit/m^2
Packet size	4000bits

The results attained by the MFFCA in terms of remaining energy level, number of alive nodes, throughput and number of clusters formed are shown in Fig. 8-10. From Fig. 10, in terms of remaining energy level, MFFCA spent its 100% energy at the 2435 round. At the same time, it is seen that the energy level is gradually reduced from its 100% to 0%. When it reaches to 0%, the node stops operating and become a dead node.

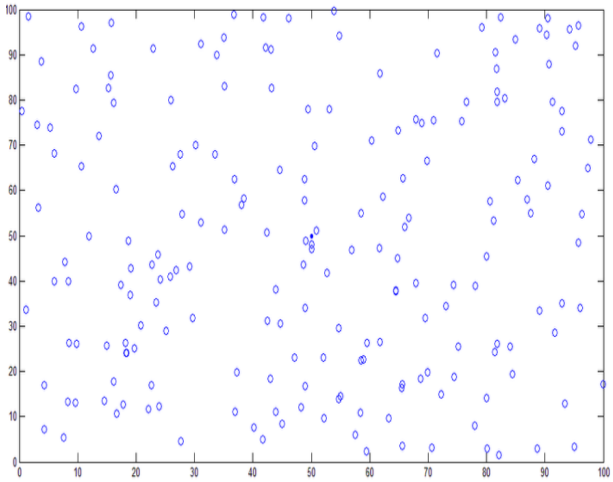


Fig. 6: Node Deployment.

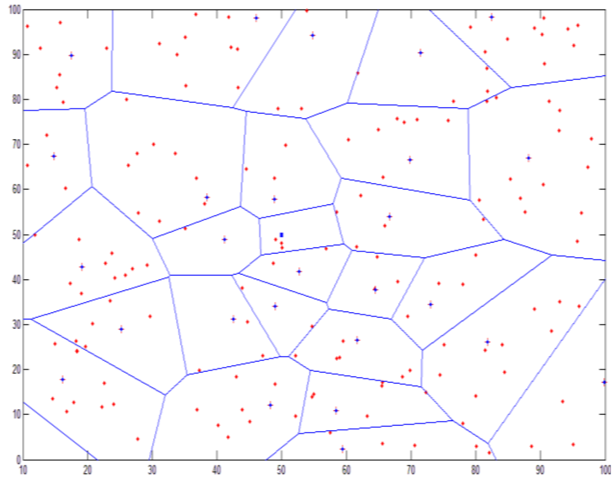


Fig. 7: Cluster Construction.

Fig. 9 depicts the count of alive nodes on rounds by the MFFCA. The MFFCA manages to perform well and kept all the nodes alive till 1474 round. After that, the alive nodes start to exhausts its total energy and become dead node. At the round n umber of 2435, each and every 200 nodes in the network becomes dead node and the network becomes inoperative.

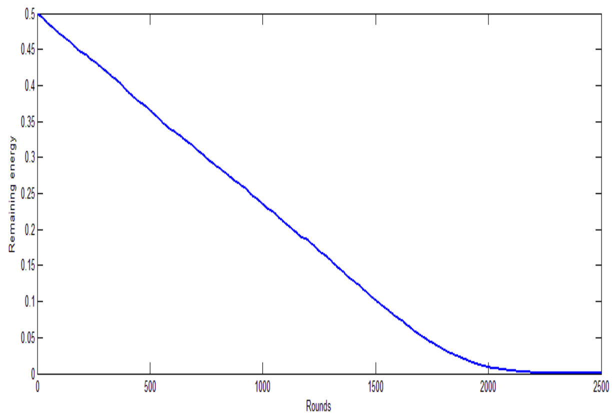


Fig. 8: Average Residual Energy By MFFCA.

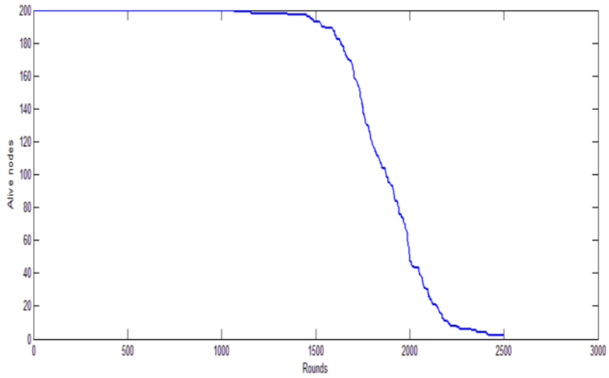


Fig. 9: Alive Nodes by MFFCA.

The throughput attained by the MFFCA algorithm is depicted in Fig 10. From the fig, it is obvious that the throughput is high. After crossing 75% of rounds, it starts gradually decreasing due to the increasing number of dead nodes. The throughput obtained by MFFCA reaches the lowest value at the round number of 2435.

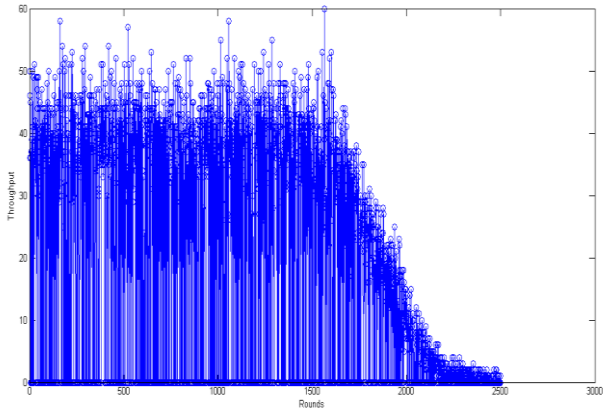


Fig. 10: Throughput by MFFCA.

Fig. 11 shows the comparative results of different clustering algorithms in terms of lifetime, i.e. count of nodes alive in order to rounds. From the Fig. 11, it is depicted that the MFFCA has higher count of alive nodes when compared to the other methods. The number of alive nodes attained by the PSO algorithm shows its inferior performance over network lifetime. At the same time, the FF algorithm shows slightly better performance than PSO algorithm, but poor performance over MFA and MFFCA. In line with, the MFA tries to higher number of alive nodes over the compared methods except the MFFCA. The MFFCA shows the highest number of alive nodes under several rounds which indicates its performance enhancement on network lifetime.

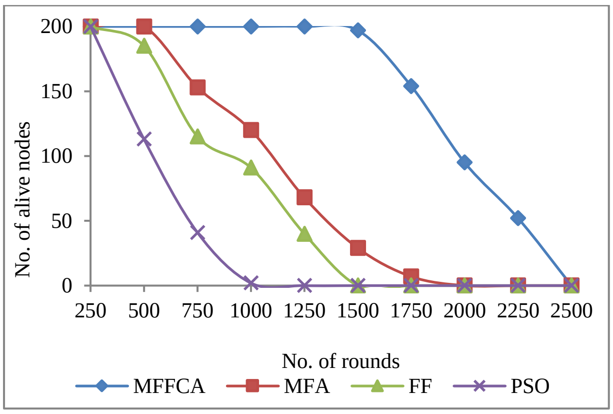


Fig. 11: Comparative Results in Terms of Alive Nodes.

Fig. 12 shows the comparative results of different clustering algorithms interms of average residual energy. For better performance, the value of average residual energy should be high. From the Fig., it is clear that the MFFCA maximum residual energy compared to the other methods. The average residual energy attained by the PSO algorithm is very low compared to the other methods. Consequently, the FF algorithm shows slightly better performance than PSO algorithm, but not better than MFA and MFFCA. In line with, the MFA tries to higher residual energy over the compared methods except the MFFCA. But, the MFFCA shows superior performance compared to all the other methods which shows its energy efficiency.

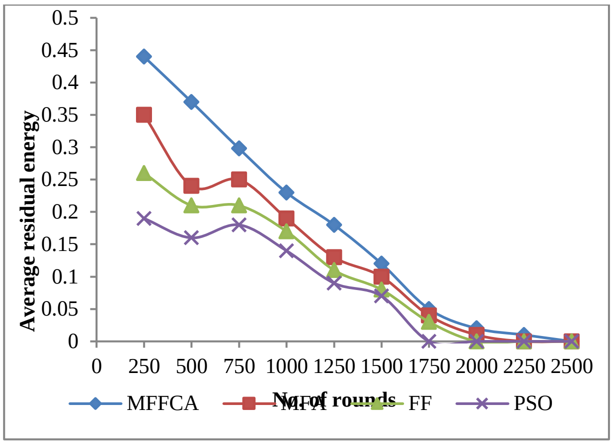


Fig. 12: Comparative Results in Terms of Energy Efficiency.

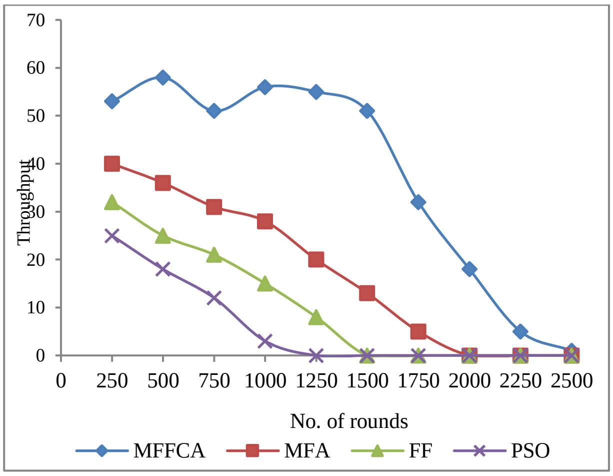


Fig. 13: Comparative Results in Terms of Throughput.

Fig. 13 shows the comparative results of different clustering algorithms in terms of Throughput, i.e. number of packets successfully reached at the destination. From the Fig., it is evident that the MFFCA achieved higher throughput compared to other methods. The throughput achieved by the PSO algorithm showed its worst performance. Likewise, the FF algorithm has slightly higher throughput than PSO algorithm, but not greater than MFA and MFFCA. At the same way, the MFA obtains better throughput over the compared methods except the MFFCA. The MFFCA shows maximum throughput which indicates the reliable performance of the projected technique. From the experimental outcomes, the MFFCA is found be energy efficient, reliable and also maximizes the lifetime of the cluster based network.

5. Conclusion

This paper has introduced a MFFCA including MFA for candidate CH selection and fuzzy logic for final CH selection. The traditional FF algorithm in enhanced using tumbling effect to develop MFA and it uses processing capability as a measure to identify the candidate CHs. Next, five input parameters named as neighboring node distance, distance to main server, node degree, residual energy and node centrality to select the final CHs from candidate CHs. From the experimental results, the MFFCA is better than the MFA, FF and PSO algorithms in terms of energy efficiency, network lifetime and throughput.

References

[1] V.S. Pai, M. Aron, G. Banga, M. Svendsen, P. Druschel, W. Zwaenepoel, and E. Nahum, "Locality-aware request distribution in cluster-based network servers", *ACM Sigplan Notices*, vol. 33, no. 11, pp.205-216, 1998. https://doi.org/10.1145/291006.291048.

[2] M. Aron, D. Sanders, P. Druschel, and W. Zwaenepoel, "Scalable content-aware request distribution in cluster-based network servers", In *Proceedings of the 2000 Annual USENIX technical Conference* (No. LABOS-CONF-2005-025), 2000.

[3] Zhu, H., Tang, H. and Yang, T., "Demand-driven service differentiation in cluster-based network servers", In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE* (Vol. 2, pp. 679-688). IEEE, 2001.

[4] W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan, An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on wireless communications*, vol. 1, no. 4, pp.660-670 2002. https://doi.org/10.1109/TWC.2002.804190.

[5] S. Arjunan, and S. Pothula, "A survey on unequal clustering protocols in wireless sensor networks", *Journal of King Saud University-Computer and Information Sciences*.

[6] R. Xu, and D. Wunsch, "Survey of clustering algorithms", *IEEE Transactions on neural networks*, vol. 16, no.3, pp.645-678, 2017. https://doi.org/10.1109/TNN.2005.845141.

[7] M.T. Hagan, H.B. Demuth, M.H. Beale, and O. De Jesús, *Neural network design*, vol. 20, Boston: PwsPub, 1996.

[8] R.S. Sutton, and A.G. Barto, *Introduction to reinforcement learning* (Vol. 135). Cambridge: MIT press, 1998.

[9] G. Klir, and B. Yuan, *Fuzzy sets and fuzzy logic* (Vol. 4). New Jersey: Prentice hall, 1995. https://doi.org/10.1109/45.468220.

[10] J. Yen, and R. Langari, *Fuzzy logic: intelligence, control, and information* (Vol. 1). Upper Saddle River, NJ: Prentice Hall, 1999.

[11] S. Arjunan, and P. Sujatha, "Lifetime maximization of wireless sensor network using fuzzy based unequal clustering and ACO based routing hybrid protocol". *Applied Intelligence*, pp.1-18, 2017. https://doi.org/10.1007/s10489-017-1077-y.

[12] X.S. Yang, "Firefly algorithm, stochastic test functions and design optimization". *arXiv preprint arXiv:1003.1409*, 2010. https://doi.org/10.1504/IJBIC.2010.032124.

[13] S. Goyal, and M.S. Patterh, "Modified bat algorithm for localization of wireless sensor network", *Wireless Personal Communications*, vol. 86, no.2, pp.657-670, 2016. https://doi.org/10.1007/s11277-015-2950-9.

[14] Deepak Gupta, Ashish Khanna, Lakshmanaprabu SK, Shankar K, Vasco Furtado, Joel J. P. C. Rodrigues, "Efficient Artificial Fish Swarm Based Clustering Approach on Mobility Aware Energy-Efficient for MANET", *Transactions on Emerging Telecommunications Technologies*, 2018. https://doi.org/10.1002/ett.3524.

[15] Pandi Selvam Raman, K. Shankar, Ilayaraja M, "Securing cluster-based routing against cooperative black hole attack in mobile ad hoc network", *International Journal of Engineering & Technology*, 7. 9 (2018): 6-9. https://doi.org/10.14419/ijet.v7i1.9.9725.

[16] Mohamed Elhoseny, K. Shankar, S. K. Lakshmanaprabu, Andino Maseleno, N. Arunkumar. Hybrid optimization with cryptography encryption for medical image security in Internet of Things. *Neural Computing and Applications.* 2018. https://doi.org/10.1007/s00521-018-3801-x.

[17] T. Avudaiappan, R. Balasubramanian, S. Sundara Pandiyan, M. Saravanan, S. K. Lakshmanaprabu, K. Shankar, "Medical Image Security Using Dual Encryption with Oppositional Based Optimization Algorithm, *Journal of Medical Systems*, 42.11 (2018) 1-11. https://doi.org/10.1007/s10916-018-1053-z.

[18] LakshmanaprabuS.K, Sachi Nandan Mohanty, K. Shankar, Arunkumar N, GustavoRamireze. "Optimal deep learning model for classification of lung cancer on CT images", *Future Generation Computer Systems*. 2018. https://doi.org/10.1016/j.future.2018.10.009.

[19] K. Karthikeyan, R. Sunder, K. Shankar, S. K. Lakshmanaprabu, V. Vijayakumar, Mohamed Elhoseny, Gunasekaran Manogaran, Energy consumption analysis of Virtual Machine migration in cloud using hybrid swarm optimization (ABC–BA), The Journal of Supercomputing, 2018. https://doi.org/10.1007/s11227-018-2583-3.