ISBN 978-981-11-0008-6
Proceedings of 2016 6th International Workshop on Computer Science and Engineering (WCSE 2016)
Tokyo, Japan, 17-19 June, 2016, pp. 88-94
doi: 10.18178/wcse.2016.06.016

Algorithm for Leader Node Selection in Vehicular Adhoc Networks at Road Intersections

Sreyash Ratna Tripathi, Shreya Mahabala Alva and Debasis Das[+]

Department of Computer Science and Engineering, NIIT University,Neemrana, Rajasthan, India

Abstract. With advancement in the field of vehicle automation and wireless communication, Vehicular Ad Hoc network (VANET) based solutions are emerging as the answer to the dire problem of road traffic management. As dedicated short-range communications (DSRC) devices, vehicles can be organized in a peer-to-peer network to manage their movement for a smoother traffic flow. Our paper targets traffic management at road intersections. Our approach calls for the selection of a leader node in every lane so that they may mutually decide a safe and efficient order of crossing the intersection. Generally leader node selection algorithms have a computational complexity of O(nlog(n)). Our proposed algorithm uses geographic routing coupled with sequential flow of communication to select the leader node with a computational complexity of O(n) where n is the number of vehicles present in a lane at Road Intersection.

Keywords: node selection algorithm, leader node, king node, intelligent transportation system, road intersection, traffic management, cooperative intersection management, cooperative vehicle intersection control, vehicular adhoc network.

1. Introduction

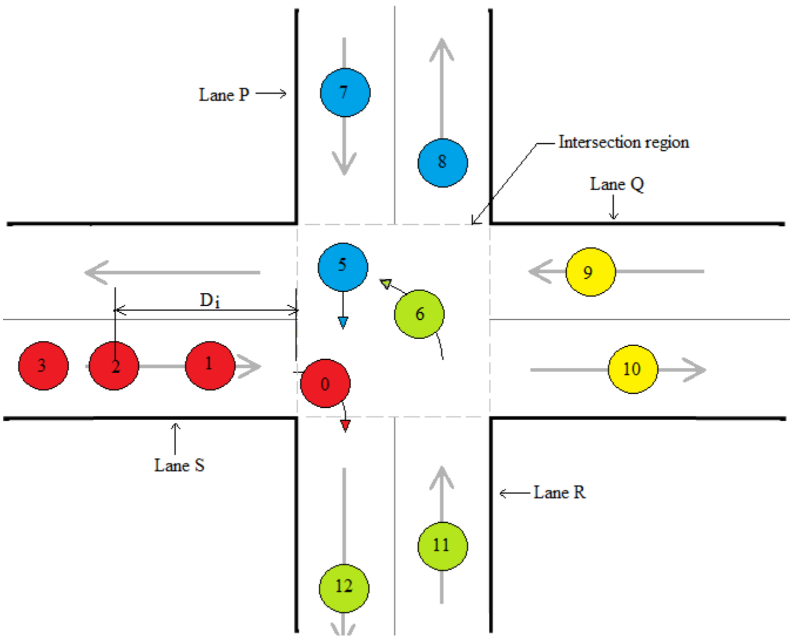

Fig. 1: A typical four-way bi-lane road intersection.

The world economy loses billions to road traffic congestion, associated delays and fuel wastage. With advancement in Vehicle-to-Vehicle(V2V)/Vehicle-to-Infrastructure(V2I) communication, in vehicle sensors

---

[+] Corresponding author. Tel.: +91 7790844892;
E-mail address: Debasis.Das@niituniversity.in.

and computer-based control, traffic management systems must evolve to harness the power of these technologies. Road Side Units (RSUs) can serve as the infrastructural support to co-ordinate vehicular movement however cost of installation and maintenance poses as a major drawback.

This paper proposes a new approach of selecting the leader node on any lane which is leading up to an intersection using only V2V communication. We have chosen geographic routing [9] (also known as position based routing) for cooperative intersection management. The leader node that will head communication for a set of vehicles. The leader nodes will communicate among themselves to decide their order of crossing. Global Positioning System (GPS) gives the location co-ordinates of each node. Each node receives this information pertaining its neighboring nodes. In every lane, the nodes that are closest to the intersection are allocated the rank 1 and chosen to be the leader nodes. Each node will deduce its own rank in the lane of vehicles based on the rank of the node that is directly ahead of it and thus closer to the intersection. Every time a node leaves its lane and enters the intersection the node that is right behind will assume rank 1 and will be communicating with the next set of designated leader nodes chosen from the other lanes.

2. Related Work

In Reference [1] J. Lee et al. designed a Cooperative Vehicle Intersection Control (CVIC) system for the Connected Vehicles environment. It relied on V2I and V2V communication for effective intersection management without a traffic signal, for fully automated vehicles. It sought to resolve all conflicting approaches with overlapping trajectories at the intersection and to chart out safe courses for all the vehicles. In Reference [2] I. Zohdy et al. used the concept of Cooperative Adaptive Cruise Control (CACC) in conjunction with the Vehicle Dynamics Model to draw up the speed distribution for the vehicles. It employed controllers at the ”smart” intersections which counseled the vehicle, at each time step, regarding the best course of action to be taken to mitigate collision and decrease the total delay. In Reference [4] F. Perronnet et al. extended sequence-based protocol, called ’Transparent Intersection Management’, for cooperative intersection management through a high-level signalization system (that is green or red displayed by an onboard screen to the driver) in which a speed limitation is determined by the sequence and adhered to by the autonomous vehicle. In Reference [5] A. Darehshoorzadeh et al. proposed Heuristic Candidate selection algorithm based on Optimum delivery Probability (HU-COP), a heuristic and fast candidate selection algorithm (in Opportunistic Routing) based on the link delivery probability a candidate and another node. HU-COP finds candidates through links whose delivery probabilities are near optimum, outperforming ExOR. In Reference [6] Mengmeng Wang et al. Mengmeng Wang et al. explored the issues of routing that are particular to urban area VANETs. They evaluated the performance of the existing routing protocols including GPCR, GyTAR, STAR, and TDR in city scenarios and also put forward guidelines for designing VANETs routing protocols. In Reference [7] Xiangjun Qian et al. adopted a priority-based coordination framework to partition the problem into a priority assignment problem and a vehicle control problem under the assigned (fixed) priorities. In Reference [8] Acarman T. et al. presented a wireless access protocol that routes the data packet towards a vehicular node in a neighbor intersection zone using navigation map data and routing information. In their algorithm, routing decision is chiefly made at the intersection zones.

3. Objective

In this paper, our focus is on road intersections, and our approach involves selecting leader nodes for regulatory purposes.

The main objectives of our proposed approach are given below:

- Communication between candidate nodes will decide the vehicles' order and speed of crossing the intersection.
- To improve upon existing methods for finding a leader node by using vehicle to vehicle (V2V) communication to establish precedence among the vehicles in the intersection.

4. Leader Node Selection in VANET at Road Intersections

4.1. Preliminaries and Data Structures

In this section, we will introduce the basic terminologies and assumptions which are used in the proposed algorithm.

- **GPS:** Global Positioning System. It is a satellite based navigation system that will provide the location and time information for the vehicle.
- T_c: It represents current time. %timer starts when the node enters the network
- T_i**:** It represents time taken by the vehicle to reach the intersection.
- T_d: It represents a short waiting time.
- T_m: Time of transmission of a message.
- D_i: It is the distance of the vehicle from the Intersection. The first value of the D_i is taken from the GPS. After that, D_i is maintained as per the formula:
  $D_i = D_{GPS} – S_{vehicle} * T_c – T_{rec}$ here, T_{rec} is the time when the GPS Data was received and D_{GPS} is the distance from intersection received from GPS.
- D_r: It is the direction of the moving vehicle.
- I_{packet}: It is an information packet broadcasted by the vehicle containing V_{id}, D_i, D_r and T_m.
- *Session-Request:* It is a message that requests for a session from another node to communicate further. It contains V_{id}, Request Message, D_i and T_m.
- *Request-Accepted:* It is a message that contains V_{id}, D_r, D_i of the vehicle and T_m.
- *Interrupt:* It is a signal generated shows that some error was found while executing the algorithm.
- *Data-Packet:* It is the message transfer packet containing the V_{id} of the requesting vehicle, R_m the modified rank, current speed of the broadcasting vehicle and T_m.

4.2. Algorithm 1: King Node Selection Algorithm

Step 1: GPS notifies the vehicle entering a lane leading up to an intersection and sends the initial value of *Di* to the vehicle.

Step 2: The vehicle starts listening for broadcasts of I_{packet} from other nodes for a Wait-Time *Td*.

Step 3: If the vehicle hears a broadcast of an I_{packet} during *Td* then execute *Algorithm 2* and returning from *Algorithm 2* execute from *Algorithm 1 Step 8*.

Step 4: End If

Step 5: Assume Rank: 1 and start broadcasting sequence of I_{packet} and continue to listen for any broadcasts of I_{packet} from other nodes as well.

Step 6: If the vehicle hears a broadcast Then execute *Algorithm 2*.

Step 7: End If

Step 8: As soon as a *Session-Request* packet is received, pause the broadcasting sequence of the I_{packet}.

Step 9: Select the node which sent the first *Session-Request* packet.

Step 10: Broadcast a *Request-Accepted* packet for the selected node.

Step 11: Execute *Algorithm 3*

Step 12: If the broadcasting vehicle receives an *Interrupt* Then it shall resume it's broadcasting sequence which was paused in *Step 8*.

Step 13: Else the broadcasting vehicle starts Message Transfer in the Session by executing *Algorithm 4*.

Step 14: End If

//The vehicle with *Rank-1* is the *King* Node while the rest of the vehicles are the Follower Nodes.

Step 15: If (*King* Node enters the Intersection, i.e $D_i = 0$, Then

Step 16: *King* node sends out a *Good-Bye* message to first follower.

Step 17: The first follower catches the *Good-Bye* message and updates its rank to 1 and becomes the next *King* Node.

Step 18: It then sends out a message to the next following node to upgrade its rank by 1 and accordingly the ranks get upgraded for all the vehicles in the lane.

Step 19: End If

4.3. Algorithm 2: Checking Broadcast Authenticity of Ipacket

Step 1: If D_i of the vehicle receiving the I_{packet} > D_i present in the I_{packet} broadcasted AND D_r of the vehicle is same as D_r present in the I_{packet}

Step 2: It means that a vehicle which is in the same lane and is closer to the intersection has broadcasted it's I_{packet}

Step 3: Drop the current assumptions and ranks

Step 4: Stop any broadcasting sequence

Step 5: Send a *Session-Request* packet to the broadcasting vehicle

Step 6: Else drop the received I_{packet}

Step 7: End If

4.4. Algorithm 3: Checking Session Authenticity

All the nodes which sent the *Session-Request* packet are expecting a reply. They will all hear any broadcast message.

Step 1: If Broadcasted Message reads *Request-Accepted* Then

Step 2: Check the D_r of the receiving vehicle with the one given in the packet

Step 3: If D_r of the receiving vehicle = D_r given in Request-Accepted Packet Then

Step 4: Take the D_i and time of message transmission T_m from the *Request-Accepted* packet.

Step 5: Use the following equation to modify the given D_i to D_m. $D_m = D_i - (S_{vehicle} * (T_c - T_m))$

Step 6: where T_c is the current time.

Step 7: This is done to compensate the change in transmitted D_i in the message versus the current D_i because of movement of vehicles.

Step 8: If $D_m > D_i$ of vehicle Then

Step 9: Generate an *Interrupt*

Step 10: Broadcast the *Interrupt*

Step 11: Exit **Algorithm 3** and all nodes execute **Algorithm 1** from **Step 2**

Step 12: Else All the non-selected nodes exit **Algorithm 3** and execute **Algorithm 1** from **Step 2**

Step 13: End If

Step 14: Else Ignore the current broadcast message.

Step 15: End If

Step 16: Ignore the current broadcast message.

Step 17: End If

4.5. Algorithm 4: Message Transfer in a Session

Step 1: The selected node with *Request-Accepted* packet replies with a *Begin-Transmission* packet to the broadcasting node.

Step 2: The broadcasting node, upon getting the *Begin-Transmission*, sends a *Data-Packet* to the vehicle with a modified rank $R_m = R_{broadcasting\text{-}vehicle} + 1$ and the speed $S_{\text{broadcasting-vehicle}}$

Step 3: The vehicle receives this *Data-Packet* and changes it's rank to R_m and changes it's speed to match $S_{broadcasting\text{-}vehicle}$

Step 4: Update T_{rec} as $T_{rec}=T_c$ and $D_{GPS}=D_i$. This will be used to update D_i.

Step 5: The vehicle then replies with an *End-Transmission* packet to the broadcasting node and start its broadcasting sequence of I_{packet} with the modified data.

Step 6: Upon receiving this *End-Transmission* packet the broadcasting node stops broadcasting. However, it does not stop listening for future broadcasts of I_{packet} from other vehicles.

Step 7: For the above condition, If the previously broadcasting vehicle hears a broadcast of an I_{packet} Then execute **Algorithm 1**, from **Step 6**.

Step 9: Else exit **Algorithm 4**

Step 10: End If

The King node selection algorithm starts executing as soon as a node enters a lane leading up to an intersection. There are two possible scenarios for a node entering such a lane:

- Case 1: Node enters in an empty lane. i.e., no other vehicle is present in that lane.
- Case 2: Node enters in a busy lane. i.e., there are one or more vehicles already present in that lane.

The King Node selection algorithm accounts for both of these cases in the following manner (refer Fig. 2) Let N_1 enter an empty lane, as in **Case 1**. As N_1 enters the lane, it will sense the approaching intersection with the help of GPS intimation which will trigger **Algorithm 1**. N_1 starts listening for broadcast from any other node on the lane for T_d time. After T_d has elapsed, N_1 assumes rank 1 and starts broadcasting $I_{packets}$ while continuing to listen for any incoming messages.

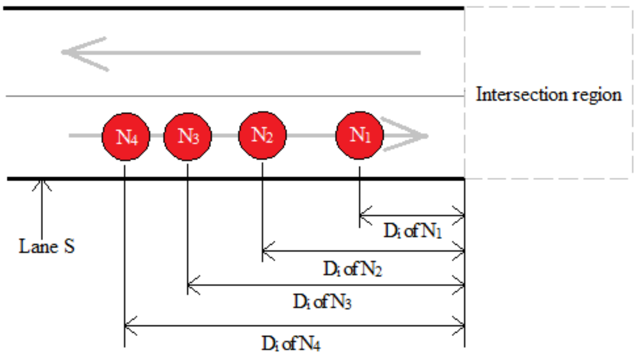

Fig. 2: Snapshot of Lane S.

Now suppose another node, N_2, enters the lane. N_2 will be entering a busy lane as in **Case 2**. N_2 will also start listening for a broadcast and will receive $I_{packets}$ from node N1 and after authentication of these $I_{packets}$, it will send a session request to N_1. However, if N_2 fails to listen to N_1 in the designated time, due to some reason, N_2 will also commence the broadcast of its $I_{packets}$ with the same assumptions as that of N_1 which creates a problem. To solve this problem, a protocol stated in **Algorithm 2** is called. The protocol specifies that a node is supposed to accept and read only those broadcast messages which come from another node whose distance from intersection D_i is lesser than their own D_i and are traveling in the same direction. Therefore, even if N_2 starts broadcasting assuming it's rank to be 1, as soon as it receives a message from N1 which has a lesser Di than that of N_2, N_2 will stop it's current broadcast transmission.

For a case where multiple nodes, N_2, N_3 and N_4, enter the lane very close to each other, then all the nodes will hear N_1 broadcasting and send a session request to N_1. In such a case, N_1 will only entertain the session request of the node that that was sent first and respond by broadcasting a Request-Accepted packet with the senders V_{id}, D_i and D_r to all the nodes. As two or more nodes may send their Session-Request packet at the same time, there is a possibility of a collision. Here, if nodes N_2 and N_3 have a broadcast storm and their Session-Request packets are lost then N_1 will only receive the session request from N_4. Now, Since N_2 and N_3 are waiting for a response, even they will read the Request-Accepted sent by N_1. If the D_i mentioned in the Request-Accepted is more than that of N_2 or N_3, then they will generate and broadcast an Interrupt which will cause the session to terminate immediately and N_1 will resume it's broadcasting sequence.

When the authenticity of the session has been checked, the Communicating nodes start the message transfer. N_2, after receiving the Request-Accepted packet, will send a Begin-Transmission packet to N_1. N_1 will send the modified rank and speed to N_2. N_2 will modify it's rank and speed accordingly and send an *End-Transmission* packet to N_1. N_2 will then adjust the values of *Trec* and D_{GPS} which is used to update D_i. When

N_1 receives the *End-Transmission* packet, it will permanently stop the broadcasting stream of its I_{packet} which was earlier paused, however, it will continue to listen for any other broadcast and *N2* will start its broadcast sequence of I_{packet}. When node N_1 enters the intersection, it will send a *Good-Bye* packet to the node with Rank 2 which is N_2. Upon receiving a *Good-Bye* message, N_2 decrements it's rank by 1 thereby becoming Rank 1 or the new *King* node. N_2 will further pass this message and so on and so forth till all the nodes in the lane adjust their ranks by 1.

5. Evaluation and Results

In order to verify the performance our proposed algorithm, we developed a simulation environment using SUMO and NS2 which models the operation of a wireless network along a single road lane, connected to a road intersection. The simulator does not model medium contention or any radio propagation effects other than random packet loss based on a delivery ratio and assumes losses are not correlated with packet size. The simulated environment consists of nodes moving towards a fixed point (the intersection region) with fixed speed of 40 km/hr.

During our experiments, we first set number of vehicles in a lane, *n*, to 3, 4, 5 and so on. Due to constraints of the simulator, we have limited the *n* to 25. The plot indicates that the number of messages transmitted to correctly assign ranks to each of the vehicles in a lane varies linearly as the number of vehicles in the given lane.

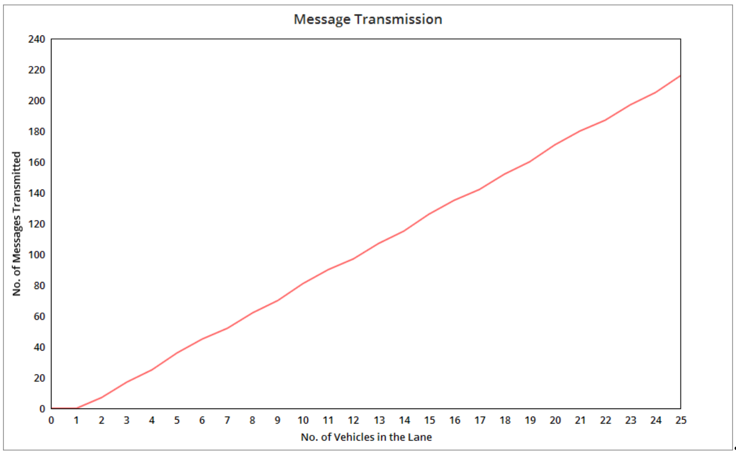

Fig. 2: Number of transmissions required to assign ranks to all nodes in a lane, averaged over number of vehicles in a single lane.

6. Acknowledgements

We'd like to thank our parents and our mentor for their constant support and guidance.

7. Conclusion and Future Work

Our approach, in this paper, was focused on V2V communication and position based routing, without relying on infrastructural support. In order to coordinate the movement of vehicles at a road intersection, the first step would be to selecting a leader node to function as the network coordinator. Existing leader selection algorithms have a computational complexity of O(nlogn) or higher and in this paper, we have formulated an algorithm called as the King Node Selection algorithm with a complexity of O(n) that does not rely on infrastructural support.

In future, we shall extend our work by devising a priority based algorithm for a safe and efficient sequence of crossing to synchronize vehicle movement at intersections via communication between these chosen king nodes.

8. References

[1] J. Lee and B. Park, “Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment,” Intelligent Transportation Systems, IEEE Transactions on, vol. 13, pp. 81–90, March 2012.

[2] I. Zohdy, R. Kamalanathsharma, and H. Rakha, “Intersection management for autonomous vehicles using icacc,” in Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on, pp. 1109–1114, Sept 2012.

[3] A. de La Fortelle, “Analysis of reservation algorithms for cooperative planning at intersections,” in Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on, pp. 445–449, Sept 2010.

[4] F. Perronnet, A. Abbas-Turki, and A. El Moudni, “A sequenced-based protocol to manage autonomous vehicles at isolated intersections,” in Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on, pp. 1811–1816, IEEE, 2013.

[5] Darehshoorzadeh, A.; Boukerche, A., "An efficient heuristic candidate selection algorithm for Opportunistic Routing in wireless multihop networks," in Computers and Communication (ISCC), 2014 IEEE Symposium on , vol., no., pp.1-6, 23-26 June 2014

[6] Mengmeng Wang; Yankang Zhang; Changle Li; Xin Wang; Lina Zhu, "A survey on intersection-based routing protocols in city scenario of VANETs," in Connected Vehicles and Expo (ICCVE), 2014 International Conference on , vol., no., pp.821-826, 3-7 Nov. 2014

[7] Xiangjun Qian, Jean Gregoire, Fabien Moutarde, Arnaud De La Fortelle. Priority-based coordination of autonomous and legacy vehicles at intersection. 17th International IEEE Conference on Intelligent Transportation Systems, Oct 2014, Qingdao, China. <hal-01059090>

[8] Acarman, T.; Yaman, C.; Peksen, Y.; Peker, A.U., "Intersection Based Routing in Urban VANETs," in Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on , vol., no., pp.1087-1092, 15-18 Sept. 2015.

[9] Debasis Das, Rajiv Misra and A. Raj “Approximating Geographic Routing using Coverage Tree Heuristics for Wireless Network”, Wireless Networks , Springer US, Vol. 21. No. 4, 2015, Pages 1109-1118.