

## Research Article

# Non-GPS Data Dissemination for VANET

**Kulit Na Nakorn and Kultida Rojviboonchai**

*Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10400, Thailand*

Correspondence should be addressed to Kultida Rojviboonchai; [kultida.r@chula.ac.th](mailto:kultida.r@chula.ac.th)

Received 12 April 2013; Revised 27 October 2013; Accepted 12 November 2013; Published 29 January 2014

Academic Editor: Xu Li

Copyright © 2014 K. Na Nakorn and K. Rojviboonchai. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Fast, reliable, and efficient data dissemination in VANET is a key of success for intelligent transportation system. This requires a broadcasting protocol which has efficient forwarder nodes and an efficient broadcasting mechanism. In this paper, we propose a self-decision algorithm that allows a node to know that it belongs to a member of connected dominating set or not. The algorithm is a combination of density based algorithm and topology based algorithm, called “DTA.” The algorithm does not require any geographical knowledge. Therefore, it can avoid violating a privacy issue. Moreover, the algorithm can resist inaccurate data than position-base algorithms that need high frequent beaconing for accurate data. The simulation results show that our algorithm provides the highest coverage results compared to existing solutions. We also propose a new broadcasting protocol, called “NoG.” NoG consists of a broadcasting mechanism, a waiting timeout mechanism, and a beaconing mechanism. The proposed protocol operates without any geographical knowledge and provides reliable and efficient data dissemination. The performance is evaluated with a realistic network simulator (NS-3). Simulation results show that NoG with DTA outperforms other existing protocols in terms of reliability and data dissemination speed.

## 1. Introduction

Vehicular ad hoc network (VANET) is one of mobile ad hoc networks (MANET). Vehicles in the network are equipped with wireless communication devices. Therefore, they can directly communicate to each other without infrastructure and without centralized control. The data can be quickly delivered to applications. This can support some applications of intelligent transportation system (ITS) such as driver assistant or safety transport applications. These applications need a fast and reliable solution for data dissemination to provide accurate and reliable services [1]. So the efficient data dissemination is one of the key successes for such applications. To achieve this, the unique characteristic of vehicle environment should be considered. Vehicle's movement changes frequently and rapidly. The speed of vehicles also affects wireless signal that leads to high intermittent connectivity occurrences between vehicles. Moreover, vehicles may be very densely packed in urban areas but are very sparse on highway roads or in rural areas.

A traditional approach for data dissemination or broadcasting for wireless ad hoc networks is simple flooding.

Simple flooding does not require any information from environment or nodes. A packet is rebroadcasted once by every received node. This approach can provide very high data dissemination speed. However, simple flooding may cause the contention and collision [2] due to its redundant transmission in dense areas and it may cause useless broadcasting as there is no neighbor to receive data in sparse area [3]. Epidemic protocol [4] was proposed to improve the performance in sparse areas by using store and forward technique. So upon receiving of a broadcasting packet, nodes will store the packet and forward it later when nodes meet a new neighbor. Then this technique has been employed to most of broadcasting protocols in VANET because it can handle the intermittent connectivity issue. As a result, reliability or delivery ratio is increased.

In VANET, several reliable broadcasting protocols have been proposed. We can categorize these reliable broadcasting protocols into 2 groups by their main algorithm. In the first group, the protocols make their decision based on node position such as EAEP [5], APBSM [6], POCA [7], and DV-Cast [8]. These protocols prefer nodes at the edge of broadcasting circular to rebroadcast the packets. All of

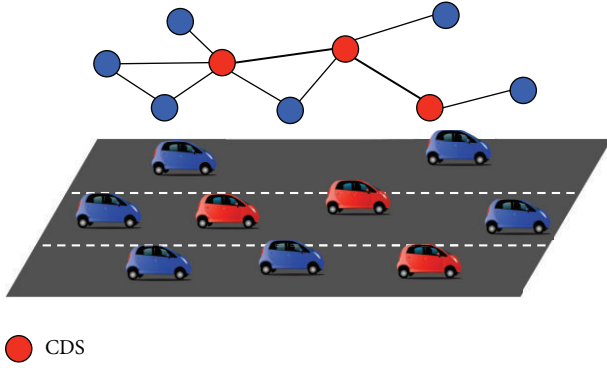


FIGURE 1: An example of connected dominating set (CDS).

protocols in this group rely on geographical knowledge. They use position or direction of nodes to make decision. In the second group, the protocols make their decision based on node's properties such as DECA [9]. The properties of nodes that are used in these protocols are number of one-hop neighbors (density) or relation between nodes and their neighbors (topology). So these protocols do not require any geographical information to make decision.

However, every algorithm in every protocol has the same goal. The goal is to minimize the number of rebroadcast nodes that can cover all of their neighbors in each group. This can minimize number of retransmissions for delivering a packet to most of nodes in networks. This problem can be solved by minimum connected dominating set (CDS). This algorithm can construct graph and select the minimum number of nodes to cover 100% of their neighbor nodes in each group as shown in Figure 1, but the algorithm requires global knowledge and the CDS computation is an NP-complete problem [10]. Therefore, an approximation algorithm is a practical solution that can construct CDS. Some previous works have been proposed for general mobile ad hoc networks such as [11–14]. These algorithms are self-decision algorithm. This means each node will decide by itself whether it is in CDS or not. Most of them make decision based on topology properties. However, these algorithms have high complexities and they are not specifically designed for vehicular environment.

In this paper, we focus on a nongeographical knowledge based CDS forming algorithms. These methods can avoid privacy issue that most of users are concerned with. Moreover, the nongeographical knowledge based algorithms can resist inaccurate data than position-base algorithms that need high frequent beaconing for accurate position data. We propose a hybrid algorithm, that is, a combination of density based algorithm and topology based algorithm (DTA). DTA has advantage points from both density based algorithm and topology based algorithm. The density based algorithm is a simple algorithm that works well in simple connection scenarios. On the other hand, a topology based algorithm is a complex algorithm that efficiently works in complex connection scenarios. So DTA is an appropriate algorithm for vehicular environment that has such a dynamic topology. We evaluate our algorithm by simulations. The evaluation

is focused on coverage results and the ratio between CDS members to total nodes. DTA can provide the highest coverage results than other algorithms.

We also propose a nongeographical broadcasting protocol (NoG). It is designed to provide the fast, reliable, and efficient data dissemination in VANET. The broadcasting protocol consists of a broadcasting mechanism, a waiting timeout mechanism, and a beaconing mechanism. NoG is implemented with our proposed DTA algorithm in NS-3. The simulation results show that NoG with DTA outperforms other previous protocols in terms of reliability and data dissemination speed. NoG also operates well with other algorithms.

The rest of this paper is organized as follows. In Section 2, the related works are discussed. Section 3 describes the overview and details of density and topology based CDS forming algorithm (DTA). Section 4 describes the overview and details of nongeographical Broadcasting Protocol (NoG). The performance evaluation is reported in Section 5. Finally, this paper is concluded in Section 6.

## 2. Related Work

We discuss more details on the previously mentioned protocols in Section 2.1 and the nongeographical knowledge algorithm is discussed in Section 2.2.

**2.1. Broadcasting Protocol in VANET.** Simple flooding is a traditional approach for broadcasting. It provides very high data dissemination speed, but all nodes will participate in rebroadcasting packets. This causes the broadcast storm problem due to redundant retransmissions. Epidemic protocol [4] is the most simple store and forward protocol. It can handle an intermittent connectivity in VANET, but all nodes still rebroadcast packets the same as simple flooding. So the broadcast storm problem is still found in epidemic protocol.

There are many previous broadcasting protocols for VANET that we have found in the works of the literature. These protocols use store and forward technique to handle intermittent connectivity that frequently occurs in vehicular environment. All protocols reduce the number of redundant retransmissions by self-decision algorithm. We can categorize these protocols into two groups based on their self-decision algorithm. The first group makes a decision based on position of node. These protocols prefer nodes at the edge of broadcasting circular to rebroadcast the packets. All of protocols in this group rely on geographical knowledge (GPS). The protocols use position or direction of nodes to make decision. This can cause privacy issue [15–17] because nodes need to broadcast their location to the others to exchange the geographical knowledge. A malicious node can track the past and current positions of these nodes. The treats that are from position tracking are discussed in [18]. The protocols in this group include PGB [19], POCA [7], EAEP [5], POCA [7], DV-Cast [8], and APBSM [6].

PGB [20] (preferred group broadcast) is a broadcasting mechanism in CAR protocol. When nodes receive a packet, they calculate the waiting timeout. A node with the shortest

timeout will rebroadcast the packet. Nodes at the edge of broadcasting circular have shorter waiting timeout than nodes that are closer to the source. However, PGB is used for routing information broadcasting, so it does not concern about a reliability issue.

EAEP [5] (edge-aware epidemic protocol) uses both the waiting timeout and probabilistic function. The waiting timeout is calculated by distance between nodes and source nodes. While the waiting timeout does not expire, nodes will count number of redundant retransmissions. The number of redundant retransmissions is used to calculate rebroadcast probabilistic value. Nodes at the edge of broadcasting circular have higher probability value than other nodes.

POCA [7] (position-aware broadcasting protocol) uses the geographical knowledge to select the next rebroadcast node. A node, that is, the furthest node to source node, will be selected by source node. The source node piggybacks the selected node's identifier to the broadcasting packet. The selected node will immediately rebroadcast once it receives the packet. This mechanism avoids the delay from waiting timeout.

DV-Cast [8] (distributed vehicular broadcast protocol) uses the broadcast suppression mechanism. A node, that is, the furthest node to source node, has the shortest waiting timeout, but if a node meets another node in the same direction of broadcasting packets, it will immediately rebroadcast the packets. This is because a node in the same direction of packets can help source node to forward the packets while it is running.

APBSM [6] (acknowledged parameterless broadcasting in static to highly mobile wireless ad hoc) is an extended version of PBSM. Nodes in APBSM use position of their neighbors to construct CDS. The CDS is calculated by Stojmenovic's algorithm [14], which is a combination of self-decision CDS forming from Wu and Li's algorithm and rebroadcast node elimination in scalable broadcast algorithm (SBA). Both of Wu and Li's algorithm and SBA will be discussed later in Section 2.2. Stojmenovic's algorithm uses geographical knowledge to select CDS members. In the case that a node is a CDS member, it will set shorter waiting timeout than other nodes. While timeout does not expire, the algorithm uses the rebroadcast node elimination the same as in SBA.

The other group of protocols makes a decision by node properties. The decision relies on comparison of node properties, so the protocols in this group can avoid using the geographical knowledge. These protocols use the density information (number of 1-hop neighbors) or the topology information, such as a list of 2-hop neighbors and relationship between neighbor nodes. The interesting protocol in this group is DECA [9].

DECA [9] (density-aware broadcasting protocol) relies on only the density information. A source node makes a decision by selecting its neighbor with the highest number of 1-hop neighbor nodes. Upon receiving the packet, the selected node will immediately rebroadcast it to avoid delay from waiting timeout. DECA also uses an adaptive beaconing mechanism to reduce overhead in dense areas.

However, most of nongeographical knowledge protocols are designed for general mobile ad hoc networks. But the CDS forming algorithms for these protocols are interesting because they can operate without any geographical knowledge.

*2.2. Nongeographical Knowledge CDS Forming Algorithm.* These CDS forming algorithms efficiently select CDS members and they also eliminate unnecessary retransmissions without any geographical knowledge.

Wu and Li's algorithm [11] proposed a self-decision algorithm to determine nodes in CDS, called gateway node. To be a CDS member, a node has to pass all three conditions. The first condition is an intermediate node condition. A node has to have at least two neighbors that are not directly connected to each other. The second condition is an intergateway node condition. A node has to have at least one neighbor, that is, not covered by its other neighbors. Let  $N_A$  be a set of node  $A$ 's neighbors and  $N_{NB}$  a set of neighbor nodes of  $A$ 's neighbors. If  $N_A \subseteq N_{NB}$ , node  $A$  will be eliminated from CDS because all of  $A$ 's neighbors can be covered by its other neighbors. The final condition is a gateway condition. A gateway node has at least a neighbor, that is, not covered by a pair of gateway node's neighbors and these two neighbors also are neighbors of each other. For example, let node  $A$  be a node that considers its gateway condition.  $A$  needs to have at least a neighbor ( $D$ ), that is, not covered by a pair of  $A$ 's connected neighbors ( $B$  and  $C$ ). If  $A$  is a gateway node, the neighbor ( $D$ ) is not covered by  $B$  or  $C$ . Therefore,  $N_A$  is not a neighbor of  $B$  or  $C$ . Let  $N_A$  be a set of node  $A$ 's neighbors,  $N_B$  a set of node  $B$ 's neighbors, and  $N_C$  a set of node  $C$ 's neighbors.  $B$  and  $C$  are neighbors of node  $A$ . If  $\{B, C\} \in N_A$ ,  $\{C\} \in N_B$ ,  $\{B\} \in N_C$  and  $N_A \subseteq N_B \cup N_C$ , node  $A$  will be eliminated from CDS. Therefore, nodes in CDS are only the necessary nodes for covering the other nodes in the group.

LENWB [12] (lightweight and efficient network-wide broadcast) uses a set of 1-hop neighbors to eliminate unnecessary rebroadcast nodes. When nodes receive a packet, they will estimate the neighbor list of source node by number of their 1-hop neighbors. If a source node has higher number of 1-hop neighbors than the received nodes, this means that the source node may cover all neighbors of received nodes so the received nodes will not rebroadcast the packets. Otherwise the received nodes will randomly set backoff delay and rebroadcast the packet. If nodes have the same number of 1-hop neighbors, the algorithm will compare with values of node identifiers.

SBA [13] (scalable broadcast algorithm) has the similar elimination algorithm as found in LENWB. Upon receiving the broadcasting packet, nodes calculate the waiting timeout. While the waiting timeout does not expire, nodes will remove the rebroadcast nodes' neighbors from their neighbor list. If the neighbor list does not empty after waiting timeout, they will immediately rebroadcast the packet.

These algorithms are based on topology properties. They use 1-hop neighbor list or 2-hop neighbor list to select the CDS members. The advantage is that these algorithms do not require any geographical knowledge, but they are designed

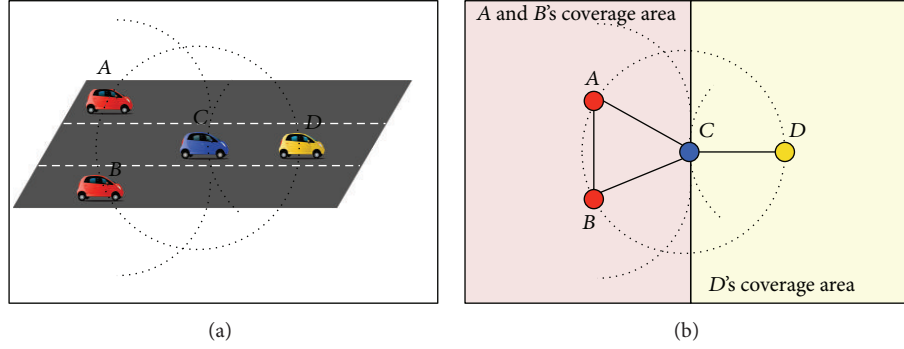


FIGURE 2: An example of gateway condition.

for general mobile ad hoc networks that may not be efficient in vehicular environment.

### 3. New Density and Topology Based CDS Forming Algorithm

Section 3.1 presents the motivation and the new density and topology based algorithm overview. Section 3.2 describes the details of the proposed algorithm.

**3.1. Motivation and Overview.** The unique characteristic of vehicular environment is the speed of nodes. Node's movement changes frequently and rapidly. So beacon messages have to be frequently broadcasted to provide the accurate geographical knowledge to position based protocols. This can cause the broadcast storm problem from beacon transmission. The information from equipment like GPS device also does not provide accurate data due to GPS drift. Moreover, broadcasting location data that can be tracked by unknown people may be concerned as privacy violation [15–17]. Therefore, we propose a new algorithm for CDS forming that does not require any geographical knowledge. It uses only density information (number of 1-hop neighbors) and 2-hop neighbor list that can be exchanged by beacon message.

Another interesting characteristic of vehicle environment is that vehicles always form groups. The vehicle environment is a nonuniform distribution and the topologies are mixed with very dynamic density environment; for example, the density is very sparse in highway scenarios, but nodes are very densely packed at the middle of intersection in urban areas. The algorithms need to be adaptable to each environment. So the algorithm should consider a node with the highest number of 1-hop neighbors to rebroadcast a packet because it can maximize a number of received nodes while minimizing a number of rebroadcast nodes. This algorithm works well for all sizes of group in every scenario. Therefore, DTA uses the number of 1-hop neighbors as a primary condition for algorithm. A node with the highest number of 1-hop neighbors is a CDS member.

However, only nodes with the highest density cannot cover all nodes in high density and complex scenarios, so DTA uses a topology based decision to increase the coverage results. In the case that nodes do not satisfy the density

condition, they will use a topology based condition for their decision. Our topology based decision is a simplified version of Wu and Li's algorithm. DTA employs only the gateway condition, that is, the most important condition especially on vehicular environment because the vehicular environment (a road) consists of narrow and long distance topology. The standard width of a road in US is 3.4 meters in each lane [18], but the maximum transmission range of 802.11p is up to 1000 meters [20]. Therefore, the width of the road is much less than the width of transmission range. For example, a pair of connected neighbors (A and B) can cover the red area behind node C as shown in Figure 2. If node D does not exist in this scenario, C will be at the edge of the group, so C is unnecessary to rebroadcast the message. Otherwise, if D exists, C is a connector between A, B (red area) and D (yellow area). In this case, C is considered as a gateway node because C has a neighbor (D), that is, not covered by a pair of C's connected neighbors (A and B). This scenario shows that the gateway condition is an important condition for CDS member selection.

**3.2. Algorithm Detail.** Upon receiving of a new beacon, a node always updates its CDS state. There are two conditions for checking CDS state. First, a node has to check a density based condition. If a node has the highest number of neighbors compared to its neighbors, it will be a CDS member. The other nodes that do not have the highest density will use a topology based condition. If they complete the condition, they will be CDS members. Otherwise they are not the CDS members. The procedure of DTA can be described as shown in Procedure 1.

## 4. New Nongeographical Knowledge Broadcasting Protocol for VANET

Nongeographical Knowledge Broadcasting Protocol (NoG) consists of three main modules: (1) broadcast mechanism that uses our DTA for CDS forming, (2) waiting timeout mechanism that is used for collision avoidance, and (3) beacon mechanism that helps nodes to exchange their local information and it helps nodes to detect the missing packet. Section 4.1 describes the protocol mechanism overview



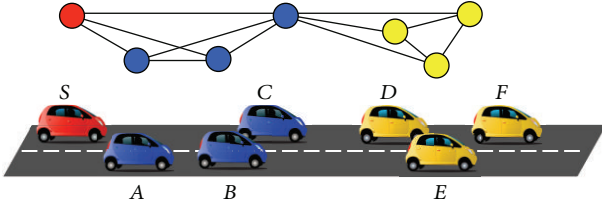


FIGURE 3: A normal broadcasting scenario.

and Section 4.2 explains the details of each module in our protocol.

**4.1. Protocol Overview.** Our proposed broadcasting protocol is a store and forward protocol with adaptive beacon intervals. A node uses beacon to exchange its information between its neighbors. The beacon includes a number of 1-hop neighbors, a 1-hop neighbor list, and a received packet identifier list. A node in protocol makes a decision by itself from this information whether to be a CDS member or not. If it is a CDS member, upon receiving the broadcasting packet, it randomly sets very short backoff delay ( $<10$  ms.). After the delay expires, it immediately rebroadcasts the packet. The nodes that are not CDS members set their waiting timeout with longer period than CDS members. While waiting timeout does not expire, they are listening to rebroadcasting from the other nodes. If they hear any rebroadcasting of the same packet in their waiting list, they will remove this packet from their waiting list to avoid redundant retransmissions.

For intermittent connectivity scenarios, NoG can detect a missing packet via an acknowledgement from the beacon. If there are some missing packets, a node will set their waiting timeout. If other nodes do not rebroadcast the packet before its waiting timeout expires, it will retransmit this packet to its neighbors.

Let us show the examples of protocol behaviors in a normal broadcasting scenario and in an intermittent connectivity scenario.

Figure 3 shows a normal broadcasting scenario.  $S$  is a source node. Let  $C$  be a node that has the highest local density, so  $C$  will be a CDS member. When  $S$  broadcasts a packet,  $A$ ,  $B$ , and  $C$  receive the broadcasting packet.  $A$  and  $B$  calculate their waiting timeout and wait for rebroadcasting from CDS members.  $C$ , that is, a CDS member, will randomly set very short backoff delay before it rebroadcasts the packet. In the case that  $C$  correctly rebroadcasts the packet,  $A$  and  $B$  will cancel their waiting timeout to avoid redundant retransmissions. On the other hand, if  $C$  does not rebroadcast the packet, one of  $A$  and  $B$  that has the shortest waiting timeout will rebroadcast the packet. Let  $B$  have the shortest waiting timeout, so  $B$  rebroadcasts the packet instead of  $C$ .  $A$  will cancel its waiting timeout not causing redundant retransmission. This mechanism will occur until all nodes in the group receive the packet or until the packet is expired.

In another case, there is an intermittent connectivity scenario. A node needs to retransmit the packet between groups of nodes. The scenario is illustrated in Figure 4. Nodes  $A$ ,  $B$ , and  $C$  already received the broadcasting packet from  $S$ . When

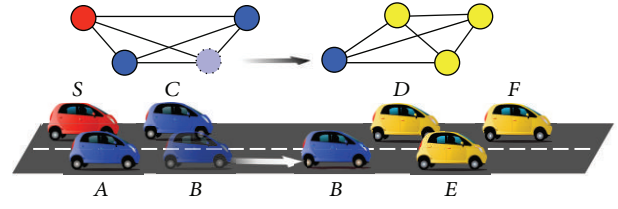


FIGURE 4: An intermittent connectivity scenario.

$B$  overtakes other vehicles, it leaves from the old group and joins a new group. Nodes in a new group are  $D$ ,  $E$ , and  $F$ . They never receive the broadcasting packet from  $S$ .  $B$  can detect the missing packet via acknowledgement from  $D$ ,  $E$ , and  $F$ 's beacon.  $B$  will set its waiting timeout and it will rebroadcast the packet to other nodes. When  $D$ ,  $E$ , and  $F$  receive the packet, then they act as the normal broadcasting scenario. The members of CDS almost immediately rebroadcast the packet and others set the longer waiting timeout than CDS members. The mechanism occurs until all of nodes receive the packet or until the packet is expired.

**4.2. Protocol Detail.** Each node in NoG has two lists: *neighbor list* and *broadcast list*. *Neighbor List* maintains identifiers of all 1-hop neighbors and their neighbor information (a number of 1-hop neighbors and a 1-hop neighbor identifier list). When nodes receive a new beacon, they will update their *Neighbor List* and they also update their CDS state. The neighbor entry will be removed if nodes do not receive an updated beacon from their neighbors within the next beacon intervals so nodes can avoid using stale information from the neighbors that currently stay out of their transmission range. *Broadcast List* maintains the identifiers of broadcasting packets and their waiting timeouts. *Broadcast List* is a list of packets that are waiting to be rebroadcasted. An entry of *Broadcast List* will be removed by two events. The first one is that nodes rebroadcast the packet when waiting timeout expires. The other one is when nodes receive the redundant retransmission from their neighbors. The entry will be removed although the waiting timeout still does not expire. Pseudocode 1 describes the pseudocode of the protocol. The details of main modules are explained as shown in Pseudocode 1.

**4.2.1. Waiting Timeout Mechanism.** Waiting timeout is a solution to avoid broadcasting collision in distributed system. Nodes will randomly set their waiting timeout as backoff delay for rebroadcasting. There are two events that use waiting timeout. The first event is when nodes receive the broadcasting packet, but they are not members of CDS. They will add the packet to *Broadcast List* and set waiting timeout. These nodes have to listen to the rebroadcasting by their neighbors that are CDS members. If waiting timeout is expired and no CDS members rebroadcast the packet, a node with the shortest waiting timeout will rebroadcast the packet. The second event is when nodes detect the missing packet from their neighbors. They add the packet to *Broadcast List* and set waiting timeout the same as the first case. As a result,

```

Procedure cds-state(a);
cds(a) = true;
//density based condition
for each neighbor b of a do{
    if noNeighbor(b) > noNeighbor(a) then
        cds(a) = false;
}
//topology based condition
if cds(a) = false then{
    cds(a) = true;
    for each neighbor b of a do{
        for each neighbor c of a, b ≠ c do{
            if b and c are neighbor to each other then
                cover = true;
            for each neighbor d of a, d ≠ b, d ≠ c do{
                if d is not neighbor of b and c then
                    cover = false;
            }
        }
        if cover = true then cds(a) = false;
    }
}
}

```

PROCEDURE 1: DTA procedure.

```

Initialize (node a)
P: received packets buffer, N: neighbor list
B: broadcast list

Event receiving a broadcasting packet p
If {p} ∉ P then{
    add p to P;
    if cds(a) = true then
        rebroadcast p with randomly delay (<10 ms.);
    else
        add p and waiting timeout to B;
} else{
    remove p and cancel waiting timeout from B;
}

Event receiving a beacon from neighbor n
if {n} ∉ N then{
    add n and beacon expire time to N;
} else{
    update n and beacon expire time to N;
}
//update CDS state of node a
cds-state(a);
for each packet p in P
    if id(p) does not contain in list of pkt. of n then
        add p and waiting timeout to B;
missPacket = false;
for each packet identifier id(q) in list of pkt. of n
    if id(q) does not contain in P then
        missPacket = true;
if (missPacket) then
    if a never send beacon within this interval then
        send beacon(a);

```

PSEUDOCODE 1: NoG pseudocode.

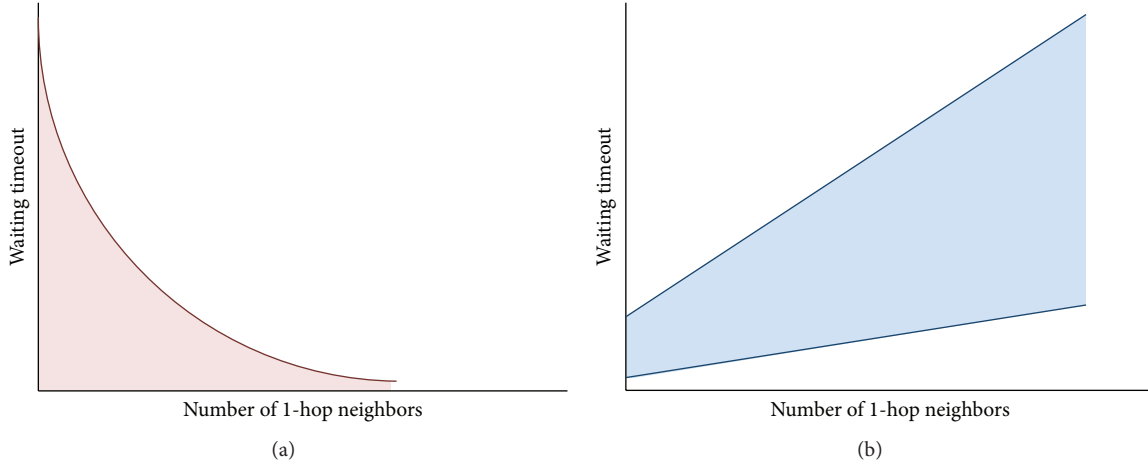


FIGURE 5: Waiting timeout function. (a) Reversed function. (b) Directed function.

a node with the shortest waiting timeout will rebroadcast the missing packet to its neighbors. These two events are explained in Pseudocode 1.

The disadvantage of waiting timeout is that it increases delay to overall system. Most of previous works calculate their waiting timeout as a reversed function to number of 1-hop neighbors. The purpose is to maximize number of received nodes in each retransmission by a node with the highest number of 1-hop neighbors, but this leads to a contention problem. It also increases extremely high redundant retransmissions in high density scenarios. The reason is that when nodes are in the dense areas, the reverse function calculates very short range of delay. For this reason, most of nodes in the same area will have the same waiting timeout. Then they simultaneously rebroadcast the packet causing collision. In order to prevent such situation, protocols should use the number of 1-hop neighbors to be directed variation of waiting timeout function. As reported in [21], the directed function can prevent collision in extremely high density scenarios. This new waiting timeout also increases the data dissemination speed in sparse areas. Since the directed function provides much shorter waiting timeout period than the inversed function in sparse area, the data dissemination speed can be increased.

The waiting timeout can be calculated by (1).  $\tau$  represents the network delay since a packet is sent by source until it is delivered to receivers.  $n$  is a number of 1-hop neighbors.  $\beta$  is a constant value used for expanding the range between minimum waiting timeout and maximum waiting timeout. The best  $\beta$  value can significantly reduce collision occurrences in dense areas while increasing only a little delay. The minimum term of waiting timeout represents the possibility delay from a beacon queuing in MAC layer. So the minimum term will be equal to total delay of all neighbors' beacon sending time. The maximum term of waiting timeout consists of two terms. The first term,  $2\tau$ , is equal to two times of network delay. This is because in the case that nodes have one neighbor, they have possibility to wait for one beacon from the neighbor and another network delay from rebroadcasting. The second

term,  $n\beta\tau$ , is the possibility delay from a beacon queuing in MAC layer, that is, multiplied by the expanding value ( $\beta$ ).  $\beta$  is used for expanding the range between minimum term and maximum term. The configuration of  $\beta$  is discussed in Section 5.2. The waiting timeout value can be illustrated in Figure 5

$$W(n) = \text{Random} [n\tau, (2 + n\beta)\tau]. \quad (1)$$

#### 4.2.2. Beacon Mechanism

(a) *Beacon Structure.* Nodes in NoG use beacon messages for discovering 1-hop neighbors and exchanging their local information. The beacon message header consists of a source identifier, a number of 1-hop neighbors, a list of 1-hop neighbor identifiers, and a list of received packets that still do not expire. The list of received packet contains an identifier of source and an identifier of the packet. This list is used for missing message detection. The beacon size will be at least 5 bytes in case there is no 1-hop neighbor and received packet. The beacon size will increase 4 bytes for each 1-hop neighbor and 5 bytes for each received packet. In order to reduce the number of beacons, nodes piggyback the beacon header with the broadcasting packet when they have a packet to rebroadcast, as shown in Figure 6. Then the next beacon will be postponed until the next beacon interval.

(b) *Beacon Interval Calculation.* The accuracy of 1-hop neighbors' position depends on the frequency rate of beacon. In fact, this can cause a broadcast storm problem in dense areas. In this paper, the nongeographical knowledge algorithms are focused. These algorithms do not require very high accurate data from beacon information. Moreover, density of vehicles has related to speed of vehicles [22], so the vehicles in dense area are moving slower than the vehicles in sparse area. Consequently, a short beacon interval is needed in sparse areas, but it is unnecessary in dense areas. NoG uses an adaptive beacon interval algorithm to appropriately calculate the beacon interval in each density environment.

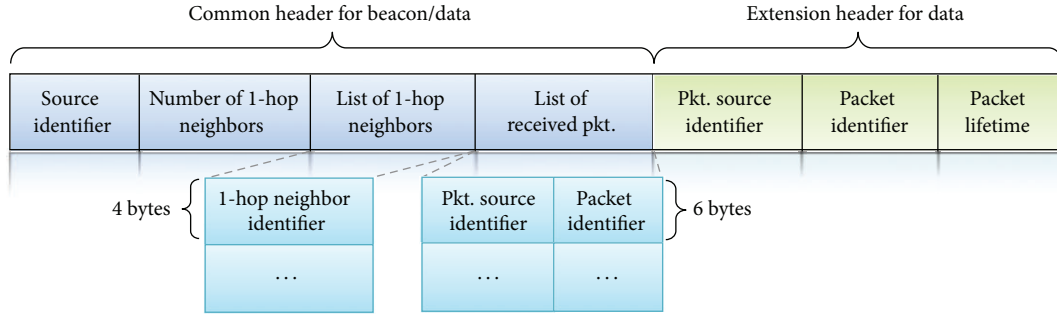


FIGURE 6: Beacon structure.

The algorithm linearly increases the beacon interval based on network density, called Linear Adaptive Interval or LIA [23]. The algorithm can reduce beacon overhead without decreasing the protocol performance.

As mentioned, the beacon interval is linearly increased depending on the network density. The network density ( $\text{netDensity}$ ) is calculated by a number of 1-hop neighbors ( $n$ ) and a number of broadcasting packets ( $p$ ) that do not expire. This can be represented by (2). The beacon interval ( $\text{beaconInv}$ ) calculation is represented by (3).  $\text{minInv}$  is a minimum beacon interval.  $c$  is a constant value.  $\text{maxInv}$  is the longest interval that does not affect the performance of protocol. The parameter setting of beacon interval calculation is explained in Section 5.2:

$$\text{netDensity} = n + p, \quad (2)$$

$$\text{beaconInv} = \min [\text{minInv} + (c \times \text{netDensity}), \text{maxInv}]. \quad (3)$$

(c) *Missing Packet Mechanism.* In VANET, an intermittent connection always occurs. In order to provide reliable broadcasting, protocols need an ability to detect missing packets. The packets can be lost due to channel error. The missing packet mechanism has two parts for its operation. The first part is that a node checks for neighbor's missing packets from incoming beacon. A node can check the missing packets by the list of received packets in the beacon. If a node detects that its neighbor has the missing packets, it will set waiting timeout and add the missing packets to *Broadcast List*. Another part is that a node checks whether there are any packets that it never receives from the incoming beacon. If it finds that it has the missing packets, it will immediately broadcast a beacon to let its neighbors know and detect its missing packets. However, this mechanism can flood many beacons to the networks, so the beaconing for missing packets will be restricted to broadcast only once within a beacon interval. This means that a node cannot rebroadcast its beacon until the next beacon schedule.

## 5. Performance Evaluation

### 5.1. Performance Evaluation of CDS Forming Algorithm

5.1.1. *Simulation Setup.* In order to evaluate performance of our algorithm, we implement a Java simulator. The simulator

TABLE 1: Parameters setting.

Simulation setup	
Message lifetime	10 seconds
Number of broadcast sources	5
Size of broadcast packet	512 bytes
Simulation time	200 seconds
Number of simulation runs	20
Max. vehicle speed (km/h)	50, 80
Vehicle density (veh/km)	2, 6, 10, 20, 30, 40, 60, 80
DECA	
Beacon interval	LIA 1.5–7 seconds $c = 0.2$
Beacon size	5 bytes + 5 bytes for each pkt.
APBSM	
Beacon interval	Constant 1 seconds
Beacon size	21 bytes + 5 bytes for each pkt.
NoG	
Beacon interval	LIA 1.5–7 seconds $c = 0.2$
Beacon size	5 bytes + 4 bytes for each 1-hop neighbor + 5 bytes for each pkt.

uses mobility traces from NS-2 [24]. The trace is generated via simulation of urban mobility (SUMO) [25]. The vehicle traces obtained from SUMO are in XML format. They are converted to NS-2 traces format by traffic simulation environment (TraNS) [26]. There are two traffic scenarios: (1) a highway scenario is a straight 4-kilometer road with two lanes per direction; (2) an urban scenario is  $2 \times 2$  kilometers Manhattan grid. Nodes are equipped with 250 meters transmission range wireless device.

The simulator samples groups of nodes every 10 seconds and then it analyses the CDS forming algorithm in terms of a coverage result and a ratio of CDS members to total nodes in groups. There are more than 2000 groups of nodes that are sampled. No real broadcasting is employed in this simulation. The real broadcasting performance evaluation is done



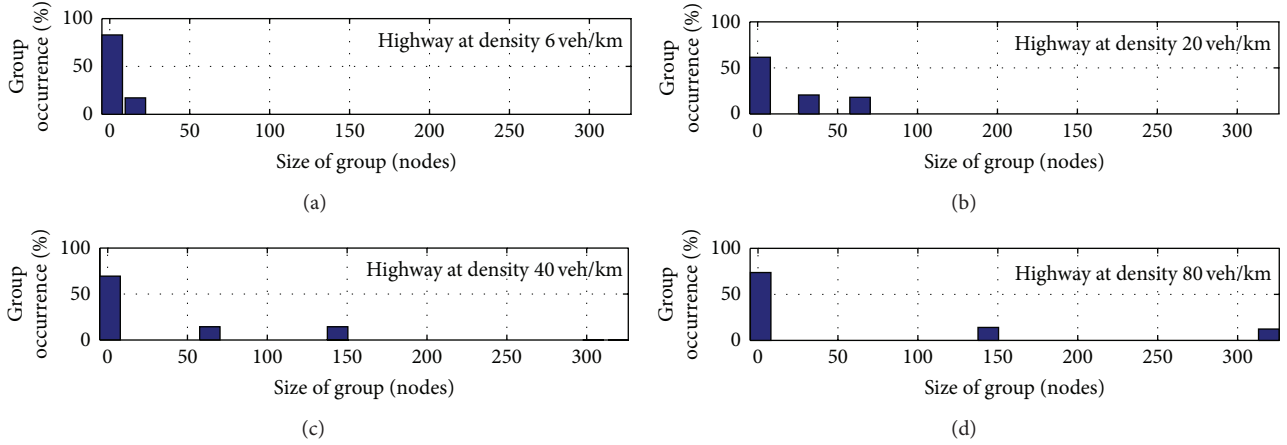


FIGURE 7: Occurrence of each size of group in highway scenarios.

on well-known network simulator 3 (NS-3) [27] in Section 5.2. The other parameters setting are shown in Table 2.

We have implemented all of following CDS forming algorithm in the simulator.

- (i) *Density based (DEN)*: only nodes with the highest number of 1-hop neighbors are members of CDS. This algorithm represents the density based algorithm.
- (ii) *Density and topology based with internode condition (DEN+IN)*: members of CDS consist of nodes with the highest number of 1-hop neighbors and nodes that can pass the intermediate condition of Wu and Li's algorithm.
- (iii) *Density and topology based with intergateway condition (DEN+IG)*: members of CDS consist of nodes with the highest number of 1-hop neighbors and nodes that can pass the intergateway condition of Wu and Li's algorithm.
- (iv) *DTA*: the algorithm that we proposed. DTA is a density and topology based with gateway condition (DEN+G). Members of CDS consist of nodes with the highest number of 1-hop neighbors and nodes that can pass the gateway condition of Wu and Li's algorithm.
- (v) *Wu and Li's algorithm (WLA)*: members of CDS are nodes that can complete all of three conditions of Wu and Li's Algorithm. This represents the most efficient topology based algorithm in our literature review.

**5.1.2. Metrics.** There two metrics considered. All simulation results are averaged from 100 of runs with 95% confidence interval. A group of nodes, that is, a complete graph connection, is not included in the results. The reason is that nodes can directly communicate to each other in this type of group. Note that an overhead result from exchanged beacon is not considered in this evaluation. However, the overhead results are discussed in Section 5.2.

- (i) *Coverage node* is measured as a percentage of the number of nodes that are covered by members of CDS to total nodes in the group.

- (ii) *Ratio of CDS members* is measured as a ratio of the number of nodes that are members of CDS to a number of total nodes in the group.

**5.1.3. Simulation Results.** Figures 7 and 8 show the occurrences of node groups in each size for highway scenarios and those for urban scenarios, respectively. For highway scenarios, vehicles are uniformly distributed although vehicles are randomly released and vehicles have the different maximum speed. This is because the highway scenario is a simple straight road with nonstructure the same as the realistic long distance highway road. On the other hand, vehicles are nonuniform distributed in urban scenarios. There are many several sizes of group in each scenario. Therefore, both scenarios and mobility traces can represent the realistic environment of vehicles in both highway areas and urban areas.

**Coverage Results.** All coverage results in both highway scenarios and urban scenarios are shown in Figure 9.

DEN that considers only the number of 1-hop neighbors provides well coverage results on low density scenarios. The coverage results decrease in high density scenarios because there are more nodes and more complex connections in dense scenarios than in sparse scenarios. The reason is that the number of members in CDS from DEN is not enough to cover all nodes in the groups.

On the other hand, WLA, that is, Wu and Li's algorithm that forms CDS by using topology information, does not operate well in sparse scenarios because the algorithm prunes too much nodes so it decreases a number of covered nodes in sparse scenarios. The advantage of Wu and Li's algorithm is it can construct the efficient members of CDS that can cover all nodes in groups in dense scenarios. WLA works well with complex connections in high density scenarios. These scenarios are similar to general mobile ad hoc scenarios

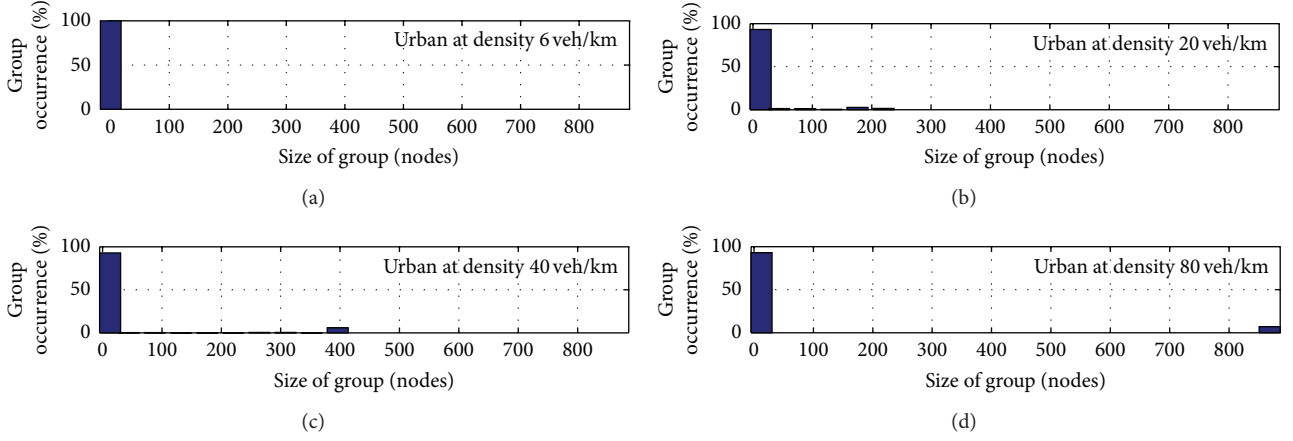


FIGURE 8: Occurrence of each size of group in urban scenarios.

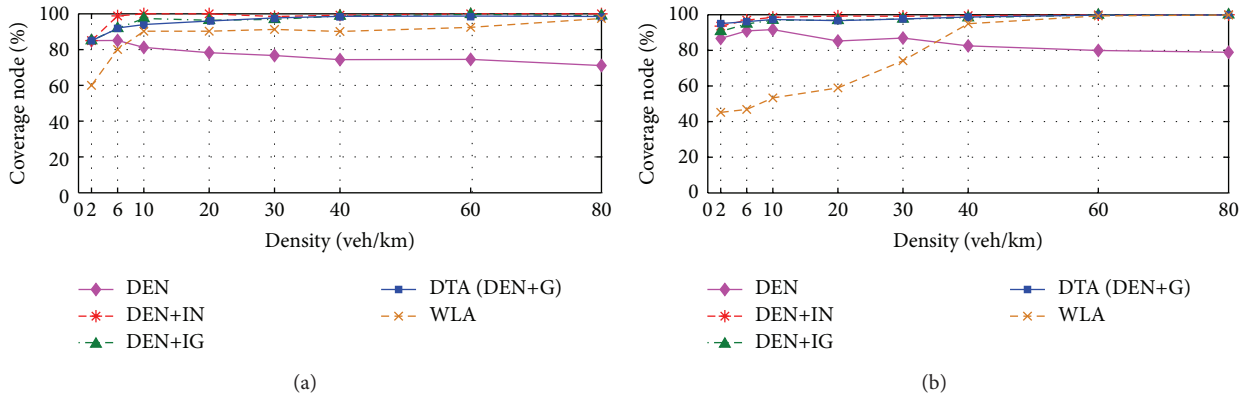


FIGURE 9: Coverage result. (a) Highway scenarios. (b) Urban scenarios.

that the algorithm is designed for. Therefore, we combine the advantages from both density based algorithm and topology based algorithm. We use the density based algorithm that can provide high coverage results in low density scenarios with a simple concept. Then we combine it with topology based algorithm that provides the efficient CDS members that can cover all nodes in groups in dense scenarios.

The combination algorithms are DEN + IN, DEN + IG, and DTA (DEN + G). These algorithms are a combination of density based algorithm and topology based algorithm. All of them provide the highest coverage results in the simulation. The algorithms can construct CDS members with almost 100% coverage results.

**Ratio of CDS Member.** The results are shown in Figure 10. The ratio results represent the efficiency of algorithm. A number of CDS members should be as low as possible, while the CDS members can cover all nodes in the group.

DEN has the least ratio results because it considers only nodes with the highest number of 1-hop neighbors. The number of CDS members converges to about 0.07 of total nodes.

WLA is the second least ratio results. It provides almost constant ratio results in every density scenario. The algorithm

is very efficient, but this leads to low coverage results in sparse areas. There are many small groups of vehicles in the sparse scenarios and the distance between nodes is longer than in dense scenarios, so the ratio of CDS members should be higher.

DEN+IN has extremely high ratio results. The results are almost 1. This means that the internode condition of Wu and Li's algorithm cannot efficiently prune nodes in vehicular environment. As a result, almost all nodes in the scenarios are CDS members.

DEN+IG also provides the efficient CDS members. It has very low ratio results, but the ratio results are higher than DTA. This is because the gateway condition can significantly prune more the unnecessary nodes than the intergateway condition as described in Section 3.1.

DTA is the most efficient algorithm because it can provide very low ratio of CDS members to total nodes. The ratio results converge to about 0.2 of total nodes. In low density scenarios, DTA has the high ratio results which are close to the results from DEN. DTA also has the ratio results that almost are the same as the results from WLA in high density. The reason is that DTA has the advantages from both density based algorithm and topology based algorithm so DTA will appropriately keep a number of CDS members depending

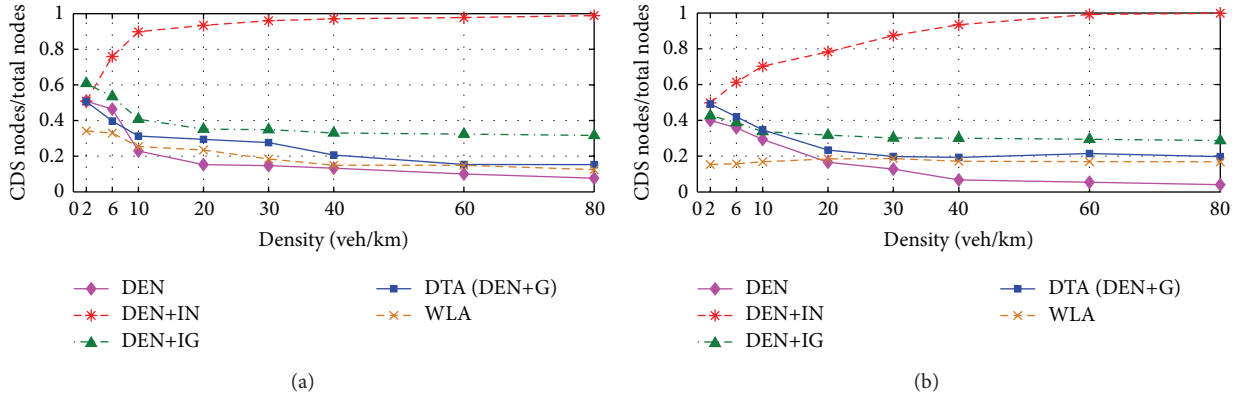


FIGURE 10: Ratio of CDS members. (a) Highway scenarios. (b) Urban scenarios.

TABLE 2: Simulation setting.

Simulation time	200 seconds
Number of simulation runs	100
Maximum vehicle speed (km/h)	50, 80
Vehicle density (veh/km)	2, 6, 10, 20, 30, 40, 60, 80

on scenarios. This can maximize the coverage results while minimizing a number of CDS members.

## 5.2. Performance Evaluation of Broadcasting Protocol

**5.2.1. Simulation Setup.** All broadcasting protocols evaluated their performance with the same road scenarios and vehicle mobility traces the same as in Section 5.1.1.

There are 5 source nodes in each simulation. After the simulation has run for 100 seconds, source nodes randomly start to broadcast their packet every 10 seconds until simulation ends at 200 seconds. The last packet will be expired at 200 seconds of simulation. All protocols use IEEE802.11b with contention for MAC. We cannot use IEEE802.11p [22] because it is under development phase in NS-3. All nodes are equipped with a wireless module with Rayleigh fading. The transmission success rate is 80% at distance 250 meters. Unless stated otherwise, parameters setting for simulations is configured as indicated in Table 1.

We have implemented all of the following protocols in the well-known network simulator NS-3.16 [27]. All of previous works are configured following their publications.

- (i) *DECA* [9]: DECA represents a protocol that uses only density information to select the next rebroadcast node. It provides very high data dissemination speed by avoiding waiting timeout.
- (ii) *APBSM* [6]: APBSM represents a protocol that uses both density and geographical knowledge to construct members of CDS by extending Wu and Li's algorithm.

(iii) *NoG+DEN*: our proposed protocol with the simplest algorithm: This algorithm uses only density information to construct CDS members. It represents a density based protocol.

(iv) *NoG+WLA*: our proposed protocol with the original Wu and Li's CDS forming algorithm: it represents a topology based protocol.

(v) *NoG+DTA*: our proposed protocol with our proposed algorithm: DTA is a combination of density and topology based algorithm for constructing CDS members.

**5.2.2. Parameter Setting in NoG.** The parameter setting for waiting timeout as mentioned in Section 4.2.1 and the parameter setting for beacon interval as mentioned in Section 4.2.2 are discussed. These parameter settings are used in our simulation.

**Waiting Timeout.** As mentioned in Section 4.2, the efficient  $\beta$  value can significantly reduce collision occurrences in dense areas while increasing only a little delay. In order to select the  $\beta$ , we performed a simulation. The simulation setup is the same setup as in Section 5.2. (Table 1). The highway scenario is used in this simulation. We evaluated the performance of NoG using the directed function with varied  $\beta$  (1–5). From the results, 3 is the best value that provides low overhead and it introduces the lowest additional delay. According to (1) in Section 4.2.1, the maximum waiting timeout depends on  $\beta$  value.

**Beacon Interval.** The efficient beacon interval should help the protocol to provide the fastest data dissemination speed, while it increases the least additional overhead to each network density. In order to select the efficient beacon interval, we performed a simulation. We used the highway scenario in this simulation. The beacon interval is varied from 0.1 to 9 seconds in different density scenarios (2–80 veh/km). The other parameters such as communication setup and packet setup are set the same as those in Section 5.2 (Table 1). From simulation results, we observed that 1.5 seconds are the beacon interval that provides the fastest data dissemination

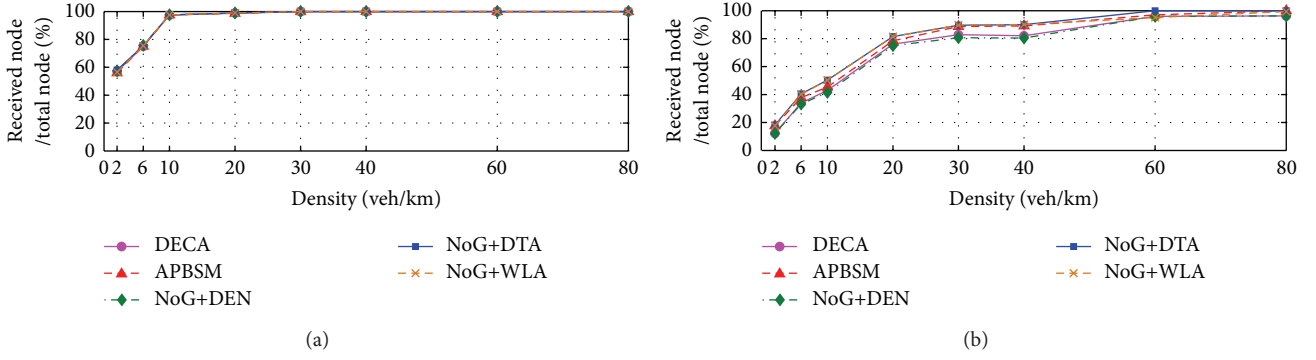


FIGURE 11: Reliability results. (a) Highway scenarios. (b) Urban scenarios.

speed with the lowest overhead in low density scenarios and 7 seconds are the longest beacon interval that provides the fastest data dissemination speed with the lowest overhead in dense scenarios. Therefore, the suitable beacon interval for NoG is between 1.5 seconds and 7 seconds. According to (3) in Section 4.2.2, (c) is equal to 0.2, minInv is 1.5, and maxInv is 7.

**5.2.3. Metrics.** Five metrics are considered. All simulation results are averaged from 20 of runs with 95% confidence interval.

- (i) *Reliability* is measured as a percentage of nodes that received the packets at the end of simulation.
- (ii) *Retransmission overhead* is measured from bandwidth consumption, that is, from packet retransmission.
- (iii) *Beacon overhead* is measured from bandwidth consumption that is from beacon transmission.
- (iv) *Source of retransmission* is measured as percentages of three sources of packet retransmission that consist of retransmission by CDS members, retransmission by waiting timeout mechanism, and retransmission by neighbor's missing packet mechanism.
- (v) *Speed of data dissemination* is measured as (4), where  $r_i$  represent number of nodes that received the packet for the first time at the time  $i$  and  $n$  is total number of vehicles in the scenario:

$$y(t) = \frac{\sum_{i=0}^t r_i}{n} \times 100. \quad (4)$$

#### 5.2.4. Simulation Results

**Reliability.** The reliability results in highway scenarios are shown in Figure 11(a). All protocols provide the same reliability in every scenario because these protocols are well designed to operate in vehicular environment. All of them employ store and forward technique that can handle the intermittent connectivity. The difference of CDS forming algorithm does not affect the reliability due to simple scenarios.

On the other hand, the difference of algorithms affects reliability results in urban scenarios as shown in

Figure 11(b). NoG+DTA provides the highest reliability results in every scenario because the rebroadcast nodes are efficiently selected to cover all of nodes in the scenarios. NoG+WLA that operates well in urban scenarios provides reliability slightly less than NoG+DTA. This is because the coverage ability of WLA is less than DTA as mentioned in Section 5.1. APBSM that uses the extended version of WLA provides reliability less than NoG with the original WLA about 1–5%. The reason is from its broadcasting mechanism and its waiting timeout mechanism. A node in APBSM has to wait for waiting timeout expiration before each rebroadcasting. Moreover, when a node detects the missing message from its neighbors, it has to wait for more than one beacon interval before each retransmission. This reduces the opportunity to increase the reliability. DECA and NoG + DEN have the lowest reliability result due to its only density based algorithm that does not perform well in high density and complex scenarios.

**Retransmission Overhead.** The retransmission overhead results are illustrated in Figure 12.

For highway scenarios, all of protocols have the same retransmission overhead except APBSM. This is because APBSM uses the inversed function to calculate their waiting timeout, so the redundant retransmissions increase in dense area. Although DECA also uses the inversed function, it avoids using waiting timeout by selecting the next rebroadcast node from source. All of algorithms on NoG protocol can efficiently operate in every highway scenario.

For urban scenarios, APBSM still has the highest retransmission overhead due to its waiting timeout calculation. Although its CDS algorithm is extended from Wu and Li's algorithm, but Wu and Li's can work better on NoG (NoG+WLA). NoG+WLA can decrease up to 35% of redundant retransmission from APBSM. For density based algorithm, DECA and NoG+DEN have the same retransmission overhead, but the results are worse than NoG+WLA and NoG+DTA by about 23%. The reason is that density based algorithm cannot work well on complex scenarios. NoG+DTA has the most efficient operation. NoG+DTA can provide the lowest overhead in every urban scenario. DTA has the advantage from density based algorithm and topology based algorithm so it is only a protocol that has the least



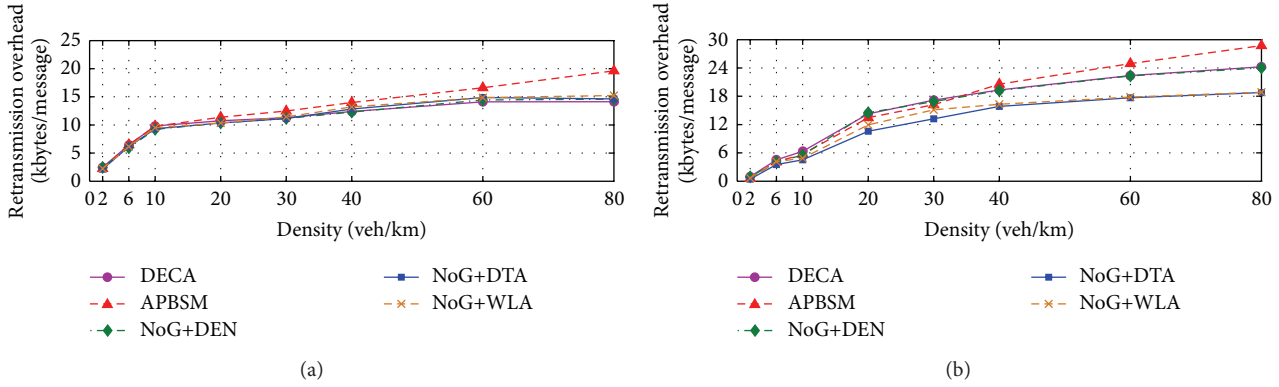


FIGURE 12: Retransmission overhead results. (a) Highway scenarios. (b) Urban scenarios.

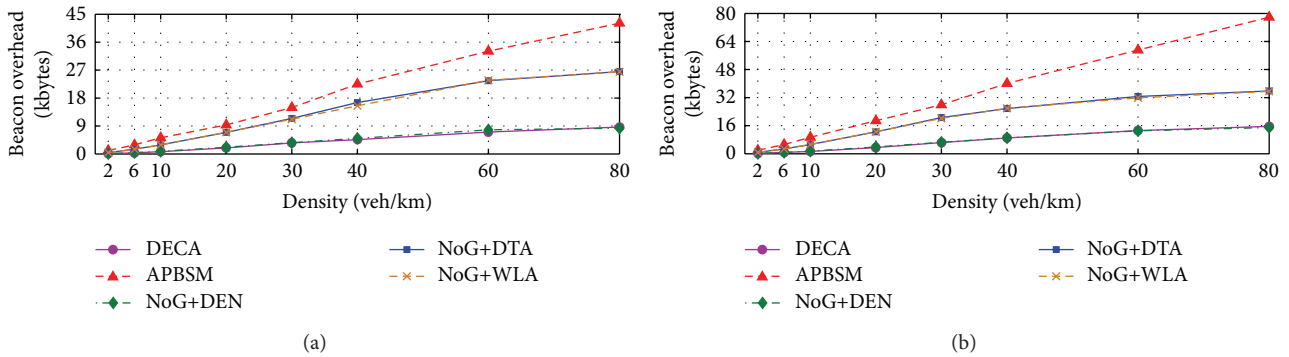


FIGURE 13: Beacon overhead results. (a) Highway scenarios. (b) Urban scenarios.

number of retransmissions in normal density scenarios that consist of many sizes of groups of vehicles.

**Beacon Overhead.** The beacon overhead results are illustrated in Figure 13.

For highway scenarios, DECA and NoG+DEN have the lowest beacon overhead results in the simulation. The overhead of DECA and NoG+DEN is very low because DECA and NoG+DEN use only density information so they require only a number of 1-hop neighbors. For NoG+DTA and NoG+WLA, their beacon messages need to contain 1-hop neighbor list. For APBSM, its beacon needs to contain position knowledge of neighbors and it has to use the constant beacon interval for accurate neighbors' position. So APBSM has the highest overhead results.

For urban scenarios, all of results are in the same trend with highway scenarios. APBSM has the highest overhead due to its constant beacon interval. DECA and NoG+DEN have the lowest beacon overhead. NoG+WLA and NoG+DTA have 55% more beacon overhead than density based algorithm. However, the difference of overhead results between density based algorithm and topology based algorithm in urban scenarios is less than the difference of results in highway scenarios. This is because the average beacon sizes in urban scenarios are larger than highway scenarios. Note that the protocol has to maintain 2-hop neighbor list for topology based algorithm. The size of beacon depends on

the size of scenario. The adaptive beacon interval significantly reduces overhead in the following case. When a node is in the dense area, the size of beacon is larger, while the beacon interval is also longer, so the large beacon will be reduced.

**Source of Retransmission.** The source of retransmission represents the efficiency of protocols and algorithms. The protocols and algorithms that have the higher retransmissions from their preferred nodes are better because these nodes are working as designed. This affects the performance in terms of data dissemination speed. The reason is that the preferred nodes can immediately rebroadcast or have the shorter waiting timeout than other nodes. The preferred node of DECA is selected by source node and the preferred node of APBSM, NoG+DEN, NoG+DTA, and NoG+WLA is a CDS member.

The results are shown in Figure 14. For density based algorithms, DECA and NoG+DEN have the best results in highway scenarios because the algorithms can operate well in simple scenarios. Both of the protocols have very close percentages of preferred node retransmissions, but DECA has better results than NoG+DEN in highway scenarios. The reason is that DECA selects the next rebroadcast node from source's perspective. The selected node is a node with the highest density of source's neighbors, so a number of selected nodes are higher than a number of CDS members from NoG+DEN. However, in urban scenarios, a number of

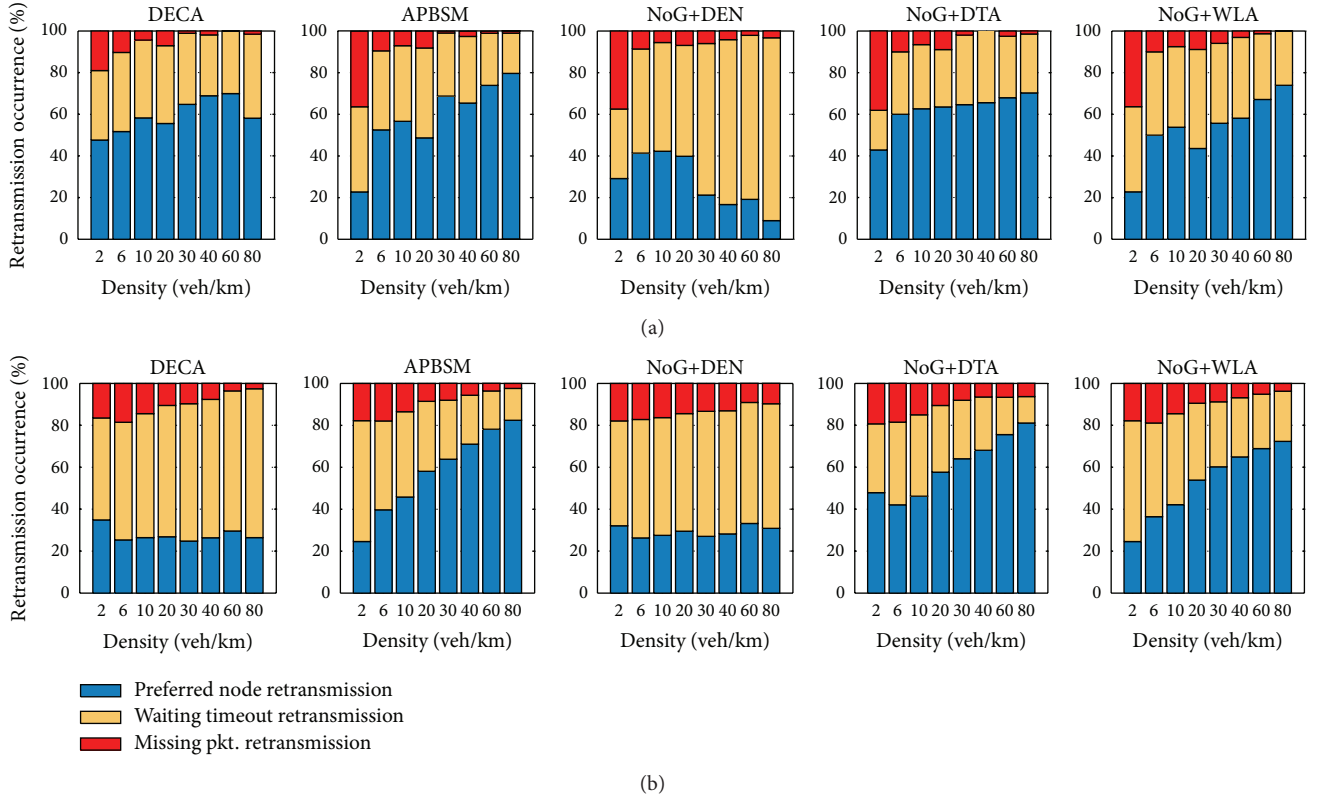


FIGURE 14: Source of retransmission. (a) Highway scenarios. (b) Urban scenarios.

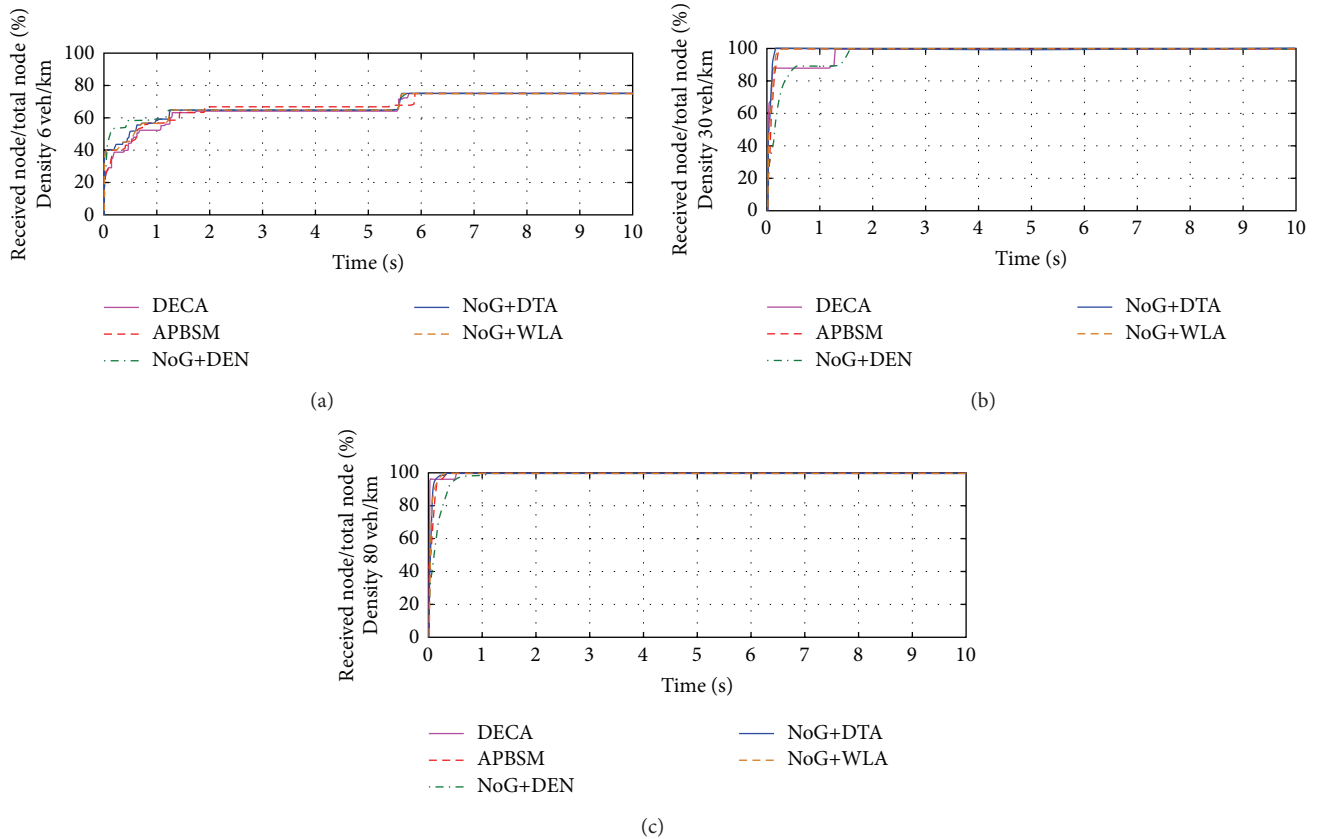


FIGURE 15: Speed of data dissemination in highway scenarios at 6 veh/km (a), 30 veh/km (b), and 80 veh/km (c).

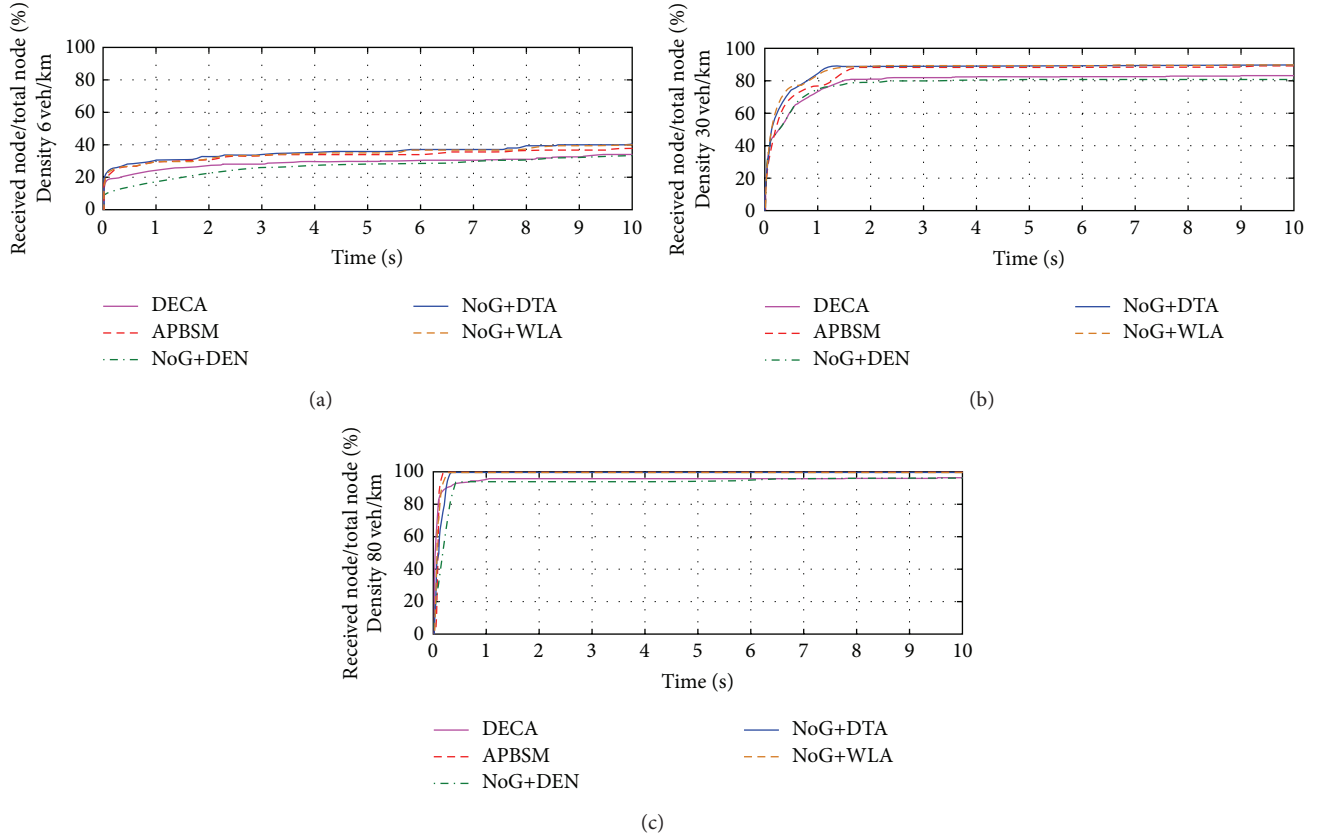


FIGURE 16: Speed of data dissemination in urban scenarios at 6 veh/km (a), 30 veh/km (b), and 80 veh/km (c).

rebroadcast nodes from both algorithms are not enough to cover all nodes in scenarios.

For topology based algorithms, the results of APBSM and NoG+WLA are the same trend with coverage results in Section 5.1. The topology based algorithm is appropriate to complex scenarios, so in higher density these algorithms have higher percentages of preferred node rebroadcasting.

NoG+DTA has the highest percentage of preferred nodes retransmission in every scenario because NOG+DTA is the combination of density-based algorithm that works well in simple scenarios and topology-based algorithm that works well in complex scenarios.

**Speed of Data Dissemination.** The speed of data dissemination results in the highway scenarios and the urban scenarios are, respectively, shown in Figures 15 and 16. The results at density 6 veh/km represent sparse scenarios (2–10 veh/km). The results at density 30 veh/km represent normal density scenarios (20–40 veh/km) and the results at density 80 veh/km represent high density scenarios.

For highway scenarios, NoG+DEN is the fastest protocol in low density scenarios, but it is the slowest one in high density scenarios due to its density based algorithm. Nodes in DECA have the results the same as NoG+DEN. NoG+DTA is the fastest protocol from simulation results. APBSM and NoG+WLA are slightly slower than NoG+DTA for all scenarios, but the difference is less than 0.1 milliseconds.

For urban scenarios, DECA and NoG+DEN are slower than topology based algorithm due to complexity of connection. APBSM is a bit slower than NoG+DTA and NoG+WLA in sparse areas and medium density areas. The reason is that rebroadcast nodes in APBSM have to wait for waiting timeout before each rebroadcasting. On the other hand, APBSM provides the fastest data dissemination in high density scenarios due to a lot of redundant retransmissions discussed in retransmission overhead results. NoG+DTA and NoG+WLA provide almost the same speed of data dissemination.

## 6. Conclusion

In this paper, we propose an approximation algorithm for constructing CDS members. It is a density and topology based algorithm, called DTA. DTA combines the advantages from density based algorithm and topology based algorithm. The density based algorithm can construct the efficient CDS members in simple connections or low density scenario. On the other hand, the topology-based algorithm can construct the efficient CDS members in complex connections or high density scenario. The simulation results show that DTA outperforms other algorithms in terms of coverage results and ratio of CDS members to total nodes. DTA has the coverage results better than other previous algorithms' results up to 50%. We also proposed a nongeographical

knowledge broadcasting protocol, called NoG. The protocol consists of a broadcast mechanism, a waiting timeout mechanism, and a beacon mechanism. It is designed to operate with high data dissemination speed and consume the least network resource as possible. The simulation results show that NoG provides the fastest data dissemination speed and the highest reliability. Currently, the beacon size of NoG depends on the size of 2-hop neighbor list which can be significantly increased in dense area. Most of broadcasting protocols in VANET uses beacon with variable size. Therefore, our future work is to reduce the beacon overhead by using fixed size beacon. The solution may be applied to other broadcasting protocols in VANET.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This research was supported in part by the TRF (Thailand Research Fund) (MRG5380164), the Ratchadaphiseksomphot Endowment Fund, and H.M. the King's 72nd Birthday Scholarship, Chulalongkorn University, Bangkok, Thailand.

## References

- [1] M. L. Sichitiu and M. Kihl, "Inter-vehicle communication systems: a survey," *IEEE Communication Survey and Tutorial*, vol. 10, no. 2, pp. 88–105, 2008.
- [2] V. Naumov, R. Baumann, and T. Gross, "An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces," in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '06)*, Florence, Italy, May 2006.
- [3] N. Wisitpongphan, F. Bai, P. Mudalige, and O. K. Tonguz, "On the routing problem in disconnected vehicular ad hoc networks," in *Proceedings of the 26th Annual IEEE Conference on Computer Communications (INFOCOMM '07)*, Anchorage, Alaska, USA, May 2007.
- [4] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Tech. Rep. CS-200006, Duke University, 2000.
- [5] M. Nekovee and B. B. Bogason, "Reliable and efficient information dissemination in intermittent connected vehicular ad hoc networks," in *Proceedings of the 65th IEEE Vehicular Technology Conference (VTS '07)*, pp. 2486–2490, Dublin, Ireland, April 2007.
- [6] F. J. Ros, P. M. Ruiz, and I. Stojmenovic, "Acknowledgment-based broadcast protocol for reliable and efficient data dissemination in vehicular ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 1, pp. 33–46, 2012.
- [7] K. N. Nakorn and K. Rojviboonchai, "POCA : position-aware reliable broadcasting in VANET," in *Proceedings of the 2nd Asia-Pacific Conference of Information Processing (APCIP '10)*, Nanchang, China, September 2010.
- [8] O. K. Tonguz, N. Wisitpongphan, and F. Bai, "DV-CAST: a distributed vehicular broadcast protocol for vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 17, no. 2, pp. 47–57, 2010.
- [9] N. N. Nakorn and K. Rojviboonchai, "DECA: density-aware reliable broadcasting protocol on vehicular ad-hoc networks," in *Proceedings of the 7th IEEE International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON '10)*, Chaing Mai, Thailand, May 2010.
- [10] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying: an efficient technique for flooding in mobile wireless networks," Tech. Rep. RR-3898, INRIA, 2000.
- [11] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks," in *Proceedings of the 3rd International Workshop Discrete Algorithms and Methods for Mobile Computing and Communication (DIALM '09)*, pp. 7–14, August 1999.
- [12] J. Sucec and I. Marsic, "An efficient distributed network-wide broadcast algorithm for mobile ad hoc networks," Tech. Rep. 248, CAIP, Rutgers University, 2000.
- [13] W. Peng and X. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '00)*, pp. 129–130, 2000.
- [14] I. Stojmenovic, M. Seddigh, and J. Zunic, "Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 1, pp. 14–25, 2002.
- [15] F. Dötzer, "Privacy issues in vehicular ad hoc networks," in *Proceedings of the Workshop on Privacy Enhancing Technologies (PET '05)*, 2005.
- [16] M. Raya and J. P. Hubaux, "Securing vehicular ad hoc networks," *ACM Journal of Computer Security*, vol. 15, no. 1, 2007.
- [17] B. Parno and A. Perrig, "Challenges in securing vehicular networks," in *Proceedings of the Workshop on Hot Topics in Networks (HOTNETS '05)*, 2005.
- [18] "The 13 Controlling Criteria," in *Mitigation Strategies for Design Exceptions*, chapter 3, US Department of Transportation Federal Highway Administration, Washington, DC, USA, 2007.
- [19] V. Naumov and T. Gross, "Connectivity-aware routing (CAR) in vehicular ad-hoc networks," in *Proceedings of the 26th Annual IEEE Conference on Computer Communications (INFOCOMM '07)*, Anchorage, Alaska, USA, May 2007.
- [20] IEEE, "Part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 6: wireless access in vehicular environments," in *Proceedings of the IEEE Standard for Information Technology Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks Specific Requirements*, New York, NY, USA, July 2010.
- [21] K. Na Nakorn and K. Rojviboonchai, "DECA-bewa: density-aware reliable broadcasting protocol in VANETs," in *Proceedings of the The Institute of Electronics, Information and Communication Engineers Transactions on Communications (IEICE '13)*, vol. 96, May 2013.
- [22] M. Artim, "Local density estimation and dynamic transmission-range assignment in vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 3, pp. 400–412, 2007.
- [23] N. N. Nakorn and K. Rojviboonchai, "Efficient beacon solution for wireless ad-hoc network," in *Proceedings of the 7th International Joint Conference on Computer Science and Software Engineering (JCSSE '10)*, Bangkok, Thailand, May 2010.



- [24] “The Network Simulator (NS-2),” <http://www.isi.edu/nsnam/ns/>.
- [25] Simulation of Urban Mobility (SUMO), <http://sumo.sourceforge.net/>.
- [26] “Traffic and Network Simulation Environment (TraNS),” <http://trans.epfl.ch/>.
- [27] “The Network Simulator (NS-3),” <http://www.nsnam.com/>.

