IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 3, September 2012
ISSN (Online): 1694-0814
www.IJCSI.org

A Stable Distributed Clustering Algorithm For Mobile Adhoc Networks

Fereydoun Ramezani Zangi[1] Sajjad mavizy[2] Javad Badali[3]

[1,2,3] Department of Computer Engineering, Islamic Azad University of Shabestar, Shabestar, Iran

Abstract

Mobile Ad hoc Networks (MANET) are distributed systems which consist of several mobile nodes; they do not include predetermined topology and have no concentrated control. The nodes connect wirelessly by the approach in which they decide themselves. To access the development in the MANET, an approach should be found such that less control messages are distributed in the MANET, and movement of the network nodes is concealed from other network nodes. One method of access the above purposes is the clustering of MANET. In this paper, a distributed weight based clustering algorithm has been proposed for the MANET. Also in the proposed method, it has been attempted that stability of the cluster is increased through solving the problem of node density in a cluster. The proposed algorithm has been compared with the WCA in terms of number of reaffiliations, number of dominant set updates, number of clusters and load balancing. The results of simulation indicate that the proposed algorithm has a better performance.

Keywords: *MANET, distributed systems, clustering, stability, reaffiliation, dominant set & load balancing*

1. INTRODUCTION

Mobile Ad Hoc Networks (MANET) consists of a set of mobile node that these networks have dynamic topology and the nodes connected via wireless links in the absence of fix infrastructure. This type of networks are used where no relational substructures exist or they cannot be used such as war area, search and salvation operation, unexpected events and so on. Some applications of the MANET can include very large dimensions like the military networks and/ or the future commercial networks. In the larger networks which there are thousands of nodes, data reserve and details of routing in the nodes cause some problems in the development of the network. Therefore the clustering algorithms have been represented so that the development problem is solved through representing the hierarchical structure. Though grouping several nodes in one node, other nodes can have the information of one cluster rather than that of several nodes [1,2].

The network nodes can be clustered through different methods in which various clusters cover the total network. These algorithms organize the network in several groups as cluster dynamically. By organizing the nodes in the clusters, less topological data transferred in the network. Every cluster forms a correlated graph, and two clusters may have overlap. The best method of categorizing the clustering designs of the MANETs is their aim [9]. So different designs have been proposed for clustering of the MANETs. One of them is clustering based on combining the various parameters which uses some parameters to form the cluster especially to determine the clusterhead, like node degree, cluster size, speed, battery, etc. By studying more parameters, the clusterhead can be selected rightly, furthermore, factors weighting can be used in different scenarios. The different approaches can be studied through these criteria, and the similarities and differences of various designs are studied for every scenario the best clustering is selected. In the proposed method, a weight is calculated for every node based on criteria of degree difference, movement angle difference, speed difference, distance from neighbors and remainder energy [2,3].

The node with the most weight between the neighbors is selected as clusterhead. In the MANET, when the nodes are not distributed uniformly, the nodes density increases in a point of network. In the distributed clustering algorithm, considering that just one node select as clusterhead between the neighbor nodes (that is two clusterhead nodes cannot be neighbors), so the nodes density increases in one cluster. Increasing the nodes density will put more load on each of the clusterheads, even if these nodes are neighbors and be in its transmission range. Covering of the network area by the least number of clusterhead imposes more responsibility on every clusterhead which necessitates using the most resources. This causes early death of the clusterhead. To solve this problem, optimization algorithms are used. However, using the centralize methods, especially in the larger networks, is not possible due to the node movement. The other method to solve this problem is that the expected member nodes of a cluster are obtained, and if the members become more than this rate, the cluster will be divided. In the proposed method, it has been attempted that the stability of clusters increases regarding this [4,7].

2. Related work

2.1. The Highest-Degree Algorithm

This algorithm was proposed by Gerla and Parekh, A node x is considered to be a neighbor of another node y if x takes within the transmission range of y. Each node sends its id to all nodes that located in the its transmission range. A node with maximum number of neighbors (Highest Degree) is selected as a clusterhead. The neighbors of clusterhead, are members of this cluster [6].

2.2. The Lowest-ID Algorithm

This Algorithm was proposed by Baker and Ephremides. It assigns a unique id for each node. A node with minimum id is selected as a clusterhead. Hence the id of the clusterhead neighbors should be more than clusterhead's id [7].

2.3. The distributed clustering algorithm

Basagni proposed two algorithms, namely distributed clustering algorithm (DCA) and distributed mobility adaptive clustering algorithm (DMAC). In these two approaches, each node is assigned a weight based on its suitability of being a clusterhead. A node is chosen to be a clusterhead if its weight is higher than any of its neighbor's weight; otherwise, it joins a neighboring clusterhead. Results show that the number of updates required is smaller than the Highest-Degree and Lowest-ID Algorithms. Since node weights vary in each simulation cycle, computing the clusterheads becomes very expensive and there are no optimizations on the system parameters such as throughput and power control [5].

2.4. The Weighted Clustering Algorithm

The WCA was proposed by SAJAL and TURGUT. It employs combined metrics-based clustering: the Degree-difference, sum of the distance with all neighbors, average of the speed, total time, are taken into account to calculate a weight factor W_v for every node v. Each node checks Weight of its neighbors, Choose v with the minimum W_v as the clusterhead [7].

3. Proposed method

In this paper, a weight based distributed algorithm has been proposed which is calculated based on 5 criteria for every weight node. The node with the most weight among its neighbors is selected as the clusterhead.

3.1. Process of selecting the clusterhead

In the following a process is represented for selecting the clusterhead whose exit is a collection of nodes (dominate set) which form the clusterheads. The process of selecting the clusterhead is recalled at the time of activating the system and when the dominate set is not able to cover all of the nodes.

Step1: We assume that every node can obtain the location of its neighbors using GPS, by which the degree of node v (d_v). Then every node sends a hello message to all of its neighbors. This message includes node id, average speed, average movement angle and d_v [3,7,8].

Hello Message	ID_v	SP_v	ANG_v	d_v

SP_v is obtained from the following relation:

$$SP_v = \frac{SP_{avg(t-1)} + SP_t}{2} \quad (1)$$

Where $SP_{avg}(t-1)$ is the average speed of node v until t-1 and SP_t is the speed of node v in the moment of t. Also ANG_v is obtained from the following relation:

$$ANG_v = \frac{ANG_{avg(t-1)} + ANG_t}{2} \quad (2)$$

Where $ANG_{avg}(t-1)$ is the average of node v movement angle until t-1 and ANG_t is the angle of node v at the moment of t.

Step 2: Every node calculates its weight. To calculate the weight, five criteria are considered as follows:

1) Degree Difference (D_V)
This criterion is calculated using the following relation for every node v:

$$D_v = |C_v - d_v| \ \ and \ \ C_v = \frac{\sum_{i=1}^{N_v}(d_{vi} + d_v)}{d_v + 1} \quad (3)$$

Where d_{vi} is degree of ith neighbor of node v.

2) Distance with neighbors (P_v)

3) Angle Difference (A_v)
Angle difference for every node v is obtained from the following relation:

$$A_v = \sum_{i=1}^{d_v} |ANG_v - ANG_{vi}| \quad (4)$$

Where ANG_v is average angle for node v and ANG_{vi} is the average angle of ith neighbor of node v.

4) Speed Difference (S_v)
To calculate the node v speed difference, the following relation is used:

$$S_v = \sum_{i=1}^{d_v} |SP_v - SP_{vi}| \quad (5)$$

Where SP_V is the average speed for node v and SP_{vi} is the average speed of ith neighbour of node v.

5) Remainder of energy (Br_v)

Parameters of weight calculation are normalized after measurement. The reason for this normalization is that the values of these measures are in different intervals, and they should be brought into a certain interval, which we consider this normalization for [0, 1] interval. Normalization is done as follows:

$$D_v = e^{-D_v} \quad (6)$$

$$P_v = e^{-\frac{P_v}{d_v * Tr}} \quad (7)$$

Where Tr is the transmission range of every node.

$$A_v = e^{(-\frac{A_v}{180 * d_v})} \quad (8)$$

The movement direction of a node in the worst case with its neighbor node is when both of them move in opposite directions, that is the movement angle difference is 180^0.

$$S_v = e^{(-\frac{S_v}{\max\_speed * d_v})} \quad (9)$$

Max_speed is the maximum speed of a node.

$$Br_v = \frac{Br_v}{E_{max}} \quad (10)$$

Where E_{max} is maximum energy of a node.

After normalization, every node obtains its weight using the following relation. The factors C_1, C_2, C_3, C_4 and C_5 are the weight factors for the corresponding parameters of the system.

$$W_v = C_1 * D_v + C_2 * P_v + C_3 * A_v + C_4 * S_v + C_5 * Br_v \quad (11)$$

Step 3: Every node sends its weight to all of the neighbors.

Step 4: Every node studies the weight of its neighbors. The node with the most weight is selected as the clusterhead.

Step5: The steps 2- 4 are repeated for the rest of nodes which has still allocated to any cluster.

3.2. Process of maintenance the cluster

A good clustering algorithm is one that possibly prevents re-clustering, that is the clusters have more stability. For this aim, in the proposed method except representing an algorithm of weight based distributed clustering, it has been tried that the stability of cluster increases regarding three processes.

1) Density

When the nodes are distributed uniformly in the network, the distributed algorithms are the best method for clustering the nodes. But as it is observed in Figure1, when the nodes are not distributed uniformly in the network, the nodes in a cluster may be more, so dominant set updates and reaffiliations are increased.

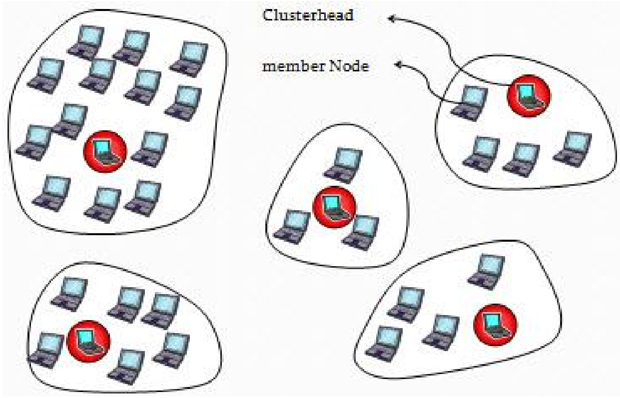

Figure1: Network with non-uniform distribution

To solve the density problem, optimization algorithms are used. But to perform these algorithms, all of the network data should be accessed, this increases the overhead of the sent messages in the large networks. Also since there is no special substructure in these networks, so one of the nodes should receive the total data of the network and perform the algorithm. Therefore using the centralize methods will not be possible in the large networks.

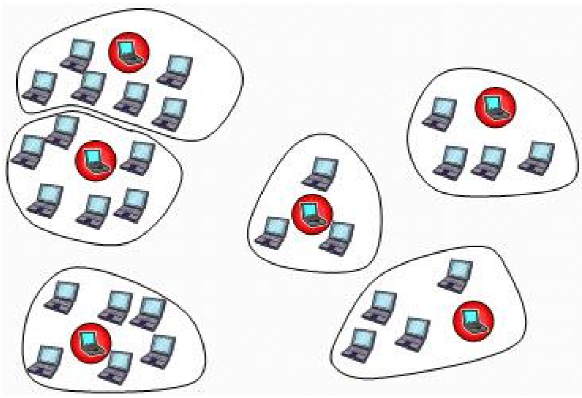

Figure2: Dividing the member nodes

As it is observed in figure2, the other method to solve the problem is that the expected members for every cluster are obtained. If the members are more than the expected rate, selecting one of the member nodes as the clusterhead and dividing the member nodes between these clusterheads. This reduces the dominant set updates and reaffiliations apart from creating load balance. To obtain the expected members for every node, the following relation is used which is based on node number, network size and transmission range of every node.

$$ED = n * \left(\frac{Tr^2 * \pi}{X * Y}\right) * \left(1 - \frac{4}{3\pi}\left(\frac{Tr}{X} + \frac{Tr}{Y}\right) + \frac{1}{2\pi} * \frac{Tr^2}{X * Y}\right) \quad (12)$$

n is the total number of the network nodes and (X, Y) are the length and width of the network. We assume that every clusterhead can support δ node ideally (a predetermined limit) so that MAC efficiency is provided, so if members of cluster i (Nm) is more than $\sqrt[2]{ED * \delta}$, in other words, if $\frac{Nm}{\sqrt[2]{ED*\delta}} > 1$, the clusterhead wants all of its members to calculate its weight using the relation (11). The member nodes calculate their weight and send to its clusterhead. The clusterhead receives the weight of member nodes and calculates a new weight for every node using the following weight relation.

$$NW_v = \omega_1 * W_v + \omega_2 * dis_v \quad (13)$$

Where W_v and dis_v are the node v weight and distance of node v with the clusterhead respectively (the node whose distance with the clusterhead is much, it has more chance to become the clusterhead) which is normalized as following.

$$dis_v = \frac{dis_v}{Tr} \quad (14)$$

The member node with maximum NW_v is selected as the clusterhead. By sending a message, the clusterhead node informs all of the new clusterhead. The member nodes distinguish their clusterhead through sending a message towards the closest clusterhead.

2) Clusterhead death

If the clusterhead dies for any reason, the first member node that realize its clusterhead death, informs all of the death of clusterhead through sending a message to the members of cluster. Of the member cluster, the nodes which cover the total cluster calculate their weight using the relation (11) and send it to all members. The node with the most weight is selected as the new clusterhead.

3) Weakening of clusterhead

If the energy of clusterhead is decreased to some extent that it cannot perform as the clusterhead, in this case the clusterhead announces. It dispense with being the clusterhead. Then like the death state of the clusterhead, those nodes of the member nodes can cover the total cluster which calculate their weight using the relation(11) and send it to all of the members. The node with the most weight is selected as new clusterhead.

4. Simulation

In this section, the proposed algorithm is simulated as the results of simulation are compared with the WCA. For simulation, Matlab software has been used. In this simulation, the mobile nodes of 10-50 have been considered, every node moves in every time unit, in an environment of 100mx 100m in dimension accidently in different directions and in variable speed between o and the maximum speed (5m/s and 10m/s). The ideal degree was fixed at δ=10 for the whole simulation. The factors C_1, C_2, C_3, C_4 and C_5 as well as w_1 and w_2 are fixed for the system. The weight factors are flexible. Considering the effect on the combined weight calculations (W_V). The weight factors for the relation (11) are as C_1=0.3, C_2=0.4, C_3=0.1, C_4=0.1, C_5=0.1, and for the relation (13), they are as w_1 =0.7 and w_2=0.3.

4.1. Performance Metrics

We consider the following metrics to evaluate the performance of our proposed algorithm [7, 10]:

1) The number of reaffiliation

A node may dissociate from its clusterhead transmission range and locate in another clusterhead transmission range due to mobility of nodes in mobile Ad hoc networks. In fact it is dissociated from its own clusterhead, connecting to the other one, which this act is called reaffiliation.

2) The number of dominant set updates

The selected clusterheads for all networksd considered as dominant sets. So, if one member node is dissociated from its clusterhead and goes into a part of network not has been covered by any of clusterheads, the act of clustering should repeat again, and new clusterheads should be selected, which is called dominant set updates.

3) The number of clusterheads

This metric is the average number of logical partitions formed in the network with the mobile nodes. The set of clusterheads in the network define as the dominant set [10].

4) Load Balancing

Load balancing factor or LBF is a measure to calculate the efficiency of an algorithm and network lifetime. In other words, LBF is a measure of balancing calculation in the clusterheads, which means that clustering algorithm should be selected in such a way that network load, i.e. different operations performed in a network, should be spreaded among all clusterheads in a balanced way. To measure load balancing factor, relation (15) is used:

$$LBF = \frac{n_c}{\sum_i (x_i - \mu)^2} \quad \text{Where} \quad \mu = \frac{(N - n_c)}{n_c} \quad (15)$$

Where n_c is the number of clusterheads, x_i the cardinality of cluster i, μ is the average number of neighbors of a clusterhead, and N is the total number of nodes in the network. With respect to what was described and also relation (15), the higher value of LBF represents a better load distribution of that algorithm.

4.2. Simulation Results

The figure 3, 4 and 5 show the simulation results for the transmission ranges of 30 and 40 m as well as the maximum speed of 5 m/s. As it is observed in the figures

3 and 4, by increasing the transmission range from 30m to 40m, dominant set updates and reaffiliations per unit time decrease. It is for this reason that by increasing the transmission range, the domain of a clusterhead increases, so the possibility of separating the member node from the cluster and being this node in a point of the network which is not covered by any clusterhead decreases.

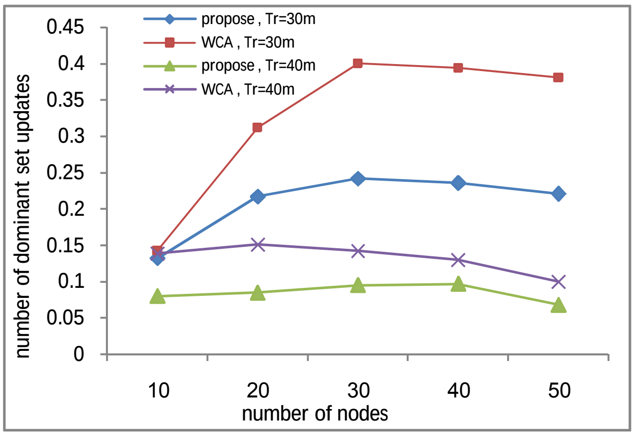

Figure3: number of dominant set updates and max_speed=5m/s.

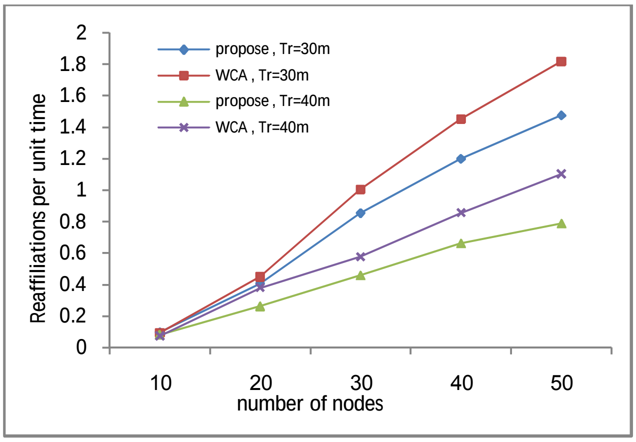

Figure3: number of reaffiliations per unit time and max_speed=5m/s.

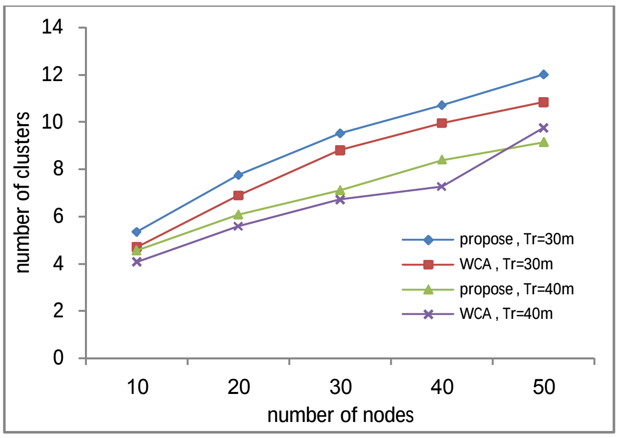

Figure5: number of clusters and max_speed=5m/s.

Considering the Figures 3 and 4, the proposed method indicates better performance in proportion to WCA regarding dominant set updates and reaffiliations per unit time.

Figure5 shows number of the created clusters. Considering that in the proposed method, the node density has been considered for stability of the cluster, so the number of created clusters for the proposed method is more than WCA.

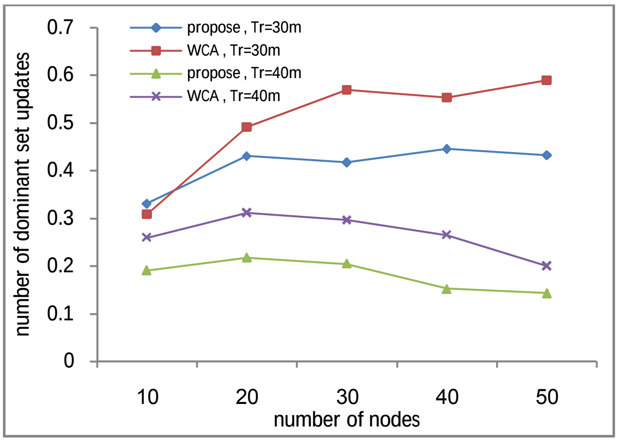

Figure6: number of dominant set updates and max_speed=10m/s.

The Figures 6, 7 and 8 show the simulation results for the transmission range of 30m and 40m as well as the maximum speed of 10m/s. As it is observed in Figures 6 and 7, by increasing the speed in proportion to the figures 3 and 4, the number of dominant set updates and number of reaffiliations per unit time increases. The reason for this increase is that by speed increase, the distance a node moves increases so the possibility of separating the node from the cluster increases. In spite of speed increases from 5 to 10m/s, the proposed method performs well in proportion to the WCA for updation of the number of dominant set updates and number of reaffiliations per unit time. Figure 8 shows number of created clusters which produce more clusters than the WCA considering the proposed approach.

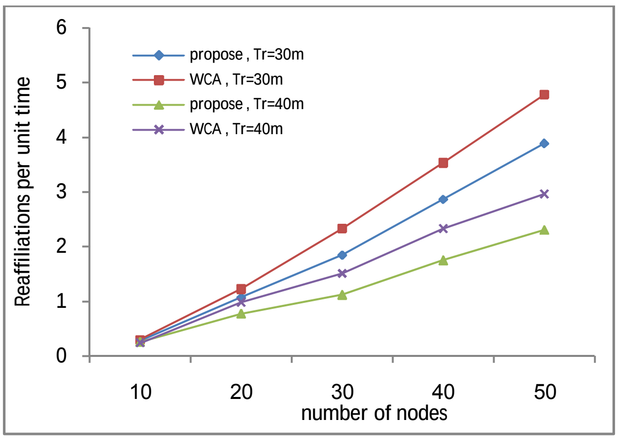

Figure7: number of reaffiliations per unit time and max_speed=10m/s.

Figure9 shows the Load Balancing Factor (LBF) for the proposed method and WCA. In the Figure9 the number of nodes is 50, maximum speed of the node is 10m/s and the transmission range of every node is 30m, the LBF for the proposed algorithm is between 0 and 0.923 and for the WCA, it is between 0 and 0.686. Also the average of LBF is 0.7129 and 0.5392, respectively; this indicates the better performance of proposed algorithm in the load distribution in proportion to the WCA algorithm.

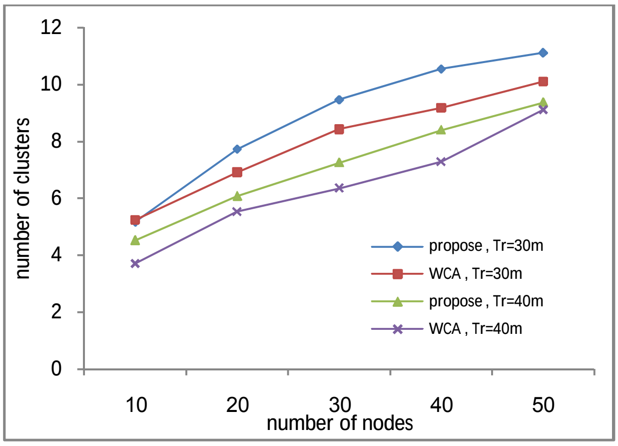

Figure8: number of clusters and max_speed=10m/s.

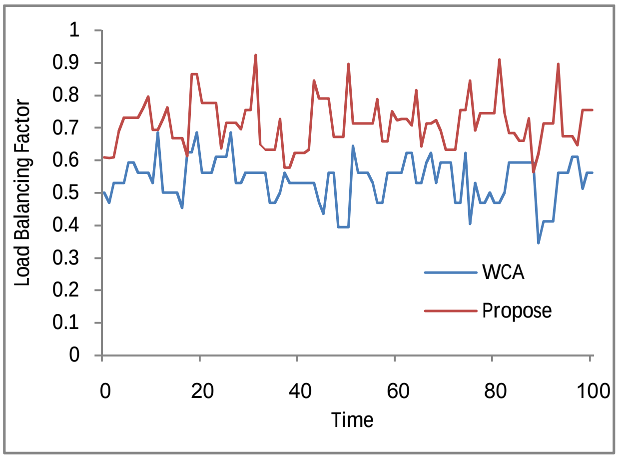

Figure9: Load Balancing Factor.

5. Conclusion

Clustering is essential for the efficient utilization of resources and balance of load in the dynamic networks in large scale as well as access to development in the mobile ad hoc networks. Therefore creating a suitable clustering algorithm for these networks is essential. In MANET, covering of the network area by the least clusterhead imposes more responsibility on every clusterhead which necessitates the most resources. Also when the nodes are not distributed uniformly in the network, density of nodes in one cluster increases which causes early death. To solve this problem, the expected member nodes can be obtained in a cluster, and if the members are more than this, the cluster is divided. In our proposed method, by considering this it has been tried that the network load is balanced and stability of clusters increases. In simulating the proposed method of four criteria, number of reaffiliations per unit time, number of dominant set updates, number of clusters and load balancing, to evaluate the proposed algorithm have been considered. The obtained results have been compared with the WCA algorithm. The results indicate that the proposed approach performs better than the WCA.

References

[1] Juan Huan, Shiguang Ju and Xiaoming Han. “Energy-aware Distributed Cluster Minimization Algorithm for Intrusion Detection In Wireless Ad Hoc Networks”, IJCSNS International Journal of Computer Science and Network Security, August 2006, pp. 106-111.

[2] Pi-Rong Sheu and Chia-Wei Wang. “A Stable Clustering Algorithm Based on Battery Power for Mobile Ad Hoc Networks”, Tamkang Journal of Science and Engineering, 2006, pp. 233-242.

[3] Suchismita Chinara and Santanu Kumar Rath, (2009) "A Survey on One-Hop Clustering Algorithms in Mobile Ad Hoc Networks", J Netw Syst Manage , 17:183–207-DOI 10.1007/s10922-009-9123-7.

[4] S.Muthuramalingam, R.Rajaram and G.Padmavathy, (2010) "Clustering Algorithm to Reduce the Overheads and to Balance the Load Using an Hybrid Algorithm in Gauss Markov Model", International Journal of Computer and Electrical Engineering, Vol. 2, No. 2, April, 2010 1793-8163.

[5] Stefano Basagni. “Distributed clustering for Ad-hoc Networks ,International Symposium On Parallel Architectures”, Algorithm and Networks (I-SPAN'99), Fremental, Australia, Juane 1999.

[6] M. Gerla and J.T.C. Tsai. “Multicluster, mobile, multimedia radio network”, ACM/Baltzer Wireless Networks, 1(3),1995, pp. 255-265.

[7] M. Chatterjee, S. K. Das, and D. Turgut. “WCA: A Weighted Clustering Algorithm For Mobile Ad Hoc Networks”, Journal of ClusterComputing, 2002, pp. 193-204.

[8] Christian Bettstetter, (2004), "The Cluster Density of a Distributed Clustering Algorithm in Ad Hoc Networks", Communications,IEEE International Conference on 20-24 June 2004, Pages 4336 - 4340 Vol.7.

[9] S.Muthuramalingam and *R.Rajaram*, (2010), "A TRANSMISSION RANGE BASED CLUSTERING LGORITHMFOR TOPOLOGY CONTROL MANET", International journal on applications of graph theory in wireless ad hoc networks and sensor networks (GRAPH-HOC) Vol.2, No.3, September 2010.

[10] Ismail Ghazi Shayeb, AbdelRahman Hamza Hussein, Ayman Bassam Nasoura. “A Survey of Clustering Schemes for Mobile Ad-Hoc Network (MANET)”, American Journal of Scientific Research ISSN 1450-223X Issue 20(2011), pp.135-15.