Distributed Weight Based Clustering with Efficient Channel Access to Improve Quality of Service in Mobile Ad-Hoc Networks (DWCA)

Mohd. Amjad[1] and M.N. Doja[2]

[1] Asstt. Professor, Department of Computer Engineering
[2] Professor, Department of Computer Engineering
Faculty of Engineering & Technology, Jamia Millia Islamia
New Delhi, 110025, India
amjad2k3@yahoo.com, ndoja@yahoo.com

Abstract. Mobile ad hoc network is a set of mobile nodes connecting with each other without physical infrastructure and centralized computing. Factors such as variable wireless link quality, propagation path loss, fading, multiuser interference, power expended, and topological changes, become relevant issues. Quality of Service (QoS) is a set of service requirements that needs to be met by the network while transporting a packet stream from a source to its destination. QoS support for Mobile Adhoc Networks (MANETs) is a challenging task due to the dynamic topology and limited resources. The clustering algorithm presents a logical topology to the routing algorithm, and it accepts feedback from routing algorithm in order to adjust that logical topology and make clustering decisions. In this algorithm we have introduced a new metric, next hop availability, which is a combination of two metrics. It maximizes path availability and minimizes travel time of packets and therefore offers a good balance between selection of fast paths and a better use of network resources. In the conclusion it provides simulation result of DWCA Algorithm performed on network simulator.

Keywords: Clusters, Quality of Service support, Ad hoc network.

1 Introduction

Mobile ad hoc network (MANET) is a collection of wireless hosts that communicate with each other through multi-hop wireless links. Due to the absence of fixed infrastructure, nodes must collaborate between them to accomplish some operations like routing and security. Some envisioned MANETs, such as mobile military networks or future commercial networks may be relatively large (e.g. hundreds or possibly thousands of nodes per autonomous system). A way to support the increasing number of nodes in MANET is to subdivide the whole network into groups, and then create a virtual backbone between delegate nodes in each group. In ad hoc network this operation is called clustering, giving the network a hierarchical organization.

N. Meghanathan et al. (Eds.): NeCoM, WiMoN, and WeST 2010, CCIS 90, pp. 260–269, 2010.

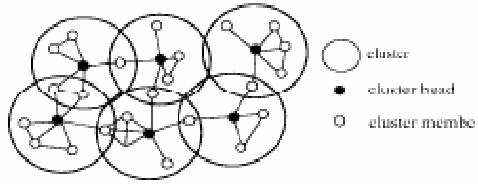


Fig. 1. Clustering in MANET

A cluster is a connected graph including a cluster head responsible of the management of the cluster, and (possibly) some ordinary nodes. Each node belongs to only one cluster. Some MANETs, such as mobile military networks or future commercial networks may be relatively large (e.g. hundreds or possibly thousands of nodes). A way to support the increasing number of nodes in MANET is to subdivide the whole network into groups, and then create a virtual backbone between delegate nodes in each group. In ad-hoc network this operation is called clustering, giving the network a hierarchical organization.

2 Clustering in MANETs

A way to support the increasing number of nodes in MANET is to subdivide the whole network into groups, and then create a virtual backbone between delegate nodes in each group. In ad-hoc network this operation is called clustering, giving the network a hierarchical organization. Several cluster based adaptations has been proposed for existed routing protocols and other protocol as ZRP (zone routing protocol), CBRP (cluster based protocol) have originally exploited this concept. Clustering for security can simplify the management of Certificate Authority in a Public Key Infrastructure (PKI) by affecting the full or a subset of Certificate Authority services to cluster heads, ensuring in this way the availability of the Certificate Authority. The Hierarchal organization consists of:

Cluster Head: A cluster head, as defined in the literature, serves as a local coordinator for its cluster, performing inter-cluster routing, data forwarding and so on. In our self-organized clustering scheme the cluster head only serves the purpose of providing a unique ID for the cluster, limiting the cluster boundaries.

Cluster Gateway: A cluster gateway is a non cluster-head node with inter-cluster links, so it can access neighboring clusters and forward information between clusters.

Cluster Member: A cluster member is a node that is neither a cluster head nor a cluster gateway.

Clustering has several advantages:

1) First, clustering allows the reuse of resource which can improve the system capacity, in the way that information is stored once on the cluster head.
2) Secondly, clustering may optimally manage the network topology, by dividing this task among specified nodes which can be very useful for routing since any node is identified by its identity and the identity of the cluster-head of the cluster to which it belongs, simplifying by this way the forwarding of messages.

3 Related Work

3.1 Highest-Degree Heuristic

The Highest-Degree, also known as connectivity-based clustering, in which the degree of a node is computed based on its distance from others. Each node broadcasts its id to the nodes that are within its transmission range. A node x is considered to be a neighbor of another node y if x lies within the transmission range of y. Experiments demonstrate that the system has a low rate of clusterhead change but the throughput is low under the Highest-Degree heuristic. As the number of nodes in a cluster is increased, the throughput drops and hence a gradual degradation in the system performance is observed.

3.2 Lowest-ID Heuristic

The Lowest-ID, also as known as identifier-based clustering, was originally proposed by Baker and Ephremides. This heuristic assigns a unique id to each node and chooses the node with the minimum id as a clusterhead. Thus, the ids of the neighbors of the clusterhead will be higher than that of the clusterhead. However, the clusterhead can delegate its responsibility to the next node with the minimum id in its cluster. A node is called a gateway if it lies within the transmission range of two or more clusterheads. The concept of distributed gateway (DG) is also used for inter-cluster communication only when the clusters are not overlapping. DG is a pair of nodes that lies in different clusters but they are within the transmission range of each other.

3.3 Node-Weight Heuristic

Basagni proposed two algorithms, namely distributed clustering algorithm (DCA) and distributed mobility adaptive clustering algorithm (DMAC). In this approach, each node is assigned weights (a real number > 0) 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. The smaller node id is chosen in case of a tie.

4 Limitations of Existing Algorithm

None of the above three heuristics leads to an optimal election of clusterheads since each deals with only a subset of parameters which can possibly impose constraints on the system. To be precise, the Highest-Degree heuristic states that the node with the largest number neighbors should be elected as a clusterhead. However, a clusterhead may not be able handle a large number of nodes due to resource limitations even if these nodes are its immediate neighbors and lie well within its transmission range. In other words, simply covering the area with the minimum number of clusterheads will put more burdens on the clusterheads. On the other hand, a large number of clusterheads will lead to a computationally expensive system. Although this may result in good throughput, the data packets have to go through multiple hops thus implying high latency.

Since the node ids do not change with time, those with smaller ids are more likely to become clusterheads than nodes with larger ids.Thus, certain nodes are prone to power drainage due to serving as clusterheads for longer periods of time.

5 Our Algorithm

5.1 Modifications over Existing Algorithms

The most of algorithms proposed in literature have giving solution to only some specific problems of ad hoc networks. However none of them deals with the entire characteristics of ad hoc networks (mobility, transmission range, size of the network, capabilities of the node). For example, the highest-degree, lowest-ID algorithms create one hop clusters, which are too small for large networks resulting on a big number of clusters, which complicate the virtual backbone management. They are also sensitive to small changing in the topology. The WCA and Mobility based algorithms try to include the mobility (stability) of nodes as a factor in the election procedure, in order to elect the most stable node as cluster-head. But their methods to compute stability are based on some assumptions which are not always valid in all ad hoc networks. Another method proposed in mobility based d-hop algorithm relies on the idea that nodes have equal antennas and transmit with the same power, to compute an estimate value of the distance between nodes. However this supposition is rarely guarantied because ad hoc networks are composed by heterogeneous nodes having different capabilities and antennas. Another observation is related to security features which are not included in the above algorithms. As mentioned in security problem must be taken into consideration in all schemes devoted to ad hoc networks.

5.2 The Design Approach

The main basic concepts used to derive the needed parameters are given below:

The Max Value: Represents the upper bound of the number of nodes that can simultaneously be supported by a cluster-head.

The Min Value: Represents the lower bound of the number of nodes that belong to a given cluster before proceeding to the extension or merging mechanisms.

D hops Clusters: As one hop clusters are too small for large ad hoc networks, therefore DWCA creates D hops clusters where D is defined by the underlying protocol or according to the cluster-head state (busy or not).

Identity (ID): It is a unique identifier for each node in the network to avoid any spoofing attacks or perturbation in the election procedure. We propose to use certificate as identity.

Weight: Each node is elected cluster-head according to its weight which is computed from a set of system parameters. The node having the greatest weight is elected as cluster-head.

Global Weight: using all parameters cited above every node in the network computes its global weight. Depending on this weight a given node can be elected as cluster-head or not.

We denote WT , WD , WB ,WM ,WS the partial weights and FT , FD , FB ,FM,FS are the weight factors corresponding respectively to Trust value, Degree, Battery, Max Value, and Stability. The global weight is computed as follows:

$$\mathbf{WG=FT\times WT+FD\times WD+FB\times WB+FM\times WM+FS\times(-WS)}$$

As we can observe the value of WS is retrieved from the global weight in order to elect the node with the greatest weight, because the stable node is the node with the smallest value of WS, which keeps the equation coherent for our assumption.

5.2.1 Election Procedure

This operation is invoked whenever a neighborhood has no cluster-head, or whenever one of the cluster-heads isn't able to achieve its responsibilities. Discovery stage: The purpose of this step is to get information about the neighborhood where the election procedure is invoked. Thus nodes desiring to be clusterhead send cluster-head_ready beacons within the radius of D hops. Each node when receiving this beacon estimates a trust value and sends it back to the asking node. After a discovery period Td, nodes having initiated this operation can derive from the received responses the following information:

- *Degree*
- *Stability:*
- *Trust value*

Computing weight: After the discovery stage, each node adds to the previous parameters the state of its battery and the max value, then combines them with the corresponding weight factors and computes the global weight.

Elaboration of the virtual backbone: Whenever the previous steps are successfully achieved, each cluster head need to discover each other to elaborate a virtual backbone to ensure inter-cluster services. Thus every new elected cluster-head broadcast a discovery request over the entire network; cluster-heads receiving this request register the certificate of the new clusterhead and send him their certificate.

5.3 Description of Algorithm

- Each node declares itself a clusterhead.
- Each node broadcasts the list of nodes that it can hear, that is, the set of nodes that are within the communication range of the original node. If a node A hears from a node B with a higher number of neighbors than itself, node A sends a message to B requesting to join B's cluster. If B already has resigned as a clusterhead itself, B returns a rejection, otherwise B returns a confirmation. When A receives the confirmation, A resigns as a clusterhead.
- When the previous step is completed, the entire network is divided into a number of clusters. This process can be repeated several times, depending on what the maximum cluster diameter is considered to be in this case. Each node keeps track of the id of its clusterhead, the first step in the path to its clusterhead, the distance to the clusterhead, the time the node has been a member of its current cluster, as well as the number of nodes in the cluster. A clusterhead also keeps track of the time each node in its cluster has been a member of that cluster.

- It is possible for a cluster to grow too large. Consider a situation when a cluster is just below the maximum allowed size, and several nodes join simultaneously.
- A node can leave a cluster, either because the situation described above, or because it is moving away from the cluster. Even if it loses contact with the node that is the first step to the clusterhead, it might still be able to connect to another node in the cluster. However, if the node is more than d-hops away from the clusterhead, it must leave that cluster. When a node leaves a cluster, it tries to find another cluster to connect to. That cluster must be smaller than the maximum allowed size, and the node cannot be more than d hops away from the clusterhead. If several such clusters are found, the node joins the largest one. If no such cluster is found, the node forms a cluster with itself as clusterhead and only member.

5.3.1 Cluster Initialization

The Init procedure is executed by each node in a no determinist status. A node with this status is a node which isn't attached yet to any cluster, this may be caused by a link failure, a roaming, or whenever a node coming for the first time to the network.

5.3.2 Cluster Division

A node can't serve for ever as CH (Cluster Head), because it has limited resources (battery, memory, etc). So, whenever it becomes busy (can't support the increasing number of nodes), the CH launches a cluster division procedure to divide the cluster into two small clusters with reasonable number of nodes. Therefore the CH broadcasts, Cluster_Division request to its CMs (Cluster Merge). Whenever this request is received, each CM computes its weight and sends it back to the CH. Then the CH chooses as a new CH the farthest node with the maximum weight and sends him a grant response. Then the new CH begins sending beacons and creates its own cluster.

5.3.3 Cluster Size Reduction

This operation is executed after the division of the cluster and aims to reduce the cluster radius from D to D-1, which means that beacons don't reach the boundaries of the cluster, resulting on the roaming of boundaries nodes to other clusters including new cluster creation.

5.3.4 Cluster Merging

In the algorithms taken from the literature, no lower bound is defined to limit the minimum number of nodes in a cluster, resulting on some clusters with two or one nodes which is not suitable. Therefore, in our algorithm we propose to merge such clusters immediately with the nearest cluster if it exists by executing the merging procedure. First, the CH begins to listen if there are any neighboring CHs; if this is the case it broadcast a merging request. Then it wait until receiving all confirmation from its CMs or the expiration of the delay TM to choose the nearest cluster and roams to that cluster.

6 Implementation and Performance Evaluation

The DWCA implemented algorithm is using GloMoSim (Global Mobile Information System Simulator), is a scalable network simulation environment for mobile ad-hoc networks, developed at UCLA Parallel Computing laboratory. And PARSEC (PARallel Simulation Environment for Complex systems) is a C-based discrete-event simulation language. GloMoSim has the capabilities to simulate thousands of mobile nodes without disregarding the details in the lower layer protocols. GloMoSim simulation library is built using the PARSEC simulation environment. With GloMoSim we are building a scalable simulation environment for wireless network systems. It is being designed using the parallel discrete-event simulation capability provided by Parsec discrete event simulation engine.

6.1 Transmitting a Hello Packet to the Nodes

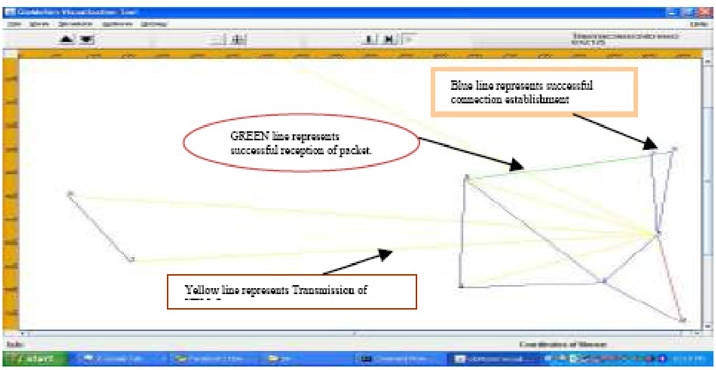
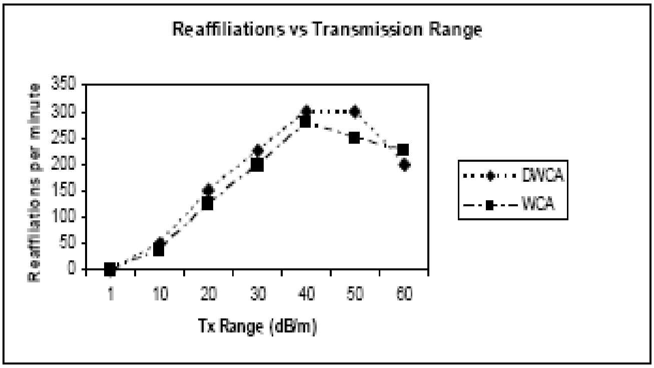


Fig. 2. Tx Packet to a node

Table. 1. Simulation Parameter

Network Size	300 X 300 m
Mobility of Nodes	20,40 and 50 Nodes
Range of each Node	625 m
Mobility Model	Random
Minimum Node Speed	5-20 m/sec
Pause Time	0,4,8 and 16sec
Data Rate	One Message per minute
Time	500 seconds



ï

Fig. 3. Reaffiliations Vs Tx Range

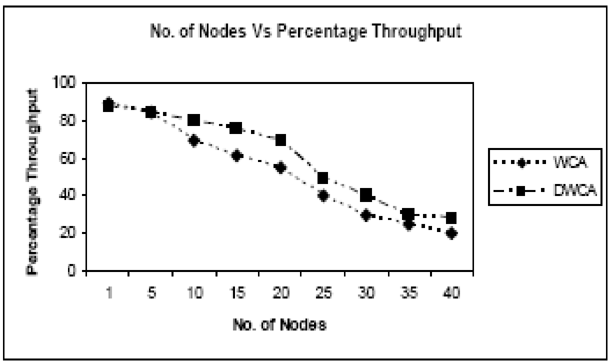


Fig. 4. Node Vs Throughput

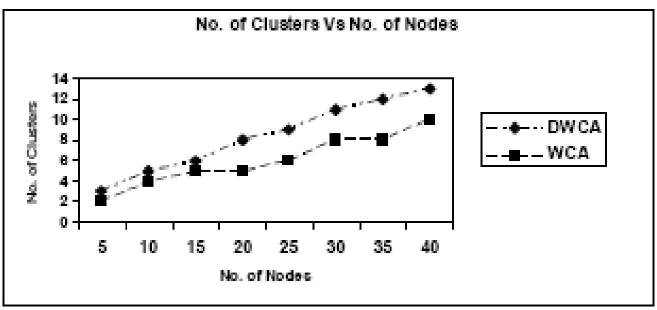


Fig. 5. Clusters Vs Nodes

7 Conclusion

We have presented a distributed algorithm, DWCA for the efficient partitioning of the nodes of an ad hoc wireless network into clusters with a clusterhead and some ordinary nodes. This is a practically important task, especially for all those network algorithms/applications that assume a mobility- adaptive hierarchical organization of the network. A new weightbased criterion is introduced for the cluster formation that allows the choice of the clusterheads based on node mobility, battery power, degree, cumulative link strength and data rate that were not available in previous clustering algorithms. It is seen that a tradeoff between network stability and effective clusterhead selection process is to be made. WCA compromises on the latter by not always selecting the best nodes as clusterheads, whereas our algorithm, DWCA improves upon the clustering process and the network organization in terms of well spread out clusters while sacrificing a bit of network stability. It also combines easiness of implementation with full adaptation to the mobility of the nodes, even during clustering set up.

References

1. Ganesan, D., Govindan, R., Shenker, S., Estrin, D.: Highly-resilient, Energy-efficient multipath Routing in Wireless Sensor Networks. Mobile Computing and Communications Review 4(5) (October 2008)
2. Mhatre, V., Rosenberg, C., Koffman, D., Mazumdar, R., Shroff, N.: A minimum cost heterogeneous sensor network with a lifetime constraint. IEEE Transactions on Mobile Computing (TMC) 4(1), 4–15 (2005)
3. Arboleda, L.M., Nasser, N.: Comparison of Clustering Algorithms and Protocols for Wireless Sensor Networks. In: Canadian Conference on Electrical and Computer Engineering (CCECE 2006), Ottawa, May 2006, pp. 256–261 (2006)
4. Akkaya, K., Younis, M.: A Survey on Routing Protocols for Wireless Sensor Networks. Elsevier Ad Hoc Network Journal 3(3), 325–349 (2005)
5. Luo, J., Hubaux, J.-P.: Joint Mobility and Routing for Lifetime Elongation in Wireless Sensor Networks. In: 24th Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2005, Miami, March 2005, pp. 1735–1746 (2005)
6. Barbeau, M., Kranakis, E., Krizanc, D., Morin, P.: Improving Distance Based Geographic Location Techniques in Sensor Networks. In: 3rd International Conference on ADHOC Networks and Wireless, Vancouver, British Columbia (July 2004)
7. Muruganathan, S.D., Ma, D.C.F., Bhasin, R.I., Fapojuwo, A.O.: A Centralized Energy-Efficient Routing Protocol for Wireless Sensor Networks. IEEE Communications Magazine 43, 8–13 (2005)
8. Dasgupta, K., Kalpakis, K., Namjoshi, P.: An Efficient Clustering-Based Heuristic for Data Gathering and Aggregation in Sensor Networks. IEEE Wireless Communications and Networking 3, 1948–1953 (2003)
9. Bandyopadhyay, S., Coyle, E.: An Energy-Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks. In: Proceedings of IEEE INFOCOM (April 2003)
10. Michiardi, P., Molva, R.: Core: A Collaborative Reputation mechanism to enforce node cooperation in Mobile Ad Hoc Networks. In: Communications and Multimedia Security Conference (2002)

11. Younis, O., Fahmy, S.: Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach. In: Proceedings of IEEE INFOCOM, vol. 1, pp. 629–640 (March 2004)
12. Mhatre, V., Rosenberg, C.: Design guidelines for wireless sensor networks: communication, clustering and aggregation. Ad Hoc Networks 2(1), 45–63 (2004)
13. Akkaya, K., Younis, M.: A survey on routing protocols for wireless sensor networks. Ad Hoc Networks 3(3), 325–349 (2005)
14. Choi, W., Shah, P., Das, S.K.: A framework for energy-saving data gathering using two-phase clustering in wireless sensor networks. In: Proceedings of International Conference on Mobile and Ubiquitous Systems, August 2004, pp. 203–212 (2004)
15. Liu, J.S., Lin, C.H.: Energy-efficiency clustering protocol in wireless sensor networks. Ad Hoc Networks 3(3), 371–388 (2005)
16. Lin, C.R., Gerla, M.: Adaptive clustering for mobile wireless networks. IEEE Journal on Selected Areas in Communications 15(7), 1265–1275 (1997)
17. He, T., et al.: SPEED: a stateless protocol for real-time communication in sensor networks. In: Proceedings of International Conference on Distributed Computing Systems, Providence, RI (May 2003)
18. Estrin, D., Culler, D., Pister, K., Sukhatme, G.: Connecting the physical world with pervasive networks. IEEE Pervasive Computing, 59–69 (January- March 2002)
19. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: A survey. Computer Networks 38(4), 393–422 (2002)
20. Vyas, N., Mahgoub, I.: Location and Mobility Pattern Based Routing Algorithm for Mobile Ad Hoc Wireless Networks. In: International Symposium on Performance Evaluation of Computer and Telecommunication Systems, Montreal, Canada (July 2003)
21. Barbeau, M., Kranakis, E., Krizanc, D., Morin, P.: Improving Distance Based Geographic Location Techniques in Sensor Networks. In: 3rd International Conference on ADHOC Networks and Wireless, Vancouver, British Columbia (July 2004)
22. Kadri, B., M'hamed, A., Feham, M.: Secured Clustering Algorithm for Mobile Ad Hoc Networks. IJCSNS International Journal of Computer Science and Network Security 7(3) (March 2007)