International Journal of Innovative Computing

Journal Homepage: http://se.cs.utm.my/ijic

Multiple Metrics Fuzzy Based Clustering Algorithm for Wireless Mesh Client Networks

Adebanjo Adekiigbe
Dept. of Computer Science, School of Applied Sciences
The Federal Polytechnic Ede, Osun State, Nigeria
aadekiigbe2@live.utm.my

Kamalrulnizam Abu Bakar
Dept. of Computer Science, Faculty of Computing
Universiti Teknologi Malaysia, Skudai, JB. Malaysia
kamarul@fsksm.utm.my)

Nnamdi Henry Umelo
Dept. of Computer Engineering, School of Engineering Technology
The Federal Polytechnic Ede, Osun State Nigeria
namo_melo@yahoo.com

*Abstract*ð **One of the variants of Wireless Mesh Networks (WMNs) architecture is Mesh Client Networks (MCNs). In the wireless network researches, little attention has been paid to MCNs to improve the quality of service of WMNs. One of the fundamental problems facing wireless network is scalability of routing protocol when the network size is grows which also apply to MCNs. Many researchers has suggested clustering approach to solve scalability problem, however, the approaches have not been able to come up with formidable approach for clusterhead selection that totally guarantee stable cluster structures and reduce clustering overhead at the same time. In this paper, we proposed fuzzy logic control approach for the selection of clusterheads by using multiple metrics (MMFBCA). In our method, three MCs metrics such as node mobility speed, traffic delivery capacity and cost of service are fuzzified. The simulation results show that stable cluster structures with minimized clustering overhead are generated with these metrics. The results were compared with two existing weighted clustering algorithm (WCA) and Adaptive Intelligent Method for Dynamic Cluster Formation (AIMDCF) using basic performance parameters such as clustering overheads, number of cluster, cluster size and reaffiliation counts for the evaluation. In the final analysis, MMFBCA performance results are better than WCA and AIMDCF in all scenarios tested.**

Keywords — Multiple Metric, Mesh Client Networks, Distributed Clustering Algorithm, Fuzzy Logic Controller, Clusterheads.

I. Introduction

Wireless networks has provided extraordinary autonomy and mobility for a growing number of computing devices such as laptop, PDA and smart phones handlers who no longer require wired connections in other to stay connected with their various place of work and or the Internet. Incidentally, the equipment that makes provision of wireless service to all clients requires certain level of wired connection to either private networks or the Internet. The installation of this wiring is expensive and requires careful planning. However, deployment of mesh client architecture which is one of the three categories to which Wireless Mesh Networks (WMNs) [1, 2] are branded can help to overcome the high cost and less time in planning for wireless network setting out. In other to overcome the scalability issue that arises due to the growing in number of clients on the Mesh Client Networks (MCNs), node clustering has been variously proposed [3-8]. Clustering algorithms have been proposed for usage to enable the accomplishment of spatial reuse, network and location management, security and quality of service support [9]. Different approaches have been used in performing clustering scheme, these include centralized clustering, distributed clustering and Hybrid Clustering [10]. Centralized clustering architecture is used in a clustering process that requires one fixed clusterhead such as Mesh Portal (MPP) in WMNs and the remaining nodes such as Mesh Clients (MCs) and Mesh Points

(MPs) in the cluster acting as member nodes. The major disadvantage of this architecture is that if the MPP fails, the whole network may crumble since all nodes depend on this singular clusterhead. Therefore, there is no assurance for dependability in centralized clustering scheme. To this end, distributed clustering architecture comes handy to sort out the problems with centralised scheme. Distributed clustering requires no fixed central clusterhead since any node can assume clusterhead status once its meet up with certain criteria. Clustering scheme of distributed architecture is ideal for WMNs since it can be deployed either in the backbone/mesh-router domain or MCs domain where router or MC failures could be experienced.

The idea of reducing scalability by clustering algorithm invariably brings up a challenge of how the best clusterhead can be chosen among the various mesh client members. Various authors have presented a number of ways to select clusterheads [11-17]. However, some of these clustering algorithms exhibit some limitations. For example, in a reasonably large MC networks where many member nodes might require storage of complete routing information details for the whole network topology, the routing tables grows immensely when all nodes acquire and store routing information details for the networks [14, 18]. Hence, the issue of scalability that was being tried to be solved is again brought to forth [19-21]. Another challenge in some of the existing clustering algorithm include the need for formidable metric to help form stable cluster especially when nodes are mobile [22, 23]. These challenges have prompted many researches into both single and multiple metric clustering algorithms. Earlier proposed clustering algorithms such as Lowest ID (LID) by Ephremides*, et al.* [24], Least Cluster Change (LCC) algorithm [25], and Mobility Based Metric for Clustering (MOBIC) [26] are single metric clustering algorithm. These clustering algorithms are simple with small overhead, ironically, many of these algorithms failed to provide stable cluster structures especially when nodes are mobile. In complex network systems such as MANETs and MCNs, single metric like node degree or node distance is far-off from dazzling the dynamism of the whole network [27], therefore, single metric cannot be a fair representation of the intertwined nodes metrics [7]. In any network clustering algorithms, maximizing only one metric usually leads to loss of generality and results in low performance when compared with other metrics [28, 29]. Multiple metrics clustering objectives is to generate cluster structures that optimize several node metrics simultaneously to ensure cluster stability [6]. Many of the existing proposals such as those presented in Chatterjee*, et al*. [30], Cheng*, et al.* [27], Lacks*, et al.* [31], Hussein*, et al.* [32], Wei-dong [33], Sahana*, et al.* [6] and Aissa*, et al.* [7], all considered multiple metrics. Weighted Clustering Algorithm (WCA) is a proposal presented by Chatterjee*, et al.* [30]. In WCA, parameters such as node connectivity, node mobility speed, transmission range and remaining battery energy formed the main metrics in the selection of clusterheads. These parameters are assigned various weights subject to the scenario under consideration. In energy constraint scenarios, the power of the battery can be assigned maximum weight over other node parameters, while in a highly mobile network scenario; it can assign higher weight to node mobility speed. Despite the prospects of producing stable cluster structures based on multiple metrics considerations, the limitation of the algorithm centres on adoption of the concept of global minima. Each node within the network area is responsible for knowing the weights of every other member nodes earlier in preparatory for clustering process. This forces the algorithm to observe pause time for this process to take place, hence, the entire processing time increases. This contributes long delay and high clustering overhead in relatively large networks. Also, it requires high pause time for a set of mobile nodes to set up cluster structures. This is because of the necessity to compute high volume of data as it concerns each member so as to derive the collective weight. At any time a re-election of clusterhead is needed, re-computation of collective weight of each member node is required; hence, clustering overhead is thereby increased. Another shortcoming is the clusterhead selection process which is started at the commencement of cluster structure formation or whenever node changes it position due to its mobility and its new position is not covered by any clusterhead. In Adabi*, et al.* [34] and Lee and Jeong [35], multiple metrics clustering algorithm were proposed using fuzzy logic control approach. InAdabi*, et al.* [34], the process of clusterhead selection was optimized to save battery energy, therefore energy consumption minimization was seen as very important challenge in their approach rather than the cluster stability.

Though, the weight based approach guaranteed to some extent, considerable cluster structure stability, but the metrics such as node degree, mobility speed and energy remaining used in assignment of weights in the majority of the proposals do not results in lower clustering overhead for the MCs networks. Nevertheless, this implies major limitations for the existing approach. On another note, weighted methods require substantial mathematical computation which increase network resources usage and add up to clustering overhead.

MCs require different traffic delivery capacity to be able to perform its role efficiently and effectively in a network topology, it is pertinent to consider this metric when chosen clusterheads. At the same time, the cost of rendering data transfer services by every member nodes in a cluster differs; this metric determines the overhead for signalling and construction of cluster structures. However, none of the existing proposals jointly considered the node mobility, traffic delivery capacity and cost of service metrics in the clustering algorithm to construct stable cluster structure with minimal overheads. Therefore, a clustering algorithm that considered three multiple metrics of the MC such as mobility, traffic delivery capacity and cost of service is required so as to help reduce the cluster maintenance overhead and provide stable underlying cluster structures for the routing protocol in MC networks. Based on the need to provide effective clustering algorithm MC networks due to high cluster instability and clustering overheads of the existing clustering algorithms, Multiple Metrics Fuzzy Based Clustering Algorithm (MMFBCA) as a clustering algorithm is therefore proposed to cater for the need of MC Networks in this paper.

The remaining part of this paper is divided into the following sections: section II gives an overview of our proposed clustering algorithm; in section III the detail of the experiment to implement proposed MMFBCA is presented. Section IV discusses the simulation results while section V concludes the paper.

II. OVERVIEW OF THE PROPOSED MMFBCA

The proposed MMFBCA is a multi-metrics clustering algorithm which considered node degree, traffic delivery capacity and cost of service as metrics required for the selection of clusterheads. The traffic delivery capacity metric is introduced to address the problem of chosen clusterheads that is liable to bottleneck at all times due to MC small buffer capacity. In the case of cost of serve metric, it is introduced to mitigate high clustering overhead. The mobility metric is to ensure cluster stability. MMFBCA has two major functional components: Cluster Formation Algorithm and Cluster Maintenance. The cluster formation algorithm contains details of computation of MC metrics, fuzzification procedures, selection of clusterheads and cluster gateway nodes respectively. The cluster maintenance ensures reconstruction of cluster structures during link failure. Detailed descriptions of these functional components of MMFBCA design are presented in sections A and B respectively.

A. Clustering Formation Algorithm

The clustering formation algorithm presented in Algorithm 1 performs two major operations which includes computation of MC metrics and construction of cluster structures. For the cluster structures to be formed with its clusterheads and gateway node selected, the algorithm requires MC metrics to determine the optimal MCs suitable for clusterhead or gateway nodes as the case may be respectively. The computation of the MC metrics and cluster construction are discussed in the following paragraph.

1) MC Metrics

In the algorithm, three MC metrics are computed for the selection of clusterheads and gateway nodes. These metrics are mobility speed, traffic delivery capacity and cost of service as previously mentioned. We present the mathematical model of these metrics in sections a, b and c, respectively. For the purpose of clarity, the following definitions are made on the notations used for the modelling of the metrics.

Definition 1: Consider MC network model which can be represented by an undirected graph $G = (V, E)$ where G consists of a finite set V of objects referred to as MCs, a finite set E of objects referred to as logical edge. Assuming $v_1, v_2, ... \in V$, a logical edge (v_1, v_2) denotes that MCs x and y are within the communication range of each other and they are one-hop neighbour.

Definition 2: A partition $C = \{C_1, C_2, C_3, ...C_k\}$ of $V = \{v_1, v_2, v_3 ....v_n\}$ is called a clustering C of graph G, C_i is a cluster where $i \in [1, k]$ and clearly, the union of all C_i is equal to $V = \{(v_1, v_2, v_3 ....v_n) \mid \cup_{i-1}^{k} C_i = V\}$.

Definition 3: If MC v_i is in cluster C_i, the number of links between v_i and its neighbour MCs in C_i is referred to as intra-cluster connectivity of v_i and this gives the values for the number of MCs that make up the cluster.

Definition 4: For $v_i \in V$, the MC degree of v_i is the number of one hop neighbour, which can be represented as $n\deg(v_i) = \sum_{d=1}^{n} v_d$.

Definition 5: For $v_i \in V$, the average distance of v_i is the average summation of distance between v_i and all its one-hop neighbour, and is defined as $AveDist(v_i)$.

Definition 6: The fuzzy score of MC v, denoted by $fs(v)$ is the score value from FLC when MC metrics are fuzzified according to fuzzy rules.

Definition 7: If $V = \{v_1, v_2, v_3 ....v_n\}$, a corresponding sequence of fuzzy score value for graph G is $\{fs(v_1), fs(v_2), fs(v_3)........fs(v_n)\}$.

Algorithm 1: MMFBCA Clustering Formation Algorithm	
1:	*Compute nMob, nTDC and nCoS for all MCs in the network*
2:	*Compute the Fuzzy Score for all MCs using nMob, nTDC and nCoS*
3:	*Build a cluster of MCs around nodes that receive message acknowledgment for each other within Transmission Range*
4:	*Select the clusterhead from the MCs based on the highest Fuzzy Score value in the same neighbourhood within the same cluster.*
5:	*If more than one MC has the same highest Fuzzy Score value, select the MC with highest Fuzzy Score value whose average distance is smallest.*
6:	*Determine the cluster gateway node for every cluster.*

a) Mobility Speed of Mesh Client

To compute MC mobility speed, the distance between every MC that is directly linked with each other is computed based on coordinate information gathered from neighbouring MCs. Every MC is aware of the position of neighbour MCs that are within its transmission range. The MCs thereby uses beacon message to inform and also acquire useful information about its neighbour MCs that falls within transmission radius (R_x) and therefore set up a bidirectional links between MCs. The MC degree is determined using the following mathematical notations:

The one hop neighbour of vertex $x \in V$ is defined in Equation 1:

$$N(x) = \{y \; such\, that \; x \neq y \; AND \quad dist(x, y) \leq 1\} \quad (1)$$

Where x and y are nodes that are one-hop distance to each other. The MC degree of x is therefore represented in Equation 2:

$$n\deg(x) = \left|N(x)\right| \quad (2)$$

This is the number of edges occurring to x. The distance $dist(x, y)$ is the measure of path length between MC x and y. This distance is calculated using the formula presented in Equation 3:

$$dist(x, y) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (3)$$

x_i,x_j and y_i,y_j are corresponding nodes i^{th} and j^{th} for x and y respectively.

The mobility speed for each MC is defined as running average of the speed until a current time T. If a MC randomly moves from one place to another in a network area over a time period T, the values of its coordinate parameter changes accordingly. Therefore, the model in Equation 4 gives the mobility speed of the MCs.

$$M_v = \frac{1}{T}\left(\sum_{i=1}^{T}\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}\right) \quad (4)$$

b) Traffic Delivery Capacity

The traffic delivery metric is modelled as a function of MC connectivity/degree and available buffer size of MC. This affords the network to be able to balance load on the basis of its MC traffic capacity. Because of the peculiarity of MCs networks and its traffic pattern that always results in congestion at some intermediate nodes, it is desirable to identify MC with higher capacity to transfer data packets. The metric is bound to add value to cluster structure by providing better stability while also eliminating hotspot in the network. To simplify the model, all MCs are assumed to be able to generate and deliver packets of relative size, and then buffer capacity setting assumed for all MCs is as shown in Equation 5.

$$B_x = \mu * nDeg_x \quad (5)$$

Where $nDeg_x$ is the degree of MC x. The MC traffic delivery capacity is considered to be directly proportional to the MCs buffer capacity, this is as presented in Equation 6.

$$TDC_x = \varpi * B_x \quad (6)$$

μ and ϖ are adjustable parameters for the traffic system and ϖ assumed value that ranges between 0 and 1. The idea of delivery capacity in traffic system engineering was first presented by Xian-Bin, *et al.* [36] and Xiang, *et al.* [37] respectively. The choice of MC that is less likely to experience congestion due to high traffic volume and buffer overflows is a good reason to determine traffic delivery capacity of every MC. A MC with higher capacity to transfer data packets can add value to the cluster structure not only for stability but also guide against congestion in the network.

c) Cost of Service

For the avoidance of contradictions, Cost of Service (CoS) is defined as overhead required for MC to carry the burden of some other MCs in the network. The CoS is defined in this paper as a function of node degree with respect to average distance covered by MC during data packet transmission over a unit time. The covered distance between two MCs within the same transmission range determines the average cost for data packets transmission. This distance also determines the success and failure rate for packet delivery. If distance between MCs is shorter, though lesser cost for routing protocol, but leads to high number of cluster being generated and high level of interference [38]. A MC with longest distance equating the maximum transmission range is facing prospect of link failure due to increase in signal fluctuation of unreliable channels of wireless network [39]. By finding average distance for all MCs guarantees the choice of MC that enjoys high centrality. Selection of MC with utmost centrality assist all neighbour MCs to access their clusterhead in just one-hop without losing connection link, therefore, reducing service cost tremendously. The average distance for MC x is hereby computed as cumulative distance of all one-hop neighbour of x divided by the degree x of MC as presented in Equation 7.

$$AveDist(x) = \left(\frac{\sum_{i}^{j}\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{|\,N(x)\,|}\right) \quad (7)$$

$\forall x = \{i \mid i = 1.......j\}$

Once the average distance is computed, it is easier to determine the cost of service by multiplying the tuning parameter with Equation 7. In this case, the tuning parameter is to adjust the average distance upwards or downwards to improve cost of service available to each MC. Therefore, cost of service (CoS) is defined as follows:

$$CoS(x) = f * \left(\frac{\sum_{i}^{j}\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{|\,N(x)\,|}\right) = f * (AveDist(x)) \quad (8)$$

f is a tuning parameter.

2) Cluster Formation

MC clusters are formed by the cluster formation algorithm after the fuzzy scores are computed for every MC in the network. The algorithm select a MC with maximum fuzzy score value (i.e. $\max fs(x)$) as clusterheads while every neighbours of MC (x) are joined as its members. The greater the fuzzy score value $nfs(x)$, the higher the chances that MC is selected as clusterhead. In a situation of more than one MC having the same $nfs(x)$ score value within the same neighbourhood, one of the MCs with maximum score value whose average distance is smaller is chosen as clusterhead. To demonstrate the workings of the algorithm, assuming, MC (y) is near cluster (x), whereas MC (n) is a member of the neighbour set of MC (y) and MC (n) is not in neighbour set of MC (x). The algorithm elect MC (n) with the highest fuzzy score value as clusterhead of subsequently cluster, this procedure continues until every $v_i \in V$ is joined to a clusterhead. The algorithm ensures that clusterheads can communicate with each other provided they are within communication range; otherwise, any border MCs which resides in the two adjacent clusters, whose average distance is minimal between the two clusters are selected as cluster gateway to facilitate communication between these clusters.

Once all MC metrics are determined, fuzzy score values for all MCs are computed. This is achieved by invoking the fuzzy logic tool based on some predefined rules using the input and output variables represented in Figure 1. The membership function for the three input variables MC mobility speed ($nMob(x)$), traffic delivery capacity ($nTDc(x)$) and cost of service ($nCoS(x)$) are represented as Low, Medium and High. These membership functions are scaled as follows: $Low[x: 0.0, 0.0, 0.2, 0.4]$, $Medium[x: 0.2, 0.45, 0.7]$, $High[x: 0.5, 0.8, 1.0, 1.0]$ respectively.

However, output variable $nfs(x)$ is represented with seven membership functions as follows: VeryLow(*VL*), Low(*L*), LittleLow(*LL*), Average(*A*), LittleHigh(*LH*), High(*H*), *and* VeryHigh(*VH*). The membership functions are scaled as follows: $VL[x: 0.0, 0.0, 0.1, 0.2]$,

$L[x: 0.1, 0.2, 0.35]$,

$LL[x: 0.2, 0.3, 0.45]$,

$A[x: 0.35, 0.45, 0.55]$,

$LH[x: 0.45, 0.6, 0.7]$,

$H[x: 0.55, 0.7, 0.85]$,

$VH[x: 0.7, 0.85, 1.0, 1.0]$.

There are twenty seven rules that were presented to help in making inference by the fuzzy logic controller. These rules are based on the need of MC networks and also the experience of the researcher to control various actions in a linguistic form. Due to data traffic pattern in WMNs, MCs with low traffic delivery capacity is not fit for consideration as clusterhead. The reason is that, at various times, MC becomes bottlenecked and packets are dropped at will while the higher cost of service for MC results in more overheads being contributed to the clustering operation and highly mobile MC constitute a problem of instability. In view of all these factors, the three metrics are therefore given equal priority in the designing of rule base for the inference engine.

Since there are three input variables, and each variable have three membership functions, the total of twenty seven (27) rules are proposed. The average of all centroids of each rule is placed as the rule consequent centroid which helps to have rules that takes the format: ñIF mobility of MC ($nMOB$) is φ_l^1, and its traffic delivery capacity ($nTDC$) is φ_l^2, and its cost of service ($nCoS$) is φ_l^3, THEN the chances that MC is selected as a clusterhead is ρ_{ave}^rò, where ρ_{ave}^r is defined as follows:

$$\rho_{ave}^r = \frac{\sum_{r=1}^{7} \delta_r^l \rho^r}{\sum_{r=1}^{7} \delta_r^l} \tag{9}$$

In this model, $l = 1, 2, 3..........27$, δ_r^l is the number of choice of linguistic label r for the consequent of rule $l = 1, 2, 3..........27$ and $r = 1, 2, ...7$ respectively and ρ^r is the centroid of the r^{th} consequent set $r = 1, 2, ...7$. The 27 rules are as presented in Table 1. The number of rules which is based on the number of metrics in MMFBCA makes a significant difference in terms of processing speed and cluster stability when compared with works presented in Adabi, *et al.* [34].

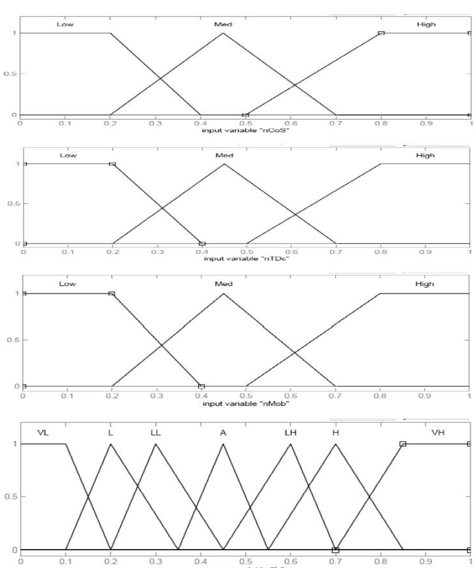

Fig 1: Input and Output Variables

Consequent upon successful input and output variables fuzzification based on the twenty seven inference rules, defuzzification process begins. In the defuzzification process of FLC, the centroid of area approach is adopted. The centroid of area only requires lower processing method which ensures that energy resources and limited processing capacity of MCs are not quickly exhausted. The deffuzification process produces numerical score values which are changed to crisp values by considering the average of centrally activated values for all linguistic variables, where score values are the membership degrees of each output linguistic variable. To acquire the output fuzzy score value for each MC, the output is computed using the defuzzification formula in Equation 10.

$$nFS = \frac{\Sigma_l^{27} \mu\varphi_l^1(nMOB) * \mu\varphi_l^2(nTDC) * \mu\varphi_l^3(nCoS)\rho_{ave}^r}{\Sigma_l^{27} \mu\varphi_l^1(nMOB) * \mu\varphi_l^2(nTDC) * \mu\varphi_l^3(nCoS)} \quad (10)$$

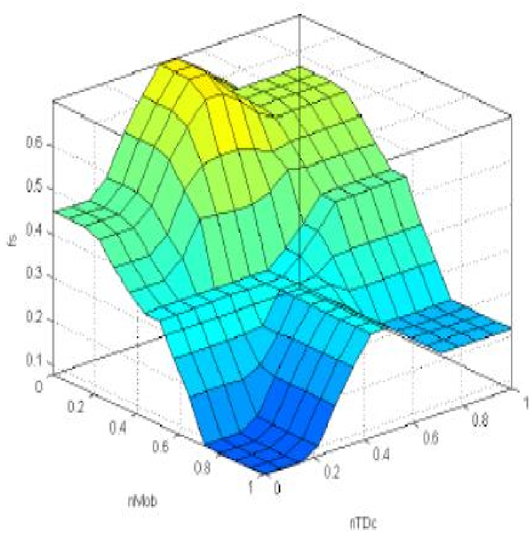

Fig. 2: Correlation Between $nMOB$ and $nTDC$ and nFS Variables

The outcome of this computation produces 4-Dimensional hyper-surface $(nMOB, nTDC, nCoS, nFS)$ which is not possible to be visually plotted using MATLAB. However, the correlation among two input variables is presented in Figures 2 and 3 respectively.

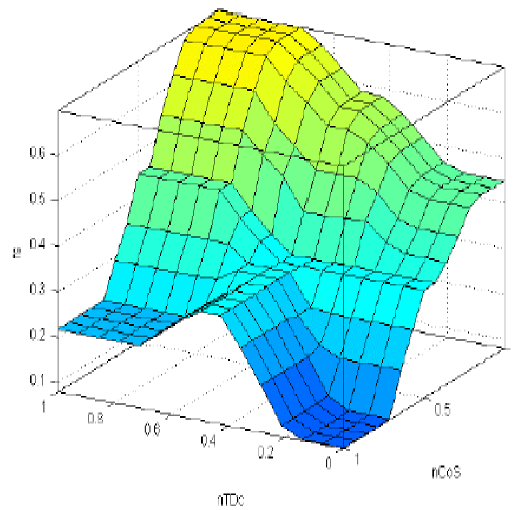

Fig. 3: Correlation Between $nTDC$ and $nCoS$ and nFS Variables

Table I: Fuzzy Rules Base

SN	***nMob***	***nTDC***	***NCoS***	***nFS***
1	High	High	High	L
2	High	Med	High	VL
3	High	Low	High	VL
4	High	High	Med	L
5	High	Med	Med	LL
6	High	Low	Med	VL
7	High	High	Low	LL
8	High	Med	Low	LL
9	High	Low	Low	LL
10	Med	High	High	L
11	Med	Med	High	LL
12	Med	Low	High	VL
13	Med	High	Med	A
14	Med	Med	Med	LL
15	Med	Low	Med	LL
16	Med	High	Low	H
17	Med	Med	Low	LH
18	Med	Low	Low	A
19	Low	High	High	LH
20	Low	Med	High	A
21	Low	Low	High	LL
22	Low	High	Med	LH
23	Low	Med	Med	H
24	Low	Low	Med	A
25	Low	High	Low	VH
26	Low	Med	Low	H
27	Low	Low	Low	A

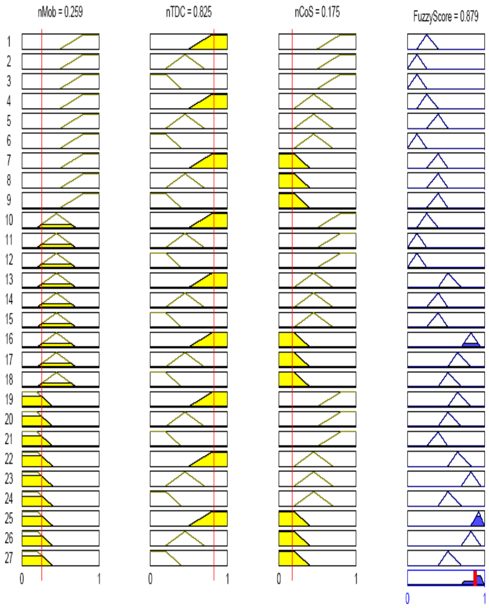

Fig. 4: Inference Engine Demonstration of Decision Making

In Figure 4, the effect of fuzzy rules on fuzzy score for every MC is as shown considering a rule ñIf *nMOB* is *Low* and *nTDC* is *High* and *nCoS* is *Low*, then the *nFuzzyscore* is *VH*ò. From the output, when *nMOB* equal to 0.259 which is a crisp value for Low, *nTDC* is equal to 0.825 also a crisp value for High and *nCoS* is equal to 0.175, crisp value for Low. Then, the output is 0.879 as *nFuzzyscore* is a crisp value for VH. The fuzzified values of *nMOB* , *nTDC* ,and *nCoS* are all represented in yellow colour while the blue colour represent the defuzzified value of *nFuzzyscore* in Figure 4. The *nFS* score 0.879 denotes fuzzy score of specific MC in the network. In this case, the inference engine uses a concessional decision based on MC metrics already defined.

B. Cluster Maintenance

The cluster maintenance is required when there are changes in topology. These changes may be caused by MC moving out of transmission range of its present clusterhead or MC energy depletion. Once the reply to periodic beacons cannot be received by source MC, then it is evident that such MC is either dead or moves out of its present place. Therefore, the invocation of cluster maintenance algorithm is inevitable. In other to reduce overhead, simple cluster maintenance algorithm is employed. There are two common assumptions made in the cluster maintenance process:

(i). Any message sent by a MC is received correctly within a finite time interval by all its neighbours.

(ii). No MC failure or link failure happens during execution of cluster maintenance algorithm.

With these assumptions, the cluster maintenance pseudo code in Algorithm 2 is presented. In the algorithm, the procedure enforces maintenance routine when any mobile MCs move out of its current cluster structure. In adopting the maintenance routine, the assumptions that all MCs remain immobile during clustering. MCs are homogenous and having the same capability is totally preserved. Once any break in communication link is noted by MC sensing its surrounding topology for any change, the re-clustering operation is activated.

Algorithm 2: MMFBCA Clustering Maintenance Algorithm

1: *Perform Periodic Calculation of Fuzzy Score by Mesh Clients*
2: *Send* Periodic Hello-msg to MCs
3: If *msg-acknowledged is not acknowledged within time inter t,*
4: *Then, member nodes are not within Transmission Range of clusterhead,*
5: *Repeat cluster formation procedure in Algorithm 1*

III. Research Method and Experimental Setup

In this paper, NS-2 version 2.33 network simulator is used. The network generator of NS2 is used to generate set of MCs for network topology to observe the performance of MMFBCA alongside WCA [30] and AIMDCF [34]. The number of MCs generated ranges between 50 and 100 over the network area of 800m by 800m. Three major simulation experiments are conducted with different network scenarios. The experiments are performed by varying number of MC, transmission range and MC mobility speed. The simulation environment parameters are chosen based on similar experiments conducted in Xing*, et al.* [40], Sonia and Fethi [41] and Sahana*, et al.* [6]. The summary of the simulation environment settings are presented in Table 2.

Table II: Simulation Parameters

Parameter	**Value**
Network Area	800m x 800m
Number of MCs	50-100
Transmission Range	50m-200m
Mobility Speed	1m/s-20m/s
CBR Packet Size	512 Byte
Simulation Time	200s
Buffer Size	50 kB

Three parameters are measured to evaluate MMFBCA. These parameters are: number of clusters, number of reaffiliations and clustering control overhead. The number of reaffiliations of MCs is considered for determination of the stability of network topology. The average number of clusterheads and control overhead determines the cluster maintenance cost and quality of clustering algorithm respectively. The proposed algorithm was evaluated based on average of ten (10) simulation trials. This is to reduce to the barest minimum the errors due to simulation.

IV. SIMULATION AND RESULTS DISCUSSION

To validate the proposed MMFBCA, several experiments were conduction by varying the simulation parameters to study the effect of number of MCs, MC speed and transmission range respectfully. The outcomes of various experimental tests are discussed based on these three performance metrics.

A. Effect of Number of MC, Mobility Speed and Transmission Range on Number of Clusterheads

The results of several experiments which involve variation of number of MC, maximum mobility speed attainable by MCs and MC transmission range are bench-marked with simulation results from WCA and AIMDCF under the same simulation environment. The choice of WCA and AIMDCF as bases for comparison was born out of the fact that these clustering algorithms are multiple metrics clustering algorithms as the proposed clustering algorithm. Figure 5 shows the performance of MMFBCA when compared with WCA and AIMDCF. The numbers of MC are varied between 50 and 100. In the scenario for the experiment that produces these results, mobility speed of the MCs is set to 10m/s while the pause time is set to zero. All clustering algorithms under investigation produce an increase in the number of clusterheads as number of MC increases. However, MMFBCA performs better because it produces lower number of clusterheads.

The reduction in number of clusterheads by MMFBCA consequently results in lower cluster maintenance overheads. The results here agrees with outcomes of the evaluation of max-Min d-clustering algorithm and Lowest ID clustering presented in Amis and Prakash [42].

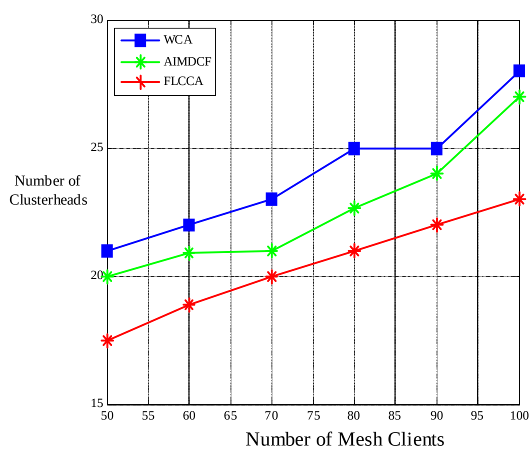

Fig. 5: Effect of Varying Number of MC on Number of Clusterhead Produced

Figure 6 shows the effect of varying MC mobility speed when the number of MC is fixed at 70 and pause time is set to zero second. The mobility speed is varied from 1m/s to 20m/s. The three algorithms show variation in number of clusterheads produced. In all, MMFBCA produces less number of clusterhead as shown in the results. The highest number of clusterhead produced by our proposal is 20 which is lower than the average produced by WCA and AIMDCF. The implication of fewer clusterhead by MMFBCA is that lower clustering overhead is generated; low clustering time is also recorded.

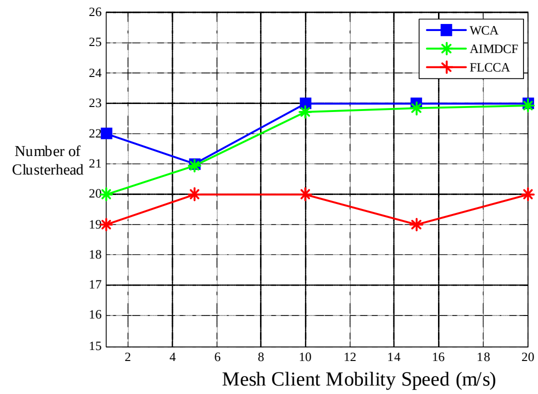

Fig. 6: Effect of Varying Mobility Speed on Number of Clusterhead Produced

In Figure 7, the network scenario has 70 numbers of MCs with maximum mobility speed of 10m/s and pause time set to Zero. The numbers of clusterheads are shown when the transmission range varies between 50 and 250 meters. The algorithms under consideration show downward trend in number of clusterheads as transmission range increases. The simulation results show that numbers of clusterheads of MMFBCA are lower than the numbers of clusterheads in both WCA and AIMDCF. In the scenarios, the decrease in number of clusterheads is as a result of increase in radius of coverage for clusterheads. This allows many MCs to be covered by few clusterheads. However, the implication for this trend is that a clusterhead may have more number of member MCs, therefore, more data traffic traverse the clusterhead than it can actually support.

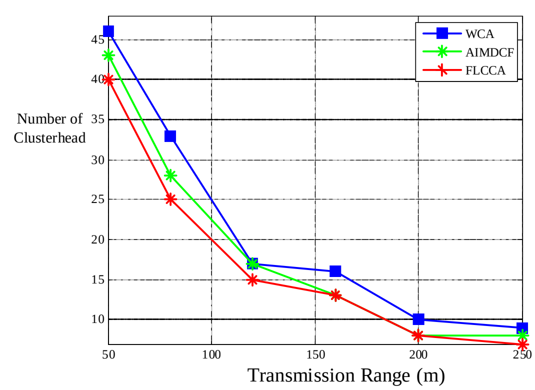

Fig. 7: Effect of Varying Transmission Range on Number of Clusterhead Produced

B. Effect of Number of MC, Mobility Speed and Transmission Range on Reaffiliation Rates

The reaffiliation rate in unit time for all the clustering algorithms is considered to study the rate at which mobile MCs leave its cluster structure to join another cluster. Figure 8 is a result of varying number of MCs when the transmission range of all MC is fixed at 100m, MC mobility speed at 10m/s and pause time set to zero. Pause time is the time in which a node paused in moving activity during clustering process. Under this scenario, MMFBCA, WCA and AIMDCF performance were compared and the results show smooth downward reduction in reaffiliation rate of the three algorithms.

The increase in number of MCs actually reduced the cases of reclustering because MCs are closer to each other with more MC population. Again, the transmission range of the populated clusterheads covers more MCs even as MC mobility is experienced. MMFBCA constantly shows better performance against WCA and AIMDCF as reaffiliation rate reduces with increase in number of MCs.

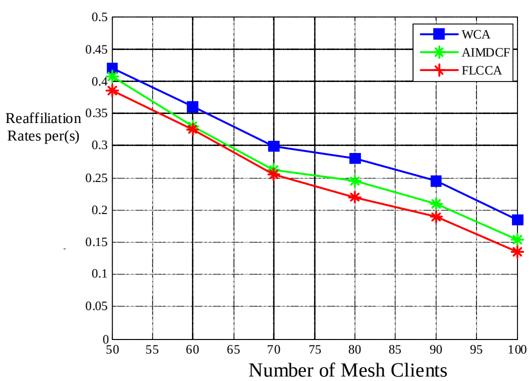

Fig. 8: Effect of Varying Number of MC on Reaffiliation Rates of Member Nodes

Figure 9 shows the relative performance of MMFBCA, WCA and AIMDCF when the MC mobility speed is varied under the scenario of 70 mobile MCs and a pause time of zero second. When the MC mobility speed increases, the MCs may likely move beyond its clusterheads, thereby disconnect itself from its clusterhead. The disconnection increases with average MC mobility speed, size and number of MCs in the networks. Hence, the MCs are forced to join another clusterhead frequently. Therefore, cluster stability decreases with the increase in mobility speed of MCs irrespective of the number of MCs within the network topology. The explanation to this statement is that, cluster stability is inversely related to the number of cluster topology changes incurred in the network [43].

The number of reaffiliations in the three algorithms under consideration is high. However, MMFBCA shows better performance than WCA and AIMDCF since MMFBCA gives more stable cluster structures than WCA and AIMDCF across all varying speed. This is achieved because the frequency of invoking clustering algorithm is low with MMFBCA. This is due to the fact that fuzzy logic controller (FLC) significantly helps in selecting MCs with perfect mix of mobility speed. Therefore, this result in the choice of clusterhead that stays for a long period of time interval under its cluster structure.

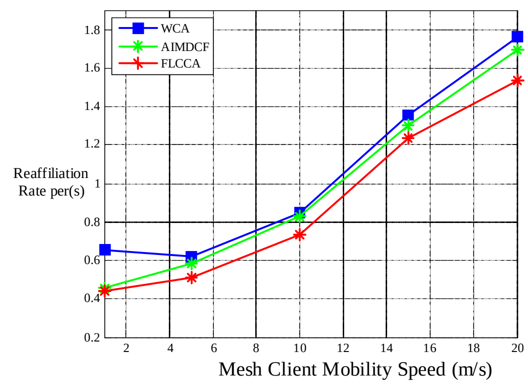

Fig. 9: Effect of Varying Mesh Client Mobility Speed on Reaffiliation Rates of Member Nodes

Figure 10 illustrates the reaffiliations rate in a time unit when the transmission range is varied between 50 and 250 meters. In the three algorithms under comparison, reaffiliation rate per unit time decreases as transmission range increases. Whereas, MCs in any cluster structure are close to clusterhead relatively when the transmission range is small, but the random mobility speed of MC tends to make MC disconnect with its clusterhead very quickly. However, further increment in transmission range brings about decrease in reaffiliations rate. This is because MCs are likely to stay within the coverage area of the clusterhead, despite the random mobility of MCs. In effect, proposed MMFBCA shows better performance than WCA and AIMDCF.

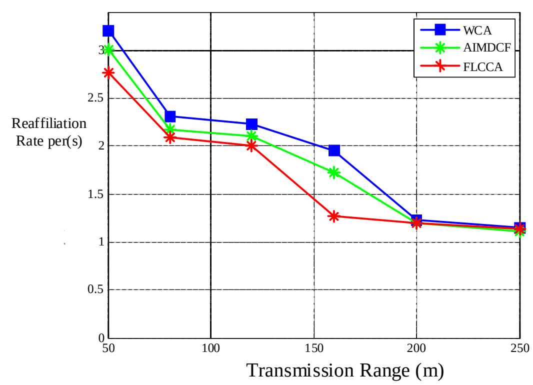

Fig. 10: Effect of Varying MC Transmission Range on Reaffiliation Rates of Member Nodes

C. Effect of Number of MC, Mobility Speed and Transmission Range on Clustering Overhead

The clustering control overhead for MMFBCA, WCA and AIMDCF are determined under a scenario with maximum mobility speed of 10m/s and pause time of zero with varying number of MC, mobility speed and transmission range. Figures 11, 12 and 13 show the different results obtained.

In Figure 11, the normalized clustering overhead of MMFBCA, WCA and AIMDCF are shown. Any increase in number of MC also increased clustering overhead. This is understandable because more numbers of MCs generates more traffic due to Hello messages during MC initialization, cluster update and maintenance. MMFBCA performs better than WCA and AIMDCF at all levels of the varying number of MCs.

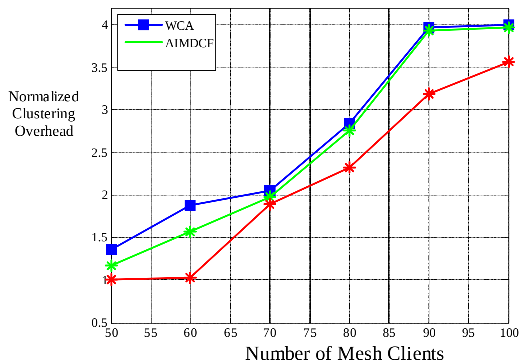

Fig. 11: Effect of Varying Number of MC on Clustering Overhead for the Network

The normalized clustering overhead of MMFBCA, WCA and AIMDCF at varying MC speed is shown in Figure 12. The sharp increment in clustering overhead for the three algorithms from mobility speed of 1m/s to 5m/s is as a result of additional overhead incurred due to MC neighbour table update when MC moves out of its present cluster. The algorithms maintain stable clustering overhead starting from MC speed of 5m/s. MMFBCA incurs lower overhead than WCA and AIMDCF for all mobility speed under consideration.

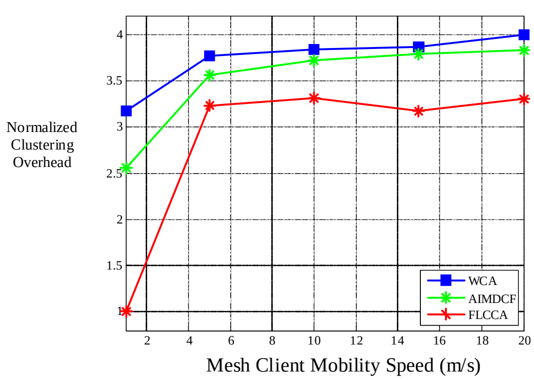

Fig. 12: Effect of Varying MC Mobility Speed on Clustering Overhead for the Network

In Figure 13, the normalized clustering overhead of MMFBCA, WCA and AIMDCF at varying transmission range is shown. Downward reduction of clustering overhead is noticed as the transmission range increases for the two algorithms. In general, MMFBCA performs better for all the transmission range considered.

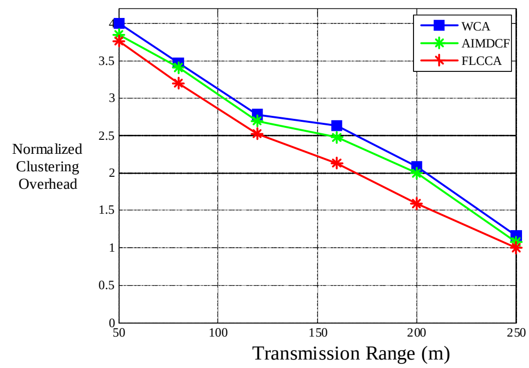

Fig. 13: Effect of Varying Transmission Range on Clustering Overhead for the Network

V. CONCLUSION

The overall comparison of proposed MMFBCA with WCA and AIMDCF shows that MMFBCA reduced the number of clusterheads by 16.38%, 13.21% and 19.16% under varying number of MC, transmission range and mobility speed parameters respectively. In the same vein, the rate of network reaffiliation is reduced by 7.07%, 11.52% and 8.72% when the number of MCs, transmission range and mobility speed are varied respectively. It is also worthy to be noted that MMFBCA clustering overhead is reduced by 18.51%, 20.29% and 8.57% with varying number of MC, transmission range and mobility speed respectively when compared with WCA and AIMDCF. It can be concluded that our proposed distributed MMFBCA has successfully fulfilled the aim to forming stable cluster structures with minimal clustering overhead.

ACKNOWLEDGMENT

The authors wish to thank the management of Federal Polytechnic Ede, Nigeria and Universiti Teknologi Malaysia for the research support provided for this work.

REFERENCES

[1] Akyildiz, I.F. and Wang, X.: "Wireless mesh networks" Wiley Publisher. 2009.

[2] Alabady, S.A. and Salleh, M., "Overview of Wireless Mesh Networks," Journal of Communications, vol. 8 (9), 2013, pp. 134-144.

[3] Wang, T. and Hung, W.N., "Reliable Node Clustering for Mobile Ad Hoc Networks," Journal of Applied Mathematics, vol.13 (2), 2013, pp.81-89.

[4] Abdelhak, B., Abdelhak, B., and Saad, H., "Survey of Clustering Schemes in Mobile Ad hoc Networks," Comm and Network, vol. 5 (1), 2013, pp. 23-31.

[5] Adekiigbe, A., Abu Bakar, K., and Ogunnusi, S.O., "Issues and Challenges in Clustering Techniques for

Wireless Mesh Networks," Journal of Computer Science, vol. 8 (1), 2011, pp. 8-15.

[6] Sahana, S., Saha, S., and DasGupta, S., "Weight Based Hierarchical Clustering Algorithm for Mobile Ad hoc Networks," Procedia Engineering Journal, vol. 38 (4), 2012, pp. 1084-1093.

[7] Aissa, M., Belghith, A., and Drira, K., "New Strategies and Extensions in Weighted Clustering Algorithms for Mobile Ad Hoc Networks," Procedia Computer Science Journal, vol. 19 (2), 2013, pp. 297-304.

[8] Malhotra, P. and Dureja, A., "A Survey of Weight-Based Clustering Algorithms in MANET," IOSR Journal of Computer Engineering, vol. 9 (6), 2013, pp. 34-40.

[9] Lin, C.R. and Gerla, M., "Adaptive clustering for mobile wireless networks," IEEE Journal on Selected Areas in Communications, vol. 15 (7), 1997, pp. 1265-1275.

[10] Abbasi, A.A. and Younis, M., "A survey on clustering algorithms for wireless sensor networks," Computer Communications, vol. 30 (14), 2007, pp. 2826-2841.

[11] Botezatu, N., Manta, V., and Vieriu, G.: "A clustering algorithm for negotiating security in wireless mesh networks". In the Proceeding of 15th International Conference on System Theory, Control, and Computing (ICSTCC'11), 2011, pp. 107-113.

[12] Fonseca, C., Mocito, J., and Rodrigues, L.: "Low-Coupling Cluster-Based Multipath Routing for Wireless Mesh Networks". In the Proceeding of 20th International Conference on Computer Communications and Networks (ICCCN'11), 2011, pp. 1-6.

[13] Krebs, M., Stein, A., and Lora, M.A.: "Topology Stability-based Clustering for Wireless Mesh Networks". In the Proceeding of Global Telecommunications Conference (GLOBECOM'10), 2010, pp. 1-5.

[14] Langar, R., Bouabdallah, N., and Boutaba, R., "Mobility-aware clustering algorithms with interference constraints in wireless mesh networks," Computer Networks, vol. 53 (1), 2009, pp. 25-44.

[15] Sharma, S., Kumar, S., and Singh, B., "Routing in Wireless Mesh Networks: Two Soft Computing Based Approaches," arXiv preprint arXiv:1307.3004, vol., 2013

[16] Tajima, S., Funabiki, N., and Higashino, T., "A WDS clustering algorithm for wireless mesh networks," IEICE transactions on information and systems, vol. 93 (4), 2010, pp. 800-810.

[17] Yoon, W. and Kim, W.: "Congestion Control for Cognitive Multi-channel Wireless Mesh Networks", In Ubiquitous Information Technologies and Applications (Ed.) Springer, 2013, pp. 659-667.

[18] Bouabdallah, N., Langar, R., and Boutaba, R., "Design and analysis of mobility-aware clustering algorithms for wireless mesh networks," IEEE/ACM Transactions on Networking (TON), vol. 18 (6), 2010, pp. 1677-1690.

[19] Kulkarni, N.S., Gupta, I., and Raman, B.: "On demand routing protocols for mobile ad hoc networks: A review". In IEEE International Advance Computing Conference (IACC'09), 2009, pp. 586-591.

[20] Kumar, M., Rishi, R., and Madan, D., "Comparative Analysis of CBRP, DSR, AODV Routing Protocol in MANET," International Journal on Computer Science and Engineering (IJCSE), vol. 2 (09), 2010, pp. 2853-2858.

[21] Bemmoussat, C., Fedoua, D., and Feham, M., "Efficient Routing Protocol to Support QoS in Wireless Mesh Network," International Journal of Wireless & Mobile Networks (IJWMN), vol. 4(5): 89-104, 2012

[22] Mitsche, D., Resta, G., and Santi, P., "The random waypoint mobility model with uniform node spatial distribution," Wireless Networks, vol. 3(2), 2013, pp. 1-14.

[23] Saha, S. and Chaki, R.: "Cluster Based Mobility Considered Routing Protocol for Mobile Ad Hoc Network", In Advances in Computer Science and Information Technology (Ed.) Springer, 2011, pp. 33-42.

[24] Ephremides, A., Wieselthier, J.E., and Baker, D.J.: "A design concept for reliable mobile radio networks with frequency hopping signaling". In the Proceeding of IEEE Conference, 1987, pp. 56-73.

[25] Chiang, C.-C., Wu, H.-K., Liu, W., and Gerla, M.: "Routing in clustered multihop, mobile wireless networks with fading channel". In the Proceeding of IEEE SICON, 1997, pp. 197-211.

[26] Basu, P., Khan, N., and Little, T.D.: "A mobility based metric for clustering in mobile ad hoc networks". In the Proceeding of International Conference on Distributed Computing Systems Workshop, 2001, pp. 413-418.

[27] Cheng, H., Cao, J., Wang, X., Das, S.K., and Yang, S., "Stability aware multi metric clustering in mobile ad hoc networks with group mobility," Wireless Communications and Mobile Computing, vol. 9 (6), 2009, pp. 759-771.

[28] Alotaibi, E. and Mukherjee, B., "A survey on routing algorithms for wireless ad-hoc and mesh networks," Computer networks, vol. 56 (2), 2012, pp. 940-965.

[29] Adekiigbe, A., Abu-Bakar, K., Amosa, B.M.G., and Umelo, N.H.: "Fuzzy Logic Approach to Clustering Algorith for Mesh Client Networks". In Application of Information and Communication Technologies to Teaching, Research and Administration, Obafemi Awolowo University, Ile Ife, Nigeria, 2013,

[30] Chatterjee, M., Das, S.K., and Turgut, D., "WCA: A weighted clustering algorithm for mobile ad hoc networks," Cluster Computing, vol. 5 (2), 2002, pp. 193-204.

[31] Lacks, D., Chatterjee, M., and Kocak, T., "Design and evaluation of a distributed clustering algorithm for mobile ad hoc networks," The Computer Journal, vol. 52 (6), 2009, pp. 656-670.

[32] Hussein, A., Yousef, S., Al-Khayatt, S., and Arabeyyat, O.S.: "An efficient weighted distributed clustering algorithm for mobile ad hoc networks". In International Conference on Computer Engineering and Systems (ICCES'10), 2010, pp. 221-228.

[33] Wei-dong, Y.: "Weight-based clustering algorithm for mobile ad hoc network". In Cross Strait Quad-Regional Radio Science and Wireless Technology Conference (CSQRWC'11), 2011, pp. 787-791.

[34] Adabi, S., Adabi, S., and Ali, R., "Introducing an Adaptive Intelligent Method for Dynamic Cluster Formation for Mobile Ad Hoc Networks," Journal of Computing vol. 3 (3), 2011, pp. 1-9.

[35] Lee, C. and Jeong, T., "FRCA: A Fuzzy Relevance-Based Cluster Head Selection Algorithm for Wireless Mobile Ad-Hoc Sensor Networks," Sensors, vol. 11 (5), 2011, pp. 5383-5401.

[36] Xian-Bin, C., Wen-Bo, D., Cai-Long, C., and Jun, Z., "Effect of adaptive delivery capacity on networked traffic dynamics," Chinese Physics Letters, vol. 28 (5), 2011, pp. 59-67.

[37] Xiang, L., Mao-Bin, H., and Jian-Xun, D., "Effects of node buffer and capacity on network traffic," Chinese Physics B, vol. 21 (9), 2012, pp. 98-102.

[38] Fasolo, E., "Efficient Data Dissemination Protocols in Pervasive Wireless Networks", PhD Thesis. Universita degli Studi di Padova., 2008.

[39] Lee, Lee, U., and Gerla, M., "Geo-opportunistic routing for vehicular networks [Topics in Automotive Networking]," Communications Magazine, IEEE, vol. 48 (5), 2010, pp. 164-170.

[40] Xing, Z., Gruenwald, L., and Phang, K., "A robust clustering algorithm for mobile Ad-hoc networks," Handbook of Research on Next Generation Mobile Networks and Ubiquitous Computing, vol., 2008, pp. 187-200.

[41] Sonia, M.G. and Fethi, F.: "A Density Based Clustering Algorithm for Efficient Channel Allocation in Multi-radio Multi-channel Wireless Mesh Networks". In the Proceeding of MESH 2011, The Fourth International Conference on Advances in Mesh Networks 2011, pp. 1-7.

[42] Amis, A.D. and Prakash, R.: "Load-balancing clusters in wireless ad hoc networks". In the Proceeding of 3rd IEEE Symposium on Application-Specific Systems and Software Engineering Technology 2000, pp. 25-32.

[43] Inn, E.I., "Mobility-adaptive clustering and network-layer multicasting in mobile ad hoc networks," Ph.D. Thesis. National University of Singapore., 2007.