Enhancement of Broadcasting in Vehicular Ad-hoc Networks (VANET) by Dynamic Adaption of Contention Window Size

Sameer Awasthi[1], Archana Singh[2]

M.Tech. (CSE) Amity University Noida[1], Assistant Professor, Amity University Noida[2]

Abstract

Broadcast transmissions are the outweigh form of network traffic in a VANET. However, since there is no MAC-layer recovery on broadcast frames within an 802.11-based VANET, the reception rates of broadcast messages can become very low, especially under saturated conditions. In this paper, we present an adaptive broad- cast protocol that improves the reception rates of broadcast messages. We rely on the observation that a node in a VANET is able to detect network congestion by simply analyzing the sequence numbers of packets it has recently received. Based on the percentage of packets that are successfully received in the last few seconds, a node is able to dynamically adjust the contention window size and thus improve performance.

***Keywords**: VANET, IEEE 802.11, Broadcast, Contension Window*

1. Introduction

Dedicated Short-Range Communications [1] (DSRC) offers the potential to effectively support vehicle-to-vehicle and vehicle-to- roadside safety communication A larger portion of messages sent on the control channel of DSRC are broadcast. Some uses for broadcasts are sending emergency warnings and periodically broadcasting a vehicle's state. The lower layers of DSRC are a variant of IEEE 802.11a [2]. However, 802.11 is known for not being able to manage the medium very efficiently, especially in case of broadcast messages. Providing reliable delivery of broadcast messages in a VANET introduces several key technical challenges. *No retransmission is possible for failed broadcast transmissions*. A failed unicast transmission is detected by the lack an of acknowledgment (ACK) from the receiver. However, acknowledgments are not used for a broadcast message. The contention window (CW) size does not change because of the lack of MAC-level recovery. The lack of detection of failed broadcast transmissions results in the size of the CW being held constant. The vehicular network must support the ability to prioritize messages. When an emergency warning is broadcast, it should be given a high priority.

The goal of this paper is to develop an adaptive broadcast protocol that improves the reliability of delivering broadcast messages in a VANET.

2. Related Works

A number of authors have addressed the problem of sending broadcast messages in a VANET. Torrent-Moreno et al. [3] show that the probability of reception of a broadcast message decreases as the distance from the sender increases and under saturation conditions the probability of reception of a message can become very low. Xu et al. [5] propose a single-hop broadcast protocol that increases the probability of a message's reception by sending the message multiple times. Yang et al. [6] propose the VCWC protocol to transmit emergency warning messages (EWM), which is based on a state machine and a multiplicative rate decrease algorithm. Both [5] and [6] aim at increasing the probability of reception by broadcasting a message multiple times.

3. Adaptive Contention Window Adjustment

According to the DSRC standard each vehicle broadcasts its status to its neighbours approximately 10 times every second [4]. As such, a node in a VANET is able to detect collisions and congestion by analyzing the sequence numbers of the packets it has recently received. While a node does not know if the packets it sent are correctly delivered, it can estimate the percentage of packets sent to him from neighbouring nodes that are successfully received.Based on the observation of packets that are recently received, a node is able to determine the current local conditions of the net- work. Therefore, a node is able to dynamically adjust its transmission parameters.

The sequence control field of an 802.11 MAC header contains a 12-bit sequence number. As a result, each node records the over- heard sequence numbers. As shown in Fig. 1, node A records that it has overheard the frames coming from node B with the sequence numbers 32, 34, 35, 36, 37, 38, 40, 41. Based on the observed sequence numbers node A could conclude that frame 33 and frame 39 were corrupted or lost and that 20% of the frames sent from node B were not received by node A.

To determine the local state of the network, a node maintains a table with entries for neighbouring nodes it has recently received frames from. One possible data structure that can be used is a hash table.The benefit of using a hash table is that lookups can be performed in near constant time (i.e, *O* (1)).

MAC Address	Sequence Number	Reception Rate	Time Stamp

Table 3.1: Data maintained in a Broadcast Table

To begin, the last sequence number is used to detect gaps between the sequence numbers. Next, the weighted reception rate is used to determine the percentage of packets that are successfully received from a specific node. Last, the timestamp is used remove entries from the table if message has not been heard from that node in a specified time period.

A weighted reception rate is used to put more emphasis on recent events. Equation 3.1 is used to calculate the weighted reception rate. Each time that a frame is successfully received the estimated reception rate is recalculated. The variable α is used to put more or less weight on the current network condition.

$$EstReceptionRate = \alpha * EstReceptionRate + (1 - \alpha) * SampledReceptionRate \quad (3.1)$$

The nodes also maintain a timer. When the timer expires the local reception rate is determined and the CW is adjusted. The local reception rate is calculated using Equation 3.2, which is an average of the weighted reception rates.

$$LocalReceptionRate = \Sigma\,(EstReceptionRate / Nodes) \quad (3.2)$$

Once a node determines the LocalReceptionRate, it compares the value against the previously stored LocalReceptionRate in order to adjust the size of CW.The following comparison is used to adjust the Contention Window (CW):

IF (average- previous average >= threshold)
Decrease the window
ELSEIF(-(average– previous average)>= threhold)
Increase the Window
ELSE
Maintain the current window.

In order to prioritize access 802.11e is used. Access priority is achieved through different choices of inter frame spacing (IFS) and CW sizes.

4. Simulation and Implementation

4.1 Algorithm Used

Input: α from traffic flow theory, Default CW [AC] values, Threshold value t2
Output: Adapted CW [AC] values

While messages are received do {

For each Time do {

Estimate the collision rate using network feedback mechanism

if collision rate>t2 then increase the corresponding CW[AC] value

else if

estimated collision rate<t2 then decrease CW [AC]value

else

maintain corresponding CW[AC] value

}

}

The ns-2 network simulator was used to evaluate the proposed broadcast algorithm. Simulation and Implementation work is divided into two phases.

Phase 1: VANET Scenario

Phase2: Protocol Design

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 6, June - 2013

4.1.1 Phase 1: VANET Scenario

In this scenario first of all we create a freeway mobility model. In this model each node is restricted to travel only within its own lane of the freeway. To simplify the complexity of this model, the model sacrifices some realism. As result, vehicles do not possess the ability to change lanes as they would on real freeway. The velocity of each node is temporally restricted based on the node's previous velocity. A safety distance is maintained so that a node cannot exceed the velocity of the node in front of it, if they are within the safety distance.

After freeway mobility model map file is created. Map file is a critical component that determining the topology of the network. Hence, the map file defines all of the freeways that exist in a simulation. As a result, it is possible to model any freeway by supplying the correct data (in the form of a map file) to the freeway program. For example, the major freeways of the Metro Detroit area can be modeled by including the data points for these freeways in a map file. To create a map file for the simulations, perl script is used. This program generates a map file which is an input for the freeway program, Generates an approximation of a circular map. The length of the inner-radius is specified for the map, and the inner-radius is the distance, in meters, from the center of the map to the inner-lane of the freeway. The program creates an eight-lane freeway, with four lanes travelling in each direction. After a map is generated by map.pl, the map file is used as input to the freeway program. The output of the freeway program is a mobility trace; mobility.tcl is generated by the freeway program. The mobility.tcl trace starts off by assigning each node an initial x, y, or z coordinate. The mobility.tcl file provides the movement of the mobile nodes.

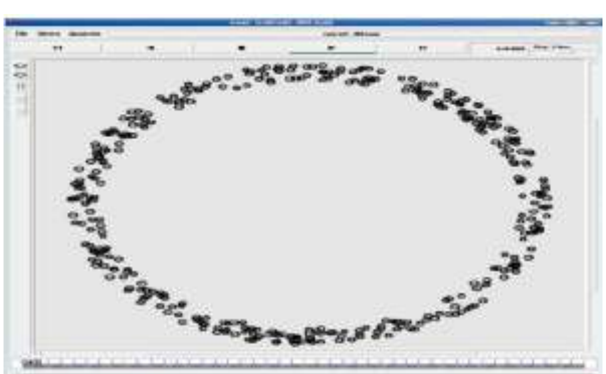

Fig 4.1.1(a): Topography for a circular road

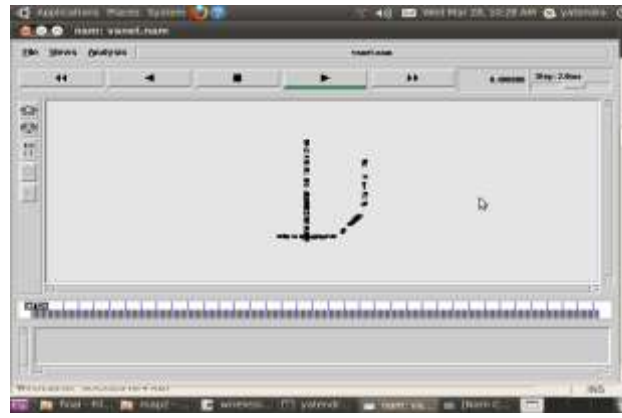

Fig 4.1.1(b): Topography for an Intersectional road

4.1.2 Phase 2: Protocol Design

Two new classes are added:

Broadcast_table and Broadcast_entry

```
broadcast_table
---------------
#table_: broadcast_entry **
#table_size_: u_int32_t
#timeout_: double
#alpha_: double
#primes[]: u_int 32t
---------------
#resize(): int
#next_size(): u_int32_t
#previous_size(): u_int32_t
#hash(mac:u_int32_t): u_int32_t
#hash(mac:u_int32_t,number_buckets:u_int32_t): u_int32_
#buckets_used(): u_int32_t
#invalidate_old_entries(current_time:double): void
#add(new_node:broadcast_entry *): void
+avg_reception_rate(current_time:double): double
+update(address:u_int32_t,seq:u_int32_t,
       time:double): void
+size(): u_int32_t
+print(): void
+broadcast_table(size:int,alpha:double,timeout:double)
+broadcast_table(alpha:double,timeout:double)
```

```
broadcast_entry
---------------
-mac_address_: u_int32_t
-sequence_number_: u_int32_t
-avg_recp_rate_: double
-time_last_heard_: double
-next_: broadcast_entry *
---------------
#next_entry(): broadcast_entry *
#remove(timeout_limit:double): broadcast_entry *
#find(mac:u_int32_t): broadcast_entry *
#add(entry:broadcast_entry *): void
#update(u_int32_t:seq,time:double,alpha:double): voi
#print_list(bucket_num:int): void
#remove_tail(): broadcast_entry *
+broadcast_entry()
+broadcast_entry(ta:u_int32_t,lsn:u_int32_t,
       time:double)
+broadcast_entry(entry:broadcast_entry &)
```

5. Performance Metric

A metric provides a standard measure for accessing the performance of a specific subject. To evaluate the simulations, metrics are needed to determine the effectiveness of the broadcast protocols. The two quantitative metrics reception rate and access delay, are used to evaluate the proposed broadcast algorithm.

5.1 Reception Rate:

Reception rate measures the percentage of packets successfully received at a specific distance.

5.2 Access Delay:

Access delay measures the amount of time it takes from, when a packet is passed down to the MAC layer, till it placed on to the wireless channel.

6. Result Analysis:

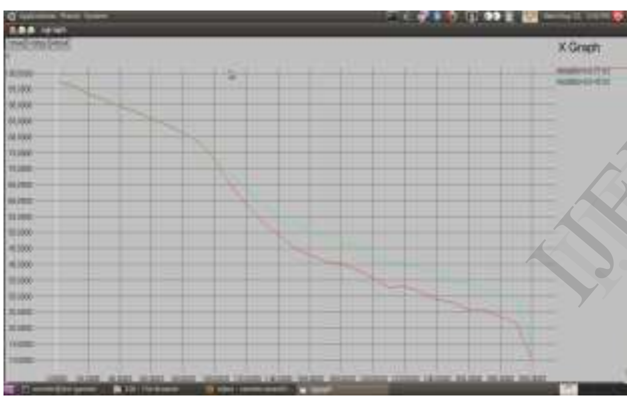

Fig 6.1(a): Static CW Simulation for 100 Nodes (Reception rate Vs Distance)

Fig 6.1(b): Dynamic CW Simulation for 100 Nodes

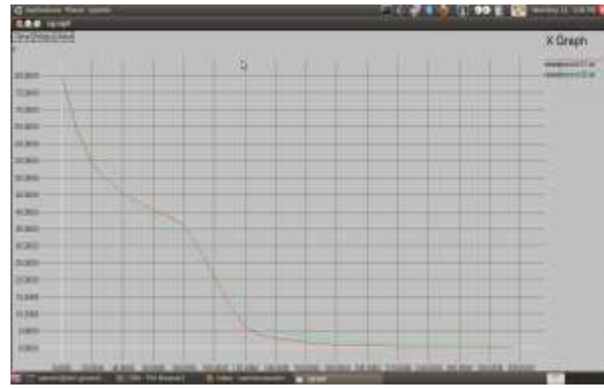

Fig 6.2(a): Static CW Simulation for 300 Nodes

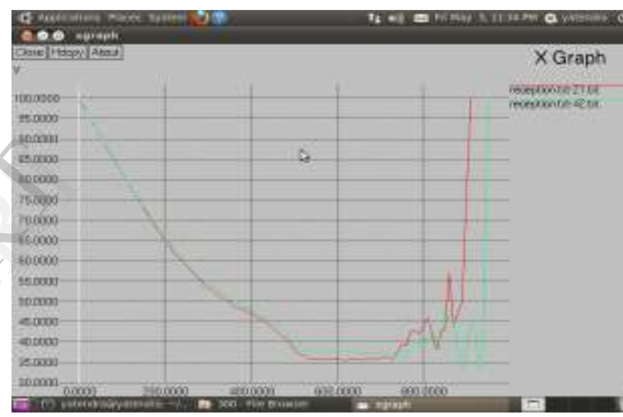

Fig 6.2(b): Dynamic CW Simulation for 300 Nodes

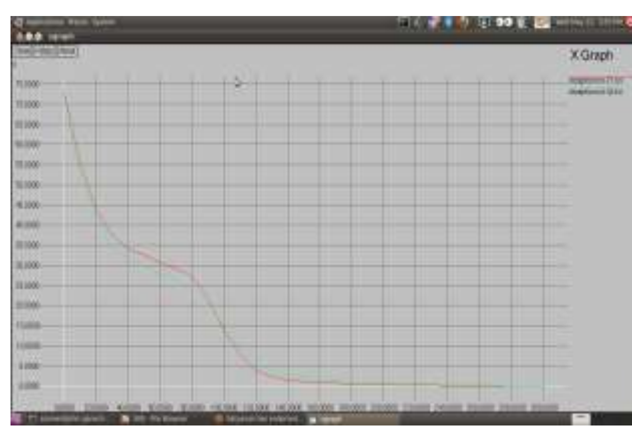

Fig 6.3(a): Static CW Simulation for 500 Nodes

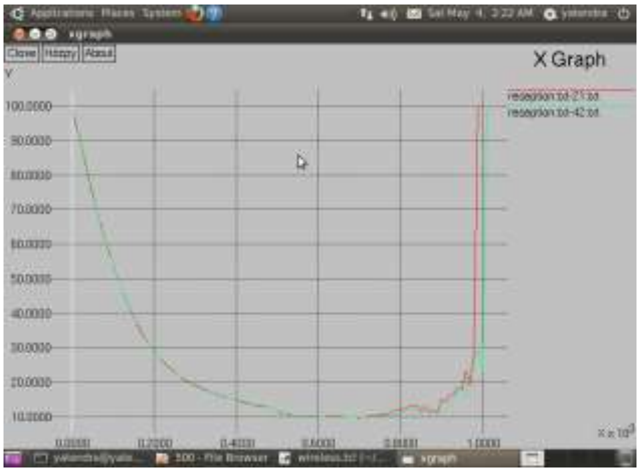

Fig 6.3(b): Dynamic CW Simulation for 500 Nodes

Table 6.1: Average Access Delay for 100 Nodes

	Standard CW	Modified CW
AC0	0.0068	$3.01\text{x}10^{-4}$
AC2	0.0069	$3.03\text{x}10^{-4}$

Table 6.2: Average Access Delay for 300 Nodes

	Standard CW	Modified CW
AC0	0.3178	$3.013\text{x}10^{-4}$
AC2	0.3419	$3.012\text{x}10^{-4}$

Table 6.3: Average Access Delay for 500 Nodes

	Standard CW	Modified CW
AC0	0.5290	5.948×10^{-4}
AC2	0.4952	6.412×10^{-4}

7. Conclusion:

Dynamically adjusting the contention window improves the reception rate under certain conditions. The results of the simulations show that adaptively adjusting the contention window can increase the reception rate of broadcast frames by approximately 35%-45% when there is moderate network traffic. And the channel access delay is less then 90ms, which is desired in VANET. The proposed additions to the broadcast protocol are not resource intensive.

8. References

[1]Armstrong. Dedicated short-range communications project.

[2]Federal Communications Commission. Amendment of the commission’s rules regarding dedicated short-range communication service in the 5.850-5.925 ghz band, FCC 02-302. Technical report, FCC, November 2002.

[3] M. Torrent-Moreno, D. Jiang, and H. Hartenstein. Broadcast reception rates and effects of priority access in 802.11-based vehicular ad-hoc networks. In *Proceedings of the First International Workshop on Vehicular Ad Hoc Networks, 2004, Philadelphia, PA, USA, October 1, 2004*, pages 10–18. ACM,2004.

[4]Vehicle Safety Communications Consortium. Vehicle safety communications project task 3 final report: Identify intellegent vehicle safety applications enabled by DSRC, March 2005.

[5] Q. Xu, R. Sengupta, and D. Jiang. Design and analysis of highway safety communication protocol in 5.9 ghz dedicated short range communication spectrum, Spring 2003.

[6] X. Yang, J. Liu, F. Zhao, and N. H. Vaidya. A vehicle-to-vehicle communication protocol for cooperative collision warning. In *MobiQuitous*, pages 114–123. IEEE Computer Society, 2004.

[7]Marios D. Dikaiakos, Andreas Florides, “Location-Aware Services over Vehicular Ad-Hoc Networks using Car-to-Car Communication,” IEEE Trans. on selected areas in communications, vol. 25, no. 8, pp. 238-249, Oct 2007.