

## A Cluster-based TDMA System for Inter-Vehicle Communications\*

TSANG-LING SHEU AND YU-HUNG LIN

*Department of Electrical Engineering*

*National Sun Yat-Sen University*

*Kaohsiung, 804 Taiwan*

*E-mail: sheu@ee.nsysu.edu.tw*

This paper presents a cluster-based TDMA (CBT) system for inter-vehicle communications. In intra-cluster communications, the proposed CBT uses a simple transmit-and-listen scheme to fast elect a VC (VANET Coordinator) and it allows a VN (VANET node) to randomly choose a time slot for bandwidth requests (BR) without limiting the number of VNs. In inter-cluster communications, when two clusters are approaching, the CBT can quickly resolve the collisions by re-allocating time slots in one of the clusters. To analyze the performance of the proposed CBT, we derive mathematical equations using probability. The performance metrics of our interests include the average number of time slots for electing a VC, the average number of time slots required for BR, and the total number of time slots required before data can be successfully transmitted. The analytical results are finally validated by a simulation. Both the analytical and simulation results show that the proposed CBT spends less time to form a small-sized cluster than IEEE 802.11p. Additionally, when the number of joining VNs is increased, CBT takes less waiting time before a VN can effectively transmit data.

**Keywords:** cluster-based, TDMA, MAC, inter-vehicle communications, VANET

### 1. INTRODUCTION

VANET (Vehicular Ad-hoc Networks) inherited from MANET (Mobile Ad-hoc Networks) employs IEEE 802.11p as its MAC (Medium Access Control) layer to support inter-vehicular communications. When there are not so many vehicular nodes (VN) in a group, 802.11p can resolve contention very quickly. However, as the nodes increases, collision and unfairness may occur accordingly. Different from the contention-based protocol as in 802.11, recently cluster-based TDMA attracts more attention for VANET to improve the transmission efficiency as the number of VNs is increased to a certain large number. In cluster-based TDMA, a cluster head needs to be selected to serve as the network coordinator. The selected cluster head is responsible for allocating time slots for data exchange among its cluster members. Through careful scheduling of time slots for its members, collision can be avoided and fairness can be achieved.

Previous works on 802.11-based VANET focused on two aspects, transmission efficiency and multi-channel techniques. In transmission efficiency, Wang *et al.* [1] proposed a method to modify 802.11 DCF. The size of contention window is reduced to half when the successfully transmitted packets have exceeded a threshold. By applying bargaining-game theory to VANET, Shrestha *et al.* [2] proposed a method that can solve the

---

Received August 18, 2011; revised November 25, 2011 & March 2, 2012; accepted April 20, 2012.

Communicated by Wanjiun Liao.

\* Part of this work was presented in IWCMC-VCS Conference, Istanbul, Turkey, July 2011.

problem of large packet loss when nodes move very fast. To minimize multi-hop delivery delay, Yu *et al.* [3] proposed a method to adjust the transmission order according to the geographical locations of vehicular nodes. In multichannel techniques, Mak *et al.* [4] proposed an access control scheme to improve the communication efficiency between AP (Access Point) and VNs. Their proposed technique combines PCF (Point Coordination Function) and DCF (Distributed Coordination Function) to maximize the channel utilization which not only guarantees the transmission of safety message but also increases the throughput of non-safety messages. In addition to multi-channel, Su *et al.* [5] further utilized the concept of cluster-based to propose a multi-channel protocol, with which non-real-time and real-time traffic can be delivered through contention and contention-free periods, respectively.

Some previous works on the cluster-based VANET studied how to use metrics to choose a cluster head. Among them, Goonewardene *et al.* [6] and Zaydoun *et al.* [7] considered relative moving speed and direction to reduce link failure rate. By using relative distance, Wang *et al.* [8] proposed a position-based prioritized clustering (PPC) to select a cluster head. Zhu *et al.* [9] selected a cluster head based on communication efficiency, network connectivity, and residual energy. Chu *et al.* [10] used the number of vehicular nodes and Guo *et al.* [11] used the transmit power and coverage as metrics to choose a cluster head. Without using a sophisticated method, Fan *et al.* [12] picked up a cluster head simply based on the smallest ID.

In cluster-based TDMA, a selected cluster head is responsible for time slot allocations. Satio *et al.* [13] proposed a spatial slot allocation scheme to dynamically reallocate time slots based on node's priority. Similarly, Du *et al.* [14] and Kostas *et al.* [15] respectively proposed slot-allocation schemes to fast deliver emergency and real-time traffic. For sensor networks, Ma *et al.* [16] proposed an NACPA (Nimble and Adaptive Control Phase Algorithm) consisting of four phases (frame synchronization, control, scheduling, and data transmission). By assuming a cluster head is known to the cluster members, a sensor node randomly chooses a time slot for bandwidth requests (BR) in control phase. If certain nodes failed in the previous round because they pick up the same slot, the failed nodes have to raise BR in the next round. Su *et al.* [17] proposed an integration of contention-free and contention-based MAC protocol for VANET. Their scheme assumed a VN always have data to send and the number of VNs is smaller than the number of time slots. When the number of VN exceeds the number of time slots, collisions cannot be avoided. Finally, Ding *et al.* [18] employed a self-organized cluster by assuming two different channels, multiple control channels and a single data channel. Unfortunately, the authors did not address time slots in TDMA.

Unlike the previous works [16-18], our proposed cluster-based TDMA, and thereafter CBT, does not require unreasonable assumptions. First, in intra-cluster communications, a VC (VANET Coordinator) is not pre-assigned, but it is elected through a simple transmit-and-listen scheme. Second, without confining the number of BR, the proposed CBT allows a VN to randomly choose a time slot for BR. In inter-cluster communications, when two clusters are approaching, the CBT can quickly resolve the collisions by re-assigning time slots in one of the clusters. To analyze the performance of the proposed CBT, we derive mathematical equations using the probability model. The performance metrics of our interests include the average number of time slots required for electing a VC, for presenting BR, and the total time slots required before data can be successfully

transmitted. Finally, the analytical results are validated by an NS-2 simulation.

The remainder of this paper is organized as follows. In Section 2, the proposed CBT is introduced. The formats of TDMA and MAC frames are subsequently defined. In Section 3, the CBT algorithms for intra- and inter-communications are presented. In Section 4, mathematical equations are derived and a simulation is performed. In Section 5, we give the concluding remarks.

## 2. CLUSTER-BASED TDMA SYSTEM

A cluster-based TDMA (CBT) system is designed for intra- and inter-cluster communications. The CBT assumes every vehicle is equipped with GPS (Global Positioning System) to synchronize TDMA slots among vehicles prior to setting up a cluster network.

### 2.1 Intra- and Inter-Cluster Communications

Basically, the proposed CBT system for inter-vehicle communications consists of two different phases, intra- and inter-cluster communications. Initially, when two or more than two vehicles approach together, the CBT of intra-cluster communications is constructed in three steps. First, all vehicular nodes compete among themselves for being elected as a VANET coordinator (VC). Second, a winner, the VC, is responsible for scheduling time slots for the rest of cluster nodes, referred to as VANET nodes (VNs). Third, data transmissions over the designated time slots can proceed without incurring any collisions.

Once a VC is elected, the coverage of a cluster is defined as the transmission range of the VC, *i.e.*,  $\omega$  meters, as shown in Fig. 1. On highway, it is highly possible that vehicular nodes in a cluster may move faster than the ones in another cluster, and eventually two VCs of the two clusters may share an overlapping area for a certain time. As shown in Fig. 2, two VCs (VC 0 and VC 1) of two clusters respectively coordinate the time slots for their own VNs (*i.e.*, VC 0 for VN 01 and VN 02, and VC 1 for VN 11 and VN 12). Once these two clusters approach together, VNs belonging to different clusters may like to exchange information as well. For example, VN 01 may send data to VN 11. Thus, it is essential to construct a mechanism for inter-cluster communications.

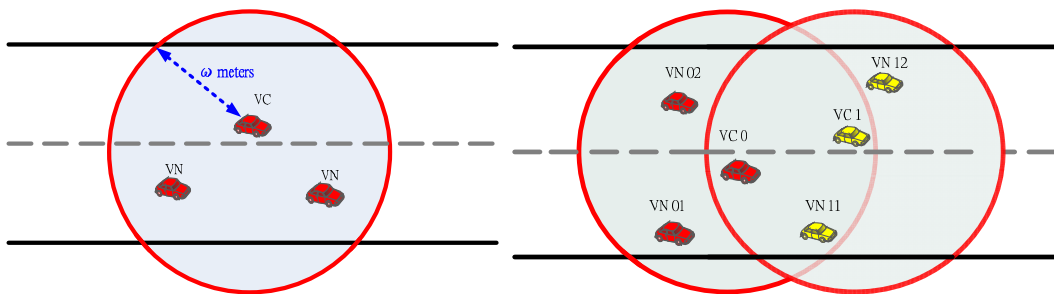
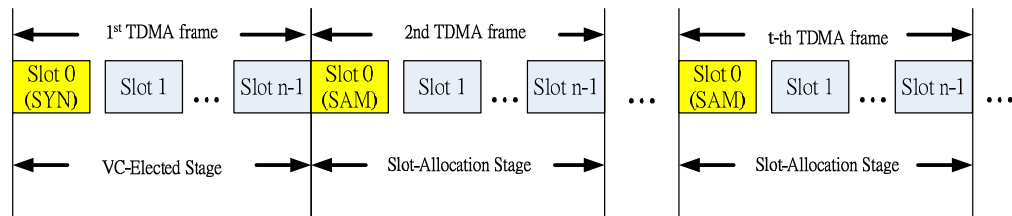


Fig. 1. Intra-cluster communications.

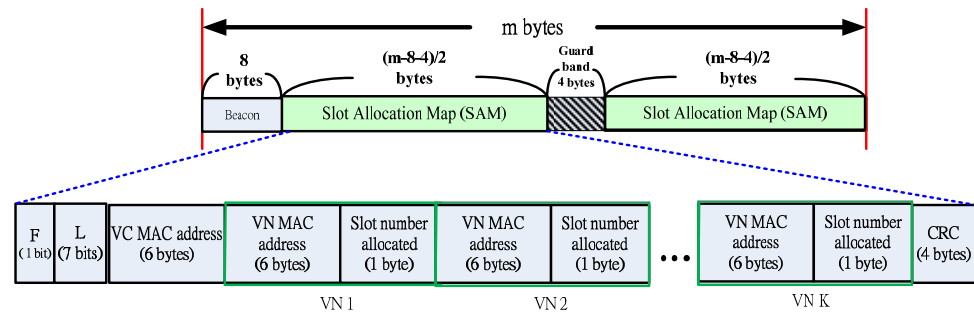
Fig. 2. Inter-cluster communications.

2.2 TDMA Time Slots

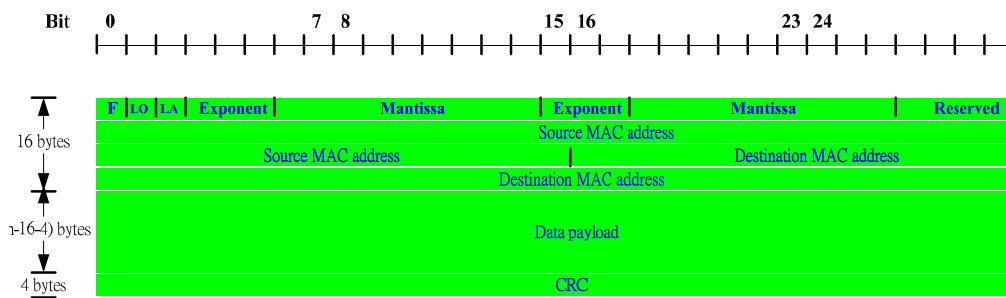
To build intra- and inter-cluster communications for the proposed CBT system, first, we need to design TDMA time slots and the associated MAC-layer frame format. As shown in Fig. 3 (a), a TDMA frame consists of  $n$  time slots (slot 0 to slot  $n - 1$ ). Among them, slot 0 serves as two different purposes in different TDMA frames; (i) SYN (in the first TDMA frame): prior to the set up of a CBT system, two or more than two approaching vehicles will issue an 8-byte beacon signal to synchronize to the start of slot 1, and (ii) SAM (in the remaining TDMA frames): once a CBT system is formed, the elected VC begins to broadcast slot-allocation map (SAM) to its VNs, so every VN has a designated time slot for transmitting data. Slot 1 to slot  $n - 1$  of the first TDMA frame (the VC-elected stage) are used for the cluster nodes to compete for a VC, while slot 1 to slot  $n - 1$  of the remaining TDMA frames (the slot-allocation stages) are designated time slots used for data transmission.



(a) Time slots in TDMA frames.



(b) MAC-frame format in slot 0.



(c) MAC-frame format in slot 1 to slot  $n - 1$ .

Fig. 3. TDMA time slots and MAC-layer frames.

Fig. 3 (b) shows the  $m$ -byte MAC-layer frame format in slot 0. It consists of three fields, 8-byte beacon, two  $(m - 8 - 4)/2$ -byte slot-allocation maps, and 4-byte guard band. The design of beacon signal serves two purposes: (i) it synchronizes the start of slot 1 during VC-elected stage, and (ii) it allows one VC to detect the existence of another VC, so that the construction of inter-cluster communication can be initiated. It is noticed that with 8-byte beacon length it can support up to 300-meter transmission coverage and up to 20-Mbps transmission rate, since  $\frac{300 \times 2}{C} \times 20 \text{ Mbps} \leq 8 \text{ bytes}$ , where  $C$  is the speed of light. The two slot-allocation maps (SAM) are designed for two VCs in two different clusters to exchange slot-allocation information. A simple transmit-and-listen scheme works in a way that a VC successfully received SAM from another VC will reschedule its time slots to avoid any collisions with another cluster. Detail procedures to build inter-cluster communications will be introduced in Section 3. Basically, SAM consists of the following fields, introduced below.

Fig. 3 TDMA Time Slots and MAC-layer Frames

- $F$  (1 bit): If  $F = 1$ , VN may access SAM; otherwise, it is for VCs to exchange their SAMs.
- $L$  (7 bits): The length of SAM (in bytes).
- VC MAC Address (6 bytes): The MAC address of a VC.
- VN MAC Address (6 bytes): The MAC address of a VN.
- Slot Number Allocated (1 byte): The ID (from 1 to  $n - 1$ ) of the allocated time slot.
- CRC (4 bytes): to protect SAMs.

As an example, if a time slot can accommodate  $m$ -byte MAC-layer payload, after deducting the 4-byte guard band, each SAM can occupy  $\frac{m - 8 - 4}{2}$  bytes, which can support up to  $K$  VNs in each cluster, and  $2K$  VNs in both clusters. Fig. 3 (c) shows the  $m$ -byte MAC frame format in slot 1 to slot  $n - 1$ . Each MAC frame consists of 16-byte MAC header,  $(m - 16 - 4)$ -byte data payload, and 4-byte CRC. Except the source and destination MAC addresses (6 bytes each), the rest of fields in the MAC header are defined as below:

$F$  (1 bit): The default value is zero, indicating that slot 1 to slot  $n - 1$  in the first TDMA frame are used for cluster nodes to compete for VC. Once VC is elected,  $F$  is set to one, indicating that slot 1 to slot  $n - 1$  are used for data transmission in the remaining TDMA frames.

- $LO$  (1 bit): East ( $LO = 1$ ) or West ( $LO = 0$ ) Longitude.
- $LA$  (1 bit): North ( $LA = 1$ ) or South ( $LA = 0$ ) Latitude.
- Exponent (3 bits): The exponent part of a 12-bit floating point number.
- Mantissa (9 bits): The mantissa part of a 12-bit floating point number.

It is noticed that  $LO$ ,  $LA$ , Exponent, and Mantissa are designed to locate a vehicular node moving on the street or highway using the Longitude and the Latitude. For example, if a vehicular node is located at East-Longitude  $179.25^\circ$ , then we have  $LO = 1$ , Exponent = 111, and Mantissa = 011001101, since  $179.25 = 10110011.01 = 1.011001101 \times 2^7$ .

### 3. THE CBT ALGORITHMS

For clarity, the proposed CBT algorithms are divided into two different communication types, Intra-cluster communications (Intra-CC) and inter-cluster communications (Inter-CC).

#### 3.1 Intra-Cluster Communications

Fig. 4 shows the state transitions of Intra-CC, which describes the procedures of setting up a CBT system. Intra-CC basically consists of five states, Initial, Competition, VC, VN, and Collision. In the Initial state, a vehicular node, referred to as a cluster node (CN), is searching for other CNs by issuing an 8-byte beacon signal, as described in Section 2.2. Once a collision of the beacon signal is detected, a CN remains idle till the start of slot 1, which deliberately synchronizes all the near-by CNs to the start of slot 1 in the first TDMA frame. Beginning from slot 1 to slot  $n - 1$  of the first TDMA frame, the transition enters Competition state, where a CN arbitrarily chooses to transmit-and-listen compete-for-VC (CFV) message, formatted as shown in Fig. 3 (c), on a slot-by-slot basis. A VC is automatically elected if only one CN intends to transmit CFV and all others are in listening CFV. On the other hand, if two or more than two CNs intend to transmit CFV at the same slot, the transition goes to the Collision state, where all the CNs will resume competition for VC at the next slot.

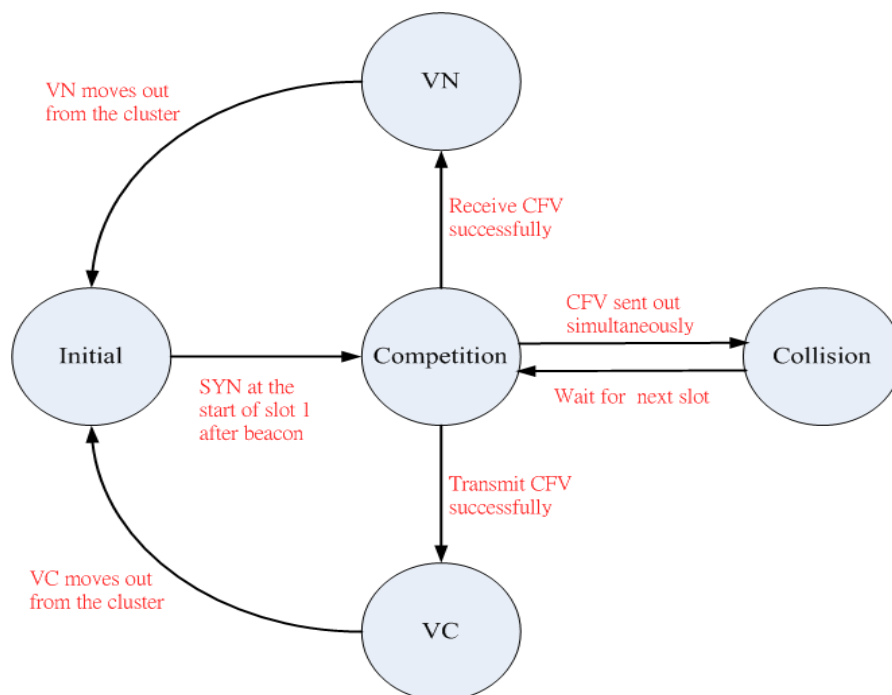


Fig. 4. State transitions of intra-cluster communications.

Fig. 5 shows the operations of Intra-cluster communications (Intra-CC). Basically, Intra-CC consists of three phases. In phase 1, a VC is randomly selected among all the VNs through VC\_elected(). In phase 2, all the VNs present their Bandwidth Requests (BR) to VC through BR() and Determine(). In phase 3, a VN begins to use its designated time slots to transmit data through Data\_transmit().

```

VC_elected()    // a VC is elected
BR()            // VNs issue bandwidth requests
Determine()     // to determine whether a BR is successful
Data_transmit() // use designated time slots to transmit data

Sloti         // Slot-ID
Input: K        // number of vehicular nodes
Input: n        // number of time slots in a TDMA frame
Input: NRand   // generate 1 to NRand time slots for data transmission

VC_elected(K) {
  Sloti = 0;
  VC = 0; // no VC is elected so far
  While (VC = 0) {
    For i = 1 to K do {
      Every node randomly chooses between 0 and 1}
      If (more than two nodes choose 1) // CFV collision
      Else if (all nodes choose 0) // all nodes are listening CFV
      Else // a VC is successfully elected
        VC = 1;
        break;
      Sloti = Sloti + 1;}
  Output: Sloti
}
BR(K) {
  all_complete = K - 1; // Initial to K - 1 VNs
  incomplete = 1;
  Remaining_slots = (Framei × n - Sloti);
  For i = 1 to K do {
    Num_of_sloti = Rand[1, NRand]; // generate multi-slots for data transmissions
  While (incomplete = 1) {
    incomplete = 0;
    If (first_round = 1) // The first TDMA frame
      For i = 1 to K do {
        Sloti = Rand[1, Remaining_slots]; //Every VN randomly selects a slot for BR
        Determine(); // Determine whether all BR are successfully
        first_round = 2; // Exit the first TDMA frame
      Else
        For i = 1 to K do {
          Sloti = Rand[1, n]; // every VN randomly selects a slot for BR

```

```

Determine();}
Output:  $Slot_i$ 
}
Determine() {
  For  $i = 1$  to  $K$ 
    For  $j = 1$  to  $K$ 
      If ( $j \neq i$  and  $Slot_j = Slot_i$ ) // determine whether two VN choose the same slot for BR
        incomplete = 1; // BR collision
      If (incomplete = 0) //BR is successful
        all_complete = all_complete - 1; // Decrement node number by one
      If (incomplete = 1) // enter the next TDMA frame
    }
  }
Data_transmit() {
  If ( $Num\_of\_slot_i = 1$ ) // using single-slot for data transmissions
  Else // using multi-slots for data transmissions
}

```

Fig. 5. Operations of intra-cluster communications.

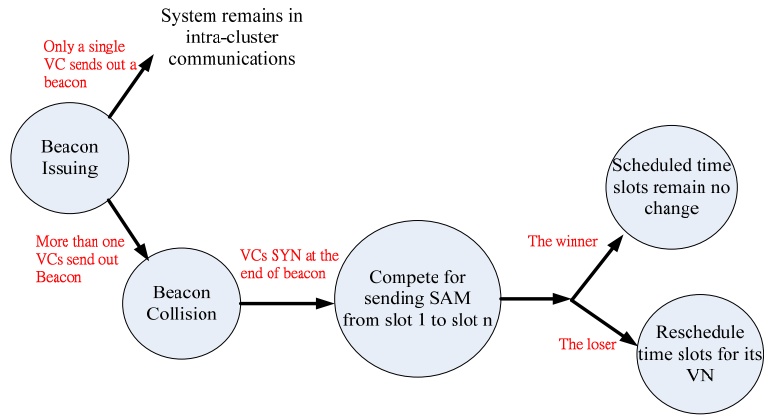


Fig. 6. State Transitions of inter-cluster communications.

### 3.2 Inter-Cluster Communications

Fig. 6 shows the five-state transition diagram of inter-cluster communications (inter-CC). The transition begins with Beacon-Issuing state. Periodically, a VC in a cluster will issue beacon signal in slot 0 of every TDMA frame. If no other beacon signal is detected, the cluster remains in the intra-cluster communications; else a collision of beacon signal implies that there is another VC of different clusters near-by. The two clusters are cooperating through the VC-to-VC contact to build the inter-cluster communications. First, these two VCs are synchronized at the end of beacon signal. From there, two slot allocation maps (SAMs), as shown in Fig. 3 (b), require for exchange. Two VC exchange SAM to each other using the simple transmit-and-listen scheme from slot 1 to slot  $n - 1$ . In other words, the one who successfully transmits SAM to the other is the winner, and the



one who successfully receives SAM becomes the loser. The winner-VC will not alter its scheduled time slots, while the loser-VC has to reschedule time slots for all the VN under its supervision. Fig. 7 shows the operations of Inter-cluster communications (Inter-CC). As it can be seen, Inter-CC consists of two phases. In phase 1, VCs are in the competition for sending SAM from slot 1 to slot  $n$  through `Compete_SAM()`. In phase 2, data are successfully transmitted using designated time slots through `Data_transmit()`.

```

Compete_SAM() // all VCs compete the sending of SAM
Data_transmit () // use the designated time slots for data transmissions

Sloti // Slot-ID
Input: KVC // number of VC
Input: n // number of time slots in a TDMA frame
Input: NRand // generate 1 to NRand time slots for data transmissions

Compete_SAM() {
  Sloti = 0;
  Success = 0; // SAM sending is not successful
  While (Success = 0) {
    For i = 1 to KVC do {
      Every VC randomly chooses between 0 and 1}
    If (more than two VC choose 1) // SAM collision
    Else if (all VC choose 0) // all VC are listening SAM
    Else // one VC has successfully sent out SAM
      Success = 1;
      The VC received SAM reallocates time slots for its VNs
      break;
      Sloti = Sloti + 1;
    }
  }
  Output: Sloti
}
Data_transmit() {
  Num_of_sloti = Rand[1, NRand]; // generate multi-slots
  If (Num_of_sloti = 1)
    // using single-slot for data transmissions
  Else
    // using multi-slots for data transmissions
}

```

Fig. 7. Operations of inter-cluster communications.

#### 4. ANALYSIS AND SIMULATION

First, we use probability model to derive mathematical equations. The performance metrics of our interests include the average number of time slots required for electing a VC, for bandwidth requests (BR), for SAM broadcasting, and the total number of time slots required for a VN to wait before data can be transmitted. To validate the mathematical results and to compare the performance with IEEE 802.11p, we conduct a simulation study using NS-2. In the simulation, we basically implement the CBT operations for intra- and inter-communications, the TDMA time slots, and the MAC frame formats.

#### 4.1 Mathematical Analysis

Prior to introducing the mathematical analysis, we list all the parameters to be used in Table 1.

**Table 1. Definition of parameters.**

Parameters	
$t$	Number of TDMA frames
$n$	Number of slots in a TDMA frame
$K_a$	Number of nodes in Cluster A
$K_b$	Number of nodes in Cluster B
$K_{VC}$	Number of Clusters
$PERM_c^d$	The permutation of selecting $d$ from $c$
$m$	MAC frame size (bytes)
$L_{SAM\ a}$	SAM size of Cluster A (bytes)
$L_{SAM\ b}$	SAM size of Cluster B (bytes)
$R$	Transmission bit rate (Mbps)
$T_s$	Duration of a slot

##### 4.1.1 Intra-cluster analysis

For intra-cluster analysis, we simply use cluster A as an example. Let  $P_{VC}$  be the probability of successfully electing a VC. Recall to Section 3, in a time slot, every competing vehicular node is in either of the two states, transmit or receive a CFV packet. Therefore, there are a total of  $2^{K_a}$  combination possibilities. Among them, only  $K_a$  combinations can be successful in electing a VC; the rest of them fail. Hence,  $P_{VC}$  can be derived from Eq. (1),

$$P_{VC} = \frac{K_a}{2^{K_a}}. \quad (1)$$

Next let  $x$  be the time slot in which VC is successfully elected. The probability density function of  $P_{VC}$  can be expressed as  $f(x) = (1 - P_{VC})^{x-1}P_{VC}$ , which implies that the election of VC fails for the first  $x - 1$  time slots and finally succeeds at the  $x$ th time slot. Let  $E[X]_{intra}$  be the average time slots required for electing a VC. We can compute  $E[X]_{intra}$  from Eq. (2). Note that one extra slot is added to the average, since in our design it requires one slot for synchronization.

$$E[X]_{intra} = \sum_{x=1}^{\infty} xf(x) + 1 = \frac{1}{P_{VC}} + 1 = \frac{2^{K_a}}{K_a} + 1 \quad (2)$$

Let  $P_{BR}^1$  be the probability of successful bandwidth requests (BR) issued from  $(K_a - 1)$  VNs by using the first TDMA frame. Since in the first frame, it requires  $E[X]_{intra}$  time slots to elect a VC, only  $(n - \lceil E[X]_{intra} \rceil)$  time slots remain for BR. For  $(K_a - 1)$  VNs, there are a total of  $(n - \lceil E[X]_{intra} \rceil)^{K_a-1}$  different combinations. An issuing BR can be suc-

successful (*i.e.*, without encountering any collisions) if and only if every VN chooses different time slots to issue its BR. Thus, there are  $PERM_{K_a-1}^{(n-\lceil E[X]_{intra} \rceil)}$  cases for all the  $(K_a - 1)N_s$  to achieve successful BR in the first TDMA frame. Let  $P_{BR}^i$ ,  $i = 2, 3, \dots, t$ , be the probability of successful BR issued from the failed VNs by using the second, the third, and up to the  $t$ th TDMA frame. It becomes more complicated to derive  $P_{BR}^i$ ,  $i = 2, 3, \dots, t$ , since a failed bandwidth request in the first TDMA frame will continue to contend with other failed VNs in the second TDMA frames, and so on. By deducting the successful BR in the previous TDMA frame, the average number of failed BR,  $N_F^i$  of the second ( $i = 2$ ), and of the follow-on TDMA frames ( $i = 3, 4, \dots, t$ ) can be expressed as

$$N_F^i = \begin{cases} (K_a - 1) - \left[ \sum_{j=1}^{K_a-1} \frac{PERM_j^{(n-\lceil E[X]_{intra} \rceil)}}{(n-\lceil E[X]_{intra} \rceil)^j} \right], & \text{if } i = 2, \\ N_F^{i-1} - \left[ \sum_{j=1}^{N_F^{i-1}} \frac{PERM_j^n}{n^j} \right], & \text{if } i = 3, 4, \dots, t. \end{cases} \quad (3)$$

Thus, we can compute the probability of successful BR issued from  $(K_a - 1)$  VNs by using the  $i$ th TDMA frame, *i.e.*,  $P_{BR}^i$ ,  $i = 1, 2, 3, \dots, t$ , from Eq. (4).

$$P_{BR}^i = \begin{cases} \frac{PERM_{K_a-1}^{(n-\lceil E[X]_{intra} \rceil)}}{(n-\lceil E[X]_{intra} \rceil)^{K_a-1}}, & \text{if } i = 1, \\ \frac{PERM_{N_F^i}^n}{n^{N_F^i}}, & \text{if } i = 2, 3, \dots, t. \end{cases} \quad (4)$$

Next, let  $E[Y]_{intra}$  be the average number of time slots required for the successful BR. We can compute  $E[Y]_{intra}$  from Eq. (5).

$$\begin{aligned} E[Y]_{intra} &= P_{BR}^1 \times (n - \lceil E[X]_{intra} \rceil) + (2 \times (1 - P_{BR}^1) \times P_{BR}^2 + \dots \\ &\quad + t \times (1 - P_{BR}^1)(1 - P_{BR}^2) \times \dots \times (1 - P_{BR}^{t-1}) \times P_{BR}^t) \times n \\ &= P_{BR}^1 (n - \lceil E[X]_{intra} \rceil) + \left( \sum_{i=2}^t (i \times P_{BR}^i \prod_{j=0}^{i-1} (1 - P_{BR}^j)) \right) \times n. \end{aligned} \quad (5)$$

After bandwidth requests, VC begins to broadcast SAM (slot allocation map) to VNs. Let  $L_{SAM_a}$  be the size of SAM (in bytes) for cluster A and  $SAM_{intra}$  be the number of slots required for VC in cluster A to broadcast SAM. Refer to the SAM format in Fig. 3 (b), in addition to 1-byte flag and length, 6-byte VC MAC address, and 4-byte CRC, a vehicular node also requires 6-byte MAC and 1-byte allocated slot number. Thus, we can calculate  $SAM_{intra}$  from Eq. (6), where  $R$  is the transmission bit rate (in Mbps) and  $T_S$  is the duration time of a time slot (in sec).

$$SAM_{intra} = \frac{L_{SAM} \cdot a \times 8}{T_S} = \frac{(7 \times (K_a - 1) + 11) \times 8}{T_S} = \frac{56 \times K_a + 32}{T_S \times R} \quad (6)$$

After the broadcasting of SAM, a VN can deliver its data over the designated time slots. Here, we assume two different types of slots are used for data transmissions, single-slot and multi-slots. Let  $E[Z]_{intra}$  be the average number of time slots required for waiting before a VN can begin to transmit its data. We can compute  $E[Z]_{intra}$  from Eq. (7).

$$E[Z]_{intra} = \begin{cases} \frac{\sum_{\tau=0}^{K_a-1} \tau}{K_a} = \frac{K_a - 1}{2}, & \text{if using single-slot,} \\ \frac{(1 + N_{Rand}) \times \sum_{\tau=0}^{K_a-1} \tau}{K_a} = \frac{(1 + N_{Rand}) \times (K_a - 1)}{4}, & \text{if using multi-slots.} \end{cases} \quad (7)$$

It is noticed that in Eq. (7), for single-slot, a node out of the  $K_a$  nodes has to wait from 0 to  $K_a - 1$  slots. Thus, the total waiting time of  $K_a$  nodes in single-slot is  $\sum_{\tau=0}^{K_a-1} \tau$ . For multi-slots, we assume the requested time slots of a VN follow Uniform distribution, where  $N_{Rand}$  is the maximum number of requested time slots of a VN. Thus, on average, every VN has to wait  $(\frac{1 + N_{Rand}}{2})$ . As a result, the total waiting time of  $K_a$  nodes in multi-slots is  $(\frac{1 + N_{Rand}}{2}) \times \sum_{\tau=0}^{K_a-1} \tau$ .

Finally, let  $S_{intra}$  be the total number of time slots counting from the time to elect a VC to the time when a node is ready to transmit its data. We can compute  $S_{intra}$  from Eq. (8).

$$S_{intra} = E[X]_{intra} + E[Y]_{intra} + SAM_{intra} + E[Z]_{intra} \quad (8)$$

#### 4.1.2 Inter-cluster analysis

Referring to Fig. 6, when two or more than two clusters are approaching, VCs of these clusters will compete for sending SAM from slot 0 to slot  $n$ . The allocated time slots of the winner-VC remain no change, while the loser-VCs have to reschedule time slots for their VNs to avoid any possible collisions. Let  $K_{VC}$  be the total number of VC and  $P_{SAM}$  be the probability of successfully transmitting SAM in a mini-slot. We can compute  $P_{SAM}$  from Eq. (9), where the numerator is the number of cases to successfully transmit SAM and the denominator is the total combinations if the number of VCs equals  $K_{VC}$ .

$$P_{SAM} = \frac{K_{VC}}{2^{K_{VC}}} \quad (9)$$

Next, let  $x$  be the mini-slot in which SAM can be successfully transmitted. The probability density function of  $P_{SAM}$  can be expressed as  $f(x) = (1 - P_{SAM})^{x-1} P_{SAM}$ , which implies that the transmission of SAM fails for the first  $x - 1$  mini-slots and eventually succeeds at the  $x$ th mini-slot. Let  $E[U]_{inter}$  be the average time slots required for a VC to successfully transmit SAM. We can compute  $E[U]_{inter}$  from Eq. (10). Notice that in our design there are two mini-slots in a time slot.

$$E[U]_{inter} = \left[ \frac{1}{2} \sum_{x=1}^{\infty} x f(x) \right] = \left[ \frac{1}{2 \times P_{SAM}} \right] = \left[ \frac{2^{K_{VC}-1}}{K_{VC}} \right] \quad (10)$$

Next, let  $L_{SAM\_a}$  be the SAM size of cluster A,  $L_{SAM\_b}$  be the SAM size of cluster B, and  $SAM_{inter}$  be the number of time slots required for two clusters (say, cluster A and B) to broadcast their SAMs. Since  $L_{SAM\_a} = 7 \times (K_a - 1) + 11$  and  $L_{SAM\_b} = 7 \times (K_b - 1) + 11$ ,  $SAM_{inter}$  can be computed from Eq. (11).

$$SAM_{inter} = \left( \frac{\frac{L_{SAM\_a} \times 8}{R}}{T_s} \right) + \left( \frac{\frac{L_{SAM\_b} \times 8}{R}}{T_s} \right) = \frac{56 \times (K_a + K_b) + 64}{T_s \times R} \quad (11)$$

Let  $E[W]_{inter}$  be the average time slots required for waiting before data can be transmitted. Similar to Eq. (7), by replacing  $K_a$  with  $(K_a + K_b)$ , we can compute  $E[W]_{inter}$  from Eq. (12).

$$E[W]_{inter} = \begin{cases} \frac{\sum_{\tau=0}^{K_a+K_b-1} \tau}{K_a + K_b} = \frac{K_a + K_b - 1}{2}, & \text{if using single-slot} \\ \frac{\left(\frac{1+N_{Rand}}{2}\right) \times \sum_{\tau=0}^{K_a+K_b-1} \tau}{K_a + K_b} = \frac{(1 + N_{Rand}) \times (K_a + K_b - 1)}{4}, & \text{if using multi-slots} \end{cases} \quad (12)$$

Finally, let  $S_{inter}$  be the total number of time slots counting from the time when SAM is successfully transmitted by a VC to the time when a node is ready to transmit its data. We can compute  $S_{inter}$  from Eq. (13).

$$S_{inter} = E[U]_{inter} + SAM_{inter} + E[W]_{inter} \quad (13)$$

#### 4.2 Analytical and Simulation Results

As shown in Fig. 8, the average number of time slots for electing a VC is exponent-

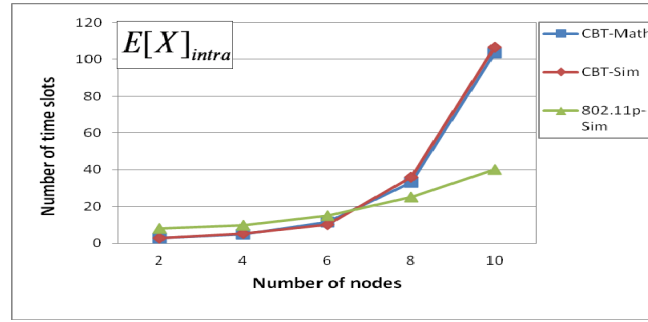


Fig. 8. Average number of time slots for electing a VC.

tially increased as the number of competing nodes increases from 2 to 10. We observe that the mathematical results, computed from Eq. (2), are very close to the simulation results; yet, simulation results are more realistic, since in the simulation, random number is used for a VN to decide whether to transmit or listen. To compare with the CSMA/CA mechanism used in IEEE 802.11p, more simulations are performed using NS-2. As shown in Fig. 8, when the number of VNs is smaller than or equal to 6, the proposed CBT requires fewer time slots in electing a VC. This is because using backoff algorithms may become too conservative for small number of VNs.

Fig. 9 shows the average number of time slots required for BR in CBT as a function of increasing VNs. When the number of time slots in a TDMA frame is small (*e.g.*,  $n = 30$ ), mathematical results are quite close to the simulation results. However, as  $n$  is increased to 70, mathematical results are increasingly far apart from the simulation results; in fact, the difference increases from 2% to 26% as the node increases from 2 to 10. This phenomenon can be understood from two aspects: (i) mathematical equations, referring to Eq. (3), can not accurately determine which nodes would successfully complete BR in a TDMA frame, and (ii) in the mathematical model, it is no way to determine which time slot is employed by which vehicular node. However, the above two uncertainty can be easily conquered in simulation by assigning an ID to each VN.

Referring to Eq. (7), in CBT the average number of time slots required for waiting before data transmission, *i.e.*,  $E[Z]_{intra}$ , can be computed for single-slot and multi-slots, respectively. From Fig. 10, it is quite straightforward that  $E[Z]_{intra}$  would increase as the number of VNs increases. For single-slot transmissions, the percentage of increase is the smallest. For multi-slot transmissions, when  $N_{rand}$ , a random number used to decide the number of time slots requested for BR, is slightly increased from 4 to 7, we observe that  $E[Z]_{intra}$  increases very significantly. Additionally, we observe that mathematical results can match with the simulation quite well.

Fig. 11 shows the total number of time slots required before a VN can begin to transmit its data on designated slots. In general, as the number of time slots in a TDMA frame (*i.e.*,  $n$ ) is increased,  $S_{intra}$  increases very significantly. From Fig. 11, we can easily observe that mathematical curves are almost in accordance with the simulation curves. For single-slot data transmissions, the reason for  $n$  to dominate  $S_{intra}$  is that the average time slots required for BR (*i.e.*,  $E[Y]_{intra}$ ) increases largely as  $n$  is increased. On the other hand, for multi-slot data transmissions, the factors that can distinguish  $S_{intra}$  come from  $N_{Rand}$  as well. It is observed that by enlarging  $N_{Rand}$  from 3 to 7,  $S_{intra}$  can greatly increase.

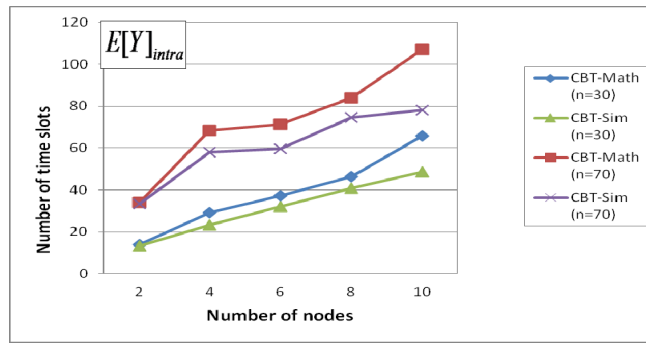


Fig. 9. Average number of time slots for BR in CBT.

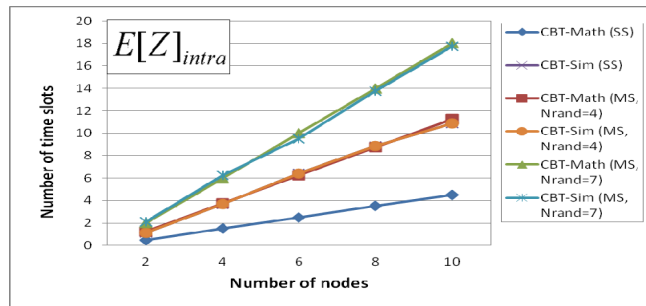


Fig. 10. Average number of time slots waiting before data transmissions.

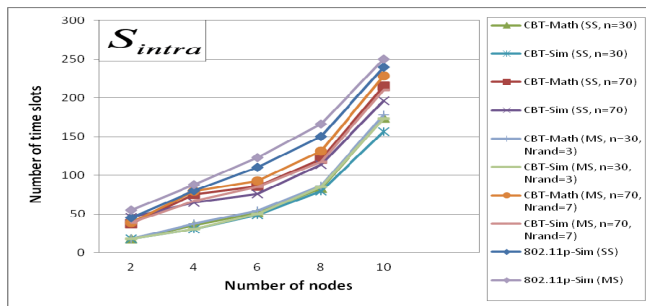


Fig. 11. Total number of time slots required before data transmissions.

This phenomenon reveals that in intra-cluster communication the major dominating factor for a VN to wait before its data can be transmitted is the number of time slots requested during BR. To compare the performance between CBT and IEEE 802.11p, more simulations are conducted using NS-2. From the simulation results as shown in Fig. 11, CBT consumes fewer time slots than 802.11p, no matter in single-slot or multi-slot transmissions. This is because CBT employs TDMA and it efficiently utilizes pre-scheduled time slots to transmit data, while using a backoff algorithm in 802.11p will exponentially increase the waiting time, particularly when the number of joining VNs is increased. Consequently, the latter waits more time slots before a VN can effectively transmit data.

Finally, Fig. 12 shows the average number of time slots required in inter-cluster communication. In this analysis, for simplicity, we assume only two clusters (Clusters A and B) exist. When the total number of VNs in these two clusters increases from 4 to 32, we observe that all curves of  $S_{inter}$  increase almost linearly. Among them, for single-slot data transmissions,  $S_{inter}$  increases very smoothly. However, for multi-slot transmissions,  $S_{inter}$  increases very rapidly no matter when  $N_{rand}$  is equal to 7 or 12. By referring to Eq. (10) to (12), since mathematical results of  $E[U]_{inter}$  and  $E[W]_{inter}$  do not make big difference, it is observed that mathematical results are only slightly different from the simulation results. To make a performance comparison, we run simulations for CBT and IEEE 802.11p using NS-2. From Fig. 12, we can observe that multi-slot transmissions in 802.11p consume the maximum number of time slots, and single-slot transmissions in 802.11p take the second. Even though CBT has an overhead that requires the exchange of SAM between two clusters, it significantly outperforms 802.11p, particularly when the number of joining VNs to the two clusters is increased. The main reason for CBT outperforming 802.11p is quite straightforward; *i.e.*, the latter employs a backoff algorithm which exponentially increases the waiting time along with the increase of joining VNs.

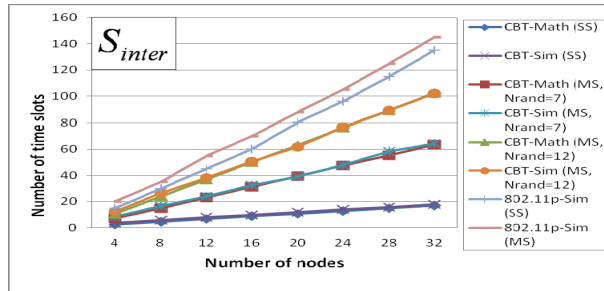


Fig. 12. Average number of time slots required in Inter-CC.

### 4.3 Protocol Overhead

As compared to 802.11p, the proposed CBT may incur extra cost and protocol overhead. First, to form a cluster, CBT must synchronize time slots among VNs. Thus, to enable CBT, a VN must be equipped with a GPS. Second, as shown in Figs. 3 (a) and (b), the first slot (*i.e.*, slot 0) in each TDMA frame is reserved for SYN and SAM; it cannot be used for transferring user data. It is noticed that this overhead is reduced when the number of time slots in a TDMA frame (*i.e.*,  $n$ ) increases. Furthermore, the overhead does not increase along with the increase of VNs. This is because in our design the number of VNs supported in a cluster (*i.e.*,  $K$ ) is fixed when the size of MAC frame (*i.e.*,  $m$ ) is determined. In fact, after a simple calculation, we can easily derive  $K$  from  $m$ ; *i.e.*,  $K = \frac{(m-34)}{14}$ .

For inter-cluster communication, in addition to the above-mentioned overhead, CBT will further require the exchange of SAM between any two clusters. The size of SAM is equal to  $\frac{m-8-4}{2}$  bytes, as it is defined in Section 2.2 and illustrated in Fig. 3 (b).



## 5. CONCLUSIONS

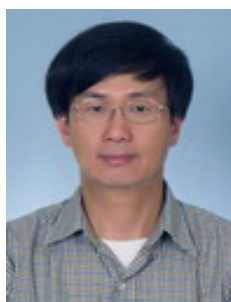
In this paper, we have presented a cluster-based TDMA (CBT) system for inter-vehicle communications. One of the novelties of the proposed CBT is right in that it uses a relatively simple transmit-and-listen scheme to timely elect a VC and to quickly resolve the collisions when two clusters are approaching together. Performance of the proposed CBT was analyzed by deriving mathematical equations to compute the average number of time slots for electing a VC, for presenting BR, and the total number of time slots required before data can be effectively transmitted. The analytical results were finally validated by a simulation using NS-2. From the analytical and simulation results, we have observed that in forming a small-sized cluster the proposed CBT spends less time than IEEE 802.11p. Additionally, when the number of joining VNs is increased, CBT takes less waiting time before a VN can effectively transmit data.

The proposed CBT may co-exist with the existing 802.11p. For example, CBT can be engaged initially for a small-sized cluster, and if many VNs are competing for a VC, 802.11p can be invoked subsequently. The protocol primitives to be added in order to switch over between these two schemes should include a threshold in terms of the number of collisions for compete-for-VC (CFV) message. In the future works, the proposed CBT can be extended by considering different traffic types with priorities. For example, real-time traffic with higher-priority should possess more privilege to acquire time slots than non-real-time traffic with lower-priority.

## REFERENCES

1. C. Wang, B. Li, and L. Li, "A new collision resolution mechanism to enhance the performance of IEEE 802.11 DCF," *IEEE Transactions on Vehicular Technology*, Vol. 53, 2004, pp. 1235-1246.
2. B. Shrestha, D. Niyato, Z. Han, and E. Hossain, "Wireless access in vehicular environments using BitTorrent and bargaining," in *Proceedings of IEEE Conference on Global Telecommunications Conference*, 2008, pp. 1-5.
3. F. Yu and S. Biswas, "Self-configuring TDMA protocols for enhancing vehicle safety with DSRC based vehicle-to-vehicle communications," *IEEE Journal on Selected Areas in Communications*, Vol. 25, 2007, pp. 1526-1537.
4. 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, 2009, pp. 349-366.
5. H. Su and X. Zhang, "Clustering-based multichannel MAC protocols for QoS provisionings over vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, Vol. 56, 2007, pp. 3309-3323.
6. R. T. Goonewardene, F. H. Ali, and E. Stipidis, "Robust mobility adaptive clustering scheme with support for geographic routing for vehicular ad hoc networks," *IEEE Journals on Intelligent Transport Systems*, Vol. 3, 2009, pp. 148-158.
7. Z. Y. Zaydoun and S. M. Mahmud, "Toward strongly connected clustering structure in vehicular ad hoc networks," in *Proceedings of IEEE Conference on Vehicular Technology*, 2009.

8. Z. Wang, L. Liu, M. C. Zhou, and N. Ansari, "A position-based clustering technique for ad hoc intervehicle communication," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 38, 2008, pp. 201-208.
9. X. Zhu, L. Shen, and T. S. P. Yum, "Hausdorff clustering and minimum energy routing for wireless sensor networks," in *Proceedings of IEEE Transactions on Vehicular Technology*, Vol. 58, 2009, pp. 990-997.
10. Y. C. Chu and N. F. Huang, "Delivering of live video streaming for vehicular communication using peer-to-peer approach," in *Proceedings of IEEE Conference on Mobile Networking for Vehicular Environments*, 2007, pp. 1-6.
11. C. Guo, R. Hekmat, and P. Pawelczak, "Analysis and optimization of energy efficient cluster forming for wireless sensor networks," in *Proceedings of IEEE Conference on Vehicular Technology*, 2007, pp. 127-132.
12. P. Fan, "Improving broadcasting performance by clustering with stability for intervehicle communication," in *Proceedings of IEEE Conference on Vehicular Technology*, 2007, pp. 2491-2495.
13. K. Saito and M. Nakayama, "The spatial reuse slot allocation with reallocating neighbor links in the TDMA-based wireless mesh network," in *Proceedings of IEEE Conference on Vehicular Technology*, 2007, pp. 1479-1483.
14. J. Du and W. Shi, "App-MAC: An application-aware event-oriented MAC protocol for multimodality wireless sensor networks," *IEEE Transactions on Vehicular Technology*, Vol. 57, 2008, pp. 3723-3731.
15. T. A. Kostas, A. H. Haddad, and C. C. Lee, "Performance analysis of a dynamic bandwidth allocation algorithm for a mixed traffic TDMA network," *IEEE Transactions on Vehicular Technology*, Vol. 56, 2007, pp. 2306-2320.
16. J. Ma, K. Yang, and S. Ou, "Nimble and adaptive time-division multiple access control phase algorithm for cluster-based wireless sensor networks," *IET Communications*, Vol. 1, 2007, pp. 179-186.
17. H. Su, X. Zhang, and H. H. Chen, "Cluster-based multi-channel communications protocols in vehicle ad hoc networks," *IEEE Journals on Wireless Communications*, Vol. 13, 2006, pp. 44-51.
18. R. Ding and Q.-A. Zeng, "A clustering-based multi-channel vehicle-to-vehicle (V2V) communication system," in *Proceedings of the 1st International Conference on Ubiquitous and Future Networks*, 2009, pp. 83-88.



**Tsang-Ling Sheu (許蒼嶺)** received the Ph.D. degree in Computer Engineering from the Department of Electrical and Computer Engineering, Penn State University, University Park, Pennsylvania, USA, in 1989. From September 1989 to July 1995, he worked with IBM Corporation at Research Triangle Park, North Carolina, USA. In August 1995, he became an Associate Professor, and was promoted to Full Professor in January 2006 at the Department of Electrical Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan. His research interests include mobile communications, wireless networks, and multimedia net-

working. He was the recipient of the 1990 IBM Outstanding Paper Award. Dr. Sheu is a senior member of the IEEE, and the IEEE Communications Society.



**Yu-Hung Lin (林昱宏)** received the B.S. degree from the Department of Information Engineering, I-So University, Kaohsiung, Taiwan, in 2007, and the M.S. degree from the Department of Electrical Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan, in 2010. During his M.S. study, his research interest was in the vehicular ad-hoc networks (VANET).