Scalable Back-Bone-Assisted Hop Greedy Routing for VANET's City Environments (SBHG)

RajaniThite
PG student,
Dr. D. Y.Patil College of
Engineering Ambi, Pune, India

A. K. Bongale
PG Guide
Dr. D. Y.Patil College of
Engineering Ambi, Pune, India

ABSTRACT

Vehicle Area Networks play very crucial role in variety of applications, such as Co-operative driving, Information sharing, other added services like Navigation, internet access etc. A lot more research is required in the optimal routing, sensing the vehicles, safely messages to be broadcasted etc. To reach to the destination flooding like technologies are unreliable in sparse regions so shortest path algorithms has been used such algorithms can cause higher hop count. On-Demand routing protocols are used by mobile nodes in an ad hoc network. It offers quick adaptation to self-motivated link conditions, low processing and Network utilization, memory overhead, and which determines unicast routes to destinations within the ad hoc network. But path generated by these protocols may deviate far from the optimal because mobile nodes and no knowledge of current topology which causes higher hop counts. In this paper we proposed self-healing and optimizing routing techniques for VANETs which improves routing optimality by monitoring routing ways continuously and changes the path which is optimal and causes less hop count.

General Terms

Vehicularad hoc network (VANET), optimal path discovery algorithm.

Keywords

Distance Vector Routing Algorithm in VANET

1. INTRODUCTION

In an Automotive industry vehicles are not supposed only a thermo mechanical machine having few electronic devices; instead vehicles are supposed an intelligent system carrier. A wide variety of applications are there of intelligent system carrier such as cash prevention, freeway management, driver assistance, infotainment to passengers or drivers, location tracking of other vehicles etc[1]. In the VANET a lot more research is required in routing, vehicular sensing, safety message broadcast etc. In a network we cannot predict the actual traffic density because in city traffic is different from the highway or delay torrent network, the structure of road is different i.e. straight or curved, available public transport, driver nature can impact on the traffic. In this paper we are going to consider about city network. If we consider connectivity parameter then the most connected road path will be overcrowded by sending the packets by the same path. If we consider hop count then the shortest path can have more change in direction causes higher hop count. In this paper we proposed an algorithm having selection of path based on both connectivity as well as hop count. The proposed routing protocol would result higher Throughput, lower energy consumption & lower latency. Throughput is the successful message delivery rate, Latency is the several kinds of delays incurred in the processing the data in a network.

In the system design we are going to divide the city map in to zones. And intersections are chosen as a boundary intersection as shown in fig. 1 In the figure 1 the city map is divided in to 16 polygons, these polygons are known as zones. It has been found that connectivity increases with increase in number of lanes [2], these polygons are formed by major roads that are having more than two lanes. Many minor roads are there inside the major roads. Minor roads are nothing but having less than two lanes. At the corner of each zone there are boundary intersections. From the boundary intersection all the packets are all the packets are routed, therefore at least one node should be there at the boundary intersection.

Connectivity should be preserved at the intersection and on road segment. First we consider about at the intersection, in the intersection the backbone nodes are used to maintain the connectivity. Figure 2 shows the backbone nodes at the intersection. Stable, primary, secondary are the three types of backbones [3]. Stable backbone is the one which is waiting at an intersection, primary and secondary backbones are those which are crossing the intersection, at the intersection primary backbone is located and outside the intersection secondary backbone is located. Secondary backbone is selected by primary backbone by considering direction of vehicle, position and speed. When primary backbone leaves the intersection suggest the secondary backbone to become primary backbone. While forwarding a packet backbone node is preferred as it can have communication history and can store a packet if there is no forwarder in the intersection region. In the absence of forwarding node the packet is buffered at the stable backbone. The stable backbone maintains the history of communication as well as timestamp. In the case of road segment if the region is void then the nodes suppose themselves as a backbone which is near to the void region.It is necessary for a back-bone node to declare its presence as soon as it enters the intersection region. For this purpose, the periodic beacons cannot be used because the beacon interval might be larger than the duration of stay of a node at an intersection. To overcome this issue, back-bone nodes use positional beacons, as described in [5].

In this network the on demand routing protocol can be used to find the shortest path discovery. But the path is continued till the connection breaks. But as nodes are moving then it can take the long route so the path takes higher hop count in that delay increases.

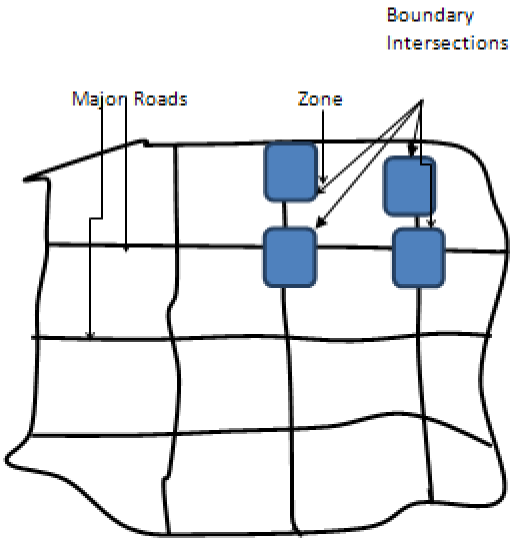


Figure 1: Zone Structure

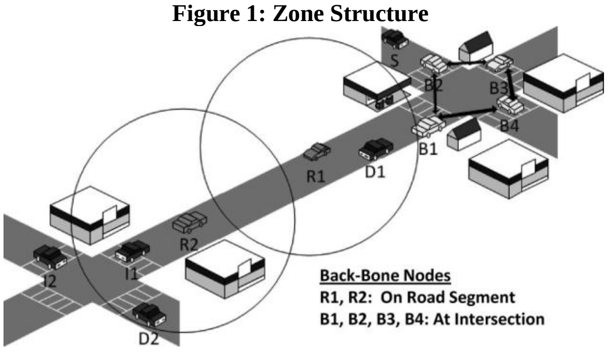


Figure 2: Backbone node structure [3]

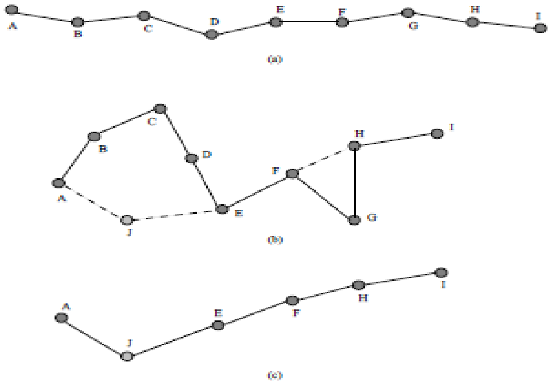


Figure 3: Examples of the changes in the routing path

There is routing path from a source node A to a destination node I as shown in Figure 3(a). This initial path is determined through the path discovery process, in which the distance between the source and destination is the shortest in terms of the number of hops, or very close to it. A packet takes eight hops while getting routed from A to I. During the course of time, the mobility of the nodes may make the shape of the routing path similar to the one shown in Figure1(b) while retaining the connectivity. In this new shape is in the transmission range of A, and E is in the trans-mission range of J. Similarly H is in the transmission range of F. However, because of the usage of route caches and the validity of the existing routing information, the routing table entries are not updated. Although functionally adequate, using the routing paths of Figure 3(b), a packet still takes eight hops to reach from node A to node I. ideally, the shortest path from A to H needs only five hops as shown in Figure 3(c). The goal of this paper is to identify such Situations and self-heal and optimize the paths dynamically by modifying the entries of the routing tables. So we can find the optimal path as in c by route AJEFHI by applying algorithm 1.

2. MOTIVATIONS

• Intersection Node Probing Problem

In city environments, intersections play crucial roles for data communications. As the intersection region is comparatively small and the probability of change of direction is very high, it will be risky to choose an unstable node as the forwarding node from this region. Protocols [5], [6], [7] that ensure connectivity on the routing path are not affected by this problem. Although CAR [8] addresses connectivity issues, it could be affected as the average connectivity does not ensure connectivity in individual road segments in a routing path. Protocols such as GPSR, GPCR, and GSR do not ensure connectivity, and hence, the foregoing problem can have a serious impact on their performances.

• Location Service Requirement Problem

Retrieving the real-time position information of the source or the destination is nearly impossible as that information has to travel a number of hops in a city area, which is generally very large in size. Further, reducing the end-to-end delay is crucial for any routing protocol. Moreover, the location service is found to be superfluous for the nodes that do not take part in any communication. Protocols like GPSR [9] and GPCR[10] take the aid of proactive location services like hierarchical location service (HLS) [11] and grid location service (GLS) [12]. In these location services, the lower beacon interval is the key factor for higher accuracy. Apparently, increased beacon messages create havoc in dense city scenarios [13]. Although the reactive location service used in GSR is an exception, neither the source nor the destination can keep a tab on each other if they change their position in the middle of data communications. As GSR relies on flooding to probe the destination position, it also suffers from drawbacks induced by flooding. As far as GyTAR [14] is concerned, it uses City Sense as the tool for location services. Sensors are deployed at the intersections to provide the actual position information of the destination. Aggregating and disseminating the position information throughout the entire network involve both computational and communication overheads. For every minor movement of the destination, there is a need for the computation of a new path to the destination from the intermediate intersection. As a result, the hop count may be increased.

• Distance- or Connectivity-Based Weighted Graph Problem

When a typical ad hoc network scenario is mapped into a graph, the nodes are considered as vertices, and the communication link between two nodes is considered as an edge. The weight of each edge is represented by the Euclidean distance between the end points. Due to the presence of high-rise buildings in city environments, vehicles are unable to communicate with their peers even if the Euclidean distance between them is less than the transmission range. A shortest path algorithm is executed on the graph to select the routing path. Many times the distance between source and destination is less but number of hop count increases.

- **Packet Swinging in Greedy Forwarding**

In greedy forwarding [9]–[10] & [15]-[18] a sender chooses a forwarding node that is closest to the destination. However, in extreme scenarios, the packet forwarding is delayed. Normally, in cities, the distance between two intersections is far less. Further, plenty of intersections are located in a small area. In such a case, the destination may move across many intersections while a data packet is on the way. Assuming that a location service provides real-time position information of the destination, the forwarding node selection depends on the updated position information of the destination. Apparently, new routing paths are computed at every hop. Hence, the packets keep on moving in search of the destination. We term this phenomenon as packet swinging in greedy forwarding

3. IMPLEMENTATION DETAILS

Once the backbone node structure has been implemented then there is need of routing protocol to reach to the destination. We have used Dijkestra's algorithm to find the shortest path. Once connection sets up communication carried out by using the path given by initialization process. But as the source and destination are moving there can be another shortest path, which gives less number of hop counts. In the EXBAHG we are implementing path aware [19] to find the optimal path so that number of hops to reach to the destination will be less, throughput will be high and the traffic will be distributed among nodes. In the algorithm1 the optimal path is find out as there is no breakage of path. Hop count (HC) need to add in the packet header. HC=0 at the source which increments by one at every hop the packet takes. In an hop comparison array <SA,DA,HC,NA> is maintained. SA is source address, DA is destination address, HC is hop count, NA is neighbor's address from which is packet is broadcasted. If the array is full then Least Recently Used policy is applied and replaced. Rest of the algorithm is explained below. In which we are going to compare the BAHG and our applied algorithm on BAHG that is EXBAHG

Algorithm 1

```
When node a receives or overhears a packet P,
BEGIN
1. IF the node a is the final destination address,
    consume the packet. GOTO END;
2. (Assume P belongs to $< SAk, DAk >$ $flow$.) $Compare$
    $< SAk, DAk >$ $to\ all\ the\ valid\ entries\ in\ the\ hop$
    comparison array;
3. IF there is no match with the entries, store
    $< SAk, DAk, HCk, NAk >$ $in\ the\ hop\ comparison\ array$;
4. IF the packet is destined to a as the next − hop
    node, process the packet for forwarding further.
    GOTO END;
5. (Assume that it matched with an entry
    $< SAk, DAk, HCj, NAj >$) $IF$ $(HCk -$ HCj $>$ 2), $a$
    short − cut is found, node a does the following:
    5.1 Send a message to NAj to update the
        routing table such that the next hop address for
        destination node DAk is modified to the address
        of node a;
    5.2 DAk will be Modified as NAk in thedify its routing table by
        making the
        next − hop address for destination DAk as NAk;
    5.3 Modify its hop comparison array, delete the
        entry corresponding to $< SAk, DAk >$;
END
```

Algorithm 1.Parh Aware Algorithm to find the optimal path [19]

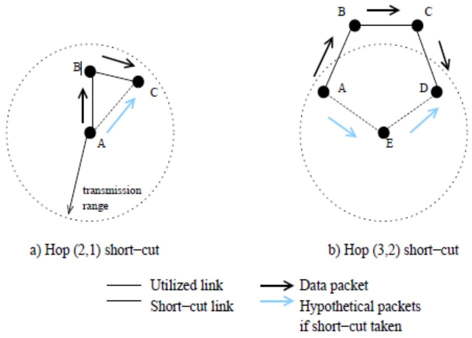
4. CONCLUSION AND FUTURE SCOPE

In EXBAHG we have covered the intersection node probing problem by using backbone nodes, as well as optimizing the routing techniques for vehicular ad hoc nodes. EXBAHG improves routing optimality by monitoring routing paths continuously.

We can use other metrics in self-optimizing operations, such as the QoS requirements

5. MATHEMATICAL MODEL

In this section, we derive the probability of the existence of the short-cut paths in mobile ad hoc networks. We model an ad hoc network by a set of random points on a two dimensional area. N points are plotted on an X-Y Cartesian plane, so that their x-coordinates and y-coordinates are uniformly distributed in the ranges of [0,X] and [0,Y], respectively. If the distance between any two points is less than 1 then they are egarded as adjacent. Among all these points, a sequence of n + 2 given points are of interest, which are labeled as v0, v1, · · · , vn+1. We are given the condition that these nodes form a path, i.e. points vi and vi+1 are adjacent for i from 0 to n. This condition limits the relative positions of vi+1 with respect to vi, since we know vi+1 is located within a unit circle centered at vi. But each point in the circle is equally probable to be the position of vi+1, so there is still a room for freedom for the shape of the path, and for the distance between v0 and vn+1. For example, if the



a) Hop (2,1) short−cut

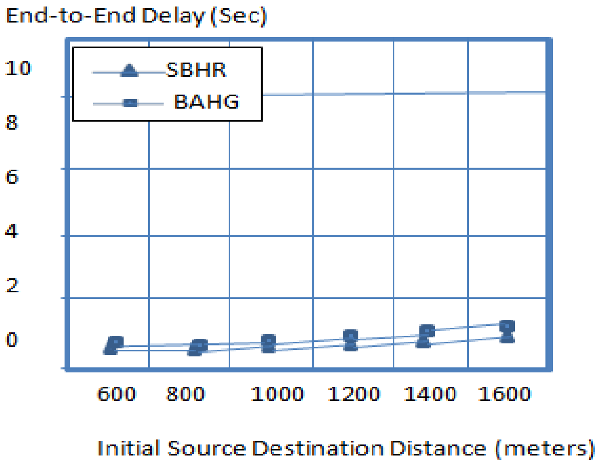
b) Hop (3,2) short−cut

Figure 4: Basic short-cut path formations.

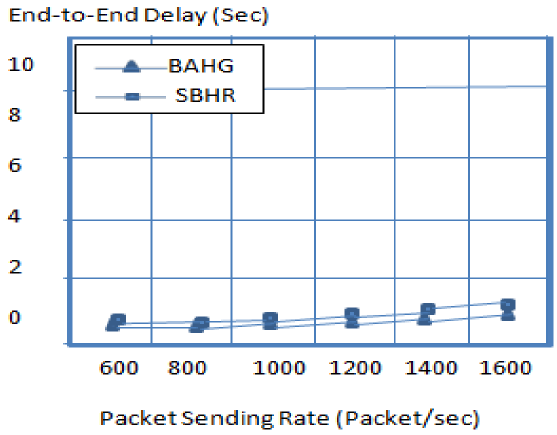
consecutive points form a straight line, then the distance can approach n + 1. On the other hand, if they form a cycle, the distance can be as low as zero. The question now is: What is the probability that v0 and vn+1 are adjacent. We denote this probability by P[n], and term it as the probability of adjacency. Let's consider the two basic short-cut configurations shown in Figure 5. Using the notations in our geometric model, the probability of hop (2,1) short-cut is just P[1] in adjacency probability. For a hop (3,2) short-cut, the adjacency probability is equal to P[3], which is illustrated in Figure 2(b). Nodes E, A, B, C, and D form a sequence of five points, which can be denoted as v0, v1, v2, v3, and v4, respectively.

6. RESULTS

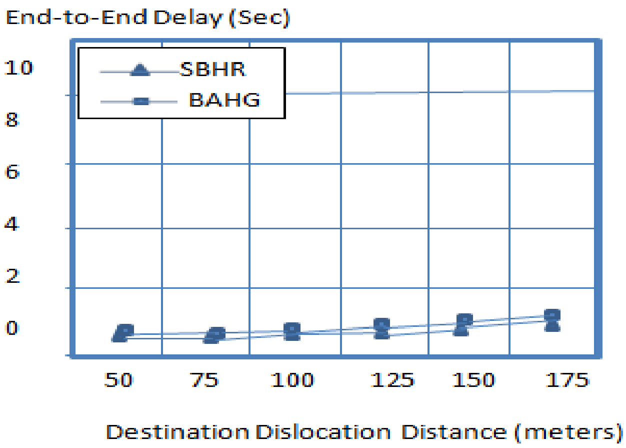
Fig. 5(a) shows the average end to end delay against source and destination distance. In graph the comparison between BAHG[3] and SAHG is shown, previously delay is same but BAHG's goes on increasing as the distance between source and destination increases but SAHG gives good results as it checks again is there any another shortest path. Fig. 5(b) shows End-to-End Delay against Packet sending Rate. Fig. 5(c) shows End-to-End Delay against Destination Dislocation Distance.



(a)



(b)



(c)

Fig.5 (a)End-to-end delay versus initial src-dst distance.(b) End-to-end delay versus Packet sending Rate (c) End-to-end delay versus destination dislocation distance.

7. ACKNOWLEDGMENTS

I would like to thank my project advisor Anupkumar Bongale ,Dr. D. Y. Patil College Of Engineering Ambi, Pune, for generating the idea for this project and for his help, advice and supported to advice proved essential to whatever small success this project may have achieved.

Also, I would like to thank my parents for their continual encouragement and the Positive support. I would also like to thank my wonderful colleagues and friends for listening to my ideas, asking questions and providing feedback and suggestions for improving my ideas.

8. REFERENCES

[1] J. Bernsen and D. Manivannan, "Unicast routing protocols for vehicular ad hoc networks: A critical comparison and classification," *Pervasive Mob. Comput.*, vol. 5, no. 1, pp. 1–18, Feb. 2009.

[2]M. M. Artimy, W. Robertson, and W. J. Phillips, "Connectivity in inter-vehicle ad hoc networks," in *Proc. IEEE CCECE*, May 2004,*pp. 293–298*

[3] Pratap Kumar Sahu, Eric Hsiao-Kuang Wu, Member, IEEE, JagrutiSahoo, and Mario Gerla, Member, IEEE, "BAHG: Back-Bone-Assisted Hop Greedy Routing for VANET' City Environments", IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 14, NO. 1, MARCH 2013.

[4]P. K. Sahu, E. H. Wu, J. Sahoo, and M. Gerla, "DDOR: Destination discovery oriented routing in highway/freeway VANETs," in *SpringerTelecommun. Syst.—Special Issue Vehicular Communications, Networks, Applications*, 2010, pp. 1–18.

[5]Q. Yang, A. Lim, S. Li, J. Fang, and P. Agrawal, "ACAR: Adaptive connectivity aware routing for vehicular ad hoc networks in city scenarios," *Mob. Netw. Appl.*, vol. 15, no. 1, pp. 36–60, Feb. 2010.

[6]G. Liu, B. S. Lee, B. C. Seet, C. H. Foh, K. J. Wong, and K. K. Lee, "A routing strategy for metropolis vehicular communications," in *Proc.ICOIN*, *LNCS*, Aug. 2004, pp. 134–143.

[7]M. Jerbi, S. M. Senouci, T. Rasheed, and Y. Ghamri-Doudane, "Towards efficient geographic routing in urban vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 9, pp. 5048–5059, Nov. 2009.

[8]V. Naumov and T. R. Gross, "Connectivity-aware routing (CAR) in vehicular ad hoc networks," in *Proc. IEEE INFOCOMM*, 2007, pp. 1919–1927.

[9] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. ACM MOBICOM*, 2000, pp. 243–254.

[10] C. Lochert, M. Mauve, H. Füßler, and H. Hartenstein, "Geographic routing in city scenarios," *ACM SIGMOBILE Mobile Comput.Commun. Rev.*, vol. 9, no. 1, pp. 69–72, Jan. 2005.

[11]W. Kieß, H. Füßler, and J. Widmer, "Hierarchical location service for mobile ad-hoc networks," *ACM SIGMOBILE Mob.Comput.Commun.Rev.*, vol. 8, no. 4, pp. 47–58, Oct. 2004.

[12]J. Li, J. Jannotti, D. S. J. De Couto, D. R. Karger, and R. Morris, "A scalable location service for geographic ad hoc routing," in *Proc. ACMMOBICOM*, 2000, pp. 120–130.

[13]P. K. Sahu, E. H. Wu, J. Sahoo, and M. Gerla, "DDOR: Destination discovery oriented routing in highway/freeway VANETs," in *SpringerTelecommun. Syst.—Special Issue Vehicular Communications, Networks, Applications*, 2010, pp. 1–18.

[18]K. C. Lee, J. Häerri, U. Lee, and M. Gerla, "Enhanced perimeter routing for geographic forwarding protocols in urban vehicular scenarios," in *Proc. IEEE GlobeCom Workshops*, 2007, pp. 1–10.

[19]Chao Gui and PrasantMohapatra, 'SHORT: Delf-Healing and Optimizing Routing Techniques for Mobile Ad Hoc Networks June 03Bowman, M., Debray, S. K., and Peterson, L. L. 1993. Reasoning about naming systems.