ISSN No. 0976-5697

Volume 2, No. 1, Jan-Feb 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

An Efficient Weight Based Clusterhead Selection Algorithm for Mobile ad hoc Network

R. Gunavathi*
MCA Department,
Sree Saraswathi Thyagaraja College, Pollachi,
India
gunaganesh2001@gmail.com

P. Sivaprakasam
MCA Department,
Sri Vasavi College, Erode,
India

Abstract - Clustering can improve the network management and energy saving. In the clustered network organization the clusters are formed by grouping the mobile nodes with one special role node termed as clusterhead. Cluster based communication protocols have been proposed for ad hoc networks for various reasons. In the proposed algorithm all clusterheads spread across the network. In this paper the high-weight node is act as a clusterhead with minimum 3-hop count distance between each other. We also investigate to minimize the clusterhead change, increase in lifetime of clusterheads, less energy conservation. The simulation results are compared with On-Demand Weighted Clustering Algorithm (ODWCA) and it shows that the Efficient Weight based Clusterhead Selection Algorithm (EWCSA) that improves the network performance for Mobile ad hoc Network.

Keywords – MANET, Clustering, Highest Degree Algorithm, Lowest Identifier Algorithm, Distributed Clustering Algorithm, Weighted Clustering Algorithm

I. INTRODUCTION

Mobile ad hoc Networks (MANET) consists of mobile devices that form the wireless networks without any fixed infra structure or centralized administration. MANET that bears great applications in variety of fields such as military applications, industrial process monitoring control, healthcare applications, etc.,

Mobile ad hoc Network is composed of large numbers of nodes (mobile devices) that form wireless networks without any fixed infrastructure or centralized administration. In these networks each node communicates other nodes via intermediate nodes. Grouping of the nodes is clustering. In each cluster a node is elected as a clusterhead [1, 2]. These nodes collect the data from various nodes and one of them will act as a clusterhead (CH). The clusterheads act as local coordinators and handle various functions such as packet routings and forwarding. Since the network is not a fixed structure, the role of the node may become a clusterhead or an ordinary node.

Mobile nodes are used only limited battery power, it is very difficult to replace or discharge it when the node dies. This will affect the network's performance. If the node consumes less energy it will increase the life time of the network. Optimize the communication range and minimize the energy usage, we need to conserve the energy of mobile nodes [3, 4]. The clusterhead within each cluster acts as the local coordinator for its member node. The primary objective of this paper is to elect an effective clusterhead, cluster formation and energy conservation. In this paper we propose a new algorithm an Efficient Weight Based Clusterhead Selection Algorithm that makes a fair distribution of cluster headship among all the nodes. The algorithm aims to form a cluster in every 3-hop and the node which has high-weight it will act as a clusterhead. To avoid the data loss we fixed a threshold value for energy, while the clusterhead reaches or less than threshold value it resigns its job and reelection of clusterhead taken place.

The rest of the paper is organized as follows. Section II describes the work done in the related area and drawbacks of existing algorithm, Section III gives a brief description about the proposed algorithm with subsections like cluster formation, clusterhead selection and cluster maintenance. Simulation results discussed in section IV followed by section V that concludes the paper.

II. RELATED WORK AND DRAWBACKS OF EXISTING ALGORITHMS

A. Highest Degree Algorithm (HDA)

In Highest Degree Algorithm [5] the node with the highest number of neighbors are elected as clusterhead. Basically, each node either becomes a clusterhead or ordinary node. Major drawbacks of this algorithm are a) re-affiliation count of the nodes is very high b) the clusterhead can not play its role for long time c) the number of ordinary nodes in the cluster is increased d) the throughput is decreased.

B. Lowest-Identifier algorithm (LIA)

In Lowest Identifier Algorithm [6, 7, 8] the node which has minimum ID is chosen as a clusterhead. Since the node *ids* do not change with time, those with smaller *ids* are become clusterheads than nodes with larger *ids*. The drawback of lowest ID algorithm is that certain nodes are prone to power drainage due to serving as clusterheads for longer periods of time.

C. Distributed Clustering algorithm (DCA)

The Distributed Clustering Algorithm is a modified version of the Lowest-Identifier algorithm. For each cluster, it chooses a node with locally lowest ID among all the neighbor nodes as a cluster head. Every node can determine its cluster and only one cluster, and transmits only one message during the algorithm [9]. Since it uses node ID for the selection of cluster heads, it inherits the drawbacks of the Lowest-Identifier heuristic.

D. Weighted Clustering Algorithm (WCA)

In Weighted Clustering Algorithm [10] the node with the minimum weight is selected as a clusterhead. The weight associated to a node with various parameters like Number of neighbors of each, Energy Consumption, Mobility of the node and Power consumption. The drawback of this algorithm is all the nodes before starting the clustering process and in draining the CHs rapidly. Since a node already serving as a clusterhead cannot accurately reflect the current level of battery power, a busy node may almost run out of power and it has never been a clusterhead.

III. PRELIMINARIES OF PROPOSED ALGORITHM

An Efficient Weight Based Clusterhead Selection Algorithm for Mobile ad hoc Network(EWCSA)

A. Clusterhead Selection

Our aim is to reduce the number of clusterheads in wireless network. The mobile node has weight values depends upon various parameters like mobility of the nodes, total number of neighbors, battery power, etc., The weight value defines which node has to become clusterhead. Each node has actions like sensing, sending and receiving data, aggregating, listening, etc., In this approach each node is assigned with an appropriate weight based on its stability of being a clusterhead.

The existing Weighted Clustering Algorithm (WCA) is based on the use of a combined weight metrics like node degree, distances with all neighbors, node speed and the time spent as clusterhead.

In this section, we describe the fundamental concepts used here; a node is defined as ‘u’. Each node should be assigned as two status ie, ordinary node or clusterhead. Every node ‘u’ in the network discovers its neighbor information periodically through Hello message and updates its neighbor table. Among the neighbor list compute node degree denoted by d_u. Node degree of a node ‘u’ is the total number of neighbors of that node ‘u’

X and Y value denotes node ‘u’ position. Within the Hello message we built X, Y value to identify the position of node. The modified hello message propagates to all nodes. The nodes maintain a neighbor table that is often updated based on the reception of this modified Hello message. Here the ‘D’ denote distance and all nodes find its distance between adjacent nodes by Du. The summation is taken over all neighbors ‘v’ of ‘u’ if d(v,u) denotes the distance between u and v then $D_u = \sum(u,v)$. ‘M’ denotes mobility and ‘M_u’ is the average speed of node. Find per second mobility of a node by

$$\text{Mu} = \frac{1}{T}\sum_{t-1}^{T}\sqrt{(X_t - X_{t-1})^2 + (Y_t - Y_{t-1})^2} \quad (1)$$

Where T is the current time. Fix a threshold value ∂ for maximum number of cluster member. ‘Δu’ defines the load of cluster head CH. To compute $\Delta u = |du - \partial|$ to find the difference of each node.

$\text{Wu} = \text{w1}\Delta\text{u}+\text{w2Du}+\text{w3Mu}+\text{w4Pu}$ (2)

Where w1, w2, w3, and w4 construct metrics of weight factors. w1 describe the changes in load of cluster, w2 explains the distance between each neighbor, mobility expressed by w3 and w4 denote if the node is a cluster head, find the time duration for which it act as a cluster head. Also it implies how much battery power has been consumed by a cluster head which is assumed to be more for a cluster head than ordinary node.

After find out the weight information of each node that message broadcast to all other nodes. Receiving nodes update their own clustering table. Subsequently it will verify its cluster table and checks for high weight node. Which node discovers its weight is high it will announce CH-announcement packet to the remaining nodes by propagation. Every node in cluster group receives the announcement packet and makes sure its cluster table. If it is not a cluster head or cluster member it will update as cluster head. Similarly next cluster head election by means of verifying previous CH with minimum 3- hop count distance. This process will do again until arrive at final node. A few of nodes may leave because of uncover area called critical nodes declare itself as a cluster head.

B. Cluster Maintenance

Cluster head (CH) announcement packet periodically sends Hello message to its all cluster members (CM) and it updates cluster head information. There are two cases for maintaining the cluster.

Case – I: Cluster member moves out of area from its cluster group due to mobility. Each node periodically checks its neighbor table for availability of cluster head. If cluster head ID is not present in its cluster table to facilitate, node sends find-CH message. Receiving node if it is cluster head it starts to send cluster head CH announcement message. If member node receives more than one cluster head message it choose cluster head CH by way of high weight node.

Case – II: Fix threshold value for energy to CH. Incase cluster head energy reduced below the threshold value it announces to send resign message instantly. Immediately the entire cluster member CM removes cluster head ID from its cluster table and which node has high weight it will act as a cluster head and sends CH announcement message.

C. Cluster formation

According to our approach, clustering of mobile nodes could be controlled the parameters like network size and density. We introduce a new approach to form a cluster. Each node discovers the neighborhood using periodic exchange of Hello messages. The nodes maintain a neighbor table that is frequently updated. Each node calculates its weight based on the equation (2). The node which has the highest weight is elected as a clusterhead. Next we move to the cluster organization step by 3-hop count. The nodes in the cluster have two different roles. One of it acted as a cluster member another is clusterhead. Once a clusterhead is elected the neighbors are the cluster members. The transmission range radius is kept as constant. Every time each node checks the neighbor table to confirm the clusterhead as its neighbor. If the clusterhead is not present in the neighbor table it has to find a new clusterhead according to the weight of the node.

Algorithm:

[a] Every node broadcasts the messages of an action of sensing, listening, aggregating, energy consumption and bandwidth.
[b] Identify neighbors using Hello messages and update it in the neighbor table
[c] Calculate weight using equation (1) and (2) and store as high-weight-node. Update it into the cluster table. Thus each node knows the highest weighted node in every 3-hop distance.
[d] During the first round high-weight-node is elected as a clusterhead with the 3-hop distance
[e] When two nodes move into each other transmission area topology changes occurs. During that each node periodically checks its neighbor table to confirm clusterhead as its neighbors. If the clusterhead moves from the transmission range, new election of clusterhead is done.
[f] Now the member node sends the find clusterhead message to its neighbor table. Then the node which has the highest-weight will elect as a clusterhead
[g] If any two clusterheads are present in the same cluster, then the node which has the lowest ID is elected as a clusterhead.

IV. PERFORMANCE EVALUATION

The proposed algorithm *An Efficient Weight based clusterhead Selection Algorithm* is implemented in NS2 [11]. NS2 is a network simulator developed at UC Berkely written in C++ and OTcl for ad hoc networking. Each simulation was executed 500 secs. Transmission range radius of node is kept constant for all nodes. In first set of simulation network density that is number of nodes fixed and average node speed varied in terms of 5ms, 10ms, 15ms, 20ms. Mobility is an important factor in selecting cluster heads. Proper reaffiliation process should be defined when one normal node moves out of a cluster and joins another existing clusters. When 2 nodes move into each other transmission range, topology change occurs. Each node checks neighbor's list entry periodically to confirm cluster head as its neighbor. If cluster head node ID is not present in the neighbor list entry node may be out of transmission range from Cluster head CH. So it has to find out new cluster head within its transmission range. Member node sends the find-CH message to its neighbor. Any one of the cluster head presents in their transmission range it will reply cluster-info message. When the node receive cluster-info message it updates the cluster head information. In second case any two cluster heads presents within the node transmission range each CH send cluster-info message. Receiving node decides which one will be the best CH based on weight information. Node choose low weight node as CH.

V. PERFORMANCE ANALYSIS

Performance analysis has done for the speed with number of cluster head changes, number of node with consumed energy, Number of nodes with Life time, and Number of nodes with End to End delay. During the simulation we found that a small value of clusterhead change it reflects on the stability of the cluster structure. The number of cluster member change is the number of mobile nodes that switch over to another clusterhead. To find the scalability of the protocol the overhead should be reduced. Clustering overhead is the number of clustering messages sent by each node in cluster formation and cluster maintenance operation.

Fig.1 shows the number of clusterhead changes based on speed compared with two algorithms ODWCA and EWCSA. At the time of simulation due to mobility many cluster heads can elect depends on energy and movement. Speed is measured in ms. Our aim is to reduce the number of clusterheads. EWCSA supports less number of cluster heads than ODWCA even in mobility.

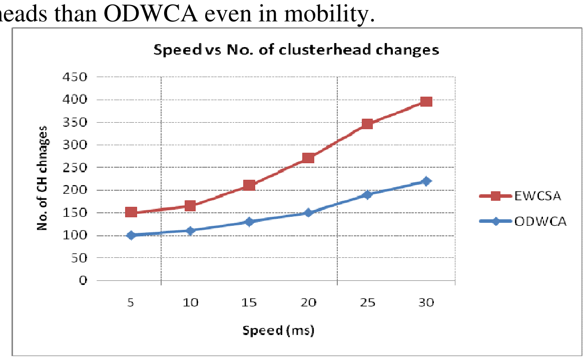

Figure.1. Speed vs Number of Cluster Head Changes

Fig.2 shows the comparison among two algorithms of ODWCA and EWCSA with Number of nodes and its Energy Consumption. Energy is measured in joules. Energy consumption varies according to the concentrations and performance of node. The graph shows that when the number of nodes increased EWCSA consumes a lesser amount of energy than ODWCA.

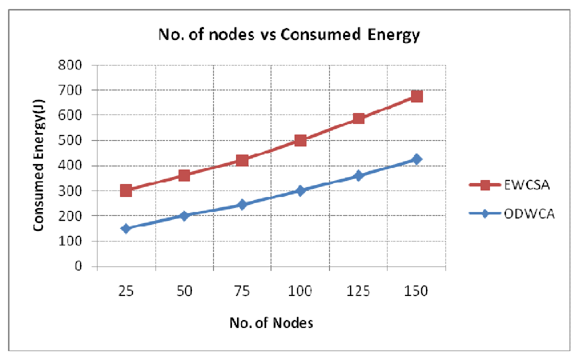

Figure. 2. Node vs Consumed Energy

Fig. 3 shows the number of nodes and its life time based on network. By comparison it is important to consider life time. If the node life time increased it will produce stable cluster formation and maintenance of cluster head CH and cluster member CM. While comparing our algorithm EWCSA with ODWCA the life time is reduced.

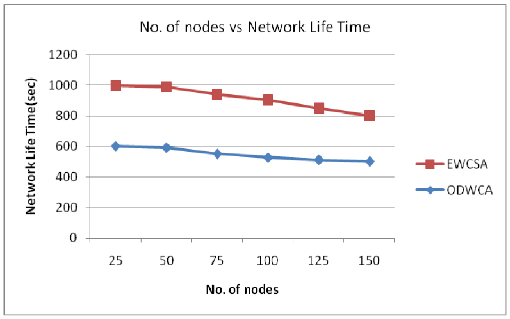

Figure. 3 Number of Nodes vs Life Time

Fig.4 shows that time taken for a packet to be transmitted across a network from source to destination is the end to end delay. Here we compared our algorithm and ODWCA for End to End data delay and number of nodes. The graph shows that the result of best performance of EWCSA algorithm. It represents the end-end data delay in the network

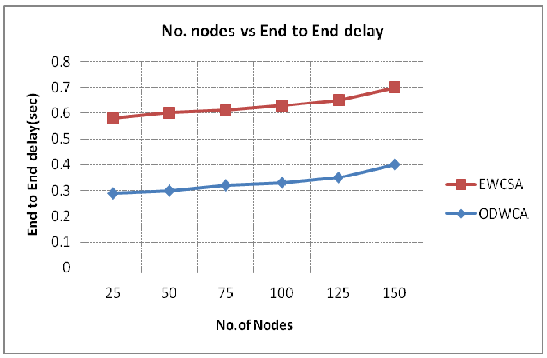

Figure.4 Number of Nodes vs End to End delay

VI. CONCLUSION

In this paper we investigated to minimize the number of cluster heads without affecting the stability of the network. All clusterheads spread across the network with minimum 3 hop count distance between each other. Through simulations we have compared the performance of our algorithm with that of On demand Weighted Clustering Algorithm (ODWCA) in terms of the clusterhead change, energy consumption, life time and time delay**.** Simulation results show that Efficient Weight Based Clusterhead Selection Algorithm (EWCSA) improves network performance.

The network lifetime in our algorithm is less while compared with ODWCA. By

VII. REFERENCES:

[1] G. Wang, H. Zhu, H. Dai, L. Wu and B. Xiong, "The Clustering Algorithm of Wireless Sensor Networks Based on Multi-hop between Clusters", Computer Science and Information Engineering, vol. 3, pp. 177–181, March-April 2009.

[2] H. Huang, Y. Xu, Yu-e Sun and L. Huang "Cluster-based load balancing multi-path routing protocol in wireless sensor networks", The 7th World Congress on Intelligent Control and Automation(WCICA), pp. 6692–6696, June 2008.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey", Computer Networks: The International Journal of Computer and Telecommunications Networking", v.38 n.4, p.393-422, 15 March 2002

[4] M. Younis, K. Akkaya, M. Eltoweissy, A. Wadaa, "On Handling QoS Traffic in Wireless Sensor Networks", Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 9, p.90292.1, January 05-08, 2004

[5] Gerla M., Tsai J. T. C., "Multicluster, Mobile, Multimedia Radio Network," ACM/Baltzer Wireless Networks Journal 95, vol. 1, Oct. 1995, pp. 255-265.

[6] Baker D.J., Ephremides A., "A distributed algorithm for organizing mobile radio telecommunication networks," Proceedings of the 2nd International Conference on Distributed Computer Systems, Apr. 1981, pp. 476-483.

[7] Baker D.J., Ephremides A., "The architectural organization of a mobile radio network via a distributed algorithm," IEEE Transactions on Communications, Nov. 1981, pp. 1694-1701.

[8] Ephremides A., Wieselthier J. E., Baker D. J., "A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling," Proceedings of the IEEE, vol. 75, no. 1, Jan. 1987, pp. 56-73.

[9] Nocetti, F. G., Gonzalez, J. S., and Stojmenovic, I., "Connectivity Based k-Hop Clustering in Wireless Networks," Telecommunication systems, Vol.22, 2003, pp.205-220.

[10] Chatterjee M., Das S.K., Turgut D., "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks," Cluster Computing Journal, vol. 5, no. 2, Apr. 2002, pp. 193-204.

[11] K. Fall and K. Vardhan, The Network Simulator (ns-2). Available:http://www.is.edu/nsnsm/ns

