Power-Aware Cluster-based Vehicular Ad hoc Network Model for Highway Communication

P. Punithavathi
Department of CSE,
BharathUniversity,
Chennai, India
p_punithavathi@rediffmail.com

AR. Arunachalam
Assistant Professor,
Department of CSE
Bharath University,
Chennai, India
ararunachalam78@gmail.com

Dr. T. Nalini
Professor,
Department of CSE
Bharath University,
Chennai, India
drnalinichidambaram@gmail.com

***Abstract*—Clustering in VANET is one of the control schemes used to make VANET global topology less dynamic. Many of the VANET clustering algorithms are derived from MANET. However, VANET nodes are characterized by their high mobility, and the existence of VANET nodes in the same geographic proximity does not mean that they exhibit the same mobility patterns. Moreover, the algorithms suitable for urban mobility models are not suitable for highway communication. There will be an increased routing, bandwidth allocation and channel access overheads, if they are used on highway. Therefore, VANET clustering schemes should take into consideration, multiple parameters to produce relatively stable clustering structure. A new multi-thread clustering algorithm for VANET has been proposed to achieve stable, long living clusters for reliable communication on Highways. Cluster formation is based on multiple parameters. Tests performed in the simulation environments show that the algorithm performs better than the popular classic approach (the Lowest ld algorithm, the highest degree algorithm, Density-based Clustering algorithm)– the cluster stability is significantly increased. The chances for re-clustering are also minimized.**

***Keywords* – Battery Power, Clustering, Cluster-head, Highway communication, Link Quality**

I. INTRODUCTION

Vehicular Ad hoc network (VANET) is called as a component of Intelligent Transportation Systems (ITS). The main goal of VANET is to provide security and comfort to the passengers.

The applications of VANET are listed as below:

- Intelligent Transportation System
- Traffic coordination and assistance
- Traveller information support applications
- Comfort applications
- Air pollution emission measurement and reduction

The communication among vehicles is performed in VANET using wireless technologies such as dedicated short-range communications (DSRC) –a type of Wi-Fi, WLAN, satellite, or WiMAX.

There are three types of VANET. They are listed as below:

- Purely ad hoc – for vehicle-vehicle communication
- Purely cellular – for vehicle-road side unit/ cellular base station
- Hybrid – combination of above both

There are several urban mobility models [4] such as Simple Model (SM), Manhattan Model (MM) and Downtown Model (DM), etc. These scenarios are valid inside a city where vehicles move slowly, and more number of base stations are available. These algorithms increase routing, bandwidth allocation and channel access overheads, if used on highway. The installation of base station on highways is also a tough job. Many of the VANET clustering algorithms are derived from MANET [6]. However, VANET nodes are characterized by their high mobility, and the existence of VANET nodes in the same geographic proximity does not mean that they exhibit the same mobility patterns.

A new multithread clustering algorithm for VANET has been proposed to achieve stable, long living clusters for reliable and power-aware communication on Highways. Hence the algorithm is called Power-Aware Cluster-based Vehicular Ad hoc Network (CBVANET) Model .Cluster formation is based on complex multiple parameters such as connectivity level, link quality, node reputation, traffic conditions and battery power.

II. RELATED WORK

In [1], enhanced cluster based routing protocol for mobile nodes in wireless sensor network, has been proposed. Since the nodes in VANET are highly mobile and dynamic when compared with the wireless sensor network, the same algorithm cannot be adopted. Hence a new solution has been proposed which performs power-aware clustering in VANET.

In [2], density-based clustering algorithm has been proposed which takes multiple parameters into account to form cluster in VANET environment. Though it outcomes the issues in Lowest ID and Highest Degree algorithms, it has no solution for power-awareness. Hence, the new solution has been proposed which performs power-aware clustering in VANET.

In [3], a load balancing energy efficient clustering algorithm has been proposed for MANETs. Though VANET is a type of MANET, the algorithm cannot be directly implemented on VANET. Hence, a new solution has been proposed that has load balancing feature. This means a cluster-head relinquishes its role when its energy-level falls behind a predetermined alert level.

In [4], a new mobility pattern generator called Citymob has been proposed. Using Citymob, urban mobility scenarios, including the possibility to model car accidents, can be designed easily. Three different mobility models: SM, MM and DM, are implemented. But all these models are suitable for urban area only. Hence the new solution has been designed vehicle-to-vehicle communication on highways.

In [5], an algorithm has been proposed that supports vehicle-to-vehicle communication on highways. But only two parameters namely speed of the vehicle and position of the vehicle for cluster-head election. This leads to lot of confusions. Hence a new solution has been proposed that considers multiple parameters for cluster-head election.

The description of the parameters used in our solution is given in section III. The algorithm is described in section V.

III. DESCRIPTION

The proposed model is appropriate for vehicle to vehicle communication on highway where the topology-changes are highly dynamic. Multiple parameters are considered for clustering. These parameters are listed as below:

- connectivity level
- link quality (SNR)
- traffic conditions (relative node position and the prediction of this position in the future)
- node reputation
- battery power

The connectivity level, which is the number of connections, of each and every node is calculated. Then the stable links from all the current links are selected. The algorithm also uses knowledge about distance between nodes, their relative speed and link quality. Finally, the communication history is used to determine node's reputation before it finally becomes a cluster member. Moreover, the chances of over- exhausting the current cluster-heads are reduced by considering battery power. Hence, it is apt to say that the model is energy-efficient.

A. Connectivity Level Estimation

The connectivity level estimation [2] is responsible for discovering density of local neighborhood of the node. We measure density in a very simple way - by counting the number of active links. Every node periodically broadcasts HELLO messages with TTL set to 1. The period is a configurable part of our algorithm. After receiving a HELLO message node sends an acknowledge response RESP. Number of the received responses is then counted and if it is greater than the given adjustable threshold (we assume that it is 4 in our simulations) then node notices that it belongs to the dense part. In case when the threshold is not exceeded the node cannot become a cluster member.

B. Link Quality Estimation

In the next phase node selects neighbors with a reliable connection. This selection is made on some prediction about future, but it also takes into account the past knowledge. In HELLO messages every node sends information about its current position, speed and movement direction - these data are obtained using for example a GPS device. The receiver of such a message is able to evaluate the potential link quality in the near future. In this evaluation it also uses the signal-to-noise ratio (SNR) of the link. The Node has knowledge about its current position, position of the node connected by the link being evaluated and velocity vectors of itself and the other node. Given these data it can make prediction about their future relative position and the approximate moment when the nodes will be out of their ranges. On that basis link quality is estimated. Estimating potential link quality helps to avoid re-clustering in many situations - for example, when groups of vehicles move in the different directions. In such case, the moment of meeting is usually very short and changing cluster structure at this moment may lead to another re-clustering, immediately after groups move outside their transmission range. Fig. 1 from [2] shows an example of such situation. It is assumed that clusters are determined by nodes shape (circle or square) and that a cluster-head is distinguished by its darker color and thicker border. At the top of the figure, nodes are re-clustered when the vehicles meet. At the bottom, a similar situation is shown, but with re-clustering avoided. Thanks to this, the clusters stability is increased.

Another example of such a situation is presented on Fig. 2. We can observe a node overtaking group of nodes. Due to the movement prediction the estimated potential link quality will be poor, and re-clustering can also be avoided, which also leads to increased clusters stability.

One remark that should be done here is that in some situations the behavior of the algorithm can be different from the one presented above. For example in the traffic jam, nodes moving in different directions will usually stay in range for longer period of time - this will be predicted by the link quality estimation procedure. Such nodes can communicate and form clusters with each other and such cluster should also be stable.

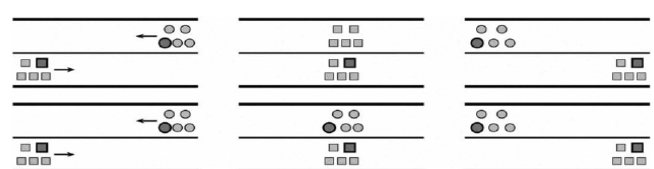

Fig. 1. Re-clustering and its avoidance thanks to link quality estimation (I).

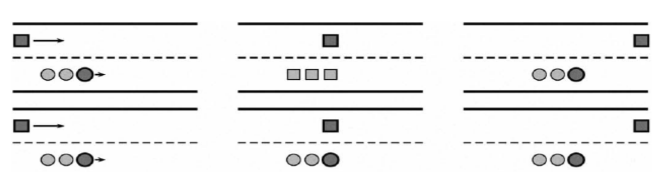

Fig. 2. Re-clustering and its avoidance thanks to link quality estimation (II).

C. Determination of node status

If number of links better than some given threshold is sufficient, then CLUSTERLIFETIME(CLT) counter is updated. The CLT counter says for how long node has communication with some group of nodes. Depending on this value node can be in one of the above states:

• CLT = 0 - node is a PASSERBY,
• CLT = {1, 2, 3} - node is a cluster CONTESTANT ,
• CLT > 3 - node is a cluster MEMBER.

The *visitor* node is a node which has just obtained connection with a group - it must be verified before becoming cluster member. The *candidate* node is a node which is being verified - it has to maintain connection with a group for some time (here it is a three time units period). After that time node becomes the full cluster *member.* The aim of introducing the CLT counter is to provide a method to check reliability of node before attaching it to cluster. Nodes have to prove that they are somehow stable and predictable. In fact in the implementation of the algorithm we also check that the groups of nodes which have assured CLT counter increment in the past are similar to the current neighborhood. We have introduced here three different states of nodes. They are used in the proposed service model. Services for the visitor nodes are limited to SMS-like single packets exchange. Nodes being cluster candidates are allowed to establish DTN sessions. And finally the full cluster members have access to all services including that based on continuous sessions. We have decided on such complex service model to enable fast start of communication without need for waiting on network topology control protocols - simple services are available all the time. Our model also has an advantage that it can adapt to the connectivity quality.

D. *HELLO packet*

The HELLO packet is comprised of the following fields as indicated in figure 3:

id	*time*	*position*	*velocityVector*	*batterypower*	TTL

Fig.3. HELLO Packet

IV. NOTATIONS

CLT – CLUSTERLIFETIME
TTL – Time To Live
Id – Identity of the Node
ClusterId – Identity of the cluster
Cluster-headId – Identity of the cluster-head

V. ALGORITHM IMPLEMENTATION

The algorithm has multiple threads that work asynchronously. The algorithm is comprised of the following threads:

- Signaling thread for transmission of HELLO packets
- Refreshing thread for Cluster maintenance
- Clustering thread for Cluster formation
- Electing thread for Cluster-head election & Gateway node selection

A. Signaling Thread

The signaling thread is responsible for transmission of HELLO packets. On the other hand, the nodes receiving the HELLO packet, update their NEIGHBORHOOD TABLE, accordingly. The outline of the algorithm looks as follows:

Source:
for each BEACONTHRESHOLD
send HELLO*(id, time, position, velocityVector, batterypower,* TTL)
Receiver:
Nodes receiving HELLO
update *(id, time, position, velocityVector, batterypower,* TTL) of source
Receiver also updates reputation (*linkQuality, lastSeenTime, numberOfTimesSeen*) of Source in NEIGHBORHOODTABLE

B. Refreshing Thread

The refreshing thread is used to maintain the cluster. It is used to check the availability of dead links for every predetermined time, say Refreshing Time Threshold, and removes the entry from the NEIGHBORHOODTABLE.
The algorithm is designed as follows:

In each node for every Refreshing Time Threshold,
Check *lastSeenTime* of all entries
Remove any entry if *lastSeenTime>predeterminedtimelimit*
Update NEIGHBORHOODTABLE

C. *Clustering Thread*

The clustering thread is responsible for all the activities connected with partitioning nodes into clusters. This thread updates CLUSTERLIFETIME and CLUSTERMEMBERSHIPSTATUS variables, it assigns the proper cluster id (which is id of the most stable node in the group) and it is also responsible for deciding which nodes should be gateways. The neighbors with good links are determined using link quality estimation procedure. Computing group similarity means that the node checks whether the groups which previously allowed to increase its CLT counter. It is just the mean value of rates of current neighboring nodes compared to neighboring groups from the latest history. We want to avoid situation when each increment of CLT counter is thanks to totally different groups of nodes.

Check NEIGHBORHOODTABLE periodically
For each node if

Neighbors with linkquality> LINKQUALITYTHRESHOLD then
CLUSTERLIFETIME++
add *goodNeighbors* to *groupHistory*
else
CLUSTERLIFETIME==0
end if
if *CLUSTERLIFETIME* == 0 then
CLUSTERMEMBERSHIPSTATUS== PASSERBY
else if *CLT* < CLTTHRESHOLD then
CLUSTERMEMBERSHIPSTATUS ==MEMBER
else
CLUSTERMEMBERSHIPSTATUS ==CONTESTANT
end if
send REFRESH *(id, weight, clustermembershipstatus, clusterId)*
wait REFRESHPERIODTIME
end while

The weight of the node is computed each time it is needed. We decided that the most stable node in the group becomes a cluster-head, but it can be easily changed to take into account different factors. We compute node reputation by tracing history of connections with nodes from the cluster – node with the most stable communication with other group members gains the highest weight and becomes a cluster-head.

D. *Electing Thread*
The electing thread is responsible for selecting cluster-head for each cluster. The cluster-head of each cluster is determined based on the weight given to each node and remaining battery power. The weight of the node is computed based on the *numberOfTimesSeen* entry in the NEIGHBORHOODTABLE. The weight of a node increases as the *numberOfTimesSeen* entry increases. The algorithm for electing thread is designed as follows:

After receiving cluster information, each node updates its NEIGHBORHOODTABLE with following:
weight, clustermembershipstatus and clusterID
From NEIGHBORHOODTABLE entries,
Node with max(*weight, batterypower*) && with *clustermembership status ==* Contestant, is elected as cluster-head with new cluster-headID
The cluster-head leads the cluster till battery power >> alertlevel
If a node receives HELLO from two cluster-headID
Then the node is GATEWAYNODE

VI. Simulation Result

We have simulated the four approaches using the network simulator NS2. Several experiments were performed to evaluate their performances using the network lifetime and the average dissipated energy as metrics while varying several parameters such as traffic, network size, link quality, etc. Results have shown that the proposed algorithm outperformed the other approaches in all the scenarios.
The selected simulation scenarios vary within several parameters:

- number of nodes: 100, 200, 300, 400 and 500,
- clustering algorithm: *Highest degree, Lowest Id* and *DBC,*
- HELLO messages per minute: 12, 20 and 60,
- radio transmission power (tx power): 6, 7 and 8,
- use of velocity vector to avoid clustering of nodes moving in different directions: *enabled* and *disabled,*
- link quality threshold (normalized to 1): 0.3, 0.5 and 0.7

The simulation result is indicated by the graph in figure 4.

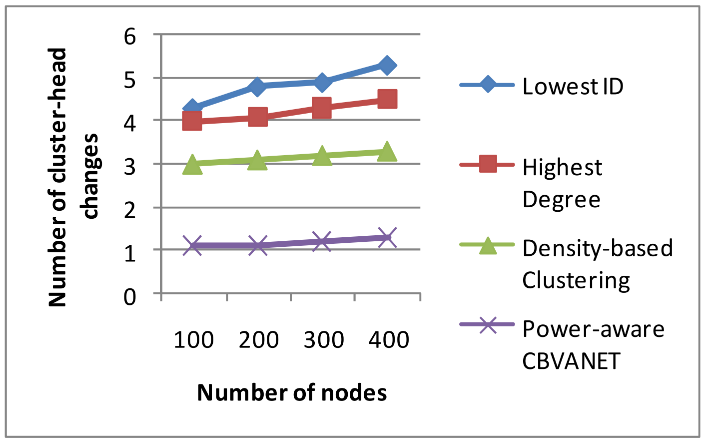

Fig.4. Average number of cluster-head changes by algorithm and number of messages per minute tx power= 7, link quality= 0.5)

The proposed algorithm has stable cluster formation. The number of changes of cluster-head is also minimized in power-aware CBVANET, when compared to other algorithms such as lowest ID, highest ID and Density-based clustering. As we have introduced several node's states VISITOR, CANDIDATE and MEMBER, nodes which are not cluster members are not taken into account, so in case of our algorithm not all nodes will be clustered at the moment. It can be seen, that with the increasing number of nodes, the percentage of clustered nodes also increases.

VII. Conclusion

A new solution has been proposed that can be effectively utilized for clustering of VANET nodes on highway. At the same time, the algorithm is power-aware, which is an added advantage. In the nearest future we plan to combine presented clustering algorithm with secured data dissemination and QoS.

References

[1] Anitha, R.U.; Kamalakkannan, P., "Enhanced cluster based routing protocol for mobile nodes in wireless sensor network," Pattern Recognition, Informatics and Mobile Engineering (PRIME), 2013 International Conference on , vol., no., pp.187,193, 21-22 Feb. 2013

[2] Kukliński, S.; Wolny, G., "Density based clustering algorithm for VANETs," Testbeds and Research Infrastructures for the Development of Networks & Communities and Workshops, 2009. TridentCom2009. 5th International Conference , vol., no., pp.1,6, 6-8 April 2009

[3] Safa, H., Mirza, O., "A load balancing energy efficient clustering algorithm for MANETs", International Journal of Communication System, VL - 23, IS – 4, John Wiley & Sons, Ltd., SN - 1099-1131,
URL - http://dx.doi.org/10.1002/dac.1084

[4] Martinez, F.J.; Cano, J.-C.; Calafate, C.T.; Manzoni, P., "CityMob: A Mobility Model Pattern Generator for VANETs," Communications Workshops, 2008. ICC Workshops '08, vol., no., pp.370,374, 19-23 May 2008

[5] Ramakrishnan B, Rajesh R S, Shaji R S, "CBVANET: A Cluster Based Vehicular Adhoc Network Model for Simple Highway Communication" , Int. J. Advanced Networking and Applications 01/2011; 755:2-4.

[6] Siva Ram Murthy C, Manoj B S, "Ad Hoc Wireless Networks, Architectures and Protocols", First Indian reprint 2005, Pearson Publication. ISBN 81-297-0945-7