

A novel weight based clustering algorithm for routing in MANET

Sunil Pathak¹ · Sonal Jain¹

Published online: 20 November 2015
© Springer Science+Business Media New York 2015

Abstract Cluster based routing in Mobile AdHoc Networks are considered one of the convenient method of routing. Existence of Cluster Head (CH) in a group of nodes for data forwarding improves the performance of routing in terms of routing overhead and power consumption. However, due to the movement of CH and frequent change in cluster members, cluster reformation is required and increases cluster formation overhead. The stability of the cluster highly dependent of stability of the CH and hence during CH selection special care should be taken so that the cluster head survives for longer time. In this paper a method of cluster formation is proposed which will take into account two most vital factor node degree and bandwidth requirement for construction of the cluster and selection of the cluster head. Further, when two clusters come closer to each other they merge and form a single cluster. In such case out of two CHs one has to withdraw the role and other will take over. A new mechanism of merging two clusters is also proposed in the paper. We call this method as an Improved Cluster Maintenance Scheme and primarily focused on minimizing CH changing process in order to enhance the performance. The stated method makes cluster more stable, and minimizes packet loss. The proposed algorithm is simulated in *ns-2* and compared with Least Cluster head Change (LCC) and CBRP. Our algorithm shows better behavior in terms of

number of clusterhead changes or number of cluster member changes.

Keywords Clustering in MANET · Routing overhead · Clustering scheme

1 Introduction

Mobile ad hoc network or MANET is a self-organized multi hop wireless network. Due to the movement of nodes, they generate dynamic topology and this dynamic nature of the topology; it imposes lots of challenges and possesses several intrinsic characteristics. As the MANET exist without infrastructure. So, this type of network is particularly useful in military/search, rescue and other tactical situations where cellular infrastructure is not available or not reliable [1, 2]. However, this class of network has lots of challenges and issues compared to infrastructure based network. Each node in MANET must forward traffic and hence act as a router [2–5]. So, maintaining routing information and forwarding packets by nodes are a challenging task. Because the dynamic topology makes node goes out of coverage from each other and hence frequent rout failure occurs. In case of route failure packet will lost and hence the node need to re discover the path to the destination. These reformations of path incur additional overhead to routing protocol for MANET. Other challenges of MANET apart from dynamic topologies are limited achievable bandwidth, heterogeneous communication links, and limited battery power. Due to these features routing in MANET is a challenging task and drawing considerable attention among researchers.

There are ample of methods suggested so far for routing in MANET [1, 6–10]. In many of the proposals, and other

✉ Sunil Pathak
sunilpath@gmail.com

Dr. Sonal Jain
sonaljain@jklu.edu.in

¹ Computer Engineering Department, JK Lakshmiapat University, Jaipur, India

literatures related to MANET routing protocols are classified as reactive, proactive and hybrid [11]. In the reactive class the path is discovered and used accordingly as and when there is a need for packet transmission. Hence, the name on-demand routing protocol is also given to this class. Proactive protocol on the other hand maintains a routing table and hence called table-driven protocol because of the existence of routing table. The hybrid protocol is a combination of proactive and reactive protocol. A special category of routing popularly known as cluster based routing falls in this category. In such routing nodes are organized as a group called cluster and a single node in a cluster called cluster head (CH) routes packet on behalf of all the nodes in the cluster. As the clustering protocol exhibits the advantages of both the other classes and hence considered as a convenient way for developing efficient routing scheme in MANET. One of the characteristics of cluster based algorithm is its implementation in hierarchical manner [12–17] in which the network is organized into subsets of nodes. The clusters are nothing but a subset in the topology. A cluster may contain N nodes and one cluster head that acts as a leader of the group. Clustering algorithms are proved to be scalable and efficient bandwidth utilization which arranges mobile nodes in a hierarchical architecture.

Cluster based routing can control the routing overhead inside a cluster by forming a group among themselves and selecting a coordinator (the CH) for routing. Clustering makes possible a hierarchical routing in which paths are recorded between clusters instead of among groups. This increases the lifetime of the network as well as improves routing efficiency. The ordinary node within a cluster communicates via the CH. A special node called Gateways exists in the topology. The Gateway is a special node in a cluster but not the CH and still it forwards on behalf of the cluster. Ordinary nodes send the packets to their cluster-head that either distributes the packets inside the cluster, or (if the destination is outside the cluster) forwards them to a gateway node to be delivered to the other clusters. By replacing the nodes with clusters, existing routing protocols can be directly applied to the network. Only gateways and clusterheads participate in the propagation of route control or route formation messages. In densely populated networks this mechanism of clustering significantly reduces the routing overhead, thus solving scalability problems for routing algorithms in large ad hoc networks.

In [17] an energy-efficient routing and scheduling scheme is proposed for vehicle delay tolerant networks. In this paper Nash Q-learning approach is used to find the optimized routing process. DRSS algorithm provides energy efficient and delay bounded data delivery by using forwarding and replication method. In [18] improve end-to-end throughput for multi-hop wireless networks in

consideration of spatial reusability of the wireless media to find the “best” path from the source node to the destination node so that network overhead is reduced. In [19] minimize energy consumption for wireless node with a small penalty in the delivery ratio. Reducing energy consumption will make network more stable and durable, which decreases network overhead for wireless network. It [20] controls the topology to satisfy the interference constraints and increases the transmit range to achieve delay requirement. These algorithms reduce the delay and improve the throughput for MANET [21]. Established routes to all source nodes with minimal cost under the constraints of packet delay and load balancing for wireless network. Each route link also assigned weights based on their battery of individual nodes.

In [22] coverage quality is a basic problem in wireless sensor network. In this paper a novel biology inspired optimized algorithm is designed for Minimal Exposure Problem [MEP]. MEP is formulated and then converted into steiner problem by dishonor monitoring area to large scale weighted grid. Based on Path finding and network formation capability of physarum new algorithm [23] was developed. In this paper physarum optimization method is applied to minimal exposure problem which is a fundamental problem corresponding to the worst case coverage in wireless sensor networks. It achieved good performance with low complexity. In [24] a CodePipe, novel reliable multicast protocol for lossy wireless networks is introduced by using LP Based opportunistic routing structure, opportunistic feeding, fast batch moving and inter batch coding, which offer improvement in throughput, energy efficiency and fairness for wireless network. The topology of MANET may change unpredictable and frequently because of its dynamic nature and infrastructure less network. In [25] multicast routing algorithm is proposed using genetic algorithms. By performing suitable values, crossover, mutation and population size, algorithm performs better compared to other methods for MANET. Existing routing strategies are discussed in [26] and useful design guidelines are identified for suitable routing protocol using for a network. Existing routing is broken into small number of common routing modules and then identified when and how which routing module can be used. MANET and Sensor network are known as Delay tolerant networks (DTNs) which form host and router mobility, and also as a result of disconnections due to power management or interference. In [27] various issues related to MANET such as networking, wireless, mobile communications, and technology analysis, an energy-aware routing protocol for DTNs, and a routing-compatible credit-based incentive scheme for DTNs, mobile peer-to-peer systems over DTNs, delay tolerant monitoring of mobility-assisted are discussed.

Although, the cluster based routing protocols possess lots of advantages, there are certain issues exist which need considerable attention. For example, the mobile nature of nodes in MANET leads to disappearance of the CH and may in turn get the cluster dismissed. In such scenario the cluster need to be reformed in order to continue routing. Frequent change in cluster formation either due to change in location of the cluster head or for some other reason may involve high cost of cluster maintenance. As the CH manage and store routing information, changes in the CH leads disruption in the entire topology. Our goal in this paper is to suggest an algorithm to create a cluster based algorithm which minimizes cluster changing probability. Few parameters are considered at the time of creation of cluster and selection of cluster head. We also propose an efficient cluster maintenance mechanism to minimize the cost of cluster re-formation. The protocol is examined in terms of various performance parameters in *ns-2* environment.

The paper is organized as follows. In Sect. 2 previous work related to cluster maintenance is described. Section 3 covers our proposed algorithm. Section 4 includes simulation description with comparative results. The paper is concluded in Sect. 5.

2 Related works

There are many cluster based routing algorithms found in literatures of MANET research. Few of such protocols related to our proposal are described in this section.

2.1 Lowest-ID (LID)

It was proposed by Baker and Ephremides in their work presented in [28]. It is also known as Linked cluster Algorithm (LCA). In this algorithm, each node is assigned to unique ID. All nodes broadcast their ID to their 1-hop neighbours at regular interval. After that each node compares its own ID value with received IDs from their direct neighbours. A node declares itself as CH if it has the lowest ID value among its neighbour IDs. This approach alarms only with the lowest value of node IDs which are arbitrarily assigned values without considering any other qualifications of a node for selection as a clusterhead. Since the value of node IDs do not change with respect to time, those with smaller IDs have more chances to become clusterheads than nodes with larger IDs. Thus, shortcomings of lowest ID scheme is that certain nodes are prone to battery drainage due to serving as clusterheads for longer time period, the packet delivery delay may become excessive, battery drainage, the selection of cluster heads has to be frequently updated because of high mobility of nodes etc.

2.2 Highest-degree (HD)

It was proposed by Gerla and Parekh [29]. It is also called as connectivity-based clustering method, which operates based on the value of node degree of a particular node. In this algorithm every node broadcast their ID in their domain. Based on the number of received IDs, each node calculates its node degree value and the one who has the maximum degree selected as cluster-head (CH). If two nodes or more have the same degree value then node with the lowest-ID is selected as the cluster head. In this method of clustering, the numbers of cluster heads are relatively low in comparison with lowest ID approach. In addition, it also shrinks the value of packet delivery delay. However, the number of re-affiliations of CHs increases when the topology changes. All these shortcomings happen because this scheme does not have any restriction on the upper limit of the number of nodes inside a cluster.

2.3 Distributed clustering algorithm (DCA)

It is an enhancement of lowest-ID algorithm. A real number above zero (a generic weight) is given to each node inside the network [3]. Then each node broadcasts its weight value to its all neighbour. A node is selected for cluster head if its weight value is higher in comparison with all its neighbor weight values; otherwise, it joins one of the neighbor clusters. If multiple neighbor nodes have the same weight value, the node with lowest-ID will be appointed as the cluster head. This algorithm inherits the drawbacks of the Lowest-ID algorithm. In addition, the number of cluster heads generated at the end will be high in comparison with previous approaches, which degrades network performance.

2.4 Incremental maintenance scheme (IMS)

The IMS as proposed in [3] uses lowest ID clustering algorithm, in which the node with lowest ID in its neighbour elected as clusterhead. IMS addresses the disadvantages of LCC, CBRP and increases delay timer to reduced CH change. In IMS when two CH are within same radio range, then they will wait till delay period not reaches up to maximum limit. As delay period reaches up to maximum limit, it will check whether both cluster head are still in same radio range the cluster head with lost ID will act as cluster head and other will hand over its charge and act as member within the cluster.

2.4.1 Least cluster head change (LCC)

There is one protocol Lowest ID node Count (LIC) proposed in [28]. In LIC, a mobile node with lowest ID among

its neighbour of a cluster is elected as CH. When a CH find another mobile node with ID lower than its own ID, the CH is forced to handover its charge to the new node. Another variation, HCC proposes similar method with a CH having highest node degree for maintaining the cluster. In HCC, extra calculation is required to check the local highest node degree of a cluster head periodically. When a cluster head finds another mobile node with highest node degree, it is forced to hand over his charge to other mobile node having highest node degree in a cluster. Both LIC and HCC suffer from frequent re-clustering because of mobility of a node in the network. To avoid frequent changes in CH, a new clustering scheme was proposed that is known as LCC.

Least Cluster head Change (LCC) is proposed as an improvement of LIC [28] and High Cluster head Change (HCC) [2]. In LCC clustering algorithms are divided into two phases, in first phase of algorithm cluster formation is done and the second phase cover cluster maintenance. First phase follows LIC method to select a CH for a cluster. Second phase, that is the cluster maintenance phase is event driven and executed only for the below two conditions.

- If two cluster head are within the same radio range, then the cluster head with lowest ID will work as cluster head and the mobile node with highest ID will release its role as cluster head to cluster member. Other simple members of the cluster are not allowed to participate in cluster maintenance scheme to change the cluster head even if they have lowest ID.
- If there is no cluster head in a cluster, the new cluster head will be formed according to LIC method.

Thus, LCC improves cluster stability by reducing the changes in cluster head for a cluster in first phase by providing some specific attribute to a cluster head. But in second phase if a cluster head movement occur, it requires complete re-calculation for a cluster structure. This involves large amount of communication overhead, when a cluster head is moving from the current position to a range of another cluster head for short duration.

2.5 CBRP

Another cluster changing method is reported in [29] defined as Cluster Based Routing Protocol (CBRP). In this method, each cluster observes the following rules for changing the cluster head:

- A member of a cluster never challenges to the cluster head for being cluster head.
- If two cluster heads came into the same radio range over an extended period of time then only one of them will act as cluster head second will lose its role as cluster head.

When a CH listens a HELLO message from another CH, having a bidirectional link, it sets timer to a predefined value `CONTENTION_PERIOD` given in seconds. When the timer expires, it checks if two CHs are still in same radio range, then it compare its own ID with other cluster head's ID. The CH with smaller ID will continue to act as cluster head and other will force to act as cluster member. These rules provide some short of stable clustering by delaying the CH change for certain time specified by the timer.

With this overview of existing cluster based routing protocols so far, in next section we are presenting our proposed weigh based cluster formation algorithm for routing in MANET to address few unaddressed issues observed in other proposals.

3 Proposed weight based clustering

We propose a Weight Based Clustering (WBC) protocol for routing in MANET. The objective of the proposed protocol is to form a cluster that sustain for longer time and CH can serve for longer duration. Each node learns about its one hop neighbours and selects a CH among them based on some predefined criteria. During the movement of node it may leave a cluster and join a new cluster. Similarly, a CH may due to its mobility may leave the cluster and in such case a new CH must be selected. The following assumptions are taken into consideration before clustering procedure take place.

- The network topology is relatively stable during the execution of the clustering procedure.
- One node can join exactly one cluster.
- Data routed only via CH and no gateway node is exists.

The algorithm comprises of three different phases pre clustering, cluster formation and cluster maintenance. All these phases are discussed below.

3.1 Pre clustering phase

Initially all nodes compute two most vital information for itself. This information includes node degree and bandwidth requirement of the node. After computing these two values the said node constructs a *node_info()* packet and broadcasts to all its neighbour. The procedures of computing these two parameters are discussed next.

3.1.1 Node degree

Degree of a node (D_i) is defined as the number of neighbouring nodes. To compute node degree the concerned device broadcast a HELLO packet. The nearby nodes that

can hear any HELLO packet record the source node's address as its neighbour node. Then the code can compute total number of neighbours by counting the number of HELLO packets that it hears. A table of nodes is maintained by each node to use it in future communication. Node Degree is very important parameter for cluster head selection as our prime concern is to maximize nodes inside single cluster and minimize the number of clusters in order to reduce inter cluster communication cost. So the node having highest degree will have more probability to play the role of cluster head.

3.1.2 Bandwidth requirement of node

The bandwidth requirement of node is another parameter in selection of efficient cluster head. Each node can estimate its bandwidth requirement based on its expected data transmission requirement. The node then communicate its bandwidth requirement to all its neighbor who are expectedly forming cluster. These information later used for cluster head selection. While selecting CH, if a node has high demand for bandwidth, then it implies that it has its own task to do and hence will get less time to pass other data. We formulate a method to select a CH with the criteria that too high bandwidth requirement has less probability of that node to be selected as cluster head as the role of cluster head itself demand much bandwidth.

After computing node degree and bandwidth requirement, a node computes its weight as

$$w_{(i)} = w_1 D_i + w_2 B_i \quad (1)$$

where, $w_{(i)}$ is the computed weight of node i , w_1 and w_2 are two weight factors associated with node degree and bandwidth requirements respectively. Values are assumed as $0 < w_1, w_2 < 1$. If the degree of the node is high then w_1 takes a value close to 1 otherwise close to 0. Similarly if bandwidth requirement of the code is high then w_2 assigned a value close to 0 otherwise closeto 1. After, computing weight of itself the node constructs a *node_info()* packet and communicate to all its neighbor. The *node_info()* packet contains identification of the node along with the weight calculated using the Eq. (1).

3.2 Cluster formation phase

During the neighbour discovery process each node creates a table of neighbours. When a node receives a *node_info* packet then it marks all the nodes in the table and inserts the weight of the node. After completing the reception of *node_info* message the node examines the weight entry in the table and the node whose weight is highest declares as CH. If two nodes have the same entry in the weight field then the node with highest ID will be the CH. The node

who notices himself as a CH now needs to send the *CH_advertisement* message. This message contains the weight value of the CH and its ID. All other nodes that hears *CH_Advertisement* responds with *Cluster_Join* message. The cluster head then have a complete information about its members and members have the CH information. An algorithm of cluster formation is given in Fig. 1.

This completes the cluster formation process. Now onwards the CH periodically broadcasts its identification and any new node coming closer to the cluster will receive the same. The new node itself computes the node degree and bandwidth requirement and computes the weight accordingly. If the weight of the newly coming mode is large than the CH whose advertise is received then the new node declares as a cluster head communicates it to the old CH. The new CH then advertise its ID to all its neighbours. All nodes hearing the *CH_advertisement* then can figure out that a new CH is coming up and responds to the new advertisement through *Cluster_join* message.

3.3 Cluster maintenance phase

The aim of the proposed Improved Cluster Maintenance Scheme (ICMS) [16] algorithm is to minimize the CH selection cost by delaying the process up to acceptable time period. It will not immediately change the CH as and when two CH comes closer, it will also not delay beyond a time limit which is very long at all. Further, the algorithm computes a priority of cluster head in order to select a new one. The working of the cluster maintenance algorithm may be described as follows (Fig. 2):

- If two cluster heads are within the same radio range, cluster head change will be delayed up to *delay_timer* expire. After *delay_timer* expire if both cluster head are still within the same radio range then calculate priority of each cluster head. If newly arrived cluster head has

```

CLUSTER_FORMATION()
{
    Received node_info from all its neighbour
    max_weight = max(weight[node(id)]) and CHid=
node(id)
    IF CHid > 1 Then
        CHid = Largest_Value_ID
        SEND "Construct CH_advertisement
message and broadcast"
    END IF
}

```

Fig. 1 Algorithm for *Cluster formation phase*


```

CLUSTER_CH_MERGING()
{
    Old_CH Received Hello from New_CH
    Delay_timer=delay_period
    Max_limt=Tran_Range/2*speed
    hello_interval=transmission_time
    Delayed clusterhead change for delay_timer.
    IF Old_ch is in the same radio range New_ch
        WHILE delay_time<=Max_limt
            Old_Priority_factor=Max_degree_ch+Battery_life
            New_Priority_factor Max_degree_CH+Battery_life
            IF New_Priority_factor>Old_Priority_factor
                New_CH=1
                Old_CH=0
                Exit(0)
            ELSE
                delay_time=delay_time+hello_interval
            END IF
        ENDWHILE
    IF Oldch_ID<=Newch_ID
        New_ch=1
        Old_ch=0
        Exit(0)
    END IF
}

```

Fig. 2 Algorithm for *Cluster maintenance phase*

high priority over old cluster head then newly arrived cluster head will act as cluster head and other will hand over its charge and act as member of the cluster.

- If newly arrived cluster head has low priority over old cluster head then ICMS follow IMS method to delay cluster head change
- Priority factor of a cluster head is calculated by addition of maximum degree of cluster head and battery life.
- If there is no cluster head in a cluster, the new cluster head will be formed according to LIC method.
- Maximum limit is calculated by dividing two times transmission range by speed.

The cluster formation in proposed ICMS depends on the priority of both CH which are within the same radio range and method of IMS. The algorithm is given below:

The proposed weight based clustering mechanism of this paper works in three phases as discussed in this section. In order to verify the correctness of the proposed algorithm we have performed a simulation of the method in *ns-2*. In the next section simulation environment is described with observed results.

4 Simulation setup and performance metric

The proposed weight bases clustering algorithm is implemented in *ns-2.31* [12] and performance is compared with LCC and CBRP. In simulation, node movement generator is used to generate the different node movement scenarios following random way point model. The movement generator takes the number of nodes, pause time maximum speed, field configuration and simulation time as input parameters. The propagation model used in the simulation is two ray ground. Complete simulation is divided into two stages. In first stage simulations is carried out by changing the mobility (pause time) and in second stage by changing node speed. Few of the important parameter and their typical data are shown in Table 1.

To measure performance of proposed weight based clustering we have considered mainly on total number of clusters formed. The purpose of the observation is to see how many numbers of clusters are formed by this algorithm. Because, less number of clusters are always desirable as it reduces network overhead for cluster maintenance. Apart from number of clusters formed, we also observed the cluster formation overhead for the protocol. Clustering overhead is the number of clustering management messages sent by each node during cluster formation and maintenance operation. This is very important factor to identify the scalability of a protocol. Again, we have also suggested a method of cluster merging the proposal. In order to see the effectiveness of the proposed merging mechanism we have also recorded the number re-clustering over time. All the above mentioned parameters are compared with LCC, CBRP and discussion is given in next section. IMS we have considered the number of CH change, number of cluster member change and cluster overhead by changing pause time and speed of mobile nodes. Results are presented in the next section.

4.1 Simulation results and discussion

This section has been focused on proving the performance of Proposed WCA over existing WCA in terms various performance metrics as mentioned in previous section.

Table 1 Simulation parameters

Parameters	Values
Number of mobile nodes (N)	100
Simulation area	500 × 500 m
Simulation time	50–300 (s)
Pause time for mobile nodes	10–40 (s)
Maximum speed for mobile nodes	2–10 mps
Transmission range for mobile nodes	25–250 m

4.2 Cluster formation

Figure 3 shows the performance of the proposed WCA for networks which are different in transmission ranges. It shows that for small ranges, most of nodes remain out of each other's transmission range, thus the number of clusters is relatively high and the network may become disconnected because there are no other choices. When transmission range increases, more nodes can hear each other. The average number of clusters formed decreases and the clusters become larger in size. Result shows the number of clusters formed is less in proposed WBC compared to LCC and CBRP. These two protocols have almost similar number of cluster formation. However, significant performance is noticed in WBC when we have considered the total number of clusters formed including reformation after end of the simulation when MN comes to a stable state (no movement). In this case WBC shows almost constant total clusters, whereas LCC shows maximum cluster reformation and CBRP shows moderate behaviour. Since, WBC uses weighted factor so it selects the most stable node as cluster head and so need less cluster reformation. Also, during merging of clusters WBC weight for some time after two CH coming close to each other. If they stay for longer time then only clusters are merged. In other two protocols two clusters are merged immediately as they come close to each other and after some times they are split as they go out of range.

In Fig. 4 cluster formation is plotted against simulation time. In this case we are recording only the total number of clusters formed at the end of the simulation. No separate recording of cluster formation at initial stage is observed. For this result transmission range is taken as 50 m and

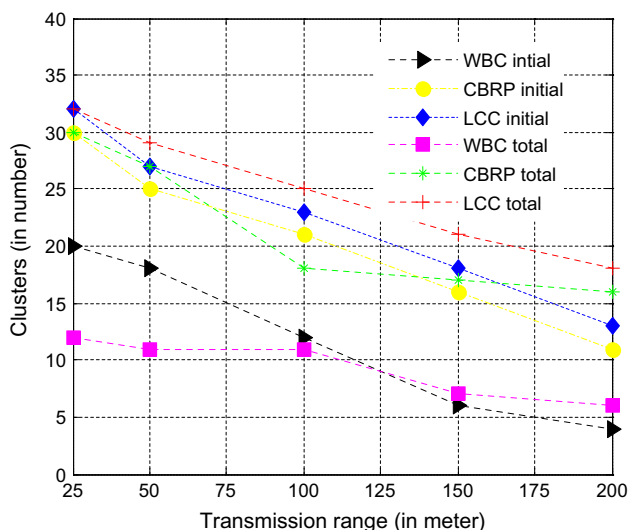


Fig. 3 Cluster formation with transmission range

speed of MN is taken as 5 mps of most of the MNs. Few of the MNs however has lower and few of them have higher speed than 5 mps. Result depicts that proposed WCA clustering algorithm's average number of clusters formation is lowest within the range of 5–25. On the other hand LCC and CBRP show significantly large number of clusters. Due to the selection of stable node as CH, the WBC shows less number of total cluster reformations and hence total clusters are low. In other two algorithms no such strategies for selection of stable cluster head is adopted hence more cluster reformation takes place.

The graphs in Fig. 5 another scenario of total cluster formation against existing mobile nodes. For this experiment transmission range is taken as 50 m and speed varies from 3–6 mps. In this observation also the proposed WCA shows considerably good behaviour. The CH covers comparatively more number of nodes as its member. Also WCA avoids unnecessary merging of clusters when nodes and clusters are moving nearby. Cluster changes into our proposed algorithmic less than the LCC and CBRP as and when MN increases. The efficient cluster maintenance scheme of our protocol is another reason of reason of invoking unnecessary re-clustering procedure.

Figure 6 shows the number of cluster head changes for varying pause time. For all the protocols for lower pause time cluster head change tendency increases. However, WBC cluster head change is lowest among all three protocols. High pause time indicates low mobile scenario and hence less cluster reformation occurs in all cases.

In Fig. 7 performance of the proposed WBC is compared with LCC and CBRP for varying MN speed. The analysis of result shows that the proposed WBC is better clustering scheme over other schemes. From the figure it is clear that as speed of mobile nodes increases the cluster head change also increases. So, it may be concluded that

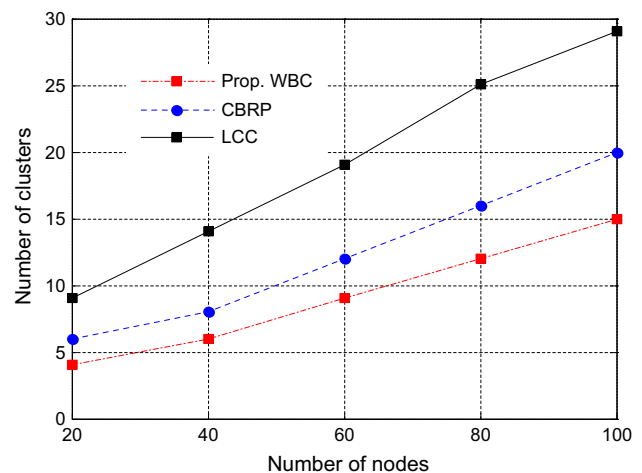


Fig. 4 Cluster Formation over simulation time

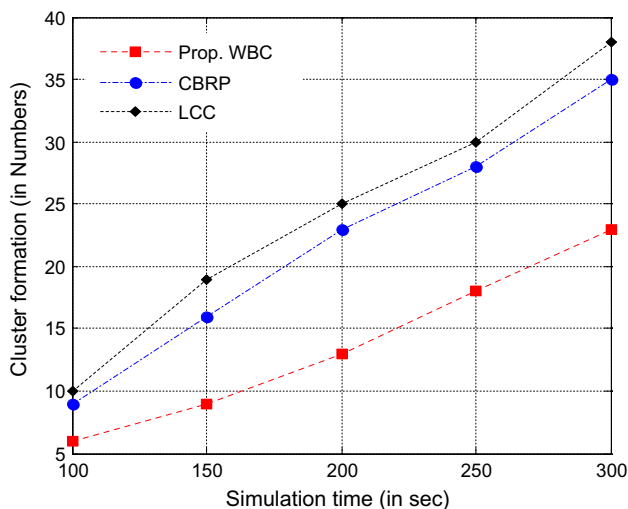


Fig. 5 No of nodes versus Number of Clusters formed

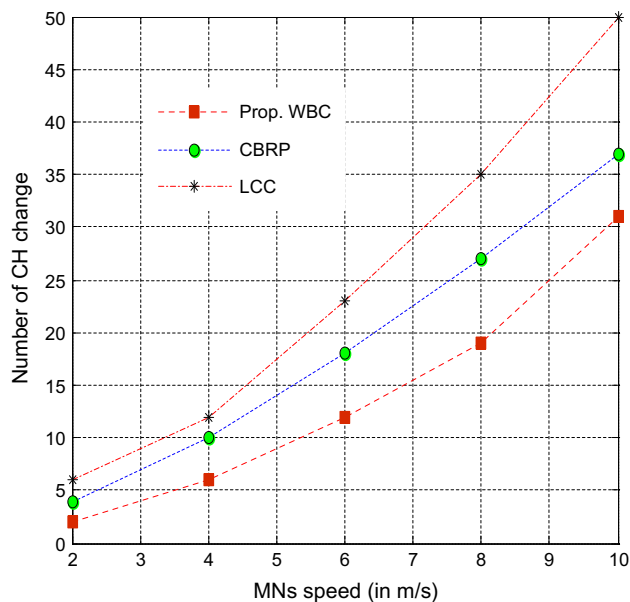


Fig. 7 Clustering member change varying speed

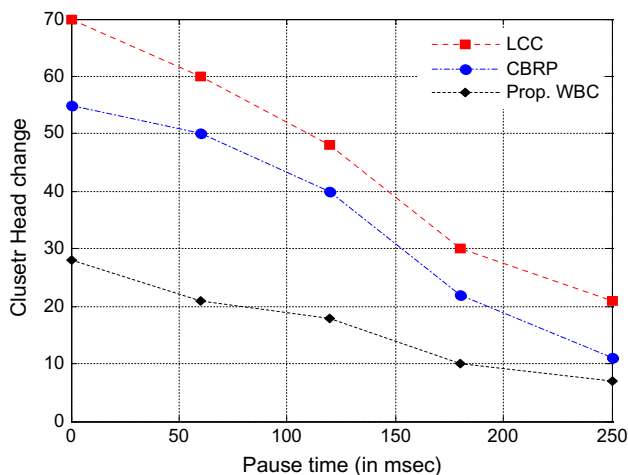


Fig. 6 Number of Cluster Head changes over pause time

the proposed cluster scheme performs better than other two approaches as speed of mobile node changes.

Clustering overhead:

Apart from total cluster formation, clustering overhead is another important performance parameter to study for any clustering algorithm. After observation of cluster formation in previous subsection, in this subsection we are showing our observation regarding cluster formation and maintenance overhead for all the three protocols.

Figure 8 shows clustering overhead with respect to simulation time. We have considered the same simulation environment as it is done for Fig. 4 (clusters against simulation time) and computed the cluster formation overhead. This calculation also includes the cluster reformation overhead. As simulation time increases, overhead also increases as more cluster formation and reformation takes place. However, WBC shows lowest overhead compared to

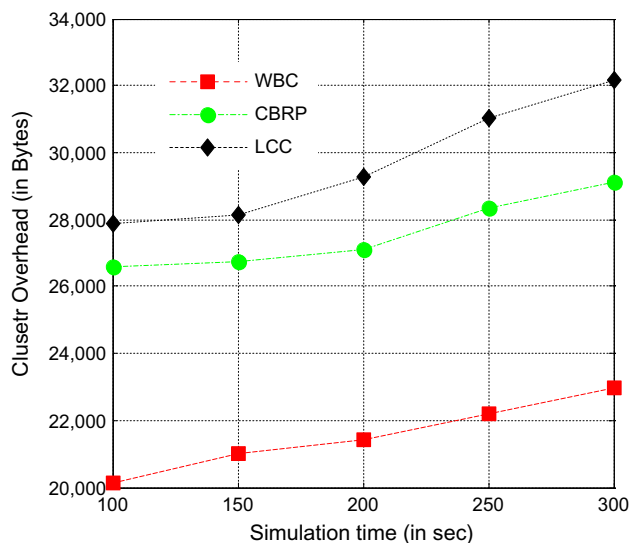


Fig. 8 Clustering overhead over simulation time

LCC and CBRP. This is justified as the WBC incurs less cluster re-formation and more stable clusters.

In Fig. 9 a measure of clustering overhead is shown against pause time.

Higher pause time shows less overhead incurred as that scenario nodes are comparatively stable and clusters are survive for longer time. WBC in this case shows much low overhead compared to other two protocols.

Figure 10 demonstrates the overhead against MNs speed. It shows that as speed increases total overhead increases. It is because of the pattern of cluster head change in all the protocols. Higher speed leads to more

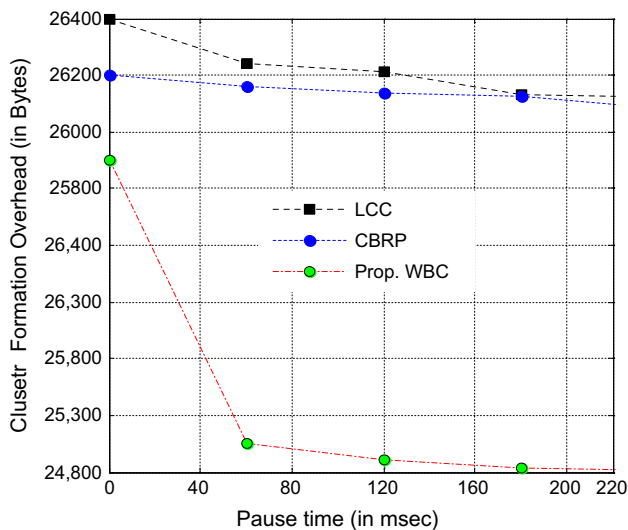


Fig. 9 Clustering overhead over pause time

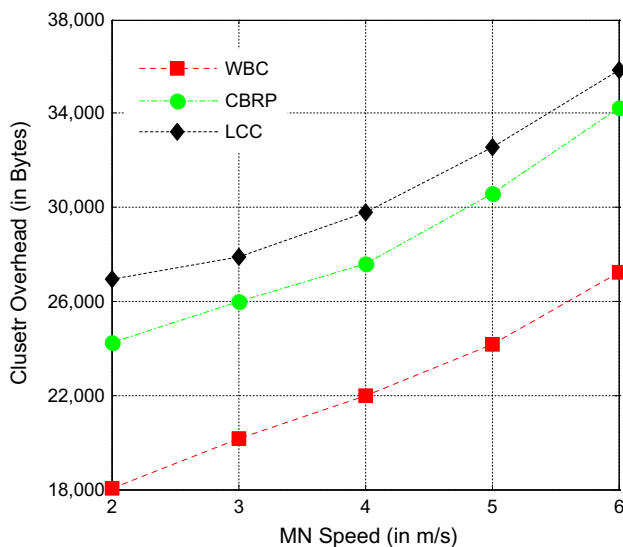


Fig. 10 Clustering overhead for varying MN speed

unstable clusters and incur more clustering overhead. LCC and CBRP shows more overhead compared to WBC.

5 Conclusion

In this paper a novel approach for cluster formation is proposed for routing in MANET. The proposed algorithm is a weight based approach that takes two parameters into account. The first parameter is the node degree that gives the idea of population of nodes in is around. The second parameter is the bandwidth requirement that gives the busyness of the node. Based on these two parameters a weight function is derived and the weight of each node is

computed. During formation of the cluster and selection of the cluster head, the protocol considers the weight of every candidate for cluster head. The assign weight ensures stability of the cluster head and hence long surviving cluster. Another enhancement of the proposal is a cluster merging algorithm. This merging algorithm proposed here compels two cluster heads to weight for a certain time interval before merging them to single cluster. This approach reduces the chances of unnecessary cluster merging in case of passing by clusters. The algorithm of WBC is simulated in ns-2 and compared with LCC and CBRP protocol. Results shows that proposed WBC performs better that the other two protocols.

References

- Royer, E. M., & Toh, C. K. (1999) A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications Magazine*, 1999, pp. 46–55.
- Gerla, M., & Tsai, J. T. (1995). Multiuser, mobile, multimedia radio network. *Wireless Networks*, 1, 255–265.
- Yadav, N. S., Deosarkar, B. P., & Yadav, R. P. (2009). A low control overhead cluster maintenance scheme for mobile ad hoc networks. *International Journal of Recent Trends in Engineering*, 1(1), 1–9.
- Chiang, C. -C. et al. (1997). Routing in clustered multihop, mobile wireless networks with fading channel. In *Proceedings of IEEE SICON'97*, 1997, pp. 197–211.
- Ephremides, A., Wieselthier, J. E., & Baker, D. J. (1987). A design concept for reliable mobile radio networks with frequency hopping signaling. In *Proceedings of IEEE*, Vol. 75, 1987, pp. 56–73.
- Abolhasan, M., Wysocki, T., & Dutkiewicz, A. (2004). A review of routing protocols for mobile ad hoc networks. *Journal of Ad Hoc Networks*, 2, 1–22.
- Gupta, P., & Kumar, P. R. (2000). The capacity of wireless networks. *IEEE Transactions Information Theory*, 46(2), 388–404.
- Hong, X. Y., Xu, K. X., & Gerla, M. (2002) Scalable routing protocols for mobile Ad Hoc networks. *IEEE Network*, 16(4), 11–21.
- Hong, X. Y., Xu, K. X., & Gerla, M. (2002) An Ad Hoc network with mobile backbones. In *Proceeding on IEEE ICC 2002*, Vol. 5, April–May 2002, pp. 38–43.
- Yu, J. Y., & Chong, P. H. J. (2005). A survey of clustering schemes for mobile ad ho networks. *IEEE Communication Surveys & Tutorials*, 7(1), 32–48.
- Abusalah, L., Khokhar, A., & Guizani, M. (2008). A survey of secure mobile ad hoc routing protocols. *IEEE Communications Surveys & Tutorials*, 10(4), 78–93.
- Fall, K., & Vardhan, K. *The network simulator (ns-2)*. <http://www.isi.edu/nsnam/ns>.
- Agarwal, R., & Motwani, M. (2009). Survey of clustering algorithms for MANET. *International Journal on Computer Science and Engineering*, 1(2), 98–104.
- Kioumourtzis, G. (2005). *Simulation and evaluation of routing protocols for mobile ad hoc networks*. Thesis, Master of Science in Systems Engineering and Master of Science in Computer Science, Naval Postgraduate School, Monterey, California, 2005.
- Raju, R., & Mungara, J. (2010). Performance evaluation of ZRP over AODV and DSR in mobile adhoc networks using qualnet. *European Journal of Scientific Research*, 45(4), 658–666.

16. Pathak, S., Dutta, N., & Jain, S. (2014). An improved cluster maintenance scheme for mobile AdHoc networks. In *IEEE 2014 international conference on advances in computing, communications and informatics*, pp. 2117–2121.
17. Zeng, Y., et al. (2013). Directional routing and scheduling for green vehicular delay tolerant networks. *Wireless Networks*, 19(2), 161–173.
18. Meng, T., et al. (2015). Spatial reusability-aware routing in multi-hop. *IEEE TMC Wireless Networks*. doi:10.1109/TC.2015.2417543.
19. Dvir, A., et al. (2011). Back pressure-based routing protocol for DTNs.ACM. SIGCOMM. *Computer Communication Review*, 41(4), 405–406.
20. Zhang, Xin Ming, et al. (2015). Interference-based topology control algorithm for delay-constrained mobile Ad hoc networks. *IEEE Transactions on Mobile Computing*, 14(4), 742–754.
21. Yao, Y. et al. (2013). EDAL. An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for wireless sensor. Networks, Mobile Ad-Hoc and Sensor Systems (MASS). *IEEE 10th International Conference 2013*, Oct 14-16, pp.182–190.
22. Song, Y., et al. (2014). A biology-based algorithm to minimal exposure problem of wireless sensor networks. *IEEE Transactions on Network and Service Management*, 11(3), 417–430.
23. Liu, L., et al. (2015). Physarum optimization: A biology-inspired algorithm for the steiner tree problem in networks. *IEEE Transactions on Computers*, 64(3), 819–832.
24. Li, P., et al. (2014). Reliable multicast with pipelined network coding using opportunistic feeding and routing. *IEEE Transactions on Parallel and Distributed Systems*, 25(12), 3264–3273.
25. Yen, Y.-S., et al. (2011). Flooding-limited and multi-constrained QoS multicast routing based on the genetic algorithm for MANETs. *Mathematical and Computer Modelling*, 53(11–12), 2238–2250.
26. Spyropoulos, T., et al. (2010). Routing for disruption tolerant networks: Taxonomy and design. *Wireless Networks*, 16(8), 2349–2370.
27. Vasilakos, A., et al. (2012). *Delay tolerant networks: Protocols and applications*. Boca Raton, FL: CRC Press.
28. Chiang, C. -C., Wu, H. -K., Liu, W., & Gerla, M. (1997). Routing in clustered multihop, mobile wireless networks with fading channel. In *Proceedings on IEEE SICON'97*, 1997, pp. 1–5.
29. Jiang, M., Li, J., & Tay, Y. C. (1999). *Cluster based routing protocol*. August 1999 IETF Draft. <http://www.ietf.org/internet-drafts/draft-ietf-manetcbrspec-01.txt>.



Sunil Pathak received the M.Tech. degrees in Information Technology from Tezpur Central University, Tezpur, Assam, India in 2006, He is pursuing Ph.D. degree in computer engineering from JK Lakshmipat University, Jaipur, India. His research interests include Routing Protocol, Security, and Power Constraints in Mobile Ad-Hoc Network. He is member of CSI, IACSIT and IEN.



Dr. Sonal Jain has over 12 years of academic experience. She is working with JK Lakshmipat University, Jaipur as Associate Professor-Computer Science Engineering. Her area of interest for Research, Training and Consultancy includes Technology for education, Information Retrieval and Natural Language Processing.