Energy Efficient Mobility Adaptive Distributed Clustering Algorithm for Mobile Ad Hoc Network

Suchismita Chinara [#1], Santanu Kumar Rath [#2]

[#] *Department of Computer Science and Engineering,*
National Institute of Technology, Rourkela, Orissa, India.

[1] suchi.nitrkl@gmail.com

[2] rath.santanu@gmail.com

***Abstract*— In this paper, we propose energy efficient mobility adaptive distributed clustering algorithm for mobile ad hoc network. Node mobility in the dynamic network has a remarkable affect on cluster stability. In order to reduce the initial cluster setup time of the dynamic network with frequently changing topology, we consider a single node parameter as the cluster head selection criteria. That is the average displacement of individual nodes in the network is computed by keeping track of its total displacement in last n time units. As the selected cluster heads form the routing backbone of the dynamic network, better stability is ensured by preferring low mobile nodes to act as cluster heads. A new energy consumption model has been proposed for the cluster heads that takes into account the network traffic, density of cluster members and the transmission power utilized to communicate the member nodes. A better cluster stability and a low maintenance overhead is aimed to achieve by electing volunteer and non-volunteer cluster heads.**

I. Introduction

Mobile ad hoc network supports multi hop routing where the deployment of central base station is neither economic nor easy. Recent developments in the technologies of laptops, PDAs and sensors as well as the reduction in their cost have exponentially raised the interest in wireless mobile ad hoc networks [13]. Situations like disaster recovery, battle field communication and law enforcement operations demand for setting up a network in no time. MANET is a proven solution for such emergencies. However, limited battery power and computing capability, scarce in radio resources and bandwidth, frequent node mobility and link failure perturbs the efficient network services [2].

In cellular network the mobile nodes communicate directly with the fixed base station reducing the wireless part of communication to a single hop problem. Many good solutions have been proposed for handling of mobility of the nodes by this base station. Thus the mapping of cellular architecture into peer to peer network leads to the concept of clustering [6]. In such a virtual cellular architecture (VCA) cluster heads are selected to play the role of base station as in cellular architecture. The cluster head along with its one-hop members form the virtual cells retaining the merits of cellular structure. The cluster head within each cluster acts as the local coordinator for its member nodes. The one-hop distance of member nodes from its cluster head guarantees faster communication [3].

Cluster heads being the communication hotspots tend to drain its battery power rapidly while serving its member nodes [8]. In this paper we propose a distributed clustering algorithm that makes a fair distribution of cluster headship among all the nodes. The algorithm aims to distribute the time for which a node is selected as cluster head in an uniform manner so that every node obtains nearly equal opportunity to act as a central router for its neighbor nodes. A faster cluster setup is aimed to achieve such that unlike ([5], [12]) the freezing time of motion for mobile nodes during clustering can be minimized. Further, energy consumption is a key factor that hinders the deploy ability of a real ad hoc and sensor network. It is due to the limited life time of the battery powered devices that motivates intense research into energy efficient design of operating systems, protocols and hardware devices. We consider the energy consumption of every node in a broadcast environment. Specially, we propose a new energy consumption model for the head nodes by considering the amount of data forwarded, the service cost for its members and the total transmission power utilized by it. The rest of the paper is organized as follows. Section II describes the work done in the related area followed by section III, which gives a brief description about the proposed algorithm with subsections like *cluster formation*, *energy consumption model* and *cluster maintenance*. Simulation result is discussed in section IV followed by section V that concludes the paper.

II. Related Work

Cluster control structure is found to be a proven solution for efficient resource management in mobile ad hoc network. There exists some one-hop clustering algorithms that emphasize either on minimizing number of cluster heads ([1], [2]) in the virtual back bone to reduce the routing delay or maximizing the cluster stability by un-altering the head nodes ([3],[7]). A combined metric clustering algorithm ([5], [10]) uses parameters like running average, degree of connectivity, transmission power and available battery power of a radio to find its suitability as a head. But obtaining so much of information to compute the combined weight for every node in the network itself needs a longer frozen period of motion before the cluster is actually formed. Moreover, a large number of message exchanges are required globally among all the nodes to yield a node with lowest weight.

The author in ([4], [9]) proposed a weight based distributed mobility adaptive algorithm DMAC that removes the nonmobility assumption of the hosts during clustering setup and maintenance. Heads are selected among the radio nodes with the knowledge of their local topology resulting in a faster cluster setup period. This algorithm adopts the topology change in the network during the cluster setup as well as maintenance period. This major advantage of DMAC makes it more realistic in ad hoc network. However, the criteria by which the nodes are selected as heads are not specified in it. Also, this algorithm does not allow two cluster heads to be neighbors of each other. That is when two head nodes come within the transmission range of each other; the node with the lower weight has to resign its role as head and is forced to become the member node of the node with higher weight. This restriction of non-neighborhood of two cluster heads increase the cost of cluster maintenance by increasing frequent reaffiliations of the member nodes and the rate of dominant set updates. Moreover, the condition of affiliation of member nodes to a head with higher weight than its present head (if it finds any at any time within its transmission range) further increases the reaffiliation rate reducing the clustering efficiency.

None of the above algorithms emphasize on the energy constraint of the mobile radios. Though the authors in ([5], [10]) takes the residual power as one of the metric for computing the combined weight, but the cluster head serving time alone can not assure a good prediction for energy consumption. Recently, Yang and Zhang [12] have proposed a new weight based clustering algorithm WBCA that considers the mean connectivity degree and the consumed energy of a node to calculate its weight. As per the algorithm the consumed energy is dependent on the degree of connectivity of the nodes. Calculating the mean connectivity degree of a node also needs to calculate the degree of connectivity of its neighbor nodes. In a dynamic network where the topology changes frequently and the degree of connectivity changes rapidly, keeping the connectivity information of every node in a local proximity increases further complexity in computation as well as the freezing time prior to the cluster setup. The authors in [18] have proposed an energy efficient cluster design which is possible either in case of a static network or where the cluster head is known a priori. For a real ad hoc network these conditions are never satisfied.

III. Preliminaries of Proposed Algorithm

The mobile ad hoc network can be modeled as a unidirectional graph G = (V, L) where V is the set of mobile nodes and L is the set of links that exist between the nodes. We assume that there exists a bidirectional link L_{ij} between the nodes i and j when the distance between the nodes d_{ij} < t_{range} (transmission range) of the nodes. In the dynamic network the cardinality of the nodes $|V|$ remains constant, but the cardinality of links $|L|$ changes due to the mobility of the nodes. A preliminary version of this algorithm appeared in [21]. The algorithm works with the following assumptions:

- Nodes in the network have equal transmission range
- Every cluster head has weight information regarding its cluster member nodes.

A. Cluster Formation

In a dynamic network the mobility of nodes can not be ignored. It has a vital role in maintaining the stable cluster structure. Thus we choose mobility or average displacement to be the deciding factor for initial cluster setup so that a better cluster stability can be achieved. A node with lower average displacement has a higher chance of being a cluster head. The weights assigned to the nodes are reciprocal to their respective displacements. i.e. a node with lower avg. displacement is assigned a higher weight and the node with higher value is assigned a lower weight.

Basically, we consider the mobility of a node by taking the average of the distances covered by it in last n time slots. That is

Total distance moved at time t is $D_t = \sum_{i=t-n}^{i=t} Dist_i$ where i= t is current time.

Thus, Mobility M = Total distance D_t / n

Here, instead of taking the most recent displacement of a node we take a statistics of last n distances covered by it. This gives a prediction of its mobility characteristic and identifies the node with minimum average displacement. Then the weights are calculated for every node which is inversely proportional to its mobility. In order to find the weight we take a very simple ratio as

$$W_i = \frac{k}{M_i}$$

Where, the value of k is a constant that can be decided depending on the network area. During the initial cluster setup, a node having maximum weight among its 1-hop neighbors is selected as the cluster head. And its 1-hop uncovered neighbors (i.e. whose role is not yet decided) become the members of the current head. These set of covered nodes are exempted while selecting the next lower weighted nodes and this process is repeated till all the nodes are assigned with their role either as a head or a member. This ensures a fast cluster setup with minimum freezing time of motion by the nodes. During this phase every node broadcasts its ID along with its weight W_i to all neighbors and stores the weights W_j that it hears from other nodes. If it does not hear another node ID with weight higher than itself then it becomes a cluster head and its one-hop uncovered neighbour nodes become its members. In case of a tie in the node weights the low ID node is preferred for the role of cluster head. Unlike [4] once a member node is affiliated with a cluster head, it does not re-affiliate to a new head unless it goes out of the range of its current head or the head drains out of battery power. This reduces the number of re-affiliations lowering the cluster maintenance cost.

We calculate the Cumulative Time T_i during which node i acts as a cluster head as $T_i = \sum_{k=1}^{q} T$ where q indicates the times of period for which a node i acts as cluster head.

We start with the algorithm by considering the following data structures:

1) ***NODESTATUS****:* this array stores the status of every node in the network. We are considering only 3 status of a node. That is a head node, a member node and an uncovered node whose status is not yet decoded.

2) ***RUNNINGDISTANCE:*** this array of size **m x n** stores the distances covered by every node for last **n** unit of time elapsed. This array is updated in every unit of time by deleting the values in the left most position of every row and shifting the rest of the values to 1 position left so that the new distance covered by every node can be inserted to the right most position.

3) ***NEIGHBORS:*** this array of size **m x m** keeps the record of the links between the nodes in the network. This array stores the binary value of 1 or 0 depending on the presence of a link or not between every pair of nodes.

4) ***AVERAGE:*** this array of size **m** stores the running average of the nodes in every unit of time. This array is updated in concurrence with the RUNNINGDISTANCE array. It calculates the average of every row of the former array and updates its corresponding field.

The steps of the algorithm are given in Table I.

TABLE I
PSEUDO CODE OF THE PROPOSED ALGORITHM

Begin Cluster Setup

Call weight_calculation
For (every $v \in V$)
If $w_v \rangle w_i$ where $i \in \Gamma(v)$ //$\Gamma(v)$ is the neighbor set of v
Then Set head= v
For (every $x \in V_{uncovered}$)
If dist (head, x) $\langle =$ head$_{trange}$ Then
Set HEAD$_x$ = head
End for
End for
End Cluster setup

Begin weight_calculation

For (i =1 to m) // for m nodes
For (j = 1 to n-1) // for last n units of time
running Distance(j, i) := running Distance (j + 1, i)
end for

running Distance(n, i) := $\sqrt{(x(i)_{t_n} - x(i)_{t_{n-1}})^2 + (y(i)_{t_n} - y(i)_{t_{n-1}})^2}$
For (i = 1 to n-1)
average (i):= average (i)+ running Distance(n, i)
end for
average (i) := average (i)/n
end for

For (i =1 to m)
weight (i):=k / average (i) //weight calculation
end for
End weight_ calculation

The example of cluster setup phase of the proposed algorithm is demonstrated with the help of Figure 1 to Figure 3. To start with the Fig.1 every node is identified with a unique ID and its associated weight in parenthesis. We assume that the weights are already being computed for every node.

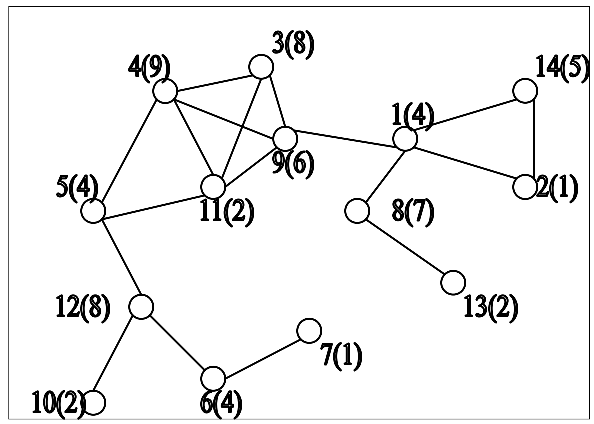

Fig.1 Initial topology of nodes after weight calculation

The link between every pair of nodes denotes that they are within the transmission range of each other and establish a bidirectional link among each other.

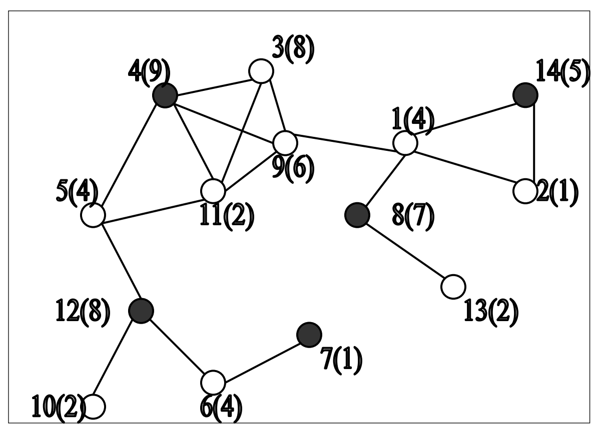

Fig. 2 Initial cluster heads are identified

Fig.2 shows the network after the cluster heads are identified after the exchange of their weights within the local topology. A node having the higher weight among its 1-hop neighbors become the head and its immediate uncovered neighbors become its members. Fig. 3 shows the network after the initial clusters are formed.

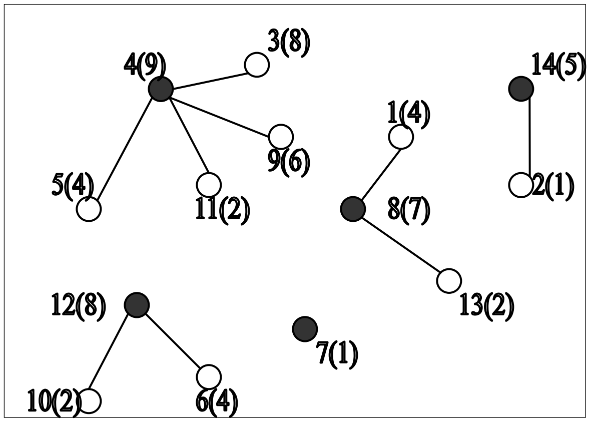

Fig.3 Clusters are formed

Here, the dark circled nodes indicate the cluster heads and the ordinary circles denote the member nodes. As we know this is only the virtual partition of the nodes. And this structure changes with every topology change in the dynamic network.

B. Energy Consumption Model

The effect of ad hoc operation on power consumption has been explained in [17]. According to 802.11, wireless network interface can work in either base station mode (BSS) or ad hoc mode. In BSS mode, every mobile host remains in the transmission range of one or more base station, which are responsible for buffering and forwarding traffic between hosts. Similarly, nodes can send outgoing traffic to the base station anytime and to receive the incoming traffic it can poll the base station periodically. The rest of the time the node can enter into sleep state, from which the interface must wake up in order to send or receive traffic. The guaranteed availability of fixed infrastructure like base station for buffering and traffic management supports this kind of energy conserving functionality.

In contrary, ad hoc mode of operation does not use any base station. So nodes communicate directly with all other nodes which are in its transmission range. This demands nodes to remain ready to receive traffic from their neighbors and does not allow a node to enter into sleep state. However, a node can enter into idle mode when it constantly listens to the wireless media and consumes energy which is almost same as the energy consumption in receiving traffic. Thus, energy consumption of the mobile devices depends on the operating mode of its wireless network interfaces.

Considering a broadcast communication between the nodes of the dynamic network, wireless interfaces can be in any of the following operating modes:

1) ***Transmit:*** for transmitting data.

2) ***Receive:*** for receiving data.

3) ***Idle:*** a default mode when the node is ready to transmit or receive.

4) ***Sleep:*** the low power consumption state when a node can not transmit or receive until woken up.

For simplicity we have considered a linear model for the energy consumption cost of mobile nodes for sending or receiving a packet. The linear model [19] consists of an incremental cost **m** and a fixed cost **c** that is represented for a broadcast communication as

$$Energy_{member} = m_{send/receive} \; X \; size_{packet} + c_{broadcast}$$

In a broadcast traffic, the sender listens briefly to the channel. If the channel is found to be free then the packet is sent and is received by all nodes in wireless range. If the channel is found easy, the sender has no choice but to back off and retry later. In order to avoid the complexity of retransmission, we have ignored the situation of retransmissions, acknowledgements as well as discarding cost of overhear packets by the hidden and exposed terminals.

The situation is little bit different for the cluster heads. In addition to the job of forwarding the packets to and from the cluster, it has an additional job of resource management for its members. Thus the cost of consumption of energy is proportional to the number of member nodes served by the cluster head. Again the radio range coverage by the head node has a considerable effect on its energy consumption. Depending on the RF environment the energy consumption can vary from p_v^2 to p_v^4 [20], where p_v is the transmission power utilized by the head node in communicating a one-hop neighbor v within its cluster. As we consider the distance between the nodes in a cluster is very small, so we setup a linear relation between the transmission power and the energy consumption of the head node. As a whole, we consider the energy consumption of a head node basically depends on the following parameters:

- The traffic forwarded by the head
- No. of members served by the head
- Total transmission power utilized by the head in serving the members

and proposed a simple linear equation as

$$Energy_{head} = \alpha * |n_i| + \beta * Traffic_{bcast} + \gamma * \sum_{v' \in n_i} dist(v, v')$$

Where $|n_i|$ represents the cardinality of the cluster, $Traffic_{bcast}$ is the cost of energy consumption in traffic forwarding and $\sum_{v' \in n_i} dist(v, v')$ is the total transmission power utilized in communicating the member nodes of the cluster head. α, β and γ are the weighing factors for the corresponding network parameters. These values are kept flexible so that they can be changed as per the network scenario. For example, when the network traffic is very high β can be given more weight age than the other two. Similarly, in a dense network where the cardinality of clusters are more, the weight age of α dominates the other factors. All three parameters are chosen so that $\alpha + \beta + \gamma = 1$.

C. Cluster Maintenance

The process of clustering can be visualized as a pack of two consecutive stages, i.e. cluster formation and cluster maintenance. Once the cluster setup phase is over, the objective of cluster maintenance is to preserve as much as of the existing clustering structure as possible. In one-hop cluster there exists a bidirectional link between the cluster head and the members till both are within each other's transmission range. When any of them moves away from each other, there occurs a link failure and the member node searches for another new head within its close proximity to get affiliated. Such a changing situation of a member node from its current head to another head is called reaffiliation. For every single reaffiliation of a member node both its old and new cluster head update their member list by deleting or inserting the ID of this changing member resulting multiple messages to flow betweens the nodes and increase the message complexity as well as congestion of the network. Thus minimizing the reaffiliation rate is the objective of any clustering algorithm.

There exists one more overhead in maintaining the linked cluster architecture, which is the average number of cluster head set updations per unit time. When the existing set of cluster heads become unable to serve the member nodes or form linked cluster architecture, an updation takes place by selecting a fresh set of cluster heads within the network. A lower cluster head updation frequency can lead to improved route stability.

For our proposed algorithm a **reelection** of cluster head or updation of cluster head set takes place when:

- A single node becomes orphan or isolated by moving away from all other nodes and declares itself as a volunteer cluster head.
- An existing cluster head consumes the threshold amount of battery power; so that it resigns from its current role and hands off its role to one of its member node with maximum available battery power. Such a forcibly elected node is called non-volunteer cluster head.

Similarly, a **reaffiliation** occurs when:

- A member node leaves its current head's transmission zone and enters into another cluster zone.
- A non-volunteer head is elected so that nodes of the existing head reaffiliates to the new head (if exists within the transmission range) or finds another head within its range.

IV. Simulation Result and Discussion

We simulate our mobility based algorithm using the Random Walk mobility model. This mobility model ([14], [15]) represents the most erratic and unpredictable movement of an entity. Here, a mobile node (entity) moves from its current location by choosing a random speed between (speedmin, speedmax) and a random direction between (0, 2π) respectively. In random walk model when a mobile node reaches a simulation boundary it bounces back with an angle determined by the incoming direction. This is a memory less mobility pattern as it retains no knowledge about its past location and speed value. This model can be used for vehicular and large scale environments.

We simulate a system of M nodes in a 100 x 100 grid area. And the simulation time is kept for 10000 seconds. The maximum mobility of nodes is kept between 0 m/sec to 5 m/sec. In our simulation the value of K was taken as 1000 and the running average of every node is calculated for 5 unit of time (i.e. n=5) elapsed. The simulation result for the cumulative time of a node acting as a cluster head is shown in Fig.4 where M is considered as 35. We made a comparison with the well known lowest ID algorithm and the recent WBCA algorithm.

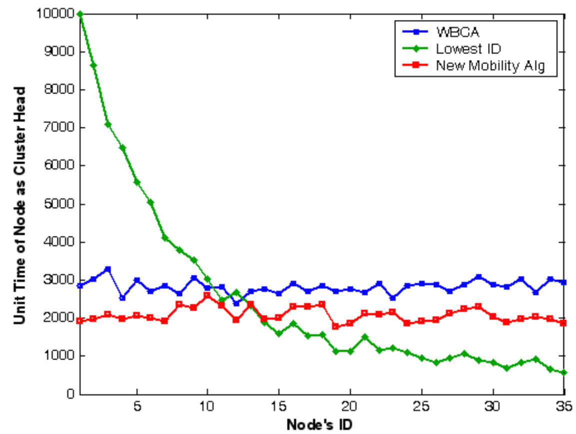

Fig. 4 The time of a node acting as a cluster head

It can be seen from the result that the distribution of cumulative time T_n for every node to act as a cluster head is not uniform in the Lowest ID algorithm. The nodes with lower ID get the opportunity to remain as head for a longer period of time. But the situation is different in other algorithms. The proposed algorithm and the WBCA algorithm gives almost equal chance to every other node to act as a cluster head which is clearly visible in figure 4.

The simulations of the new energy consumption model for the algorithms are shown in the following figures. For the member nodes the measurements of the energy consumption during different operating modes are considered ([17], [19]) as:

Broadcast send= 1.9 μ W.s/byte * size $_{packet}$ +250 μ W.s

Broadcast recv= 0.50 μ W.s/byte * size $_{packet}$ +56 μ W.s

Idle = 808 mW

and the packet size is taken as 1024 for the simulation. Figure 5 represents the result for the bench mark Lowest ID algorithm for different transmission ranges. It is clear from the figure that higher transmission range of a node consumes more energy than that of in lower transmission range. Moreover, the nodes with the lower IDs consume maximum energy where as the energy consumption of higher ID nodes is almost negligible. The result shown in figure 5 is a proof of the result of figure 4 that the period of time for a node to serve as a cluster head is directly proportional to its energy consumption.

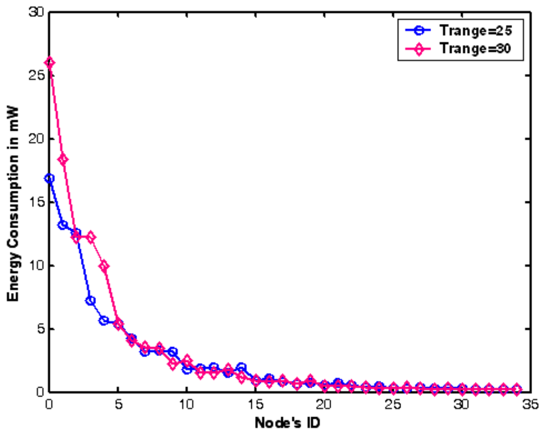

Fig. 5. Energy consumption model of Lowest ID algorithm for different transmission range and Max. displacement =5.

As the lower ID nodes remain as heads in most of the simulation time, so the energy consumption is greater for them. The energy consumption of nodes of our proposed algorithm is shown in figure 6. Here, the period of getting the headship by the nodes is randomly distributed among them. So the energy consumptions by the nodes are also randomly distributed. The spikes of the graph represent the consumption by the head nodes as they do the traffic forwarding as well as the member management within a cluster.

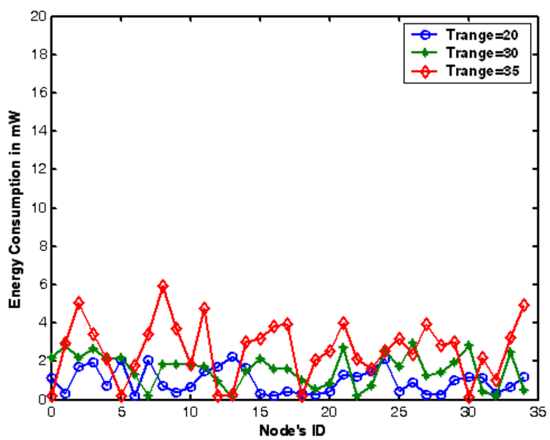

Fig. 6 Energy consumption model of proposed algorithm for different transmission range and Max. displacement =5 .

The comparison of the simulation result for both the algorithms is shown in figure 7. It is clearly visible that in the lowest ID algorithm the nodes with lower IDs are biased for the headship and drain their energy sooner reducing the network lifetime as well as cluster stability. Our proposed algorithm makes a random selection of head nodes according to their average speed in previous n seconds. This removes the biasness of any node to act as a head so that the consumption of energy is fairly distributed among them. This improves the network lifetime as well as cluster stability by preventing a single node to become overloaded and finally dead.

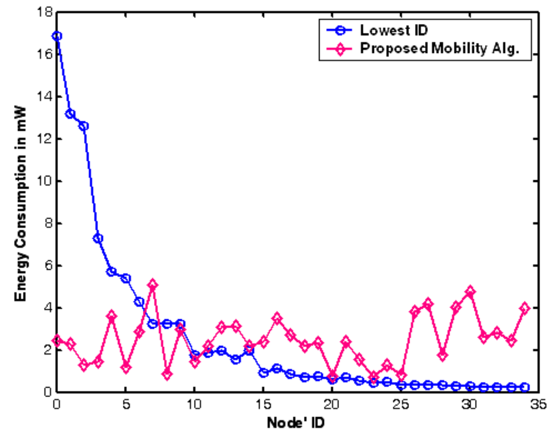

Fig. 7 Energy consumption of nodes, Trange=25 and Max. displacement =5.

As explained earlier, cluster head updation takes place from time to time by electing the volunteer or non-volunteer nodes in the dynamic network. It can be observed from Fig. 8 that the node density does not have much effect on the head updation rate so that the maintenance cost is also unaffected. Similarly the nearly flat structure of the graph indicates that the rate of updation remains marginally affected with increase in the transmission range. Thus to have low power consumption by the cluster heads, the range of the nodes can be kept low.

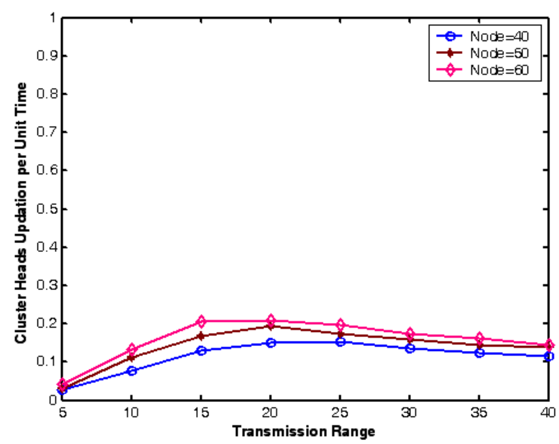

Fig. 8 Cluster head updation for node densities with Max. displacement =5.

Figure 9 denotes the rate of reaffiliation by the member nodes with various transmission ranges. It can be seen that though the transmission range does not have much effect on the reaffiliation rate (for higher value the graph is almost flat) but the node density changes the value considerably. This can be easily understood that higher node density increases the energy consumption of the heads and lead to non-volunteer head election. This indirectly causes several reaffiliations to occur.

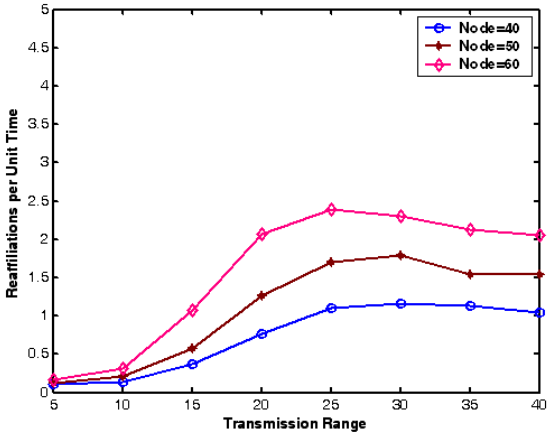

Fig. 9 Cluster reaffiliation for node densities with Max. displacement =5.

Comparison of head updation rate for different algorithms is indicated in Figure 10. The updation cost of our proposed algorithm is much lower in comparison to that of lowest ID algorithm. In comparison to WBCA, the result has a considerable difference for low transmission range though it almost merges for high transmission range. A lower cluster head updation frequency indicates better routing stability as the cluster heads take major role of routers in forwarding the packets.

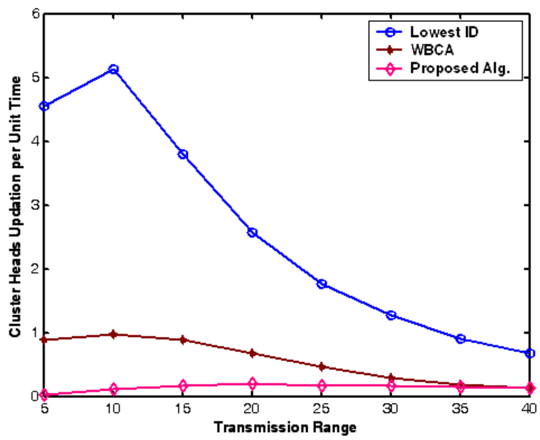

Fig. 10 Cluster head updation of algorithms with Max. displacement =5 and Node=50.

Figure 11 shows the comparison of the reaffiliation rate of the algorithms with various transmission ranges when 50 nodes are considered. The result indicates that the number of reaffiliations is marginally higher that the WBCA though considerably lower than lowest ID algorithm. So it is concluded that if the reaffiliation cost can be compromised slightly, then the proposed algorithm can achieve better cluster stability with minimum energy consumption.

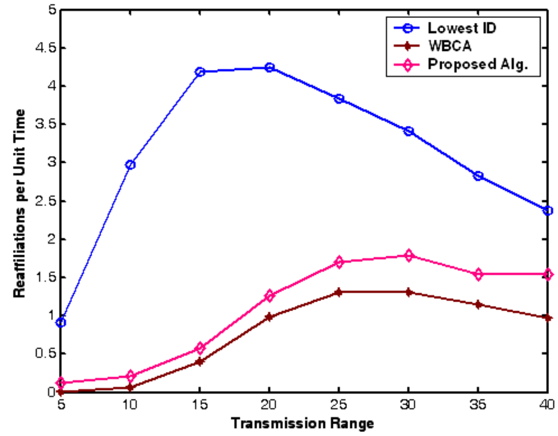

Fig. 11 Cluster reaffiliation of algorithms with Max. displacement =5 and Nodes=50.

V. Conclusion

An energy efficient and mobility adaptive distributed clustering algorithm is proposed in the dynamic network in order to reduce the freezing time of motion of mobile nodes during the initial cluster setup. We keep a record of previous n set of movements of every node to predict their average mobility. A node with lower mobility is assigned higher weight value so that cluster stability can be improved. The simulation result proves that the nodes get the role of cluster heads in a fairly distributed fashion. A new energy model has been proposed for the cluster heads that considers several factors like the traffic load, network density and the total transmission power utilized in handling the cluster members. Cluster maintenance overhead is aimed to reduce by electing Volunteer and non-volunteer cluster heads from time to time. It is seen in the result that the cluster head updation frequency is very low in comparison to any other algorithm developed so far.

References

[1] M.Gerla and J.T.C.Tsai, "multicluster, mobile,multimedia radio network", *wireless networks*, 1,3,pp.255-265,1995.

[2] A.K.Parekh, "Selecting routers in ad hoc wireless network", *Proceedings of the SBT/IEEE international telecommunication symposium*, August 1994.

[3] D.J.Baker, J.E.Wieselthier and A. Ephremides, "A design concept for reliable mobile radio networks with frequency hoping signaling", *proceedings of IEEE*, vol. 75, no.1, Jan. 1987.

[4] S.Basagni, "Distributed and mobility-adaptive clustering for multimedia support in multi-hop wireless networks", *proceedings of vehicular technology conference*, *VTC*, vol.2, pp. 889-893, 1999.

[5] S.Das, M.Chatterjee, D.Turgut, "Wca: A weighted clustering algorithm for mobile ad hoc networks", *Journal of Cluster computing (special issue on mobile ad hoc networks)*, vol. 5, no. 2, 4, pp. 193-204, 2002.

[6] S. Basagni, I. Chlamtac, A. Farago, "A generalized clustering algorithm for peer-to-peer networks", *Proceedings of Workshop on Algorithmic aspect of communication* (satellite workshop of ICALP), July 1997.

[7] D.J.Baker and A. Ephremides, "The architectural organization of a mobile radio network via a distributed algorithm", *IEEE Transactions on Communications COM-29*, pp. 1694-1701, 11(Nov.1981).

[8] J.H.Chang and L.Tassiulas, "Energy conserving routing in wireless ad hoc networks", *Proceedings of INFOCOM 2000*, Tel-Aviv, Israel, March 2000.

[9] S. Basagni,"Distributed clustering for ad hoc networks", *Proceedings of International Symposium on Parallel Architectures, Algorithms and Networks*, pp.310–315, June 1999.

[10] S.Das, M.Chatterjee, D.Turgut, "An on demand weighted clustering algorithm(WCA) for ad hoc networks", *in proceedings of IEEE GLOBECOM 2000*, pp.1697-1701, San Francisco, Nov. 2000.

[11] T.Ohta, S.Inoue, Y.Kakuda, "An adaptive clustering scheme for highly mobile ad hoc networks", *in proceedings of sixth international symposium on autonomous decentralized systems(ISADS'03)*, April 2003.

[12] Wei-dong Yang and G-z Zhang, "A weight-Based Clustering Algorithm for mobile Ad Hoc Network", *proceedings of Third International Conference on Wireless and Mobile Communication* (ICWMC'07), 2007.

[13] S.Basagni, M. Conti, S.Giordano and I.Stojmenovic, *Mobile ad hoc networking,* A John Wiley & Sons Inc. publication, 2004.

[14] T.Camp, J.Boleng, V.Davies. "A Survey of Mobility Models for Ad Hoc Network Research" Wireless Communications and Mobile Computing (WCMC): Special issue on *Mobile Ad Hoc Networking: Research, Trends and Applications,* vol. 2(5), pp. 483-502, 2002.

[15] G.Lin, G.Noubir and R. Rajaraman, "Mobility models for ad hoc network simulation", *IEEE INFOCOM 2004*.

[16] E.Lochin, A. Fladenmuller, J-Y Moulin and S. Fdidda, "Energy consumption models for ad-hoc mobile terminals", *Paris Cedex 05*, France, 2005.

[17] L.M. Feeney and M Nilsson, " Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment", *Proceedings of IEEE INFOCOM 2001*, Anchorage, Alaska, vol.3, pp. 1548-1557, 2001.

[18] C-F Chiasserini, I. Chlamtac, P. Monti, A. Nucci, " Energy Efficient Design of Wireless Ad Hoc Networks", *proceedings of European wireless*, February 2002.

[19] L.M. Feeney, " An Energy Consumption Model for Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks" , *Mobile Networks and Applications , Kluwer Academic Publishers*, vol. 6, pp. 239-249, 2001.

[20] Q. Gao, K.J.Blow, D.J. Holding , I.W.Marshall and X.H.Peng, "Radio range adjustment for energy efficient wireless sensor networks" , *Ad hoc networks* , vol4, pp.75-82, 2006.

[21] S. chinara, S.K.Rath, P. Bhattad and A. Dey, "Mobility Based Distributed Clustering Algorithm for Mobile Ad Hoc Network", *Proceedings of NCNTE-2008*, Vashi, Mumbai, pp. 169-172, 2008.