Master's Thesis

Reliable Data Dissemination for Car Safety Application in VANET

By

Padmanaban Elumalai
Prabhakaran Murukanantham

Department of Electrical and Information Technology
Faculty of Engineering, LTH, Lund University
SE-221 00 Lund, Sweden

Abstract

In our master thesis work we have investigated and simulated a broadcast algorithm for wireless communication between vehicles, known as Vehicular Ad-hoc network . The main aim of Vehicular Ad-hoc Network is to enable dissemination of safety warnings and traffic information as detected by independently moving vehicles. Reliable dissemination of data in VANET will improve the quality of driving in terms of safety and time. This Master thesis work includes comparative study of different reliable data dissemination methods and then we have also implemented two data dissemination algorithms called The last one and Adaptive probability alert algorithm using NS-3 and our main aim is to evaluate the reliability of the algorithm. Hence the results are using three parameters called Success ratio, Normalized Throughput of received packet and Propagation Time. A simulation result shows that Adaptive probability alert algorithm gives better performance results compared to The Last One in terms of Success ratio and Propagation Time.

Acknowledgments

We are extremely grateful to our Supervisor *Kaan Bür* at Lund University, School of Electrical and information technology for his guidance throughout various stages of our Master thesis and his painstaking efforts in proof reading our drafts are greatly appreciated. Also would like to extend our Heartfelt thanks to our Examiner *Maria Kihl* for giving a positive feedback to our report. Indeed only with both of their support this Master thesis made possible. Thanks to both of you.

Prabhakaran Muruganantham and Padmanaban Elumalai

I owe my deepest gratitude to my friend *Saravana Kumar Sankaramoorthy* for guiding and supporting me throughout entire thesis. My Thesis would have been a dream had it not been for his relentless support. i wish to thank my childhood friends *Siddharth, vinoth* for their continuous moral support and encouragement towards me in completing this thesis. And I am indebted to many of my friends who always stood by my side at difficult times and always would be grateful to them.

Last but not least, I would like to thank my parents for their unconditional support, both financially and emotionally throughout my degree. In particular, the patience and understanding shown by my mother, dad and brother during the entire Masters is greatly appreciated.

Prabhakaran Muruganantham

This thesis is dedicated to my parents *Ealumalai* and *Rajeswari* and my brother *Athimoolam Elumalai* who have given me the opportunity of an education from the best institutions and support throughout my life.

Padmanaban Elumalai

Table of Contents

CHAPTER 1

1 Introduction

In the **Vehicular Ad Hoc Network (VANETs)**, vehicles form a self-organised network without the help of a permanent infrastructure network. Data dissemination should be established efficiently between the communicating nodes for safety applications. Data dissemination is a process of broadcasting data packets over a distributed wireless network. While during broadcasting, the number of vehicles equipped with computing technologies and wireless communication devices is going to increase dramatically. This can enable a wide range of applications, such as emergency message dissemination, collision avoidance, dynamic route scheduling, and various kind of entertainment application. However, it is very important to consider several parameters when approaching any kind of broadcast in a VANET because nodes are not fixed but can move also in this scenario. Unlike the mobile ad-hoc networks (MANETs), where nodes can freely move in a certain area VANET’s vehicles' movements are constrained by streets, traffic and specific rules.

The main problem of VANET is to exchange the information in a scalable fashion [1] and this can be done either by using a proactive method i.e. broadcast method (push model) or by on demand (pull model) method. The main objective of data push model is to share the information between vehicles frequently in order to assess the traffic conditions ahead of it. Two main types used to achieve the objective are flooding and routing (including multi and broadcast). In flooding, information is broadcasted periodically by each individual vehicle.

In this work, we will give an overview of different techniques used in solving the problems faced by the data dissemination in VANET i.e. the different types of broadcasting algorithms used to enhance the data dissemination in VANET. The main objective of this research work is to study in detail about two rival broadcast algorithms namely TLO (The Last One) and APAL (Adaptive Probability Algorithm). The nature, operation, performance of these 2 algorithms is discussed and evaluated here.

1.1 *Layout of thesis*

Following is a brief introduction to the chapters discussed in this report.

Chapter 2: Background

A clear introduction to VANET, and the differences and similarities between VANET Vs MANET, unique features of VANET and VANET applications such as safety applications and user applications are discussed.

Chapter 3: Related work

The different types of broadcast algorithms in VANET such as flooding based algorithms, probability based algorithms, counter based algorithms and position based algorithms are discussed in chapter 3.

Chapter 4: Two Competing Algorithms

A detailed explanation of the TLO and APAL algorithm is given in this chapter. An illustration of the algorithm and the programmable implementation of the algorithm are also discussed.

Chapter 5: Simulation setup and Results

The NS3 simulator which is used in the evaluation of TLO and APAL algorithms, the simulation model of NS3 simulator and classes used in NS3 for evaluating this algorithm are discussed. The simulation settings used for the NS3 simulator are the 'packet size description', 'physical layer parameters' and 'road side dimensions'. Also in this section we have compared the results of these two algorithms with parameters such as success ratio, throughput of received and sent packets, and propagation time.

Chapter 6: Conclusion and future work

The results obtained in the chapter 5(Simulation setup and results) and the performance of the two algorithms are discussed and analysed. Also improvements to improve and enhance the performance of the algorithms in the future are suggested.

1.2 *Individual contributions*

Both participants have provided vital contributions in analysing and testing the discussed algorithms. The maximum inputs in analysing the TLO & APAL algorithms were provided by Mr.Padmanaban & Mr. Prabhakaran respectively. And for the testing part, there was an equal share of contributions from both participants.

CHAPTER 2

2 Background

In this section, a clear background of VANET is discussed and the differences and similarities about VANET & MANET are compared, also unique features of VANET, VANET applications namely safety applications and user applications are discussed.

2.1 *What is VANET?*

VANET (vehicular ad-hoc networks) is a wireless network that is formed between vehicles on an as needed basis[1]. Vehicles must be equipped with wireless transceivers and computerised control modules in order to participate in VANET and acts as network nodes.

VANET is also called as inter-vehicle communications (IVC) or vehicle to vehicle (V2V) communications [2]. VANET turns each participating vehicles into a wireless router or nodes to connect to form a wide network, since each individual node's wireless network range may be limited to few metres, so by hopping through several nodes it provides end to end communication across longer distances[1].

VANET does not require any network infrastructure, although it can use ad-hoc network infrastructure as roadside units. These roadside units can serve as a wide range of applications like serving geographical data, acts as a gateway to internet and can serve as a drop point for messages in highly populated roads.

Hence VANET is a special subset of Mobile ad-hoc networks and can be formed either with vehicles and infrastructure communication or vehicles with vehicle to vehicle (V2V) communication as shown in fig 1 [3].

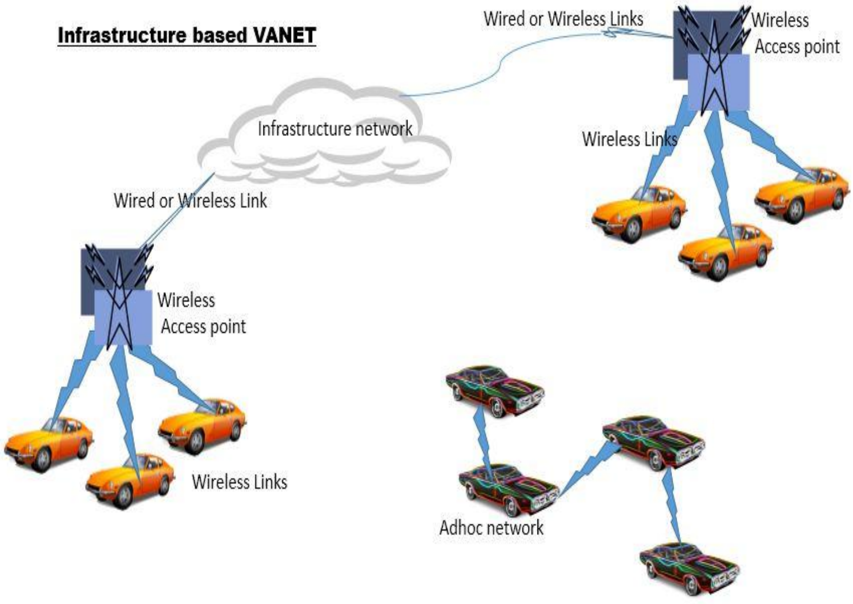


Fig.1. Two basic kinds of VANET, infrastructure based ad ad-hoc based [3].

2.2 *VANET vs. MANET*

VANET is a subset of Mobile ad-hoc networks (MANET). MANETs consist of both mobile, semi-mobile nodes and it does not have any pre-established infrastructure. They establish multi-hop routes and they connect in a self-organising decentralised manner[4]. The main property which distinguishes MANET from VANET is that the nodes in VANET are larger in number and moves in high average speed. Even though many unicast routing protocols[5][6] have been created for MANET, it cannot be directly implemented on VANET due to its unique characteristics. It is unrealistic and very expensive to install 802.11 access points to cover all the road area, and any nodes can be easily deployable without the involvement of infrastructure, which makes vehicular ad-hoc networks highly favourable in vehicular environments.

Like MANET, VANETs self-organize and self-manage information in a distributed fashion without a server dictating the communication norms or the presence of a centralized authority. In VANET, nodes employ themselves as both servers and clients, thereby exchanging and sharing information like peers. Moreover, nodes are mobile, thus making data transmission less reliable and suboptimal.

2.3 *MANET Characteristics*

Dynamic topologies:

Nodes are free to move randomly and may change randomly and rapidly at irregular times and may consist of both bidirectional and unidirectional links.

Controlled-Bandwidth and variable capacity links:

Wireless links have considerably a lower capacity than their hardwired links and the throughput of wireless communications is less than a radio's maximum transmission rate. As the mobile network is often an extension of the fixed network infrastructure, MANET users will demand similar services. These demands will continue to increase as multimedia computing and collaborative networking applications rise.

Energy-constrained operation:

Most of the nodes in a MANET may rely on batteries or other exhaustible means for their energy. The most important system design criteria for optimization may be energy conservation for these nodes.

Restricted physical security:

Comparatively mobile networks have more physical security threats than the infrastructure based networks. To reduce physical security threats, some of the existing link security techniques are often applied within mobile networks. This gives the decentralized nature of network control in MANETs and enhances them to be robust against the single points of failure of more centralized approaches.

2.4 *Unique characteristics of VANET*

Even though Vehicular ad-hoc network (VANET) share some common characteristics with normal ad-hoc networks like decentralisation of infrastructure networks, self-organising but it still differs from them due to the changes in the design of the communication system and protocol security which are impacted by its unique challenges. Some of the unique characteristics of VANET[4] are as follows ,

High number of nodes:

Since VANET is the technical base for ITS (Intelligent transportation system), a large number of vehicles with communication capabilities is required to deal with potentially high number of nodes and in addition to that, it also deals with the potential road side units which makes VANET to deal with larger number of nodes.

Mobility of nodes:

A VANET node moves with high mobility and in some cases when vehicles cross each other they have less time duration for exchange of packets and in certain cases the movement is restricted by the road structure and traffic rules.

Reliable and timely delivery:

Since VANET's application is mainly to avoid accidents and to save lives, it requires highly realistic and reliable application for data delivery as even a small end to end delay can cause the safety message meaningless.

No confidentiality:

For safety applications, the information contained in the alert message is available and applicable for all vehicles in the road as the main aim is to avoid accidents; hence there is no confidentiality in the messages.

Bandwidth issue:

The bandwidth issue **[7]** is highly aggregated due to the traffic jam, intersections between roads especially in a high dense urban traffic. VANET tends to grow on a large scale. Let us consider a normal scenario, inter- vehicular distance is just 75 m and within a radius of 1km, we have around 70 vehicles around a given car. When a traffic jam occurs, with inter vehicular distance of around 5-10 m there will be around more than 1000 vehicles within 1 km radius.

2.5 *VANET APPLICATIONS*

VANET applications can generally be categorised into two types namely safety applications and user applications. The former deals with the safety of vehicles and the latter deals with the value added services like for example entertainment [3].

2.5.1 Safety applications

Safety applications can play a significant role in reducing the number of accidents. With reference to a study[8], more than 50 percent of the accidents can be avoided if the driver is informed with a warning half a second before the moment of accident. Safety applications play a vital role in 3 major scenarios, which are as follows

Accidents warning

Vehicles on road generally travel at very high speed which gives the driver only a short span of time to react to vehicles in front of them. As per the results presented in [9], the driver –perception response time of 95 percentile of people is 1.6s, and as the results shown in [10] indicates that this reaction time is very less to avoid an accident in many emergency cases, especially when the driver cannot see the vehicles upfront due to some bad weather condition, sharp hair pin bends and when they violate traffic rules etc.

The safety applications warns the driver early by giving a alert message of an accident occurred further ahead of the road, thus preventing the accident by giving some extra time for the driver to react.

Intersections warning

The possibility of accidents is very high as two or more traffic flows intersect in junctions which makes it a complex challenge for the driver. According to the latest reports from Department of Transportation from United States, in 2009, total fatalities across the nation are around 33,808. Out of the total fatalities occurred, fatalities caused in intersections are around 7043 which is contributes to 21% of the total fatalities[29]. Large number of accidents could be prevented if the driver is warned early by the safety application.

Road Congestion warning

Safety applications also provides the best route to the drivers to their destinations and also decreases road congestion, ensures smooth traffic flow and in turn prevents traffic jams**[7]**. It eases the driver’s job by providing a clear mind-set and makes them less frustrated and in turn indirectly it reduces the number of accidents.

Passive safety applications

Passive safety applications work inside the vehicles and protect the passenger from injuries during the accident occurrence **[11]**. Air bags and safety belts are some examples of passive safety applications. It doesn’t help to avoid accidents but it is quite useful in case of criminal attack, accidents and it finds the exact locations of the people affected and provides assisstance to them effectively. Post crash emergency applications are an effective subset of passive applications

2.5.2 User applications

User applications can provide road users some valuable information, entertainment services, advertisements etc. Some applications are related only to user entertainment and cannot be linked to safety applications. The main role of the user applications is to comfort the passengers, improving traffic system, adding entertainment but make sure that, it doesn't affect the safety applications. Some of the examples of the user applications are discussed below.

Parking availability services

By providing a clear picture on a empty parking slot in a specific region, provides safety and saves time by finding an exact place to park your vehicle in crowded places like shopping mall, restaurants, theatres etc.

Internet connectivity

Nowadays constant internet connectivity is unavoidable for many users and vehicle occupants as it is the backbone for many other VANET applications to function properly **[7]**. This also means that without a requirement of a specific re-development, usual business framework should be seamlessly present in the vehicle.

Other entertainment applications

In order to alleviate boredom, especially for the vehicle occupants, some of the entertainment applications like sharing the movies, music, chatting with each other could be done during their long journeys.

CHAPTER 3

3 Related Work

In VANET, broadcasting is the most effective way to disseminate the alert or the warning messages in the accident situations [12]. Applying traditional ad-hoc algorithms for VANET will degrade the performance due to VANETs unique features. Researchers have proposed many broadcast algorithms to suit the reliability requirements of VANET.

Broadcasting techniques used in vehicular ad-hoc networks and mobile networks are generally classified as:

1) Flooding based algorithms
2) Probability based algorithms
3) Counter based algorithms
4) Position based algorithms

3.1 Flooding based algorithms

Flooding is a method to transfer a packet of data from a source to one or multiple destinations over a network topology of a VANET[13]. Flooding method is relatively simple compared to other algorithms and especially suited when the network topology changes frequently like scenarios of high speed nodes [12].

3.1.1 Simple flooding

The first method is the simple flooding algorithm[14]. Once if a packet is generated by the node, it is then relayed by each node to the next node as it receives the packet. In order to avoid the problem of multiple processing and relaying, each node contains a duplicate table, which contains the list of all Ids of the packets received during the predefined waiting time. This method is usually used by the reactive routing protocols to find its effective route to the destination.

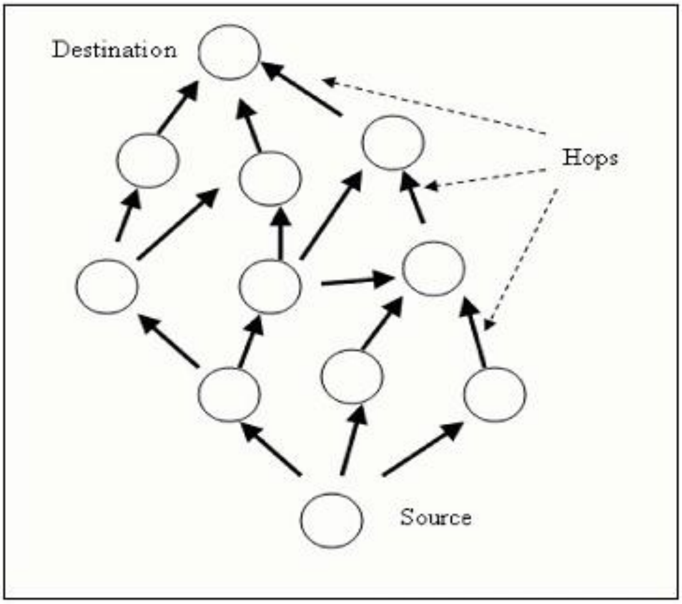


Fig. 2. Simple flooding in networks [13]

3.1.2 Geography aware flooding algorithm

The next method is the geography aware flooding algorithm. This algorithm is a direction aware flooding algorithm and was designed for the vehicle environment based on [10].

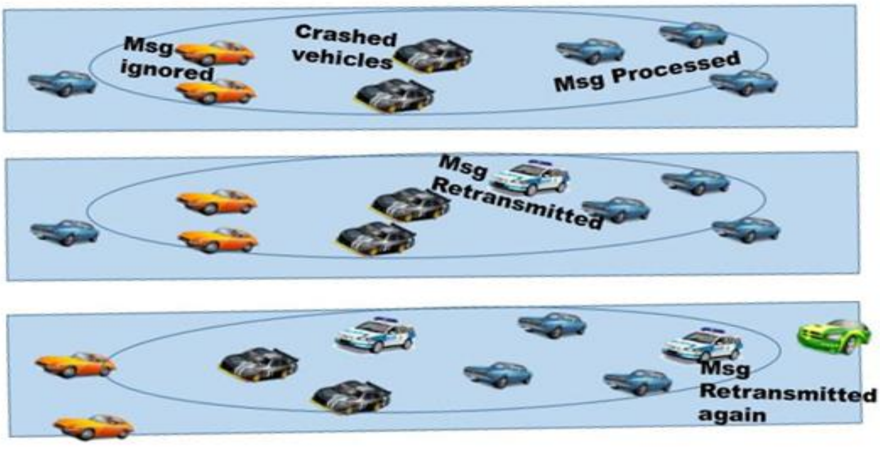


Fig. 3. Geographical aware flooding algorithm [14].

First when the vehicle receives the message, it checks the direction of the message, if it is from the back, it ignores the message. If the message is received from the front, it checks whether it has received the same message before or not. If it has received the same message before, it ignores the message and if it receives the message for the first time, the vehicle decelerates and waits for a random duration to check if any message has been received from behind for the same event. If it didn't receive any message from behind during the waiting period it retransmits the message to the next hop.

3.1.3 Optimised linked state routing (OLSR)

The next algorithm is the optimised linked state routing which is an proactive protocol that inherits the link state routing algorithm and has the advantage of having the routes immediately. It reduces the size of the control packets by only declaring the subset of links within its neighbours called Multipoint selectors[30]. Flooding has been optimised using the selected nodes called the Multi point relays(MPR) as only the multipoint relays of the node retransmits the message[30]. This method highly reduces the amount of retransmissions in a flooding related algorithm.

OLSR algorithm is highly related to table which has to regularly maintain the updates about receiving information and time period during which the information is valid [15]. Only based upon these tables, all route calculations are done. With the usage of MPR optimisations, OLSR defines a default mechanism for forwarding the messages.

So whenever the node broadcasts its packet to its neighbour, only the MPR's of the node relay the packet to the two hop neighbours and it continues the same way. Each node maintains a duplicate table where the sequence numbers of all received packets are stored. The relay is carried further only during their first reception and it discards the other receptions. And also the node doesn't relay, if it receives the packet from the non-MPR selector node for the first time [30]. However these flooding methods, network sources are wasted and each packet is transmitted multiple times and it creates a problem known as Broadcast storm.

3.1.4 Broadcast Storm Problem

In traditional simple flooding method, a node re-broadcasts immediately and provides a very high speed of data dissemination[16]. However it doesn't have good performance in dense and sparse areas. Especially when we have traffic jam during the rush hours of dense and sparse areas, it costs 'n' transmissions in a network of *n* hosts which may lead to following problems [31].

Redundant rebroadcasts:

When a node decides to re-broadcast its message to its neighbours and if the neighbours have already received the same message, then it is called redundant rebroadcast.

Contention:

When a node decides to re-broadcast the message and at the same time the neighbours also decides to re-broadcast the message, here they contend with each other.

Collision:

When 2 or more nodes attempts to re-broadcast the message at the same time, collision of data occurs, which leads to loss of data and affects the performance of the network and leads to low reliability with a lot of redundant broadcast messages.

Collectively we refer the above problems as broadcast storm.

3.2 Probability based algorithms

In order to suppress the broadcast storm problem, basic broadcast technique follows either a 1-persistence or p-persistence rule[17] most routing protocols designed for multi hop ad-hoc networks follow the exhaustive search strategy of 1-persistence flooding rule, due to its low complexity and high penetration rate, it requires all nodes to retransmit the packet with probability 1.

On the other side some algorithms which follows the p-persistence flooding rule requires the node to retransmit with a predetermined probability p which is also known as probabilistic flooding. Repeated reception of the messages is discarded in both the schemes.
In the following we discuss three different broadcast schemes, in all these methods only based on its local information, each node calculates its own rebroadcasting node.

3.2.1 Weighted p-persistence technique

In this method [17], when a node j receives a message from a node i , it checks the message and retransmits with the forwarding probability P_{ij} and it discards the repeated reception of the messages and accepts only if the message received during the first time.

The forwarding probability is calculated using the expression

$$P_{ij} = D_{ij} /R.$$

If a node j receives the duplicate messages before retransmission from various sources during its waiting time (wait-Time), it selects the smallest Pij value as its re-forwarding probability .i.e. each node should use the relative distance with the nearest ones to ensure, the nodes which are farther away should transmit with higher probability. If node j decides not to rebroadcast, then it has to buffer for some extra additional waiting time (wait-Time +δ ms), where δ -one hop transmission and propagation delay and (δ< wait-Time).if node j does not receive any message from its neighbours during the waiting time (wait-Time+ δ), it re-broadcasts with probability 1, in order to ensure the 100 percent guarantee of the message reachability and to prevent message die out.

3.2.2 Slotted 1-persistence technique

When node j receives a message from node i, it checks the message and retransmits with the forwarding probability 1 at the assigned time slot T_{sij} and it discards the repeated reception of the messages and accepts only if the message received during the first time and it should not have received before its assigned time slot.

Some of the terms which are used in this technique are,

D_{ij} - Relative distance between nodes i and j
S_{ij} - assigned slot number
R- Average transmission range.
Ns-predetermined number of slots.
τ – Estimated one hop delay.

With the given terms, T_{sij} can be calculated as,

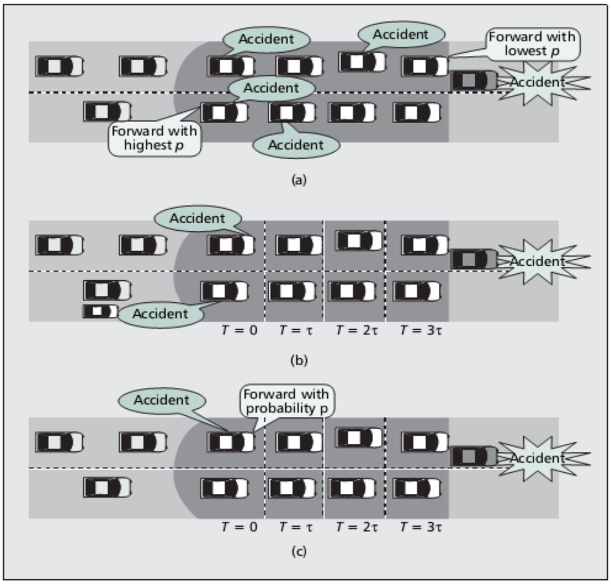
$$T_{sij} = S_{ij} *\tau.$$

And S_{ij} can be expressed as,

$$S_{ij} =Ns (1-[min (D_{ij}, R]/R)$$

Slotted 1 persistence scheme is similar to the persistence broadcast scheme, but it uses the GPS information to calculate its waiting time instead of calculating the re-forwarding probability. For example in fig 2b, broadcast coverage area is divided into four regions, node located in a farther region will be assigned the shorter waiting time. When the node receives the duplicate messages from multiple senders, it selects the one with the smallest D_{ij} value.

This method highly requires the transmission range information to decide on a certain value of slot size and the number of slots similar to the persistence scheme method. Ns, the design parameter and the function of traffic density is carefully chosen such that higher the traffic density, smaller the slot size, larger the number of slots. Network designer should fix this to value over time since it’s hard for each vehicle to predict the traffic density.



Fig, 4, Slotted 1 persistence scheme [17].

3.2.3 Slotted P-persistence technique

When node j receives a message from node i, it checks the message and retransmits with the predetermined forwarding probability P_{ij} at the assigned time slot T_{sij} and it discards the repeated reception of the messages and accepts only if the message received during the first time and it should not have received before its assigned time slot.

Each node in this method has to buffer the message for certain waiting time like ([Ns-1]*wait-time+ δ ms), and during this waiting time if it doesn't hear any re-broadcast message from the neighbours, it re-broadcasts with the probability of 1 to prevent the dying out of the message. Similar to the p-persistence scheme the performance of this method highly depends on the chosen re-forwarding probability P, the concept of slotted p-persistence method with four slots is clearly illustrated in fig 2c.

3.3 Position Based Algorithms

Position based algorithms perform well especially in highly dynamic regions[18] , routing decisions are based on the geographical coordinates of the node. Some of the examples of position based algorithms are TLO[19], DREAM[32], SIFT[20], FACE-2[21]. The geographical neighbour's location is updated in a table by each node.

When nodes move, they have to inform about their new position to the neighbours by sending a control message in order to maintain the nodes up to date. Position methods reduces the control overhead since the amount of data needed to build a link is larger than the amount of data needed to build a location table [20]. In this section, some of the examples of position based algorithms are briefly discussed, which are as follows

3.3.1 DREAM

Dream algorithm is directional, restricted flooding and a position based algorithm. It uses several techniques to reduce the control overhead rate[20]. In DREAM, two algorithms are used.

The first one is used to disseminate the location information packets and the second one is to disseminate the data packets. The former algorithm which deals with the dissemination of the location packets, is strictly a restricted flooding algorithm, in which each node periodically floods location packets to other neighbour nodes which are in its range (i.e. by introducing a distance threshold) in order to update the location .

The flood is also restricted by controlling the frequency of the location updates sent by the nodes. This frequency at which the location updates is sent by the nodes is directly proportional to the mobility rate of the nodes, higher the speed of the nodes, more frequent the updates.

The second algorithm which deals with dissemination of data packets is directional flooding type. Here if a node S wants to send the packet to node D, then node S looks the position of D from the location table and based on the information S selects the neighbours of the node D which are in the same direction and the forwards the packets in the same direction of node D. This process is repeated by each node until it reaches the node D. This algorithm is a directional one since the flooding is done only in small sectors which are in the radio range of node S, and the sector lies in the direction of node D. As per the simulation results of [33], DREAM is able to reduce the sum of bandwidth and transmission power compared to simple flooding and it assures that DREAM can find a route to the destination 80 percent of the times.

3.3.2 Reactive Location Services

In this method, if a node S needs the position information of node D, node S floods a request containing the id of the node D and it also contains the id and position of node S. So when node D receives the request with its own id, it replies to the node S which is present in the request [18].

An expanding ring search is performed in order to reduce the range, first the flooding starts with a range of 2 hops, when it doesn’t receive any response during a certain time period, the flooding is repeated with a greater range and it is increased linearly or exponentially.

With the reactive location services method the main part is only the overhead compared to the payload data. With the existing location services it would produce an overhead data which is not directly related to the payload data. Thus results highly depend on the transmitted payload data.

Clearly the overhead data will be generally high in reactive location services when the communication partners are changed frequently which makes reactive location services much inferior to the other existing location services and it can be optimised using caching and petition of nodes future location based on its heading and speed.

3.3.3 **Greedy Perimeter Stateless Routing (GPSR)**

In greedy perimeter stateless routing method, with the help of beacons the node knows the position of its neighbours and with the help of the location services the node knows their position of packets destination.

With the help of information provided by beacons and the location services, nodes forwards the incoming packets to the neighbour nodes which are in the direction of the destination
This process is repeated until it reaches the destination node. But unfortunately the node represents a local optimum and it is the closer node

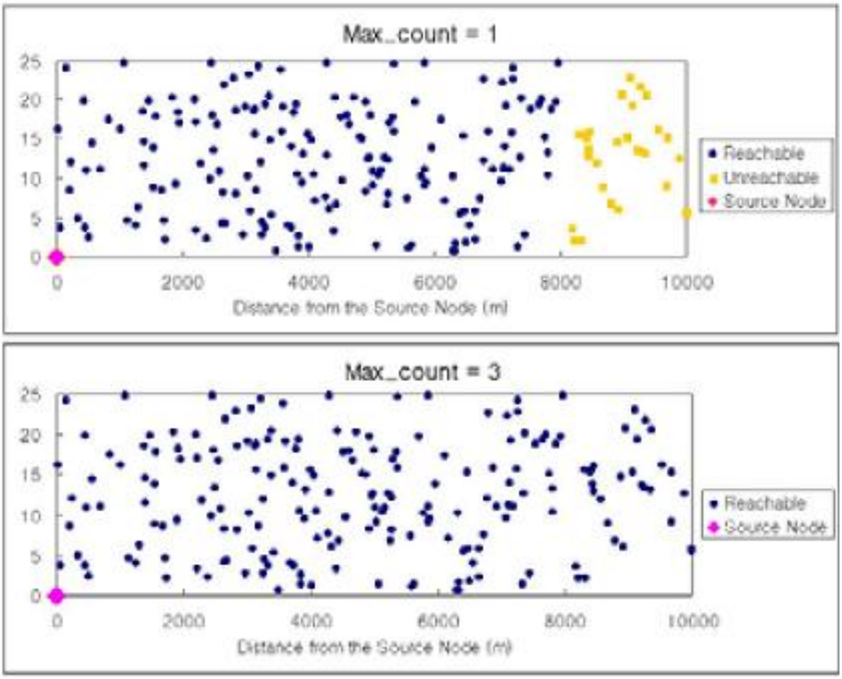
to the destination than any other neighbour node. In this case, an Algorithm named *Perimeter routing* is used to come out from the local optimum by using the planar graph traversal method.

3.4 Counter Based Algorithms

The counter based method is an variant of a probability based method [22]. The counter based algorithms takes the help of network usage into account in order to forward the packets to other nodes, while in probability based method, regardless of network status it makes a probabilistic choice on packet forwarding to the nodes.

In the counter based method, each node has been set a timer for each non duplicate message it receives. Each time when a node receives non duplicate message, delay time for each timer is set randomly and decremented afterwards.

Whenever a node overhears the duplicate messages from its neighbours, the counter gets increased. If the counter exceeds the threshold (*Max-Count*) when the timer expires, then the node discards the packet and suppresses the forwarding in order to prevent the repetition of packets. Thus counter based method is more robust in various network wide broadcasting scenarios due to its adaptive ability in controlling the probability based packet forwarding in conjunction with the node density[23], but it cannot completely eliminate the redundancy of the forwarding[22]. In the below figure, the connectivity of the network for various *Max-counter* values is compared in the counter based method.



Fig, 5, explains the connectivity of the network of counter based method with Max-counter values [22].

The nodes which has received the emergency packets is marked as reachable nodes, and the nodes which did not receive the emergency packets is marked as non -reachable node, from the above figure it is clearly evident that, larger the *Max-counter* value, better the connectivity of the network but it also increases the redundant packet transmission of the network.

CHAPTER 4

4 Two Competing Algorithms

In order to evaluate the efficiency and reliability of broadcast algorithms, two competing algorithms namely TLO (The Last One) and APAL (Adaptive Probability Algorithm) has been chosen and evaluated. The nature of two algorithms TLO and APAL is different as the former is a position based algorithm and the latter one is a probability based algorithm. TLO algorithm is highly dependent on position updates of the fellow neighbour nodes whereas APAL is independent of the position updates.

4.1 TLO algorithm

TLO(The Last One) [19] is a position based algorithms which provide a proper data dissemination in an ad-hoc network. TLO algorithm provides an effective solution to reduce end to end delay and broadcast storm problem.

In TLO, there are a series of assumptions to be made which makes the algorithm yield better results. In this algorithm it is assumed that every node in the network is completely equipped with GPS, i.e. each vehicle knows the exact geographical location of the other vehicles which are in the communication range. Frequent updating of information should take place at closer intervals. Relative velocity between vehicles should change slowly in order to have longer updating interval.

4.1.1 Explanation

Whenever an accident takes place, the accident vehicle broadcasts an alert message to all the other vehicles within its communication range. So all the vehicles within the communication range of the accident vehicle will be receiving the alert message, after receiving the message it will not rebroadcast it immediately. It will wait for some time and will perform TLO algorithm and it will select the last vehicle within that particular communication range.

Now comes the core part of the TLO algorithm, during the threshold waiting time, each vehicle updates their own location and the location of neighbours and it is updated as in table 1.

TABLE 1 : NEIGHBOUR LOCATION TABLE

Vehicle index	Vehicle location from start of the road in metres.	Calculated distance from the emergency initiator(vehicle A) in metres
Vehicle A	50	0
Vehicle B	100	50
Vehicle C	150	100
Vehicle D	200	150
Vehicle E	250	200

With the updated Table, each vehicle calculates their distance from the emergency initiator vehicle or the repeater vehicle (from accident table) and also it compares their neighbour distances from the emergency initiator vehicles. Each vehicle which receives the alert message does this operation, the vehicle with the largest difference in the distance from the emergency initiator is considered as the last one.

From the table it is clear that, vehicle E (250 m from the start of the road segment), is 200(250-50) m farther away from the emergency initiator (vehicle A (50 m from the start of the road segment)).The other vehicles B, C, D are 50, 100, 150m far away from the emergency initiator vehicle A. Now when the vehicles compare their difference of their own position to the emergency initiator position, it is clear that Vehicle E with difference of 200 m from the emergency initiator position is considered to be the last one. So the Vehicles B, C, D will know that Vehicle E is the last one, the farthest one from the emergency initiator and also within the transmission range of Accident Vehicle.

Now that last vehicle which was selected by the TLO algorithm will retransmit the alert message to the next set of vehicles and when the threshold waiting time expires and the other nodes didn't receive the alert message, it thinks that there is no relay node behind them and TLO is run again to find the next last node. Until a successful broadcast is carried out, this process is repeated. The below figure shows the pseudo code of TLO algorithm,

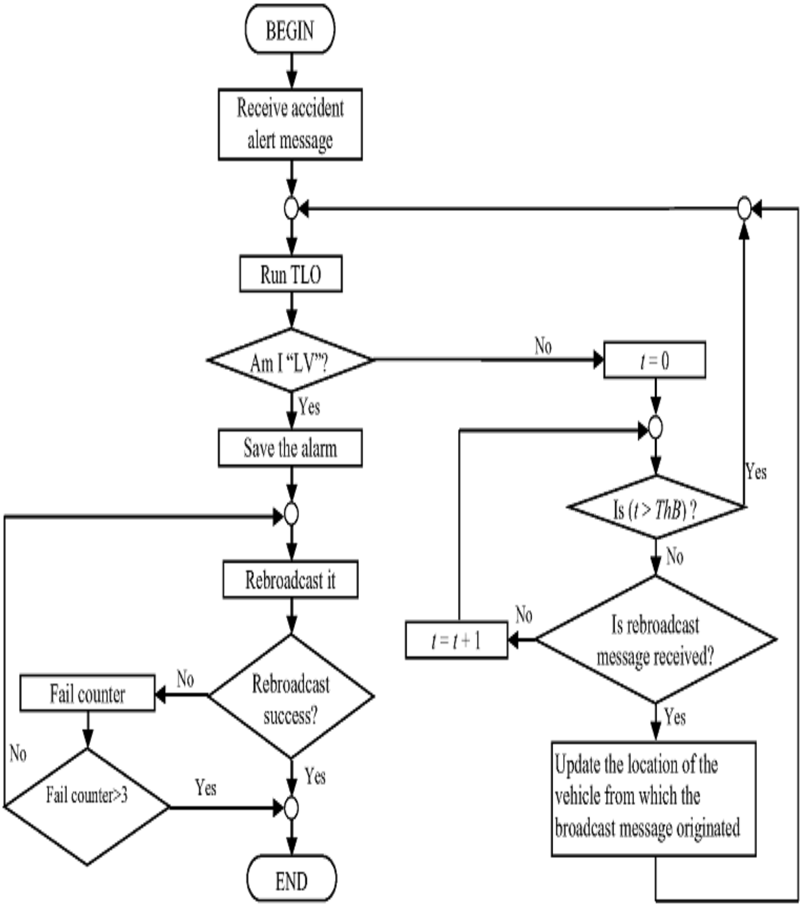


Fig.6. Flow chart of TLO algorithm [19].

4.1.2 Illustration

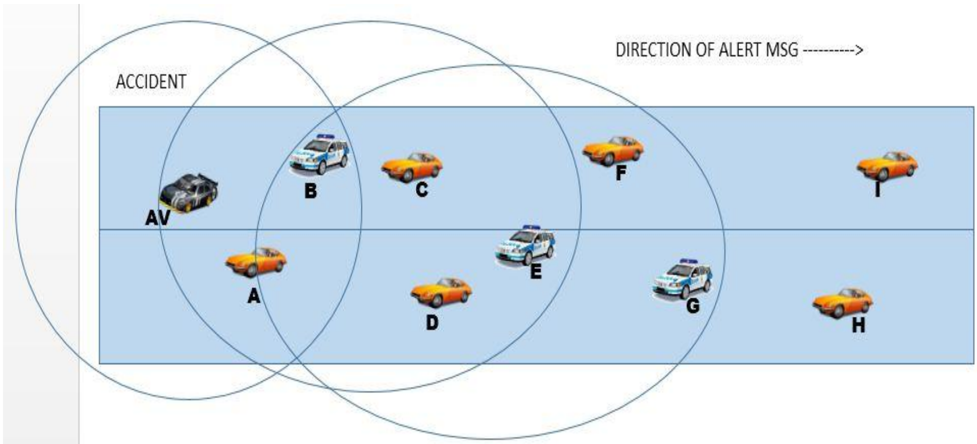


Fig. 7. Diagrammatic Explanation of TLO algorithm[19]

Let us consider the above figure as an example of an emergency scenario, let us consider AV is the accident vehicle and now AV needs to inform other vehicles coming along its way. AV sends an alert message to all the vehicles within its range, from the above figure it is clear that Vehicles A and B are in the range of the accident vehicle AV. So now vehicles A and B have received the alert message from the accident vehicle.

Now vehicles A and B will perform TLO algorithm before retransmitting to the other vehicles, i.e. Both A and B will check which one is the last one from the accident vehicle, to be more precise both A and B will compare their distances from the accident vehicle and will prepare a table. The farthest from the accident vehicle will be decided as the last one (TLO) and that designated node will carry on retransmission of the alert message to the nodes following them. From the above figure, B is designated as the last one, so vehicle B will retransmit the message to the next set of vehicles.

Normally vehicle A will wait for the threshold time to get expired and it make sure that it has received the alert message from B. If A didn't receive any messages from B, then vehicle A will retransmit the message by itself in order to avoid losing the alert messages and gives more reliability to the algorithm. Now vehicle B which is declared as a TLO transmits its message to the next hop containing vehicles C, D, E. Also vehicles A and the accident vehicle (AV) which are in the range of vehicle B receives the message. Here vehicles C, D, E, A, AV will compare their distance from the position of vehicle B and decides which one is farther and also backward from vehicle B. From the figure 7, vehicle E is the farthest and is in the backward direction of vehicle B. So vehicle E is considered as TLO and transmits it to the next hop and the process continues.

4.1.3 Program implementation

In this section we discuss about the programmable implementation of TLO algorithm. We mainly concentrate on the broadcast part used in the algorithm. Firstly we have created 3 sockets for the socket communication namely broadcast communication, Background communication, Hello communication.

All this socket communication described here is of socket programming type which is event driven programming type. Broadcast communication deals with the communication of the accident emergency messages between the nodes. Background communication deals with the communication of the unsafe applications between the nodes. Hello communication is used to update the current position of the node at certain interval of time.

Out of the three communication types, broadcast communication is the most important one as it focuses on the communication of accident emergency messages.

In broadcast communication, first the broadcast socket is created and initialized and call back function is called for broadcast receive function. Net device and Ipv4 address is bonded to socket to the emergency port. The node's information index, address, position, velocity is updated and stored in a variable. Broadcast timer is set for some milliseconds. Once the broadcast timer get expires, the node checks its own node index with the node index of the broadcast initiator and if the node's index is of the broadcast initiator, then the message packet is created with the nodes information as sender's information and the broadcast packet is sent.

Once the packet is received, broadcast receive function is called, in broadcast receive function, information from the packet is captured and stored in a variable. We have used an accident table in order to check whether the packet is received from the same origin address or not and it also has the information of the repeater address.

Now the position of each node is captured and subtracted from the sender’s address and stored as TLO address. The TLO address of the neighbouring nodes is compared with that of the present node's TLO address. If the present node's TLO address is greater than neighbouring node's TLO address, then the node decides to broadcast the packet as a repeater node.

Background communication and Hello communication is almost similar to that of the broadcast communication in the socket initialization stages and socket communication of background communication takes place in a different port named as Application port, whereas Hello communication takes place in Hello port.

In Hello port, a hello timer is been set for a few milliseconds, and once it expires it sends a packet which has the information of the neighbour nodes position and index. This hello message is sent every ten milliseconds and it is received in hello port.

4.2 APAL algorithm

The next algorithm which we are going to discuss here is APAL algorithm (Adaptive probability alert protocol)[24]. This algorithm is mainly designed to curve the problems caused by both flooding of messages and restricted transmission. In detail, when a message is broadcasted indiscriminately, like in the case of flooding algorithms it leads to various broadcast storm problems [25], like collapse of ad-hoc networks, serious contention and collision. On the other side, restricted transmission may lead to immature death of the alert message[24].

To curve the problems caused by flooding algorithms and restricted transmission, APAL algorithm is introduced.

In contrast with the TLO algorithm which was discussed previously, APAL algorithm doesn’t require any GPS location information. APAL becomes highly favourable than TLO, since accurate location information is difficult to calculate when vehicle moving at a high speed at real road environment.

Thus APAL algorithm doesn't need any location information and the probability of the broadcast alert message is chosen adaptively to avoid the lost alert message problem and to minimize the broadcast problem.

4.2.1 Explanation

Before going into the detail of the algorithm, there are certain terms which need to be understood to get the clear understanding of the algorithm. Each node after receiving their alert message takes certain time to make decision whether to transmit the message or not. This is done at certain interval of time and after expiry of this interval, each node's decision whether to transmit or not is changed or chosen adaptively. These time durations are named as *intervals.* i^{th} such time interval is denoted by $\Delta\tau_i$.

The decision process of each node is taken in a step wise manner.

Step1:

Whenever a node receives the message, it will wait for a random time interval, this time interval $\Delta\tau_1$ and is chosen randomly (in our case the interval is set from 1-100msec).

$$\Delta\tau_1 = \text{rand (1-100ms)}$$

After the expiry of this time interval, the node begins to retransmit the message based on the checking that it has received the transmitted message from some other nodes or not. If it has not received the alert message from any other node, then it will transmit it with a high probability P_i (In our case it is chosen between (0.7-0.9)). If the node with the expiry of the interval has already received some duplicate alert messages, then it goes to

Step 2:

Every time after the time interval has expired and in case if the node has already received the duplicate alert message during the time interval $\Delta\tau_i$, i.e, the vehicle will stop from rebroadcasting when it receives the same message from its neighbours during the time interval. Each time it receives the duplicate messages during the time interval it will count the number of duplicate messages number and it is termed as *duplicate number* and the next P_{i+1} and τ_{i+1} will be updated.

i.e.,

$$P_{i+1} = P_i / \text{duplicate number.}$$

$$\Delta\tau_{i+1} = \Delta\tau_i * \text{duplicate number.}$$

Only when it receives the duplicate messages, it performs the above equation, and if a node doesn't receive any duplicate message during the interval it goes to step 3.

Step 3:

When the node does not receive any duplicate message during the time interval, the node then decides to retransmit with the high probability P_i depending on the success of the rebroadcast and the next P_{i+1} and $\Delta\tau_{i+1}$ will be updated.

If the vehicle is successful to rebroadcast,

$$P_{i+1} = P_i /2; \Delta\tau_{i+1} = \Delta\tau_i.$$

And if the vehicle is not successful to rebroadcast,

$$P_{i+1} = P_i *2; \Delta\tau_{i+1} = \Delta\tau_i /2.$$

In case if P_i exceeds 1, it is clipped to 1.after completing step 3 it moves to step 4.

Step 4:

In step 4, two more parameters are introduced namely β and δ. β, the life time limit is defined as the total time duration a vehicle is allowed to handle a particular alert message. 'β' is set to a fixed value (in our case, we have set to 5 sec), which depends on the criticality of the alert message. And another fixed parameter δ, which is the number of duplicated messages, a node can handle (in our case δ is set to 5). Every node will check and decide with the condition statement mentioned in the line 8 of pseudo code ((count time < β) && (duplicate number< δ)).

In order to stay in contention as member of nodes to propagate or to exit from the member. Where count time is the total time calculated from the first time inception of the alert message to the vehicle i.e. sum of all time intervals till present time.

Count time is calculated as follows,

$$\text{Count time} = \sum i * \Delta\tau_i,$$

After checking the condition statement, if the condition is true, the node remains in the selection process to propagate the alert message and go back to step 2. And if the condition statement is false, the node exits from the process.

```
When Receive Alert message
  IF (Receive alert message for First time)
      Δτ_i Random between 1 – 100ms
      P_i random  probability between 0.7 – 0.9
  END IF
  CountTime =0
   DuplicateNumber=0
WHILE (CountTime < β && DuplicateNumber  < δ)
   WHILE (Δτ_i is not expired)
              Listen for duplicate alert message
             Count = number of received duplicate alert message
   END WHILE.
      IF (received duplicate alert message)
         DuplicateNumber = DuplicateNumber + Count
         P_{i+1} = P_i/DuplicateNumber
         Δτ_{i+1} = Δτ_i * DuplicateNumber
      ELSE
          Rebroadcast with P_i
               IF (Rebroadcast is successful)
                   P_{i+1} = P_i/2
                   Δτ_{i+1} = Δτ_i
               ELSE
                   P_{i+1} = P_i*2; P_{i+1}  is clipped to 1;
                   Δτ_{i+1} = Δτ_i/2
               END IF.
      END IF.
 CountTime = CountTime + Δτ_i
 END WHILE
```

Fig. 8. Flow chart of APAL algorithm[24]

4.2.2 Illustration

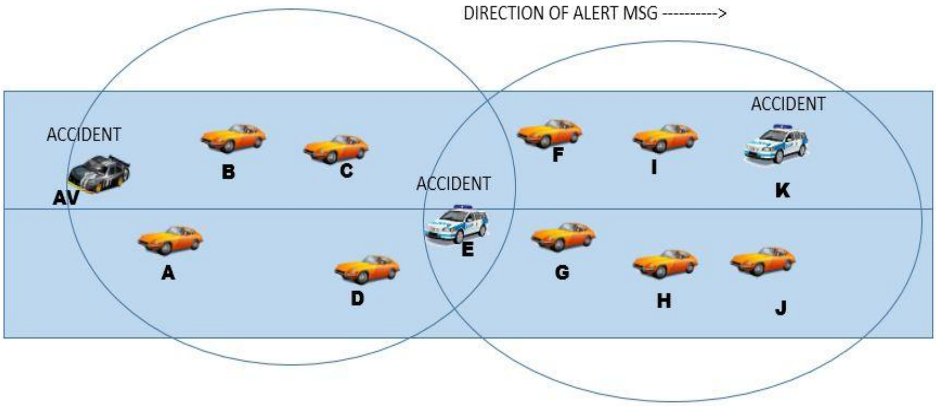


Fig. 9. Diagrammatic explanation of APAL algorithm[24]

Let us consider an example, in which let us assume that vehicle AV is the accident vehicle and it sends the alert message to the vehicles which are in its communication range, so the vehicles B, C, D, E, A which are in its communication range will be receiving the alert message from the node A.

After receiving the alert message, vehicles B, C, D, E, A will start APAL algorithm in order to retransmit the message and goes to step 1, where each vehicle is assigned a random waiting time interval. The vehicle with the least waiting time interval and with the high probability will be retransmitting the alert message .

In our example, let us assume that vehicle E is the one with the least time interval compared to the vehicles A, B, C, D . So vehicle E successfully rebroadcasts the message to the vehicles which are in its range. Now vehicles B, C, D, and F will receive the duplicate message from E and vehicles G, F, H, I, J, K will be receiving the alert message for the first time. Then vehicles G, H, I, J, K will start APAL algorithm and go to step 1 and whereas vehicles B, C, D, F will go to step 2 as they have received the duplicate message their probability P_i is decreased and will be less but not to zero and their time interval is increased.

These Vehicles B, C, D, F will not quit now and continue to remain in selection process till they satisfy the condition ((count time < β) && (duplicate number< δ)) , when it fails to satisfy the condition it gets exit from the selection process. This exiting condition makes the probability of loss very low and it highly increases the success rate of the alert message .

4.2.3 Program implementation

In this section we discuss the programmable implementation of APAL algorithm. The procedure for socket initialization of both broadcast communication and background communication is almost the same as of TLO algorithm.

Broadcast timer is set to a few milliseconds. Once the broadcast timer get expires, the node checks its own node index with the node index of the broadcast initiator and if the node's index is of the broadcast initiator, then the message packet is created with the nodes information as sender’s information and the broadcast packet is sent.

Broadcasted packet is received in emergency port and broadcast receive function is called. In broadcast receive function, each node of its first reception of the messages initiates a timer called Delay timer. This delay timer is not the same for each node and it sets a random value for each node.

As soon as the delay timer gets expired, the node checks the following information, its current probability, current delay time, current duplicate count and current total duplicate count. The probability for each node is randomly set between 0.7-0.9.

If the node's duplicate count (the number of duplicate messages) is greater than zero, the probability and delay time of the node will be updated (pi+1=pi/ duplicate number , Δti+1= Δti*duplicate number).

If the node's duplicate count is not greater than zero, then based on the probability of the node, the node decides whether it is successful to rebroadcast or not.

If the probability of the node is greater than 0.3 (as per our algorithm), then the node decides to rebroadcast the message. The probability and delay time will be updated as ($P_{i+1}=p_i/2$; $\Delta t_{i+1}= \Delta t_i$). Now the node sends the message to the next hop with its own information as the sender's information.

If the probability of the node is lesser than 0.3, then the node decides not to rebroadcast and the new probability and the delay time has been updated as ($P_{i+1}=p_i*2$; $\Delta t_{i+1}= \Delta t_i/2$).

The whole process continues only when the node's total duplicate count and the count time is less than the values of β and δ. If the node's total duplicate count exceeds the value of β and δ, the node exits from the selection process of the next broadcast node.

CHAPTER 5

5 Performance Evaluation

For the evaluation of TLO and APAL algorithms, we have used NS3 simulator and the simulation model of NS3 simulator and classes used in NS3 are discussed in this section. This section also includes a brief description of the simulation settings used for NS3 simulator and the results of these two algorithms are compared with parameters such as success ratio, throughput of received and sent packets, and propagation time.

5.1 Network simulator 3

The Network Simulator 3 (NS-3) [34] is a discrete-event network simulator for research and educational use. It is not backward compatible with NS-2, instead it is built from the scratch and it is a replacement for NS-2. The NS-3 is publically available for development, research and use. The goal of NS-3 is to create an open simulation environment for real-time network research.

The NS-3 is completely written in C++ language and it can be optionally used by Python programming language as an interface. The NS-3 is trying to rectify the problems which are present in NS-2 such as lack of memory management, coupling between different models etc. The NS-3 project has started around mid-2006 and still under heavy development [34].

NS-3 is intended to give better support than in NS-2 for the following items

- Scalability of simulations
- Integration/reuse of externally developed code and software
- Emulation
- Tracing and statistics
- Validation
- Modularity of components

5.2 Basic Model

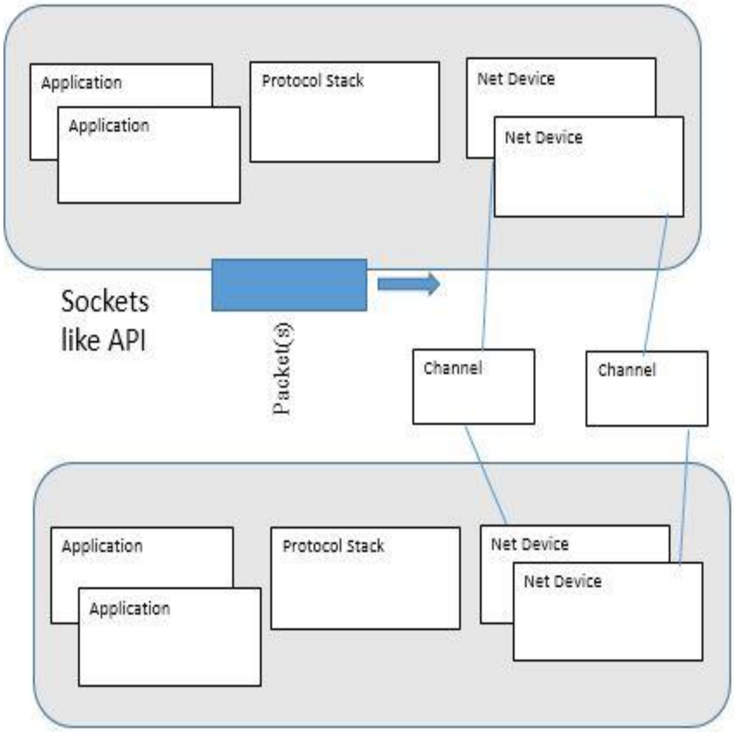


Fig. 10. Basic Model of NS-3[34].

5.3 Key Simulation Objects in NS-3:

Node: A node is an abstract base class in NS-3. It contains few objects like Unique Integer ID, System ID, a list of Net Devices and a list of applications. NS-3 will provide few subclasses like Internet Node which implements a basic UDP/IPv4 stack but still user can create their own subclasses **[34]**.

Due to the IP version used in the Nodes and implementation details of the IP stack, the design tries to avoid using too many dependencies on the base class Node, Application, or Net Device.

Therefore, the design uses the design pattern of software encapsulation to allow Net Devices and Applications to talk to implementation- independent interfaces of the underlying TCP/IP implementations.

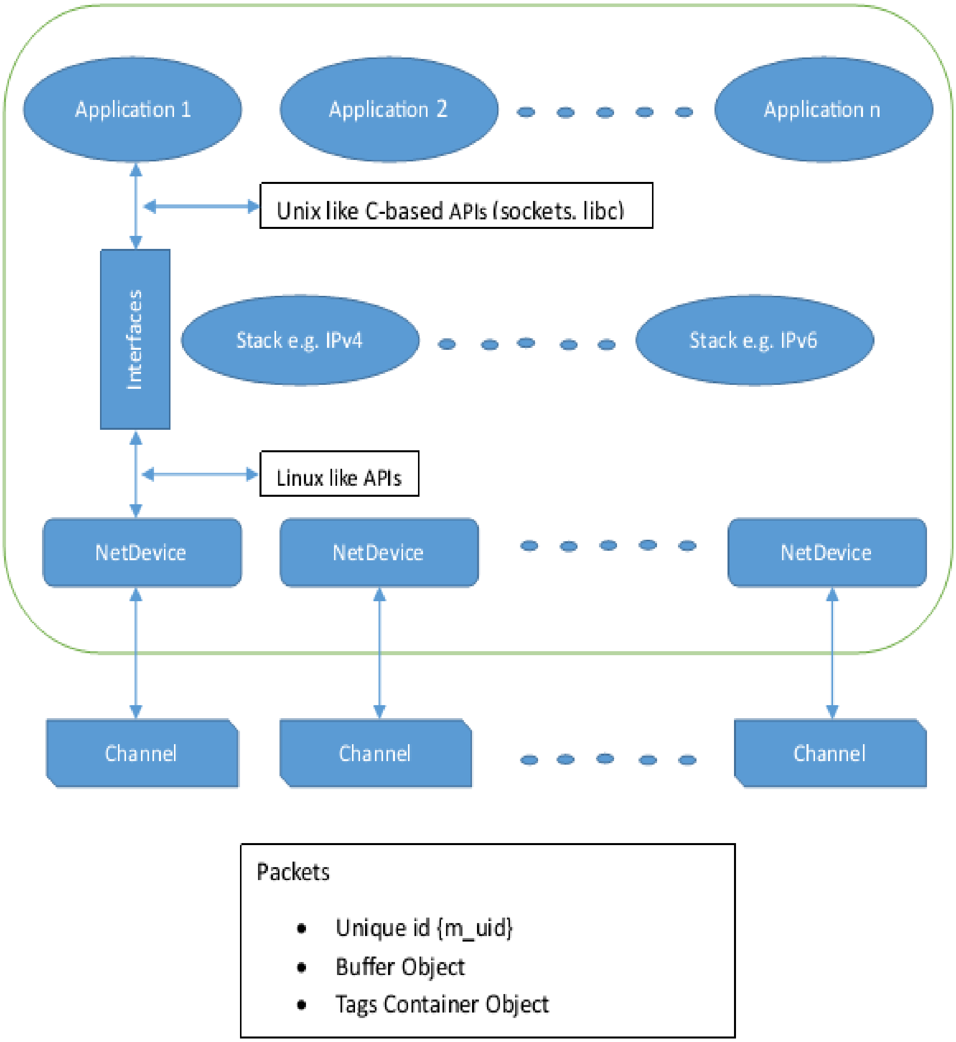


Fig. 11. High level node Architecture [34]

For example, we support a native NS-3 version of TCP/IP as well as ported Linux. If users want to try out with non-IP stacks, they can do so without having IP dependencies on the Net Devices, Channels, and Applications **[34]**.

Net Device and Channel: A key node object is ***class Net Device***, which represents a physical interface on a node and ***class Channel*** is another object which is closely attached to the Net Devices.

In stack, packets which are traversing the outbound direction call the base class ***Net Device::Send ()*** which forwards the packet to the suitable subclass method. Packets which are traversing inbound direction will call the call-back registered with ***m_receiveCallback*** when the Net Device is done processing with the packet and wants to hand it to the higher layer.

Packet: NS-3 Packet objects contain a buffer of bytes: protocol headers and trailers are sequential in this buffer of bytes using user-provided serialization and de-serialization routines. The content of this byte buffer is expected to match bit-for-bit the content of a real packet on a real network implementing the protocol of interest **[34]**.

To implement fragmentation and defragmentation are quite natural within this context because we have a buffer of real bytes and we can split it in multiple fragments and reconstruct these fragments. This will make it really easy to wrap our Packet data structure within Linux-style or BSD-style to integrate real-world kernel code in the simulator.

Memory management of Packet objects is entirely automatic and extremely efficient: memory for the application level payload can be designed by a virtual buffer of zero-filled bytes for which memory is never allocated unless explicitly requested by the user or unless the packet is fragmented. In addition, adding, removing and copying headers or trailers to a packet have been optimized to be nearly free through a technique known as Copy on Write.

Application: It is a user defined processes that produce traffic to send across the networks to be simulated. The base class of application allows one to define new traffic generation patterns via inheritance from this class. NS-3 provides a structure for developing different types of applications that have different traffic patterns. Application simply creates the application and associates it with a node and the application will send traffic down the protocol stack. On a node, application communicate with the node's protocol stack is via sockets.

Sockets: The API exported to NS-3 from sockets attempts to imitate the standard BSD sockets API. The main difference in the implementation is that when NS-3 socket API calls return the BSD socket calls the synchronous. Because in a simulation environment where one machine is simulating probably thousands of socket calls across different simulated machines simultaneously, so the simulator basically cannot afford to wait for the socket function call to return. The way the software handles the situation instead is by returning immediately, then using call-backs when other portions of the code need to be notified of a socket event. For example, when in the course of the simulation a socket is focussed to listen () on a specific port, the caller also provides a call-back to handle when the socket receives a connection request. The listen () method returns immediately and then whenever the socket receives the connection, it invokes the call-back to handle the connection. Similar things happen for the other common socket APIs, like send (), connect (), and bind ().

5.4 NS-3 Classes Used in Simulation

Broadcast Protocol: It complements the MAC protocol of NS-3 to able the broadcast function [26]. Broadcasting protocols are designed on two categrories ,one to deliver a broadcast message to nodes with in a single communication range with highest reliablity (reliable protocol). The second one is to deliver the broadcast message to the entire network (dessimination protocol). the first one is used with applications related to direct neighbours (collision avoidance) and the second is used with applications related to the entire network (traffic management).

Broadcast Packet: It defines the packet format used in broadcast.

Wi-Fi Net device: Network layer to device interface and this interface defines the API which the IP and ARP layers need to access to manage an instance of a network device layer. It models a wireless network interface controller based on IEEE802.11 standard [33].

It has 4 levels of its modular implementation namely the PHY layer model, Mac high level model, Mac low level model and a set of rate control algorithms. The low level configuration of the Wifi-netdevice is powerful but complex due to its modular implementation. To reduce the complexity, helper classes are provided to perform common operations in a simple manner.

YansWifiPhy: It is the object within the net device which receives the bits from the channel. The Physical layer can be in any of the 3 following states,

- Transmitter (the physical layer is currently transmitting a signal)
- Receiver(The Physical layer is synchronized and currently waiting for a signal)
- Idle(The Physical layer is neither transmitter nor receiver state)

YansWifiPhyHelper: It is a rework of NS-3 class called WifiPhy [27]. It is used to manage the different properties of the channel. The helper makes it easy to create and manage PHY objects for the YansWifiPhy**.**

YansWifichannel: This channel is used to implement the propagation model described in[27] . This subclass is to connect together several Wi-Fi-net devices network interface. Wi-Fi channel contains a propagation loss model and propagation delay model into consideration.

YansWifichannelHelper4: The reference is taken from the model Yans(Yet another network simulator). This helper helps in creating a Wi-Fi channel with the specific Propagation loss and delay model.

Propagation delay model: It calculates the propagation delay according to the mobility of node and propagation path. The propagation delay models available are

- Constant speed propagation delay model
- Random propagation delay model.

Propagation loss model: Models the propagation loss through a transmission medium. Calculate the receive power from a transmit power and a mobility model for the source and destination positions. The various propagation loss models available are

- Random propagation loss model
- Friss propagation loss model,
- Jakes propagation loss model,
- Long distance propagation loss model and
- Composite propagation loss model [33].

Mobility Model: It defines the mobility of each node and keeps track of the current position and velocity of an object.

The figure below represents the relation between the above mentioned classes.

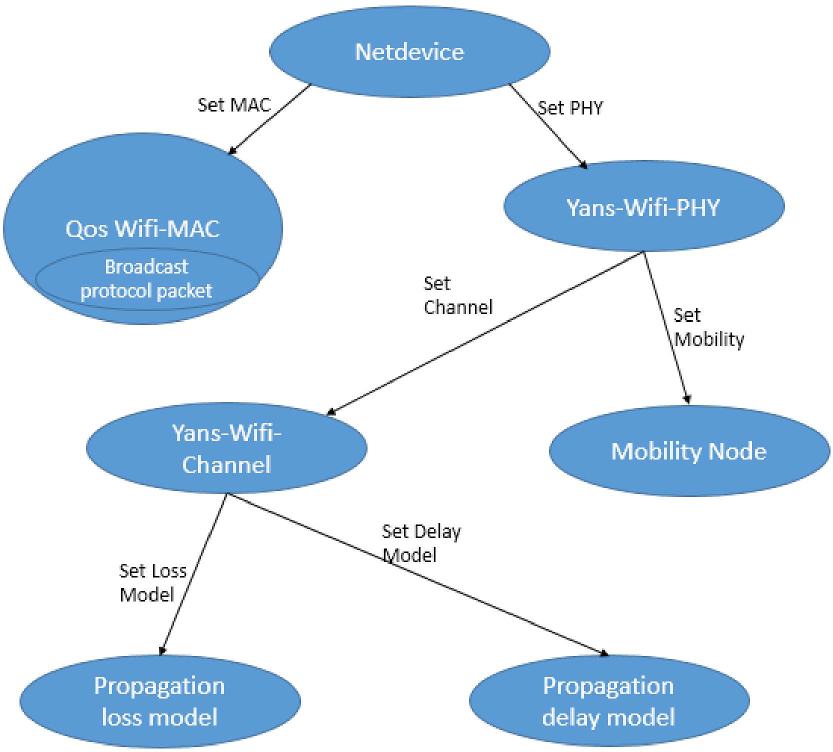


Fig. 12.Relation between the classes used in the simulation [34]

5.5 Simulation settings

With the simulation settings as described in the below table we also assume that all nodes perform properly as expected, i.e. without any malfunction. These simulation settings are setup for two contrasting algorithms in its nature namely TLO and APAL, the former algorithm TLO, which is a position based algorithm and the latter APAL, which is strictly a probability based algorithm.

TABLE 2 : EMERGENCY MESSAGE PACKET DESCRIPTION

Start time of emergency message	1000ms
Emergency Message size	80 B
Emergency Message Interval Time	100ms
Background Message size	800 B
Background message interval time	500ms
Waiting time for non-periodic emergency messages	1000ms
Hello message interval	10ms

The TLO algorithm, which is a position based algorithm, relies heavily on node's position.

TABLE 3 : ROAD SEGMENT PARAMETERS DESCRIPTION.

Road segment length	1000m
Road segment width	9.9m
Road segment lane width	3.3m
Minimum car speed	16.66 m/s
Maximum car speed	33.33 m/s
Deceleration speed	4 m/s*s
Reaction time	1.6s

The nodes are arranged randomly with **X** -coordinate representing the length of the road segment and the **Y-**coordinate representing the width of the road segment with the limits (maximum values) set as given in Table 3. With the (**X**, **Y**) coordinate put up, each node's position is precisely set on the road segment. The node which is regarded as emergency initiator is placed at the end of the road segment that is the node position (X-coordinate) of the emergency initiator should be nearer to 1000 m. As per the assumption, the **X**- coordinate value of other nodes which follows the emergency initiator node will be lesser than 1000m as it follows or comes behind the emergency initiator node. The nodes are randomly placed within the range in order to have a realistic vehicular distribution.

The mobility model which we use in our simulation is Constant velocity mobility model, in which the current speed does not change once it has been set during the simulation. The speed of each node ranges between the maximum and minimum values set as per the simulation setting table.

TABLE 4. PHYSICAL LAYER PARAMETERS DESCRIPTION

No of nodes	20,40,60,80,100(variable)
Transmission range	200 m(Constant)
Energy detection threshold	-92dbm
CCA mode1 threshold	-98dbm
Tx gain	4 dB
Rx gain	4 dB
Rx noise figure	4 dB

In the wireless physical layer, we have used IEEE 802.11p standard with 5 GHz frequency range, 10 MHz data rate at 6Mbps at the physical layer. Reception of the packets is influenced by various factors which include vehicle density, channel conditions, transmitter and receiver power, gain etc [28]. Wi-Fi channel model includes both propagation delay model and propagation loss model.

We have used Constant propagation delay model and long distance propagation loss model. We have set the value for energy detection threshold, CCA mode 1 threshold, transmitter gain, receiver gain as per the table to achieve the transmission range of 200 m. Number of nodes have been kept as a variable and varied between 20 till 100.

5.6 Evaluation criteria

Our main aim is to achieve reliable data dissemination on VANET. In order to achieve that we have compared and evaluated our algorithms using certain performance metrics namely success ratio, Normalized Throughput of sent and received packets, Propagation Time.

5.6.1 Average Success ratio

Success ratio is defined as the total number of nodes warned by the algorithm to the total number of nodes actually participated in the algorithm. This clearly gives us an idea of how many vehicles have been exactly warned during complete run of the algorithm. If the success ratio is high, it indicates the better performance of the algorithm. We have used 5 different random seed numbers and performed the simulations for each random seed number and finally the average of all simulations have been taken and marked as Average success ratio.

$$\textbf{Success Ratio} = \frac{(\textbf{Total number of nodes warned by the algorithm})}{(\textbf{Total number of nodes participated in the algorithm})}$$

5.6.2 Throughput of received packets

Throughput of received packets is defined as the total number of packets received at the destination out of the total transmitted packets to the total simulation time. Throughput is measured in packets per second or bytes per second. More the throughput of received packets better is the performance of the algorithm. Simulations were performed using 5 different random seed number and an average is taken of all.

$$\textbf{Throughput of Received Packets} = \frac{\text{(Total number of packets received at the destination)}}{\text{(Total simulation time)}}$$

5.6.3 Propagation Time

It is the time taken by the emergency node to reach the last reachable node or the total simulation time. It is measured in seconds. Propagation time gives the time for algorithm to complete and throughput is directly proportional to the propagation time.

Propagation Time = Time to reach the last reachable node from the emergency node.

5.7 Comparison of Algorithms

In this section we evaluate our algorithms using NS-3 with the performance metrics such Success ratio, Normalized Throughput of sent and received packets and Propagation Time which were described above. We have dealt with two cases, in the first case we used the number of nodes as a variable and Transmission range is used as a constant value which corresponds to 200m and in the second case we used the transmission range as a variable one and made the number of nodes as a constant which corresponds to 100. We ran our algorithm 50 times and changed the random seed number each time and final value is the average of each simulation run to avoid the same biased results.

5.7.1 Average Success ratio

Success ratio is defined as the total number of nodes warned by the algorithm to the total number of nodes actually participated in the algorithm. If the success ratio is high, it indicates the better performance of the algorithm.

Case I: Success Ratio vs. Number of nodes

From the figure 12 it is clearly evident that APAL has a better success ratio than the TLO algorithm which in turn tells that APAL has a better performance than TLO. When the node density is less, (i.e. at no of nodes 20,40) both the algorithms share almost equal success ratio ,but when the node density increases the gap between each of the algorithm increases and at number of nodes 100, it is clear that APAL has much higher success ratio compared to TLO algorithm.

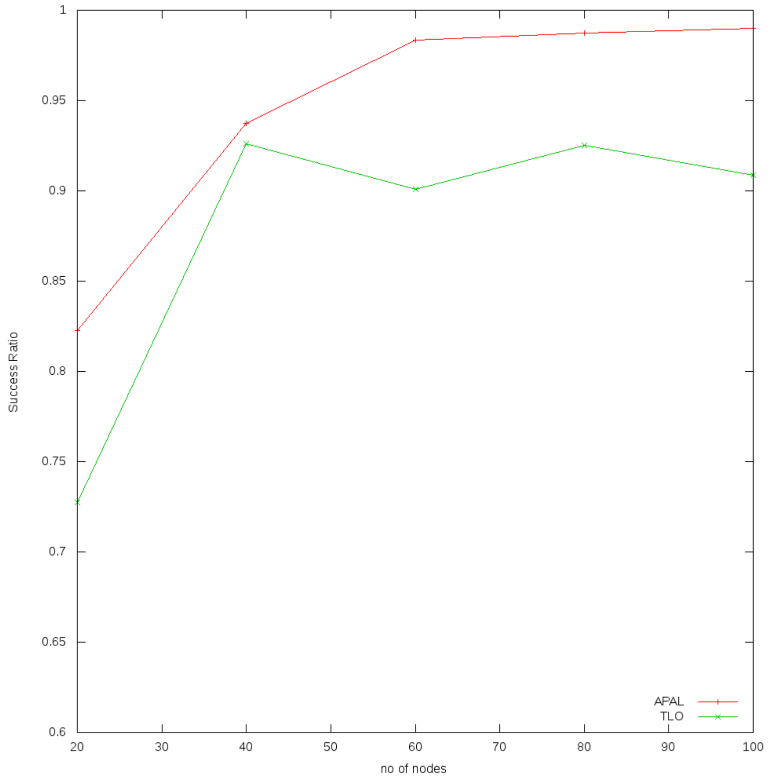


Fig. 13. Success Ratio vs. Number of nodes

Case II: Success Ratio vs. TX Range

In this case we have used the transmission range as a variable and the number of nodes as constant, from the figure 13 it is clearly evident that success ratio of APAL algorithm is almost equal to the maximum after the transmission range of 200m and remains constant when we increase the transmission range, whereas in TLO, the success ratio seems to fluctuate and it is not constant. This shows that TLO performs well only in small transmission range and when the transmission range increases, it fails to produce constant success ratio which makes APAL better compared to TLO in terms of Success Ratio.

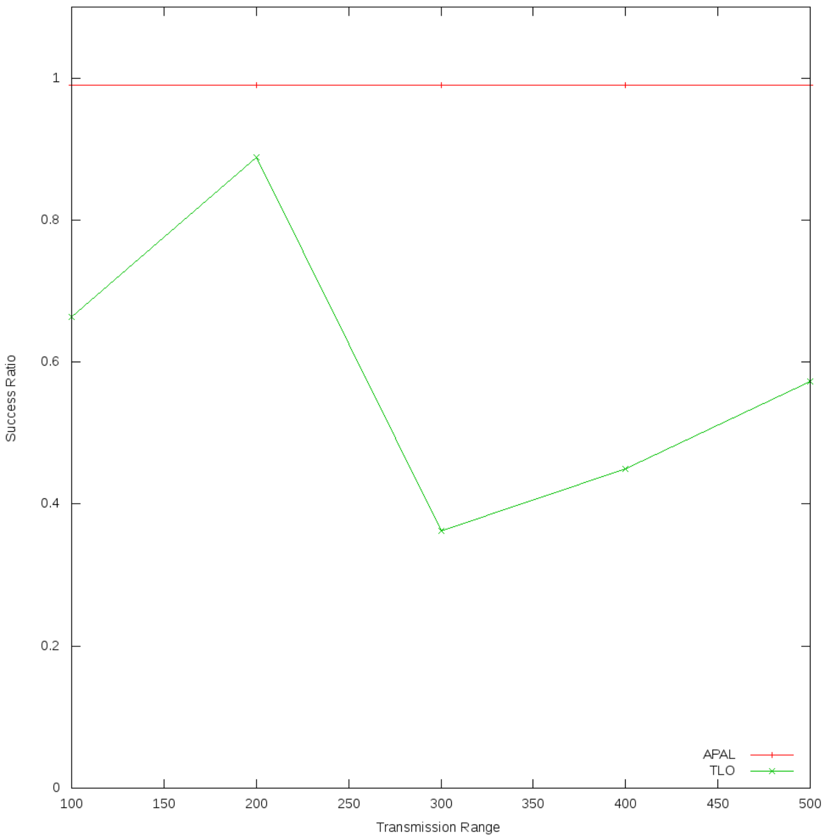


Fig. 14.Success Ratio vs. Tx range

5.7.2 Throughput of received packets

Throughput of received packets is defined as the total number of packets received at the destination out of the total transmitted packets to the total simulation time. Throughput is measured in packets per second or bytes per second.

Case I: Throughput of received packets Vs No of nodes.

The throughput of the TLO algorithm is much higher than the throughput of APAL algorithm which can be seen clearly from the figure. Node density and throughput are directly proportional to both of the algorithms.

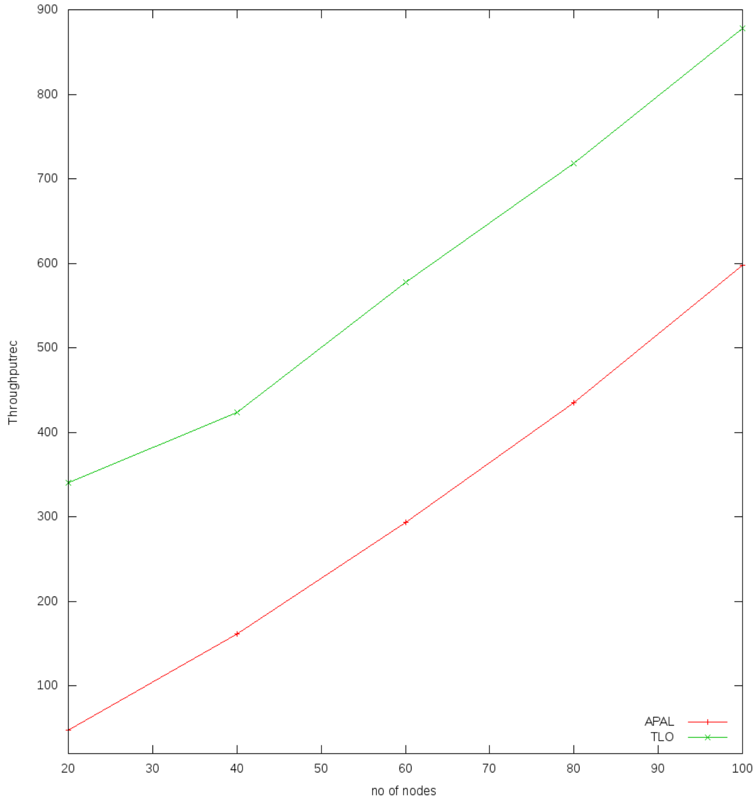


Fig. 15. Throughput of received packets Vs no of nodes.

Case II: Throughput of received packets Vs Tx range

Throughput of received packets are almost same for both APAL and TLO when the transmission range is equal to 100m. When the transmission range is increased to 200m, both the throughputs increase with TLO being the better one. But as the transmission range increases, the throughput of TLO starts decrementing whereas in contrast, the throughput of APAL constantly increases.

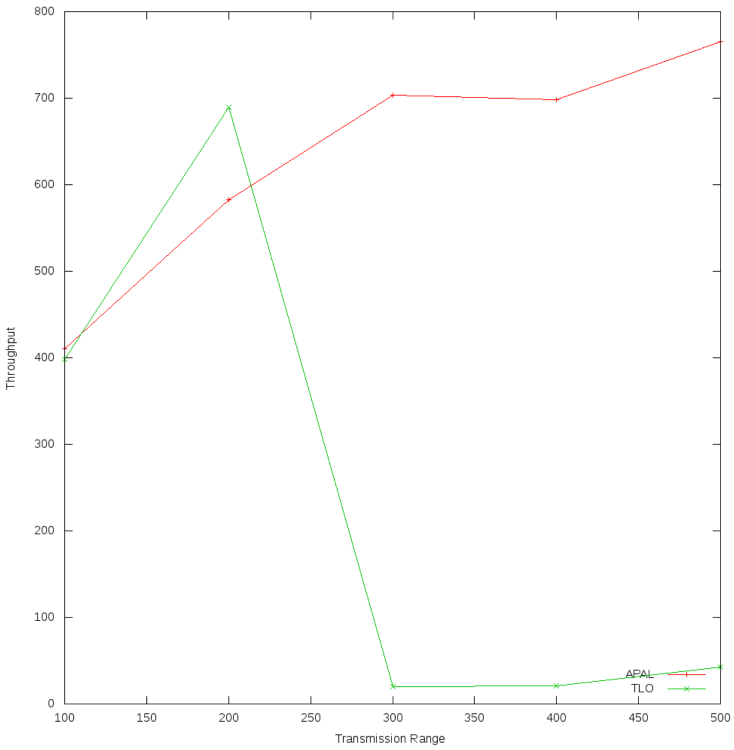


Fig. 16. Throughput of received packets Vs Tx range.

5.7.3 Propagation Time

It is the time taken by the emergency node to reach the last reachable node or the total simulation time. It is measured in seconds. Propagation time gives the time for algorithm to complete and throughput is directly proportional to the propagation time.

Case I: Propagation time Vs no of nodes.

From the figure we can see that, The TLO algorithm consumes more propagation time compared to APAL algorithm. Propagation time is less for TLO algorithm during less density of nodes and for higher nodes it gets increased, whereas in APAL the propagation time is greater at lesser density of nodes and starts decreasing towards larger density of nodes. This shows the high Reliability of APAL algorithm towards TLO.

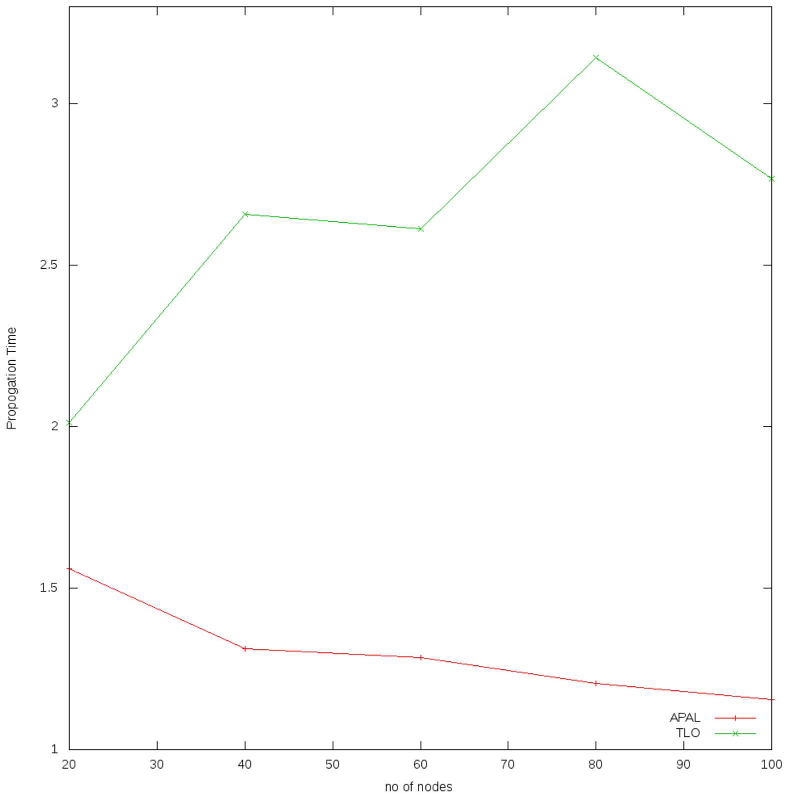


Fig.17. Propagation Time Vs no of nodes

Case II: Propagation Time Vs Tx range.

Propagation time of TLO is high compared to that of the propagation time of the APAL with the varied TX range. The Propagation time of TLO is high when the Tx range is at 100m and starts decreasing at 200m and again starts increasing at 300m. In TLO Propagation time is neither constantly decreasing nor constantly increasing with increase in Transmission range which shows that TLO is not proportional with the Propagation time. Whereas in APAL the propagation time is high in 100m and starts decreasing in 200m and it decreases gradually with the increase in TX range.

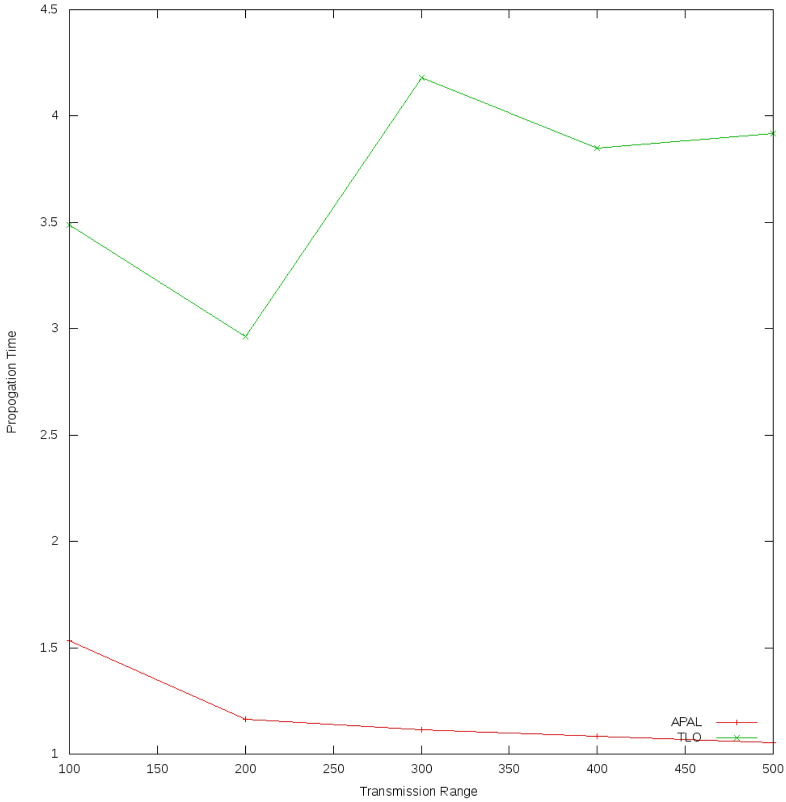


Fig. 18.Propagation Time vs. Tx range

CHAPTER 6

6 Conclusion and Future Work

Safety applications in VANET provide a key role in ensuring road safety and Broadcast data dissemination is a key factor for safety and emergency applications [36].A proper evaluation should be carried out to have reliable data dissemination. For that we have selected two algorithms TLO and APAL from the literature and evaluated them for the reliable data dissemination using QOS parameters such as

- Success Ratio
- Throughput
- Propagation Time.

In comparison with the results from our simulation of both the algorithms, TLO algorithm which is a position based algorithm has a better throughput compared to that of the APAL algorithm. Whereas success ratio of both the algorithms are same for lesser density but for the larger density APAL has the better performance compared to TLO. Propagation time is less for TLO algorithm during less density of nodes and for higher nodes it gets increased, whereas in APAL the propagation time is greater at lesser density of nodes and starts decreasing towards larger density of nodes. This shows the high reliability of APAL algorithm towards TLO. On the whole, APAL shows a better performance result compared to TLO.

Both algorithms will be able to handle only single emergency situation and it fails to handle multiple accident situation or from multiple accident sources. In order to make it more efficient in multiple handling situation, for TLO algorithm we have a concept of accident table which is carried by each node. Accident table contains the information of node address of emergency initiator and the node address of the TLO node. With the help of accident table, each node checks whether the emergency messages are received by the same emergency initiator or from different emergency initiator and it can handle according to them. Similarly, APAL algorithm should be tested to handle multiple emergency scenarios.

Reliable data dissemination in vehicular networks involves a lot of constraints mainly due to the characteristics of the transmission medium and the lack of synchronization and nodes are moving at high speed in a wide area surrounded of buildings, hills and many other architectural structures which will affect the propagation.

So we have made few assumptions like constant speed between the nodes, in the simulation time, but in practice, cars will try to reduce their speed after they received an emergency message from the neighbouring vehicles. One more assumption we have made is, that all the nodes in the simulation will have details of their node position, so for the future, we can introduce more realistic traffic models for urban areas where the assumptions are not valid. Finding better approximations for performance metrics in more general setting and relaxing some of the assumptions remain as open problems.

References

[1] J. Bernsen, "A_Reliability-Based Routing Protocol for Vehicular Ad-Hoc Networks," *Master's Theses*, Jan. 2011.

[2]F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," *Vehicular Technology Magazine, IEEE*, vol. 2, no. 2, pp. 12 –22, Jun. 2007.

[3]Y. Toor, P. Muhlethaler, and A. Laouiti, "Vehicle Ad Hoc networks: applications and related technical issues," *Communications Surveys Tutorials, IEEE*, vol. 10, no. 3, pp. 74 –88, quarter 2008.

[4]"A Comparative study of MANET and VANET Environment." [Online].Available: http://www.scribd.com/JournalofComputing/d/34832829-A-Comparative-study-of-MANET-and-VANET-Environment#download. [Accessed: 18-Jun-2012].

[5]J. Boleng and T. Camp, "Adaptive location aided mobile ad hoc network routing," in *Performance, Computing, and Communications, 2004 IEEE International Conference on*, 2004, pp. 423 – 432.

[6] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on*, 1999, pp. 90 –100.

[7] Y. Toor, P. Muhlethaler, and A. Laouiti, "Vehicle Ad Hoc networks: applications and related technical issues," *Communications Surveys Tutorials, IEEE*, vol. 10, no. 3, pp. 74 –88, quarter 2008.

[8] C. D. Wang and J. P. Thompson, "Apparatus and method for motion detection and tracking of objects in a ...," U.S. Patent 561303918-Mar-1997.

[9]P. L. Olson and M. Sivak, "Perception-Response Time to Unexpected Roadway Hazards," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 28, no. 1, pp. 91–96, Feb. 1986.

[10]S. Biswas, R. Tatchikou, and F. Dion, "Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety," *Communications Magazine, IEEE*, vol. 44, no. 1, pp. 74 – 82, Jan. 2006.

[11]"User requirements model for VANET applications." [Online]. Available: http://mimos.academia.edu/JamalullailAbManan/Papers/963337/User_requirements_model_for_VANET_applications. [Accessed: 20-Jun-2012].

[12]W. Xue-wen, Y. Wei, S. Shi-ming, and W. Hui-bin, "A Transmission Range Adaptive Broadcast Algorithm for Vehicular Ad Hoc Networks," in *Networks Security Wireless Communications and Trusted Computing*

(NSWCTC), 2010 Second International Conference on, 2010, vol. 1, pp. 28 –32.
[13] M. D. Nuri and H. H. Nuri, “Strategy for efficient routing in VANET,” in *Information Technology (ITSim), 2010 International Symposium in*, 2010, vol. 2, pp. 903 –908.
[14]P.Muhlethaler, A. Laouiti, and Y. Toor, “Comparison of Flooding Techniques for Safety Applications in VANETs,” in *Telecommunications, 2007. ITST ’07. 7th International Conference on ITS*, 2007, pp. 1 –6.
[15]S. K. Dhurandher, M. S. Obaidat, and M. Gupta, “A reactive Optimized Link State Routing protocol for Mobile ad hoc networks,” in *Electronics, Circuits, and Systems (ICECS), 2010 17th IEEE International Conference on*, 2010, pp. 367 –370.
[16] K. Na Nakorn and K. Rojviboonchai, “Comparison of reliable broadcasting protocols for vehicular ad-hoc networks,” in *Communication Technology (ICCT), 2010 12th IEEE International Conference on*, 2010, pp. 1168 –1171.
[17] N. Wisitpongphan, O. K. Tonguz, J. S. Parikh, P. Mudalige, F. Bai, and V. Sadekar, “Broadcast storm mitigation techniques in vehicular ad hoc networks,” *Wireless Communications, IEEE*, vol. 14, no. 6, pp. 84 –94, Dec. 2007.
[18]H. Füßler, M. Mauve, H. Hartenstein, M. Käsemann, and D. Vollmer, *A Comparison of Routing Strategies for Vehicular Ad-Hoc Networks*. 2002.
[19]K. Suriyapaibonwattana and C. Pomavalai, “An Effective Safety Alert Broadcast Algorithm for VANET,” in *Communications and Information Technologies, 2008. ISCIT 2008. International Symposium on*, 2008, pp. 247 –250.
[20]M. G. de la Fuente and H. Ladiod, “A Performance Comparison of Position-Based Routing Approaches for Mobile Ad Hoc Networks,” in *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, 2007, pp. 1 –5.
[21]P. Bose, P. Morin, I. Stojmenović, and J. Urrutia, “Routing with Guaranteed Delivery in ad hoc Wireless Networks,” in *WIRELESS NETWORKS*, 2001, pp. 609–616.
[22] *Location-Based Flooding Techniques for Vehicular Emergency Messaging Sangho Oh*. .
[23] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, *Contention-Based Forwarding for Mobile Ad Hoc Networks*. 2003.
[24]K. Suriyapaiboonwattana, C. Pornavalai, and G. Chakraborty, “An adaptive alert message dissemination protocol for VANET to improve road safety,” in *Fuzzy Systems, 2009. FUZZ-IEEE 2009. IEEE International*

Conference on, 2009, pp. 1639 –1644.
[25]M. Sheng, J. Li, and Y. Shi, "Relative degree adaptive flooding broadcast algorithm for ad hoc networks," *Broadcasting, IEEE Transactions on*, vol. 51, no. 2, pp. 216 – 222, Jun. 2005.
[26]K. Bür and M. Kihl, "Evaluation of Selective Broadcast Algorithms for Safety Applications in Vehicular Ad Hoc Networks," *International Journal of Vehicular Technology*, vol. 2011, pp. 1–13, 2011.
[27] M. Lacage, "Yet Another Network Simulator," in *In WNS2'06: Proc. of the 2006 workshop on ns-2*, 2006.
[28]H. Hartenstein and K. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164 – 171, Jun. 2008.
[29]http://safety.fhwa.dot.gov/intersection/".
[30]http://www.cs.jhu.edu/~dholmer/600.647/papers/OLSR.pdf
[31]http://www.cs.berkeley.edu/~culler/cs294-f03/papers/bcast-storm.pdf
[32]http://www.comp.nus.edu.sg/~bleong/geographic/related/basagni98dream.pdf
[33]http://www.nsnam.org/docs/release/3.13/manual/NS-3-manual.pdf
[34]http://www.nsnam.org/docs/architecture.pdf
[35]http://rgconferences.com/proceed/acct11/pdf/053.pdf
[36]http://www.comlab.uniroma3.it/Paper_AMVEGNI/FITCE2012_LupiPalmaVegni.pdf