CMS: Application Layer Cooperative Congestion Control for Safety Messages in Vehicular Networks

Kyu-haeng Lee*
Software R&D Center, Samsung Electronics
Seoul, 137-140 - Korea
[e-mail: strsurv@snu.ac.kr]
*Corresponding author: Kyu-haeng Lee

Received April 18, 2017; revised July 15, 2017; accepted October 8, 2017; published March 31, 2017

Abstract

In this paper, I propose an application layer cooperative congestion control scheme for safety message broadcast in vehicular networks, called CMS, that adaptively controls a vehicle's safety message rate and transmit timing based on the channel congestion state. Motivated by the fact that all vehicles should transmit and receive an application layer safety message in a periodic manner, I directly exploit the message itself as a means of estimating the channel congestion state. In particular, vehicles can determine wider network conditions by appending their local channel estimation result onto safety message transmissions and sharing them with each other. In result CMS realizes cooperative congestion control without any modification of the existing MAC protocol. I present extensive NS-3 simulation results which show that CMS outperforms conventional congestion control schemes in terms of the packet collision rate and throughput, especially in a high-density traffic environment.

Keywords: V2X, congestion control, broadcast, safety message, reservation

http://doi.org/10.3837/tiis.2018.03.010 **ISSN : 1976-7277**

1. Introduction

Safety message broadcasting is one of the key functions essential for vehicular communication in ITSs (Intelligent Transportation Systems). Vehicles broadcast their driving information such as GPS, heading angle, and speed, such that they can create a more efficient and safe driving environment; a vehicle may receive a crash warning when changing lanes, or it may announce an emergency such as a brake failure or road hazard. Above all things, to successfully obtain such benefits, the safety messages should be delivered in a stable manner, however it is very challenging since wireless transmissions may be significantly unreliable in vehicular networks.

In order to handle this issue, two main ITS standards, WAVE [1] and ITS-G5 [2], employ a simple method of periodically (e.g., at 10Hz) broadcasting a safety message. Periodic message broadcast may appear to be an efficient strategy in vehicular networks, but such simple-minded broadcast can cause severe packet collisions, in particular, in high-density traffic areas. Several researchers have addressed this problem by adjusting transmit parameters such as transmit power, timing, and rate, according to the network congestion state [3]–[5]. New TDMA MAC protocols [6]–[8] and reservation schemes [9] also have been proposed to avoid packet collisions by allocating unoverlapped transmit slots to vehicles.

Those schemes, unfortunately, still have significant limitations. First, most of them generally require substantial modifications of the existing standard; since the IEEE 802.11p [10] has been already approved for the underlying radio technology for both WAVE [1] and ITS-G5 [2], TDMA based MAC layer solutions or modified multi-channel operation schemes are basically impractical; additional APIs should be defined and implemented to access MAC layer information such as CBR (Channel Busy Ratio), since the application generally has no direct cross-layer communication with the MAC layer. Second, they rely on only a local measurement of the network congestion state (e.g., CBR) without information sharing between vehicles, with the result that vehicles suffer from the hidden terminal problem. Worse yet, the measured channel congestion state may be under-estimated, which brings ineffective collision resolution.

In this paper, I propose a new cooperative congestion control scheme, called CMS, that adaptively controls a vehicle's safety message rate and transmit timing based on the channel congestion state. Motivated by the fact that all vehicles should transmit and receive application layer safety messages in a periodic manner, the safety message itself is directly exploited as a means of cooperative estimation of the channel congestion state. More specifically, vehicles can determine wider network conditions by appending their local channel estimation result onto safety message transmissions and sharing them with each other. By doing this, vehicles easily estimate the number of hidden vehicles which may potentially affect their transmissions, as well as the number of vehicles in carrier sensing range. Finally, based on the estimation result, each vehicle adapts its transmit rate and timing to maximize the message transmission success probability. In CMS all these procedures can be done in application layer, and therefore we do not require any additional modification of the legacy MAC protocol.

The design of CMS is challenging for the following reasons. First, the size of the local estimation result must be reduced when appending it to safety messages. To do this, **I** develop a novel compression scheme, effectively reducing the appending overhead without any loss of information. Second, when adapting the safety message rate, the actual MAC layer operation should be taken into account, since all transmissions are eventually affected by MAC. The optimization framework for message rate control used here, thus includes the effect of the MAC channel contention. Extensive NS-3 simulation results show that CMS outperforms recently proposed congestion control schemes in terms of the packet collision rate and throughput, especially in a high-density traffic environment.

I summarize the main contributions as follows. The main contribution is to propose a new application layer cooperative congestion control scheme for vehicular networks. I understand that the proposed scheme is a lot different than conventional congestion control schemes for the following two reasons. First, the proposed scheme effectively exploits the legacy safety message to realize cooperative congestion control in vehicular networks, while most previous schemes use non-cooperative methods. Second, CMS can be fully implemented by middleware-level software, different to other schemes that need to modify the existing MAC protocol and hardware. In addition, a mathematical analysis to derive the optimal probability to successfully transmit a safety message in CMS is provided. Lastly, to increase the fidelity of the evaluation, I implement five schemes in NS-3 and compare their performances.

The remainder of the paper is organized as follows. Section 2 provides the related work, and I describe the CMS mechanism in a greater detail in Section 3. Section 4 shows the performance evaluation based on NS-3 simulation. I finally conclude this paper in Section 5.

2. Related Work

I survey the research results on congestion control schemes for vehicular networks related to CMS.

To overcome the limitation of the CSMA/CA based IEEE 802.11p [10], numerous TDMA based MAC protocols [6]–[8], [20] have been proposed. In these schemes, unoverlapped transmit slots are assigned to vehicles, leading to less packet collisions. VMESH [6] employs a conventional reservation technique in which a vehicle occupies a transmission slot once it wins the channel contention at the slot. RES [9] is similar to VMESH in that it also uses the reservation technique, but the reservation is made at the application layer, not at the MAC layer, so the MAC modification is not required. In VeMAC [7], transmission slots are assigned to vehicles according to the direction of vehicle movement. In PTMAC [20], more various moving patterns are utilized for predicting and avoiding possible packet collisions. Nguyen *et al.* propose a hybrid TDMA/CSMA multi-channel MAC protocol [8]. Unfortunately, such reservation based schemes are effective only when the network capacity is big enough.

Both WAVE and ITS-G5 mandate that safety messages should be broadcast over the CCH (Control Channel) [1], [2], which limits the channel resources for the safety messages. In the following studies, SCH (Service Channel) resources are also utilized for safety message transmissions. Ni *et al.* allow vehicles to use SCH resources for improving the reliability of safety message transmission [11]. In VER-MAC [12], the authors propose that the safety messages should be sent twice over the CCH and the SCH. Wang *et al.* study a variable CCH interval to improve the efficiency of the multichannel operation [13]. These techniques, however require huge modifications of the existing MAC protocol and system structure.

One of the most widely used methods for congestion control is to adaptively change various transmit parameters according to congestion states [3]–[5]. The ETSI (European Telecommunications Standards Institute) has recently published a framework called DCC (Decentralized Congestion Control) [3], which envisions a variety of potential controls, including message rate, transmit power, data rate, and receive sensitivity. It defines a simple threshold based control by using mapping of CBR to a combination of parameters. In LIMERIC [4] each vehicle adapts its transmit rate in order to achieve fair and high channel utilization. Different to DCC, LIMERIC gradually changes the transmit rate and thus enables a more fine-grained rate control. However, they depend on only local channel estimation results, and thus the performance gain can be limited. Even though PULSAR [5] exploits information sharing between vehicles, modifications of the legacy protocol are still required. In DTB-MAC [21], a safety message can only be transmitted if a vehicle acquires a token, which helps increase reliability. However, there still exists an additional overhead of manipulating tokens and managing groups.

Advanced physical layer technology [4], [14]–[16] (e.g., multi-antenna) and using infrastructures [13], [17] (e.g., RSU) greatly improve the packet reliability. Vehicles are considered to have two radios so that they can send and receive packets over CCH and SCH simultaneously [4], [14], [15]. Moser *et al.* examine the performance of a MIMO-extended IEEE 802.11p system [16]. These schemes are complementary to CMS, so we can employ them in CMS for further improvements.

3. CMS Design

3.1 System Model

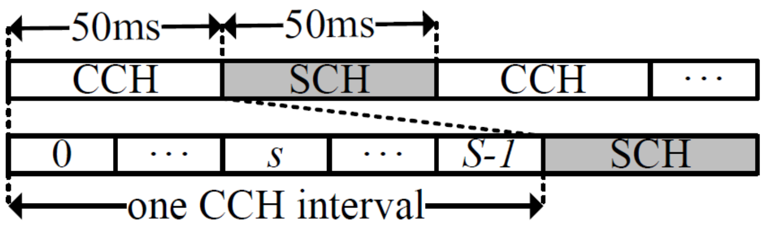


Fig. 1. Channel Model. The CCH consists of multiple CCH intervals, where each of intervals has *S* slots.

A vehicular network with *V* vehicles is considered. Each vehicle periodically transmits an application layer safety message, which has a payload of 200 bytes, at a maximum rate of 10 Hz. According to the legacy protocols [1], [2], the safety messages are broadcast on the CCH. I assume a time-slotted CCH (**Fig. 1**); the CCH is divided into multiple intervals of 50 ms, and each CCH interval consists of *S* transmit slots, where at least one safety message transmission can be completely finished in one slot. Note that this slot is defined and used in the application layer, not in the MAC layer. In general, one application layer slot is much longer than one MAC layer slot (500 μs > 9 μs (when $S = 100$)). To the end, I use the term ‘slot’ as the meaning of an application layer slot, unless otherwise stated.

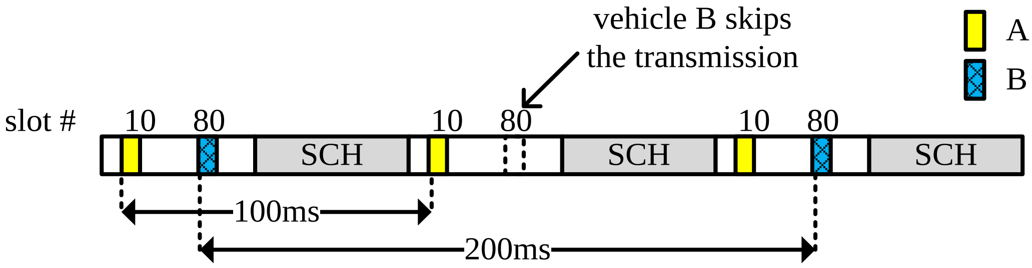


Fig. 2. An example of safety message transmission. Assume that the transmit slots of two vehicles A and B are slot 10 and slot 80, respectively. In this example, vehicle A always transmits at slot 10 every CCH interval, while vehicle B skips one transmission in the second CCH interval according to the transmit probability.

In this model, each vehicle's transmission is considered to belong to one transmit slot. Once a vehicle sets a transmit slot, it continues to use the slot until the slot is changed by the slot selector. If the vehicle detects that the slot is too crowded, it may not be able to transmit a safety message even if the vehicle reaches the slot. For this purpose, each vehicle uses the transmit probability according to the channel congestion state, which will be described later. Consider the example in **Fig. 2**, and let the transmit probability of vehicle A and vehicle B be 1 and 0.5, respectively. Then, vehicle A always transmits at slot 10 (i.e., message rate = 10Hz), while vehicle B may not transmit in some CCH intervals, (i.e., average message rate = 5Hz).

I assume that channel intervals can be synchronized between vehicles; for example, in WAVE the intervals can be synchronized by the UTC (Coordinated Universal Time) that can be acquired from GPS or other vehicles (e.g., all vehicles shall monitor the CCH beginning at the start of a UTC second [1].) Even though the actual operation of two ITS standards (i.e., WAVE and ITS-G5) are different from each other, CMS can be easily applied to each standard by varying the length of the CCH interval and S.

3.2 CMS Overview

This section provides an overview of CMS. CMS consists of three main modules, the cooperative channel monitor (**C**), message rate controller (**M**), and slot selector (**S**), which may be described as follows:

- **C**ooperative channel monitor: measures network congestion by *estimating the number of vehicles* during each transmit slot. A vehicle first locally gathers the information of visible neighbor vehicles when receiving safety messages - this is called *Visible Vehicle Information* (VVI) - and then shares it with other vehicles by appending it onto its safety message, so that vehicles' monitoring results can be more effective. This is called *Effective Vehicle Information* (EVI) (Sec. 3.3).
- **M**essage rate controller: adapts a safety message rate based on the VVI and EVI values. In order to control the message rate, a transmission probability τ is introduced, which is used to maximize the packet transmission success probability (Sec. 3.4).
- **S**lot selector: changes the transmit slot when the number of vehicles using the same transmit slot is significantly increased (Sec. 3.5).

In the following subsections, the design of CMS is described in detail in the context of tagged vehicle v and its transmit slot s.

3.3 Cooperative Channel Monitor

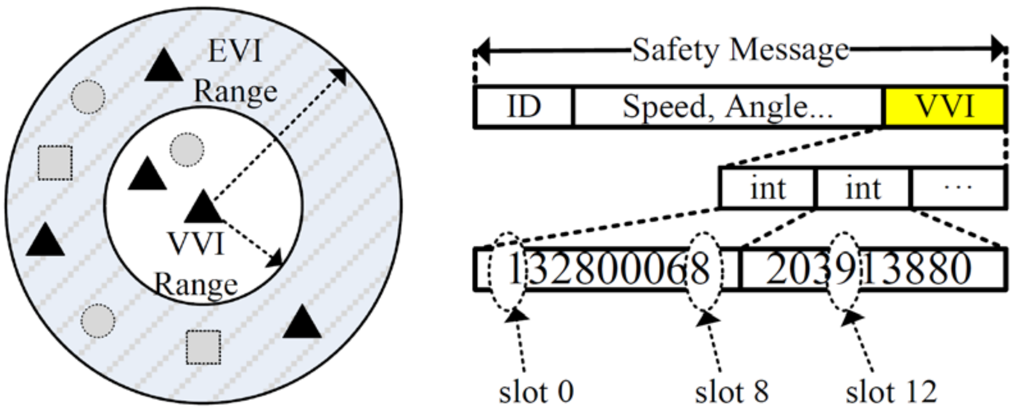
3.3.1 Visible Vehicle Information (VVI)

When a vehicle receives a safety message, it extracts the vehicle ID from the message and records it with the corresponding slot number. In this way, each vehicle can count the number of vehicles of each slot. Let N_s be the number of distinct vehicles observed in slot *s*, then VVI of tagged vehicle *v* (i.e., N_v) is represented as the vector of N_s:

$$\mathbf{N}_v = [N_0 \ \cdots \ N_s \ \cdots \ N_{S-1}]^T. \tag{1}$$

Since a vehicular network generally changes very frequently, VVI may soon become invalid. Therefore, only the vehicle information obtained during a sliding window *M* is used. Note that if a vehicle receives a safety message from the same vehicle in the same slot (say, *s*) during the sliding window, it does not increase N_s.

3.3.2 Effective Vehicle Information (EVI)



(a) Concept of VVI and EVI (b) VVI Compression

Fig. 3. An example of VVI and EVI. (a) In the figure, there are 10 vehicles (illustrated as five triangles, three circles and two squares), and the same shaped vehicles mean that they use the same transmit slot. In this example, for a tagged vehicle (the black triangle in the center), there is one vehicle within carrier-sense range (i.e., visible vehicle), and three vehicles outside carrier-sense range (i.e., hidden vehicle). (b) Each single digit number of the integer numbers shows a N_s. For example, according to this figure, the number of vehicles in slot 12 is 9.

Even though VVI includes the number of vehicles and their transmit timings, it only provides information about vehicles in carrier-sense range and thus cannot capture the effect of hidden vehicles. For this reason, as mentioned earlier, I make vehicles share their VVI with each other by appending them to their safety messages (**Fig. 3 (a)**). When receiving VVI from other vehicles, a vehicle updates its own VVI (thus EVI) by comparing with the received VVI. Let $\mathbf{N}^{eff}_v$ be the EVI of vehicle *v*, then $\mathbf{N}^{eff}_v$ is updated as follows:

$$\mathbf{N}_v^{eff} = max(\mathbf{N}_v^{eff}, \mathbf{N}_{v'}), \ \ v' \neq v \tag{2}$$

, where *max*( ) represents a pairwise max operation.

There are several works that exploit the idea of information sharing. For example, in PULSAR [5], the authors propose to exchange CBR between vehicles for rate adaptation. While several ideas of PULSAR may bear some resemblance to the proposed scheme, in PULSAR it is difficult to explicitly control vehicles' transmit timings since the CBR has no information about them. In contrast, the proposed scheme enables vehicles to adapt the transmit timing, as well as the transmit rate, thanks to VVI which carries more detailed information than CBR.

3.3.3 VVI Compression

One practical challenge in cooperative channel monitoring is that the size of VVI may be too large, thus incurring a large overhead when appending to a safety message. For example, in the case of S = 100, 400 bytes are required to represent VVI, which is greater than the typical safety message size (about 200 bytes).

To solve this issue, VVI is compressed when appending. The idea is very simple: limit the maximum number that N_s can be. The driver behind this idea is that the actual effective value that N_s can be is generally less than 10. For example, consider N_s of 10, which means there are 10 vehicles broadcasting safety messages in slot s. In this case, controlling (or, in fact, reducing) the message rate to avoid a packet collision may be no longer effective, due to the message rate and throughput being too low. In this case, changing to a less congested transmit slot is a more appropriate strategy. For this reason, CMS limits the maximum value of N_s to 9.

If the value of N_s is limited to a single digit number, and mapped to one digit of a certain integer value, we can express multiple N_s using just a few integers; a 4 byte integer variable can represent 9 N_s values at a time, and an 8 byte unsigned long integer variable can represent 19 N_s values. As a result, in the case where S = 100, the compression reduces the VVI overhead by 88% (i.e., 12 integer variables (48 bytes) are required), compared with using 100 integers to express VVI, as shown in **Fig. 3 (b)**.

3.4 Message Rate Controller

As mentioned earlier, in CMS each vehicle uses a transmit probability τ to control the transmit rate. In other words, when transmit slot s starts, vehicle v broadcasts a safety message with probability τ_v. In this paper, I am interested in maximizing the aggregate probability of successful packet delivery, and therefore the global optimization for transmit probabilities of vehicles may be required.

Unfortunately, obtaining a global optimal value (i.e., a set of τ) is impractical and also ineffective, since the optimal value may shortly become outdated due to the frequent network topology changes. In addition, the search space of the optimization problem grows too fast as the number of probabilities to be optimized increases.

For this reason, I employ a local optimization. More specifically, each vehicle v finds τ_v to maximize its packet transmission success probability based on 1) the number of contending vehicles in the same slot and 2) transmit probabilities of them. First, from VVI and EVI, vehicle v easily knows the number of visible vehicles (i.e., N_s - 1) and that of hidden vehicles. I denote the number of hidden vehicles as N^{hv}_s , and it can be computed by N^{eff}_s - N_s. Second, for the transmit probabilities of other vehicles, I apply the same probability with vehicle v to them, based on the study result of PULSAR [5]; the authors show that it is possible to synchronize the channel congestion state (e.g., CBR) of all vehicles within two hops by performing inter-vehicle message exchanges during two sliding windows.

Now I define an objective function $f(\tau_v)$ as the conditional packet transmission success probability, then the optimal τ^*_v can be computed as follows:

$$\tau_v^* = \underset{0<\tau_v\leq 1}{\operatorname{argmax}} f(\tau_v). \tag{3}$$

Since each transmission is affected by both visible and hidden vehicles, Eq. (3) can be represented as:

$$f(\tau_v) = \tau_v \cdot f_{vv}(\tau_v) \cdot f_{hv}(\tau_v) \tag{4}$$

, where $f_{vv}(\cdot)$ and $f_{hv}(\cdot)$ are the probabilities that visible and hidden vehicles do not affect the tagged vehicle's transmission, respectively.

1) $f_{vv}(\tau_v)$

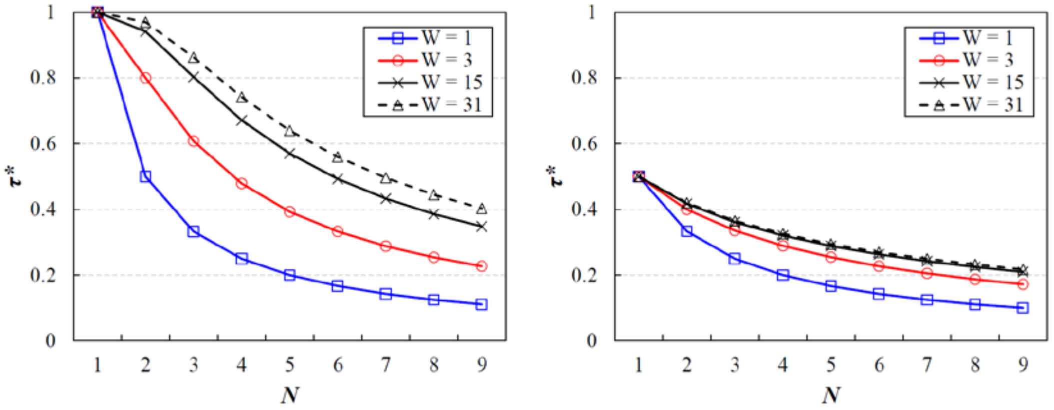
There are two cases where visible vehicles do not affect the tagged vehicle's transmission; first, in the case where no visible vehicles attempt to transmit (i.e., $(1 - \tau_v)^{Ns-1}$); second, all visible vehicles lose the MAC contention against the tagged vehicle v. Let W be the MAC contention window size, then $f_{vv}(\tau_v)$ can be expressed as the following:

$$\begin{aligned} f_{vv}(\tau_v) &= (1-\tau_v)^{N_s-1} \\ &+ {}_{(N_s-1)}C_1 \cdot \tau_v \cdot (1-\tau_v)^{N_s-2} \cdot \frac{1}{W}\left\{\left(1-\frac{1}{W}\right)+\cdots+\left(1-\frac{W-1}{W}\right)\right\}+\cdots \\ &+ {}_{(N_s-1)}C_{(N_s-1)} \cdot \tau_v^{N_s-1} \cdot \frac{1}{W}\left\{\left(1-\frac{1}{W}\right)^{N_s-1}+\cdots+\left(1-\frac{W-1}{W}\right)^{N_s-1}\right\} \\ &= (1-\tau_v)^{N_s-1} + \sum_{n=1}^{N_s-1} {}_{(N_s-1)}C_n \cdot \tau_v^n \cdot (1-\tau_v)^{N_s-1-n} \cdot \frac{1}{W}\sum_{w=1}^{W-1}\left(1-\frac{w}{W}\right)^n. \end{aligned} \tag{5}$$

2) $f_{hv}(\tau_v)$

In this case, the only way for the tagged vehicle to successfully transmit is that no hidden vehicles attempt to transmit:

$$f_{hv}(\tau_v) = (1-\tau_v)^{N_s^{hv}}. \tag{6}$$



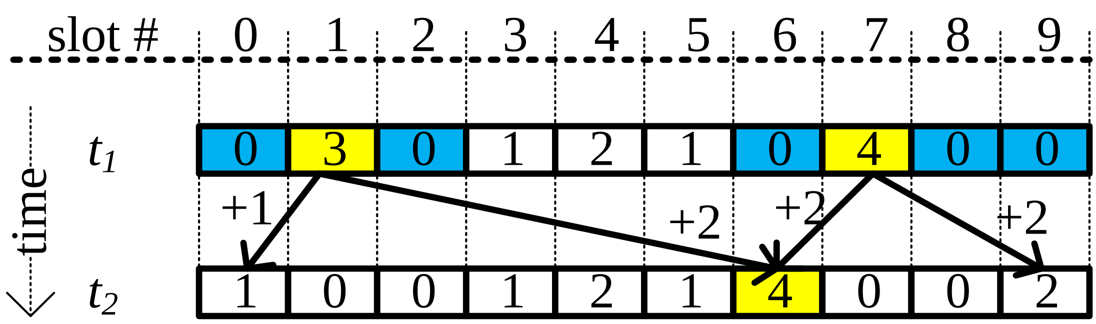
(a) $N^{hv}_s = 0$ (b) $N^{hv}_s = 1$

Fig. 4. τ^*_v according to N_s. I omit v and s. In case of W=31, the minimum transmit probability and the maximum probability are 0.111 ($N^{hv}_s = 8$) and 1, respectively. In CMS, vehicles adapt their message rates based on VVI and EVI as well as the effect of MAC contention.

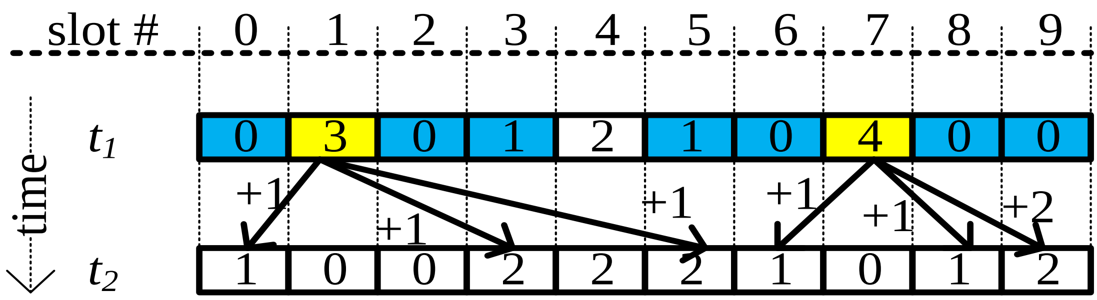
Note that $f(\tau_v)$ is a N^{eff}_s - degree function of τ_v, and we can use a hill-climbing method [18] to find the optimal τ_v. As an example, τ^*_v is provided by varying N_s and N^{hv}_s in **Fig. 4**. In both cases τ^*_v monotonically decreases with N_s, but increases with W, which indicates that a higher MAC collision avoidance capability leads to more aggressive transmissions. In addition, from the comparison between **Fig. 4 (a)** and **(b)**, we can observe that the hidden vehicle affects τ^*_v significantly, due to the fact that the MAC collision avoidance cannot handle packet collisions with hidden vehicles.

3.5 Slot Selector

The slot selection of CMS is invoked when a vehicle newly joins a network, or senses that the number of vehicles using the same transmit slot exceeds a threshold (i.e., $N^{eff}_s > \alpha$, where α is a threshold.). As mentioned before, in this case a vehicle changes its transmit slot to a less congested transmit slot. Note that the slot selector does not always choose the least congested transmit slot as a new transmit slot, since a packet collision could also happen in the new slot. Instead, CMS randomly selects one of the top k less-congested transmit slots.



(a) k=5



(b) *k*=6

Fig. 5. An example of the slot selection. The figures represent the EVI at time t_1 and t_2 according to different *k* values. A *k* value has a tradeoff between the successive collision and the transmit probability; in case of *k*=5, after changing slots, four vehicles in slot 6 should again find another non-congested slots; in case of *k*=6, no slot selection is called, but the vehicles originally existing in slot 3 and slot 5 may reduce their message rates because the slots become congested.

The slot selector tries to change to a better slot than the current one, but in some cases, it can also cause congestion on the changed slot. Let us call this a successive collision. If this happens, the slot selector will again find another non-congested slot, which can cause transmission delays.

Fig. 5 shows the EVI at time t_1 and t_2 according to different *k* values. Assume that there are 10 transmit slots (i.e., S=10) and α=2. Since in both cases at t_1 slot 1 and slot 7 are reported as congested, 7 vehicles in these slots change their transmit slots at t_2. In case of *k*=5 (**Fig. 5 (a)**), the candidate slots are slot 0, slot 2, slot 6, slot 8 and slot 9 according to the least 5 less-congested slots. The best way here is to appropriately distribute the 7 vehicles over the 5 slots such as (1,1,1,2,2), but the successive collision may occasionally happen (e.g., in slot 6).

One way to handle this issue is to apply a higher *k* value to reduce the chance of successive collisions; when applying *k*=6, vehicles have more candidates to go to (**Fig. 5 (b)**). However, using a high *k* value may also have the effect of reducing the message rate of some vehicles (e.g., the vehicles originally existing in slot 3 and slot 5), hence we need to carefully control the value of *k*. I leave this issue as my future work.

3.6 Summary

Algorithm 1: CMS

while *an application on* **do**
  $(\mathbf{N}_v, \mathbf{N}_v^{eff}) \leftarrow$ Cooperative Channel Monitor (M);
  if *transmit slot* s *is not yet set or* $N_{s,v}^{eff} > \alpha$ **then**
    $s \leftarrow$ Slot Selector $(\mathbf{N}_v^{eff}, k)$;
  end
  if *the current slot is* s **then**
    $\tau_v \leftarrow$ Message Rate Controller $(\mathbf{N}_v, \mathbf{N}_v^{eff})$;
    $\mathbf{N}_v \leftarrow$ VVI Compression $(\mathbf{N}_v)$;
    Broadcast a safety message appending $\mathbf{N}_v$ with probability τ_v;
  end
end

Fig. 6. CMS algorithm

CMS is summarized in **Fig. 6**.

4. Performance Evaluation

4.1 Simulation Setting

Table 1. Simulation Parameters

Parameter	Value
PHY Data Rate (Mbps)	6
V	10 ~ 300
K	5
S	100
α	$\lceil V/S \rceil$
M (ms)	200
DCC [3]	
CBR	**Message rate**
< 30%	10 Hz
30% to 39%	5 Hz
40% to 49%	2.5 Hz
50% to 59%	2 Hz
> 60%	1 Hz

In this section, the performance of CMS is evaluated. The vehicular network, as well as the mobility of vehicles, is simulated by using the NS-3 [19]. An intersection scenario with 6-lane roads is evaluated, with the width of each road set to 3 m. In this scenario, a two-ray ground propagation loss model is used. At beginning, V vehicles are randomly distributed in the roads, and then start to move along the road at random speed ranging from 0 to 80 km/h. **Table 1** summarizes the simulation parameters. In addition to CMS, the following schemes are implemented for comparison:

- Legacy [1]: is the legacy WAVE with no congestion control.
- DCC [3]: changes the message rate according to mapping of CBR values to message rates, as shown in **Table 1**.
- LIMERIC [4]: controls the message rate according to the following rule: $(1 - a)\cdot r + b\cdot(\text{CBR}^{target} - \text{CBR}^{current})$, where r, CBR^{target} and $\text{CBR}^{current}$ denote the message rate, target CBR (set to 79%), and the current measured CBR, respectively, and a and b are the algorithm parameters which are set to 0.1 and 0.033, respectively.
- RES [9]: exploits transmit slot reservation, similar to the slot selection of CMS. If the number of vehicles exceeds S, then random slot selection is invoked.
- CMS (w/o EVI): uses VVI instead of EVI for message rate control and slot selection.

4.2 Results

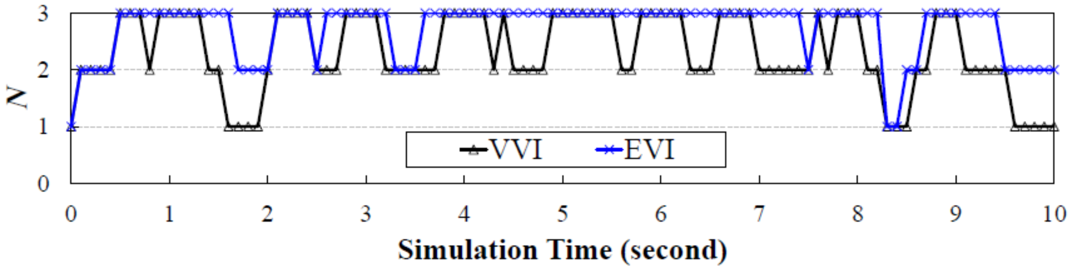
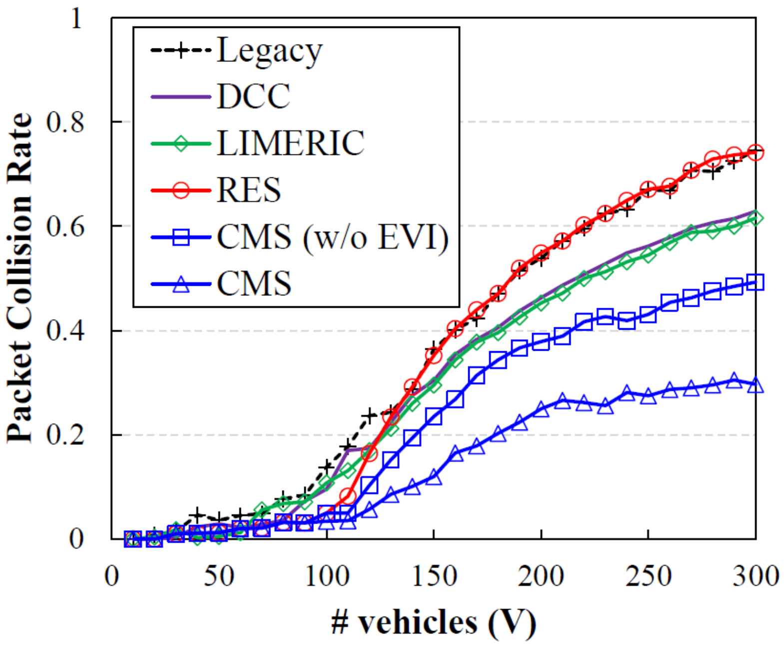
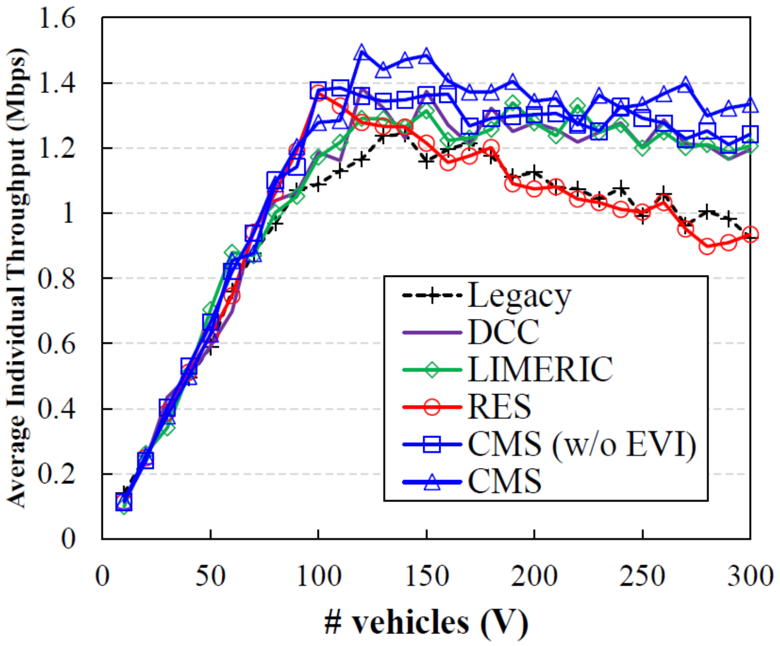


Fig. 7. VVI and EVI values of a certain vehicle when $V = 300$. The difference between VVI and EVI values indicates that hidden vehicle cases exist in this scenario.

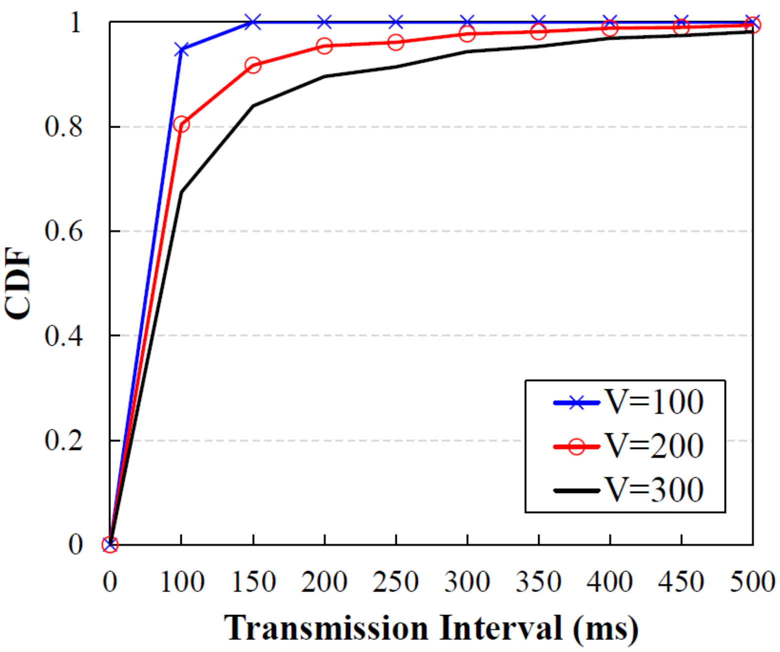
Before going into details on the performance evaluation of CMS, I first show the characteristics of the traffic model of the simulation. To do this, I randomly select one vehicle and compare its VVI and EVI values in **Fig. 7**. From this result, the difference between the VVI and EVI values can be observed (EVI value: 2.73, VVI value: 2.32 on average), and it appears frequently during the simulation, which shows the scenario is with hidden vehicle cases.



(a) Packet Collision Rate



(b) Throughput



(c) Interval

Fig. 8. Overall performance evaluation results. CMS outperforms other schemes in terms of PCR and throughput, thanks to the message rate and timing control. In addition, the interval increase of CMS is negligible.

Now, I compare each transmission scheme. **Fig. 8 (a)** shows the PCR (Packet Collision Rate) of each scheme according to V. In Legacy, PCR continuously increases with V, since it has no congestion control, while RES shows a better performance thanks to the effect of slot reservation. However, after *V* exceeds the capability of reservation (i.e., $V > 100$), the PCR increases as it does in Legacy. Compared to RES, LIMERIC and DCC show a higher PCR when *V* is small, with better PCR results at higher *V*, which indicates that slot reservation is more effective than message rate control, when it can be appropriately used. In particular, in the simulation LIMERIC shows a slightly better performance than DCC, because a more fine-grained control is available in LIMERIC. CMS outperforms the other schemes in terms of the PCR. Compared to LIMERIC and Legacy, CMS reduces the collision rate by a maximum of 32% and 46%, respectively. This is because CMS controls both transmission timing and message rate, unlike LIMERIC and DCC, which mainly control the message rate, or RES which uses slot reservation only.

In result, the PCR affects the throughput performance of each scheme, as shown in **Fig. 8 (b)**. In case of Legacy and RES, as *V* increases, the throughput first increases, and then after reaching a certain point it starts to decrease, due to the unmanaged packet collisions. In contrast, we can see that LIMERIC and DCC have a higher throughput even when *V* is high, since the message rate control effectively reduces the packet collisions. Similarly, CMS shows a larger average throughput than other schemes, and, in particular, its gain becomes greater when considering the effect of hidden vehicles (i.e., EVI) than when using only VVI. Taking the PCR result together, we can clearly see that CMS is very energy-efficient.

While CMS cleverly reduces packet transmission attempts when the network becomes congested, one concern may be that such reduced attempts lead to negative impacts on the interval between consecutive transmissions. However, the interval increases are very subtle as shown in **Fig. 8(c)**; in the high-density traffic scenario ($V = 300$), 90% of the total transmission intervals are less than 200 ms.

5. Conclusion

In this paper, an application layer cooperative congestion control scheme, CMS, is proposed for safety message broadcast in vehicular networks. This scheme adaptively controls a vehicle's safety message rate and timing based on the cooperatively estimated channel congestion state, without any modification of the existing MAC protocol. Extensive NS-3 simulation results show that CMS can reduce the packet collision rate by maximum of 32% compared to other recent advances, while providing a comparable throughput performance.

References

[1] "IEEE 1609 - Family of Standards for Wireless Access in Vehicular Environments (WAVE)," IEEE Std 1609.0-2013. Article (CrossRef Link)

[2] "ETSI ITS-G5 - EN 302 663 Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band." Article (CrossRef Link)

[3] "Intelligent Transport System - Decentralized Congestion Control." Article (CrossRef Link)

[4] G. Bansal, J. B. Kenney, and C. E. Rohrs, "LIMERIC: A Linear Adaptive Message Rate Algorithm for DSRC Congestion Control," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 9, pp. 4182–4197, Nov 2013. Article (CrossRef Link)

[5] T. Tielert, D. Jiang, Q. Chen, L. Delgrossi, and H. Hartenstein, "Design methodology and evaluation of rate adaptation based congestion control for Vehicle Safety Communications," in *Proc. of 2011 IEEE Vehicular Networking Conference (VNC)*, Nov 2011. Article (CrossRef Link)

[6] Y. Zang, L. Stibor, B. Walke, H. J. Reumerman, and A. Barroso, “A Novel MAC Protocol for Throughput Sensitive Applications in Vehicular Environments,” in *Proc. of 2007 IEEE 65th Vehicular Technology Conference-VTC2007-Spring*, Apr 2007. Article (CrossRef Link)
[7] H. A. Omar, W. Zhuang, and L. Li, “VeMAC: A TDMA-Based MAC Protocol for Reliable Broadcast in VANETs,” *IEEE Transactions on Mobile Computing*, vol. 12, no. 9, pp. 1724–1736, Sep 2013. Article (CrossRef Link)
[8] V. Nguyen, T. Z. Oo, P. Chuan, and C. S. Hong, “An Efficient Time Slot Acquisition on the Hybrid TDMA/CSMA Multichannel MAC in VANETs,” *IEEE Communications Letters*, vol. 20, no. 5, pp. 970–973, May 2016. Article (CrossRef Link)
[9] Y. Park and H. Kim, “Collision Control of Periodic Safety Messages With Strict Messaging Frequency Requirements,” *IEEE Transactions on Vehicular Technology*, vol. 62, no. 2, pp. 843–852, Feb 2013. Article (CrossRef Link)
[10] “IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments,” IEEE Std 802.11p-2010. Article (CrossRef Link)
[11] M. Ni, Z. Zhong, and D. Zhao, “A novel multichannel multiple access protocol for vehicular ad hoc networks,” in *Proc. of 2012 IEEE International Conference on Communications (ICC)*, Jun 2012. Article (CrossRef Link)
[12] D. N. M. Dang, C. S. Hong, S. Lee, and E. N. Huh, “An Efficient and Reliable MAC in VANETs,” *IEEE Communications Letters*, vol. 18, no. 4, pp. 616–619, Apr 2014. Article (CrossRef Link)
[13] Q. Wang, S. Leng, H. Fu, and Y. Zhang, “An IEEE 802.11p-Based Multichannel MAC Scheme With Channel Coordination for Vehicular Ad Hoc Networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 449–458, Jun 2012. Article (CrossRef Link)
[14] T. Tsuboi, J. Yamada, N. Yamauchi, and M. Hayashi, “Dual Receiver Communication System for DSRC,” in *Proc. of 2008 Second International Conference on Future Generation Communication and Networking*, Dec 2008. Article (CrossRef Link)
[15] Y. Kim, Y. H. Bae, D. S. Eom, and B. D. Choi, “Performance Analysis of a MAC Protocol Consisting of EDCA on the CCH and a Reservation on the SCHs for the IEEE 802.11p/1609.4 WAVE,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 5160–5175, 2017. Article (CrossRef Link)
[16] S. Moser, L. Behrendt, and F. Slomka, “MIMO-enabling PHY layer enhancement for vehicular ad-hoc networks,” in *Proc. of IEEE Wireless Communications and Networking Conference Workshops*, Mar 2015. Article (CrossRef Link)
[17] T. K. Mak, K. P. Laberteaux, R. Sengupta, and M. Ergen, “Multichannel Medium Access Control for Dedicated Short-Range Communications,” *IEEE Transactions on Vehicular Technology*, vol. 58, no. 1, pp. 349–366, Jan 2009. Article (CrossRef Link)
[18] S. Boyd and L. Vandenberghe, “Convex optimization,” *Cambridge university press*, 2004. Article (CrossRef Link)
[19] “NS-3.” [Online]. Article (CrossRef Link)
[20] X. Jiang and D. H. C. Du, “PTMAC: A Prediction-Based TDMA MAC Protocol for Reducing Packet Collisions in VANET,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 11, pp. 9209-9223, Nov 2016. Article (CrossRef Link)
[21] A. Balador and A. Böhm and C. T. Calafate and J. C. Cano, “A reliable token-based MAC protocol for V2V communication in urban VANET,” in *Proc. of 2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sep 2016. Article (CrossRef Link)

Kyu-haeng Lee received his B.S. degree in computer science and technology from Tsinghua University, Beijing, China, and Ph.D in computer science and engineering from Seoul National University in 2009 and 2015, respectively. He has been with Samsung Electronics, as a software engineer since 2015. His research interests include MIMO/OFDM systems, IEEE 802.11, system optimization, and IoT.