Smart traffic flow design with platoons navigation

Prakash Rajendran [1*], Balasaravanan T K [2*], Dr.P.V.Manivannan [3*]

Indian Institute of Madras, Chennai, India*

prakash.rs295@gmail.com [1], **balatk95@gmail.com** [2], **pvmanivannan@gmail.com** [3]

***Abstract*— This paper present a novel algorithm for dynamic node addition and deletion for communication of the information between fixed road side units (RSUs) and moving vehicles in an Intelligent Transportation Systems (ITS). The vehicle navigation is controlled by the RSUs based on position information received by the vehicle nodes. The AODV communication protocol is used for message transfer between the individual nodes and RSUs. The algorithm's performance has been verified and optimized using simulation with the Network Simulation-2 (NS-2) software. The whole system has been experimentally validated through a laboratory scale setup design using Arduino Uno controller boards and Xbee communication modules. The tracking of the moving nodes has been implemented through a motion capture system using web camera and an Image processing algorithm developed using OpenCV.**

Keywords— ITS, WSN, VANET, V2V, V2I, NS-2, RSU, OBU

I. INTRODUCTION

Intelligent transportation systems (ITS) aims to provide services relating to transport and traffic management and enable various users to be better informed and make safer, more coordinated, and 'smarter' use of transport networks. One of the fields in Intelligent Transport System is Vehicular Adhoc Network (VANET). This Adhoc Network aims at establishing communication between mobile vehicles in a traffic system and thus the name Vehicular Adhoc Network. Vehicular Adhoc Networks (VANETs) are the promising approach to provide safety and other applications to the drivers as well as passengers. It becomes a key component of the intelligent transport system. Various forms of wireless communication have been proposed to achieve Vehicular Adhoc Network. Of these proposed forms of communication, the Radio Frequency model is used widely for short and long range communication. There are two types of communication involved in VANET

- Between Vehicles (V2V)
- Between Road Side Units (RSUs) and Vehicles (Infrastructure and Vehicles, V2I) and vice versa.

Road Side Units (RSUs) are stationary bodies which are present at the edges of a road at certain pre-calculated distances. These are used to convey messages to nodes which are in the out of range zone of communication for a particular node. The vehicles within the range of a RSU is said to be a platoon. Communication of platoons are done through the RSUs.

RELATED WORK

Wireless Sensor Network (WSN) is widely in many applications like remote data acquisition, data monitoring and distributed control, etc. Self-driving vehicles (driverless car) is gaining importance in defense and civilian applications. At present Global Positioning System (GPS) is widely used for tracking and guiding the unmanned vehicles. However, researchers around the world are trying to use Wireless Sensor Network (WSN) for tracking and guiding the intelligent vehicles, as they provide vehicle-vehicle local communication with faster data sharing and better connectivity. ITS involved with so many real time challenges, requirements, standards, architectures have been described detailed in [1]. Data routing in vehicular Ad Hoc networks and described about different network architecture for VANET in [2] and [4]. Sensor network layers and WSN application discussed in [3]. Different types of routing protocols detailed in [5] and route the packets to long distance in VANET by using geographic forward [6]. NS2 simulation for backbone routing also discussed in [6]. Group of vehicles (Platoons) maintains at constant speed which implies fuel consumption and continuous data flows [7]. Fast and large data flow algorithm [8] gives efficient outcome for better communication.

II. VEHICULAR NETWORKING DESIGN

A. *Node Creation*

In the proposed system, node is created using Arduino UNO controller boards and Xbee (Series1) communication modules. XBee is used for establishing communication between nodes through RF model approach. Each node is set to have an Arduino UNO programmer which acts as the brain of the node. This Arduino UNO processes the information which is sent to and from a particular node. The algorithm used for communication is also stored in these Arduino UNO boards which is then used to establish a route to a particular destination node. Application Program Interface (API) communication mode is used as it is very reliable and robust. The algorithm for communication is written in the Arduino development system. The following Fig.1 shows a single node represents the On Board Unit (OBU)

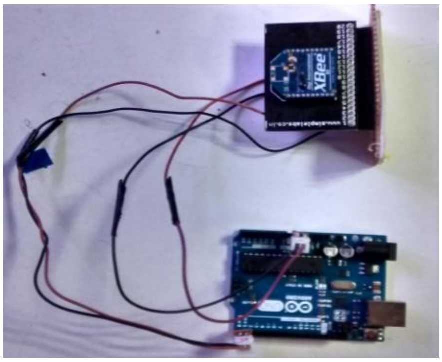

Fig.1. Node On Board Unit (OBU)

B. *Node Addition and Deletion*

In this system, one Stationary Farm Robot (SFR) is designated as a leader node which is generally the coordinator of the network. The leader node may not be in transmission range of every node in the network. Therefore the communication between certain nodes and the leader, has to be routed via intermediate nodes. New nodes has to be added to a network dynamically and nodes that wandered away, has to be detected and removed from the network automatically.

An Algorithm used is as follows: It involves the maintenance of a 2D array. The dimensions of the array is No. of nodes + 1 then have 4 nodes and the nodes [5][5] array would be

	0x21	0x33	0x42	0x56
0x21	1	0	1	1
0x33	0	1	0	1
0x42	1	0	1	0
0x56	1	1	0	1

Fig.2. [5][5] array for 4 Nodes

Element [0][j] and [j][0] contains the address of the jth node of the network. Element [i][j] will represent, if jth node is within range of the ith node. If transmission range of all nodes are the same, then the matrix would be a symmetric matrix.

Node Addition:

For run time addition of new node (NN) into the network, NN sends a message with message type as ADD_REQUEST. The message is then routed to the leader node. The leader node receives add request from various nodes. The leader adds NN's address to the nodes array as mentioned before and updates nodes array based on the origin address of the ADD_REQUEST. It then broadcasts the nodes array's elements as bytes. In case of repeated message from same source, but different intermediate XBees, the message is neglected.

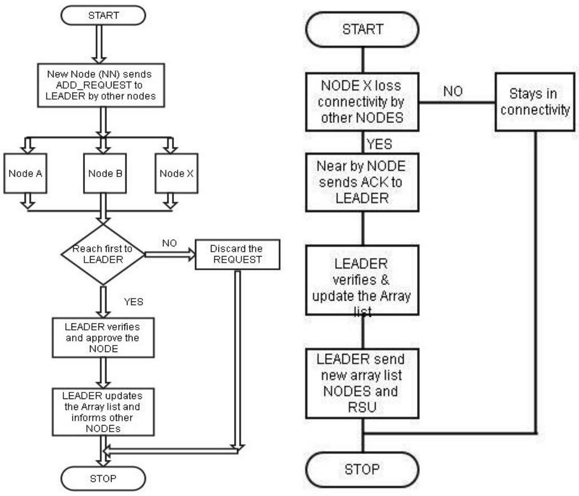

Fig.3. Algorithm for Node Addition and deletion

Node Deletion:

For runtime detection and deletion of nodes, above algorithm used. If a node loses contact with a specific node, it then informs the leader and makes the leader update the nodes array. If all elements of a specific column/row becomes (except diagonal element) zero, then it means that the node associated with that column/row has wandered away from the network. Leader removes the node, updates the nodes array, and broadcasts the information to the other nodes in the network. If the contact with node is lost inadvertently, then the lost node is coded to sound a buzzer and send a message to a global receiver like a GSM network.

C. *Indoor Positioning System*

A laboratory based setup has been developed for vehicle navigation. Mobile Robots and an Intex webcam based system is act as indoor vehicle position monitoring which can identify the location of Mobile robots based on their position. OpenCV library has been used for image processing and to identify the moving vehicles.

Initially, in the raw image, the closer objects measure greater amount of pixel than the farther objects.

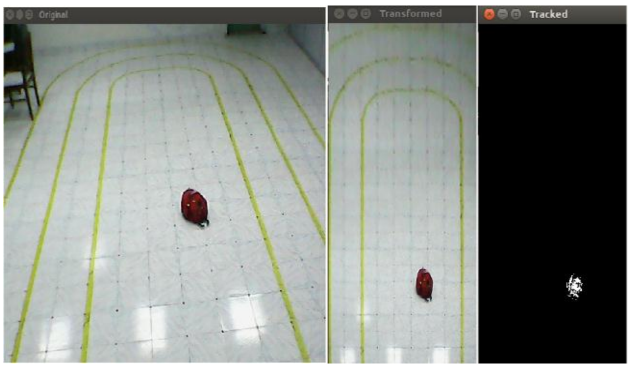

Fig.4. Single Amigobot used for Position tracking

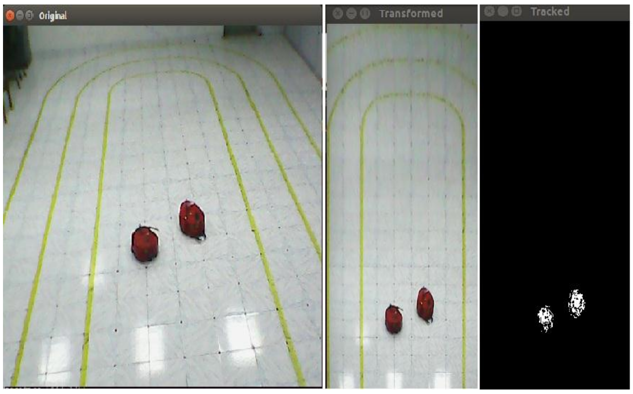

Fig.5. Amigobots used for position tracking

Raw image's perspective is transformed so that distance in the image corresponds to a linear scale. This linear scale is correlated with actual distance to find the spatial coordinate of any point in the image. The Robot tracking algorithm provides the current coordinates of the Mobile nodes. Amigobots are used as mobile nodes and the figures 4 & 5, shows the implementation of robot tracking system.

D. *Communication Protocol*

The API command mode in the XBEE can be used to create our own frame of data. Every data packet has the following frame format

Source	Present Destination	Final Destination	Data

If a node has to communicate to another node, there are two cases

- Destination node within the communication range of Source node.
- Destination node out of communication range of Source node.

Case 1: Destination Node within the communication range of source node

The source node will broadcast the information with *Present destination and Final Destination* as the destination node. All the nodes in the network will receive this packet. The received node processes the packet only if the *present destination* field matches with the address of the node. Thus the packet is processed only by the destination node and after receiving it, the destination node sends an acknowledgement to the source node.

Case 2: Destination Node out of communication range of source node.

The source node will broadcast the packet to all nodes in the network with *Present destination and Final Destination* as the destination node. Since the destination node is out of range for the source node, it will not receive this packet and send an acknowledgement. Moreover, other nodes in the network will also not process the packet since the address shows mismatch with the *present destination* field of the packet. Thus the Source node does not get an acknowledgement from any other node and hence understands that the destination node is out of its communication zone.

The source node generate another packet with *data, source and final destination* fields as the same but present *destination field* alone changed to the next farthest node (the node closest to the destination node). Now, this node (say n^{th} node) receives the packet and sends an acknowledgement to the source node. With the *final destination and present destination* not being the same, it understands that it has to send to the destination node. Thus it generates a new packet with *data* field alone unchanged from the packet it received and broadcasts it. The destination node which is within the communication zone of the n^{th} node will receive the packet and send an acknowledgment to the n^{th} node. Thus information reaches the destination node from the source node.

E. *Deployment of node*

X-CTU software is used to configure the XBEE. For Transfer of data from XBEE to Arduino, Transmit TX and Receive RX pins of XBEE and Arduino UNO (Red circle indicated in the figure) are used. TX of Arduino UNO is connected to RX of XBEE and RX of Arduino UNO is connected to TX of XBee. In this project, 7 nodes has been considered. Out of this 7 nodes, one is treated as RSU and other 6 are divided into 2 platoons of 3 vehicles (nodes) each. Consider a data has to be sent from one platoon to another (platoon 2 is out of communication range of the platoon 1) via the Road Side Unit.

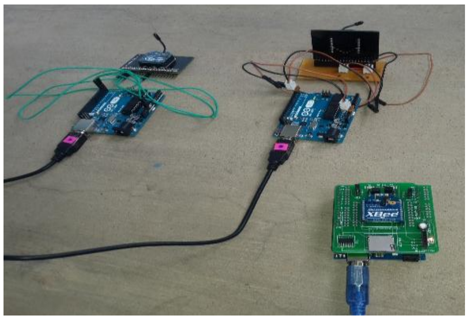

Fig.6. A Platoon 1 with 3 nodes

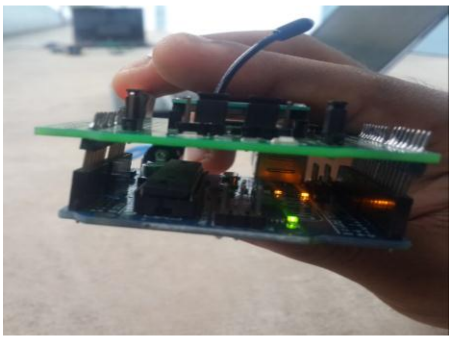

Fig.7. RSU communicate with Platoons.

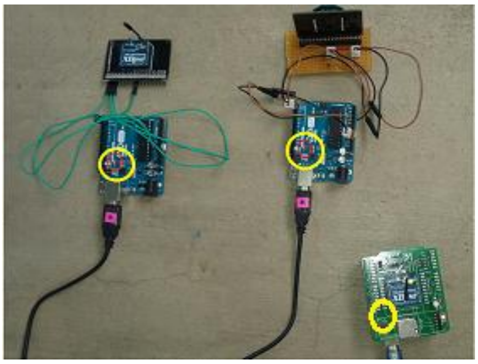

Fig.8. LED glows in platoon 2 after receiving message from platoon 1 via RSU

The Arduino UNO in the node of the first platoon generates an information, which it sends to the RSU through the XBEE attached to it. Now the XBEE in the RSU receives it and the Arduino UNO in the RSU reads the information from its serial monitor. Now this information is sent to the second platoon via XBEE attached to the RSU. The XBEE attached to the nodes in the second platoon hence receives the information sent by the RSU and delivers it to the Arduino UNO boards attached to the nodes in the platoon 2. This Arduino UNO boards process the information and upon the truth of a test condition the internal LED (Pin 13) glows.

F. Simulation results

There are numerous network simulators available as free software over the Internet. Out of these simulators, Network Simulator 2 (NS-2) is found to be commonly used by researchers to simulate various network environments. In this simulation, 10 nodes has been considered, out of which 4 nodes (node 6, node 7, node 8 and node 9) act as RSUs. The other 6 nodes are divided into 2 platoons (3 nodes in each platoon). Platoons are a group of vehicles which communicate among themselves and with the RSUs. The Adhoc on Demand Distance Vector Routing (AODV) is used as the routing protocol. In this type of routing, a route between the source node and destination node is created based on demand by broadcasting a route request from the source node. The nodes which have a route to the destination node sends a route reply and backward pointers are setup from this point of time. These backward pointers are finally used as routes to send the actual data to the destination node from the source node.

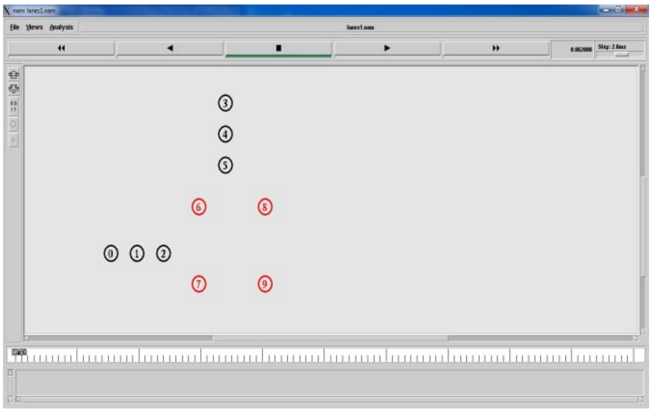

Fig.9. Initial position of Platoons and RSUs

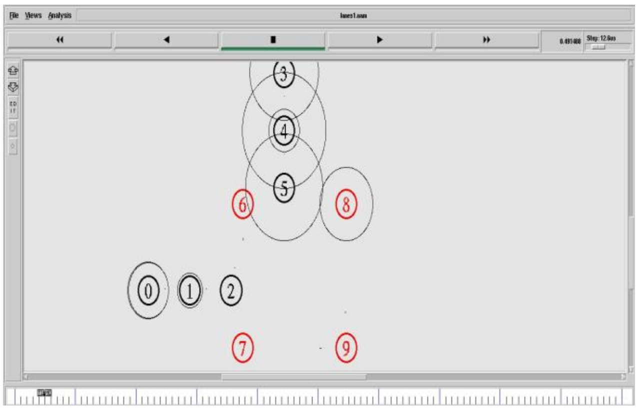

Fig.10. Communication between Platoons and RSUs

The two platoons start moving towards the junction at constant speeds. However, as and when they move, communication is established between vehicles (V2V) in a given platoon and also between the vehicles and the nearby Road Side Units (RSUs).

The platoon 1 (with Nodes 0, 1 and 2) starts its communication with the nearby RSUs (RSUs 6 and 7). Meanwhile the platoon 2 (with Nodes 3, 4 and 5) starts its communication with its own nearby RSUs (RSUs 6 and 8). Thus RSU gets to know the position of vehicles in both platoons and hence sends the information of platoon 2 to platoon 1 and vice versa.

Now platoon 1 sends data to RSU 6 that it will move first and requests platoon 2 to stop. RSU 6 transmits this information to platoon 2. Thus platoon 2 waits until platoon 1 crosses the junction. Moreover, as and when the platoon 1 moves across the junction it starts it communication with the RSUs 8 and 9 as and when it comes in the range of the RSUs. The four RSUs viz RSU 6, RSU 7, RSU 8, RSU9 keep communicating continuously throughout the simulation time.

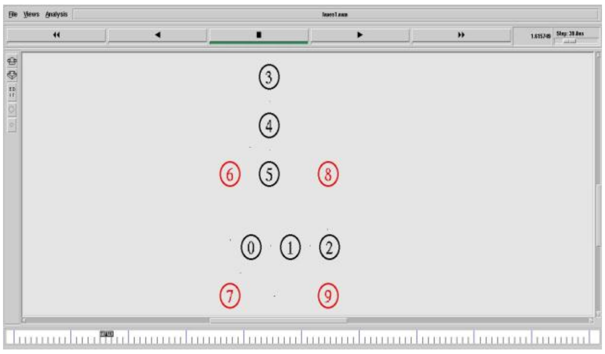

Fig.11. Platoon 2 waiting for Platoon 1 to cross the junction

After platoon 2 receives the information that platoon 1 has crossed the junction, they start moving against their destination. As and when they move towards their destination, platoon 2 starts communicating with 7 and 9 when they come within the communication zone.

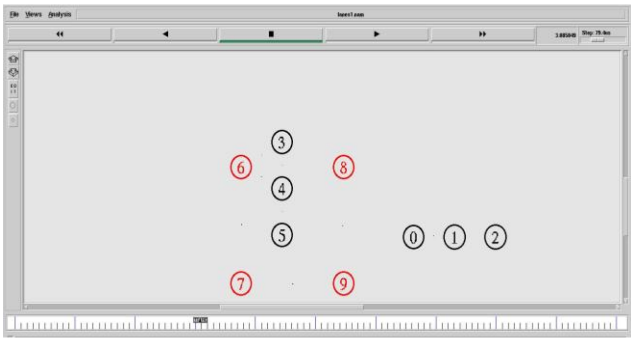

Fig.12. Platoon 2 starts moving after Platoon 1 to cross the Junction

Finally both platoons have crossed the junction safely. However, there is still communication between vehicles (nodes) within a platoon and between Road Side Units (RSUs).

CONCLUSION

In this paper, a design of Vehicular Adhoc Networking (VANET) is presented which results with more coordinated traffic flow control. The performance have been tested through simulation and also implemented in hardware. An algorithm for dynamic node addition in case of expansion of the platoon and node deletion within the range of RSU has also been validated. This system requires navigation of vehicles for continuous communication between each node in a platoon and also between platoons through RSUs which has also been validated through a laboratory setup. The proposed system design can be expanded to any number of nodes in the real time ITS.

REFERENCES

[1] Georgios Karagiannis et al, "Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions", IEEE Communications Surveys & Tutorials, Vol. 13, No. 4, Fourth Quarter 2011 pp.584–616.

[2] BaraaT.Sharef, Raed A.Alsaqour, Mahamod Ismail, "Vehicular communication ad hoc routing protocols: A survey", Journal of Network and Computer Applications (2014) pp.363–396.

[3] I.F. Akyildiz, W. Su*, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey", Elsevier Science, Computer Networks (2002) pp.393–422.

[4] Fan Li and Yu Wang, "Routing in Vehicular Ad Hoc Networks: A Survey", IEEE Vehicular Technology Magazine, June 2007, pp.12-22.

[5] Yun-Wei Lin, Yuh-Shyan Chen and Sing-Ling Lee, "Routing Protocols in Vehicular Ad Hoc Networks: A Survey and Future Perspectives", Journal of Information Science and Engineering 2010, pp.913-932.

[6] Khaleel et.al, "ROAMER: Roadside Units as message routers in VANETs," Elsevier, Adhoc Networks, 2012, pp.479-496.

[7] Pooja Kavathekar, YangQuan Chen, "Draft: Vehicle Platooning: A Brief Survey and Categorization" IDETC/CIE 2011, August 29-31, 2011, Washington, DC, USA

[8] Shyamakshi Ghosh, Shrisha Rao, " Sensor Network Design for smart Highways", 5th Annual IEEE conference on Automation Science and Engineering, 2009, pp353-360.