การแพร่ข้อมูลโดยปราศจากความรู้ด้านภูมิศาสตร์สำหรับเครือข่ายไร้สายแบบแอดฮอกบนยานพาหนะ

นายกุลิสร์ ณ นคร

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2556

NON-GEOGRAPHICAL KNOWLEDGE DATA DISSEMINATION ON VEHICULAR AD HOC NETWORKS

Mr. Kulit Na Nakorn

A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy Program in Computer Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2013

Thesis Title	NON-GEOGRAPHICAL KNOWLEDGE DATA DISSEMINATION ON VEHICULAR AD HOC NETWORKS
By	Mr. Kulit Na Nakorn
Field of Study	Computer Engineering
Thesis Advisor	Assistant Professor Kulitda Rojviboonchai, Ph.D.

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirements for the Doctoral Degree

....Dean of the Faculty of Engineering

(Professor Bundhit Eua-arporn, Ph.D.)

THESIS COMMITTEE

....Chairman

(Professor Prabhas Chongstitvattana, Ph.D.)

....Thesis Advisor

(Assistant Professor Kulitda Rojviboonchai, Ph.D.)

....Examiner

(Assistant Professor Natawut Nupairoj, Ph.D.)

....Examiner

(Assistant Professor Krerk Piromsopa, Ph.D.)

....External Examiner

(Associate Professor Anan Phonphoem, Ph.D.)

กุลิสร์ ณ นคร : การแพร่ข้อมูลโดยปราศจากความรู้ด้านภูมิศาสตร์สำหรับเครือข่ายไร้สายแบบแอดฮอกบนยานพาหนะ. (NON-GEOGRAPHICAL KNOWLEDGE DATA DISSEMINATION ON VEHICULAR AD HOC NETWORKS) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร. กุลธิดา โรจน์วิบูลย์ชัย, 109 หน้า.

การแพร่ที่มีความเชื่อได้บนเครือข่ายไร้สายแบบแอดฮอกสำหรับยานพาหนะเป็นเรื่องที่มีความท้าทายจากลักษณะเฉพาะของการเชื่อมต่อที่เป็นช่วงๆและสภาพแวดล้อมของการจราจรที่มีการเปลี่ยนแปลงอย่างรวดเร็ว ดังนั้นแอพพลิเคชั่นและบริการบนระบบจราจรอัจฉริยะจึงต้องการโพรโทคอลสำหรับการแพร่ที่มีประสิทธิภาพ มีความเร็ว และมีความเชื่อถือได้ ในงานวิจัยนี้จึงนำเสนอโพรโทคอลสำหรับการแพร่ที่มีความเชื่อถือได้โดยปราศจากความรู้ด้านภูมิศาสตร์ การทำงานของโพรโทคอลประกอบด้วยส่วนสำคัญสองส่วน คือ ขั้นตอนการเลือกโหนดส่งต่อข้อมูลที่มีประสิทธิภาพและวิธีการแพร่ข้อมูลที่มีประสิทธิภาพ ขั้นตอนการเลือกโหนดส่งต่อจะใช้ข้อมูลทางด้านความหนาแน่น และข้อมูลลักษณะการเชื่อมต่อของโหนด การตัดสินใจของแต่ละโหนดจะกระทำด้วยข้อมูลภายในโหนดนั้นเองไม่ใช่การตัดสินใจจากภายนอก การใช้ข้อมูลดังที่กล่าวนั้นทำให้ขั้นตอนการเลือกโหนดมีความทนทานต่อการเพี้ยนของข้อมูลมากกว่าการใช้ข้อมูลทางด้านภูมิศาสตร์ และไม่เป็นการละเมิดสิทธิส่วนบุคคลจากการใช้ข้อมูลตำแหน่งของผู้ใช้ นอกจากนี้งานวิจัยยังมีการนำเสนอวิธีการแพร่ข้อมูลที่มีความเชื่อได้ ซึ่งประกอบด้วยส่วนสำคัญสามส่วน คือ ส่วนของการแพร่ข้อมูลเพื่อให้มีค่าใช้จ่ายในการแพร่น้อยที่สุด ส่วนของการคำนวณเวลาคอยเพื่อหลีกเลี่ยงการชนกันของข้อมูล และส่วนของการแลกเปลี่ยนบีคอนเพื่อใช้ในการแลกเปลี่ยนข้อมูลระหว่างโหนด ขั้นตอนการเลือกโหนดส่งต่อและวิธีการแพร่ข้อมูลที่นำเสนอนั้นถูกทดสอบสมรรถภาพบนโปรแกรมจำลอง NS-3 จากการทดสอบพบว่าขั้นตอนการเลือกโหนดส่งต่อและวิธีการแพร่ข้อมูลที่นำเสนอสามารถทำงานได้เร็วโดยมีความเชื่อถือได้สูงกว่าและมีค่าใช้จ่ายในการทำงานที่ต่ำกว่าเมื่อเปรียบเทียบกับงานอื่นๆที่มีการนำเสนอไปแล้วก่อนหน้า

ภาควิชา	วิศวกรรมคอมพิวเตอร์	ลายมือชื่อนิสิต	....
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก	....
ปีการศึกษา	2556		

5471402021 : MAJOR COMPUTER ENGINEERING
KEYWORDS: VEHICULAR NETWORKS / AD HOC / NON-GEOGRAPHIC / STORE-AND-FORWARD / RELIABLE BROADCASTING

KULIT NA NAKORN: NON-GEOGRAPHICAL KNOWLEDGE DATA DISSEMINATION ON VEHICULAR AD HOC NETWORKS. ADVISOR: ASST. PROF. KULITDA ROJVIBOONCHAI, Ph.D., 109 pp.

Reliable broadcasting in vehicular ad hoc networks is challenging due to its unique characteristics including intermittent connectivity and various vehicular scenarios. Applications and services in intelligent transportation systems need an efficient, fast and reliable broadcasting protocol. We propose a non-geographical knowledge reliable broadcasting protocol that has an efficient data forwarder selection and an efficient broadcasting mechanism. The forwarder node selection algorithm is a self-decision algorithm that lets 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. 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, overhead, and data dissemination speed.

Department:	Computer Engineering	Student's Signature	....
Field of Study:	Computer Engineering	Advisor's Signature	....
Academic Year:	2013		

ACKNOWLEDGEMENTS

First, I would like to express my deep gratitude to Asst. Prof. Kultida Rojviboonchai, my advisor, for her patient guidance, enthusiastic encouragement and useful critiques of this Ph.D. dissertation. She has advised me the way to study and also the way to live. I am so proud that I have been under her supervision for five years since my master degree.

I also would like to express my very great appreciation to Prof. Dr. Prabhas Chongstitvatana, Assoc. Prof. Dr. Anan Phonphoem, Asst. Prof. Dr. Natawut Nupairoj and Asst. Prof. Dr. Krerk Piromsopa for being my dissertation committee and giving many useful comments and suggestions to improve this dissertation.

For supporting all tuition fee and monthly expense during my Ph.D. program, the scholarship from the Graduate School, Chulalongkorn University to commemorate the 72nd anniversary of his Majesty King Bhumibala Aduladeja is greatfully acknowledged.

I thank all members of UbiNet research group and all members of Mass researches group for any comments and discussion on my dissertation and for the wonderful time that we spent together.

I would also to thank to people in Department of Computer Engineering that raised me up for nine years. I got a wonderful and invaluable experience while I was here.

Finally, I wish to thank my family for their support and encouragement throughout my study. They always understand me, stand beside me, help me solve any problems and walk through all the things together.

CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

Page

Page

Page

Page

Page

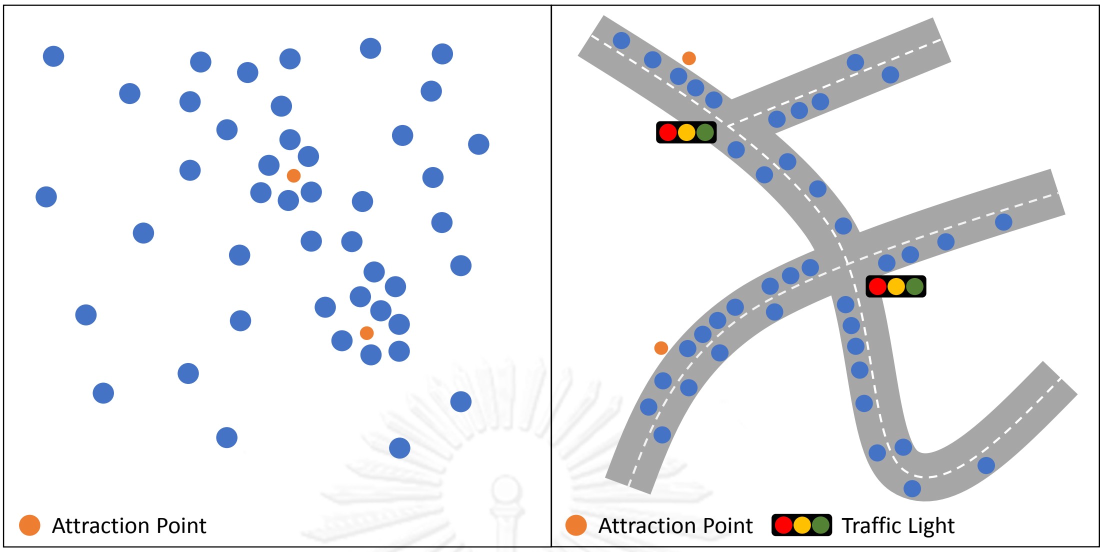
CHAPTER 1

Introduction

Vehicular ad hoc network (VANET) is one of mobile ad hoc network (MANET). Vehicles in VANET 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. VANET can support some of 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]. Moreover, VANET can support delay sensitive applications that do not require a critical real-time data delivery such as streaming or entertainment applications so the efficient data dissemination is one of the key successes for such applications.

Challenges in designing data dissemination or reliable broadcasting protocols for vehicular environment arise from the unique characteristics of ITS applications and vehicular movement. These applications need quick response and fast data dissemination because vehicles can change direction and connect frequently to their neighbors intermittently. Moreover, vehicles may be very densely packed at traffic light areas and very sparse on the highway or in rural areas. The speed of vehicles also affects to wireless signal that leads to channel occurrences between vehicles.

Figure 1.1 shows the differences between general mobility models for MANET and vehicular mobility models for VANET. In order to evaluate performance, most of researchers use a random waypoint model with or without attraction points for MANET but we have to concern the characteristics of vehicular environment for VANET because a traffic light, a size of road and a road structure can effect to mobility of nodes. This also causes the different results from performance evaluation as reported in [2]. Therefore, VANET need a specific protocol that has been designed to support its application and evaluated in accurate topology to assure its performance in real world.



(a) *(b)*

Figure 1.1 Mobility model (a) Random waypoint mobility model and (b) Vehicular mobility model.

A traditional approach for data dissemination for wireless ad-hoc networks is simple flooding. Simple flooding does not require any information from environment or nodes. Every received node rebroadcasts a packet once. 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 protocols in this group rely on geographical knowledge (GPS). 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 APBSM [6] and 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 may or may not require any geographical information to make decision.

However, every algorithm in every protocol has the same goal. The goal is to minimize number of rebroadcast nodes that can cover to 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). The algorithm constructs graph and selects the minimum number of nodes to cover 100% of their neighbor nodes in each group as shown in Fig. 1.2 but the algorithm requires global knowledge and the CDS computation is a NP-Complete problem [10]. Therefore, a heuristic 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.

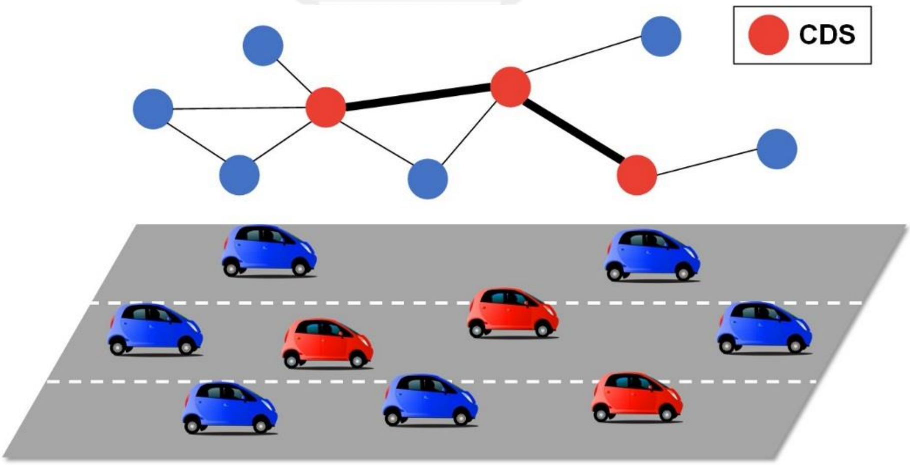


Figure 1.2 Connected dominating set (CDS).

This dissertation focuses on a non-geographical knowledge based CDS forming algorithms. These methods can avoid privacy issue that most of users concern. Moreover, the non-geographical knowledge based algorithms can resist inaccurate data than position base algorithms that need high frequent beaconing for accurate position data. 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. The topology based algorithm uses a k-common neighbor property (k-cn) that relates to a member of CDS. As a result, DTA can provide higher coverage results than existing algorithms. It is an appropriate algorithm for vehicular environment that has such a dynamic topology.

A broadcasting protocol is also a key for data dissemination performance. A protocol should support the CDS forming algorithm with efficient mechanism. In this dissertation, a proposed broadcasting protocol is an improvement version of DECA [9]. The improvement version is called DECA-bewa [15] that has the same selection algorithm as DECA. Then the protocol is redesigned to support non-geographical knowledge based CDS forming algorithms. This protocol is called "Non-Geographical broadcasting protocol" or "NoG". Nodes in NoG have a self-decision not like DECA that source/precursor will select the next rebroadcast node. NoG has three main modules, which are a forwarder selection algorithm, a waiting timeout calculation module and a beacon module. The forwarder selection algorithm can be implemented by any non-geographical knowledge node selection algorithms. We use DTA for this dissertation. The waiting timeout calculation is used for collision avoidance. The calculation function is a directed function between the number of 1-hop neighbor nodes and time. The directed function can handle more nodes in environment than the reversed function that is applied on the most of previous works. The beacon module is used for exchanging local information between neighbor nodes. We apply a Bloom Filter [16] technique to reduce the size of beacon. This can significantly reduce the overhead from beacon that other previous works [16-20] while it does not affect the performance of protocol. Therefore, NoG improves reliability, overhead and data dissemination speed from the previous protocols. Therefore, this work will support applications and services that do not sensitive to delay such as navigator, advertisement and entertainment. We believe that this work will be a part of success services on the intelligent transportation systems.

1.1 Design Goals

The goals are set as guidelines for our work design and development. These desirable properties are flexible, reliable, minimal and practical. The details of properties are mentioned as following explanations.

- *Flexible*: Our proposed work does not require any knowledge from geographical device (GPS). The GPS device can be interfered with obstacles, such as high buildings and bridges and it cannot operate in tunnels or closed areas so our work resists to these mentioned environment.
- *Reliable*: Many lost data can be found in a vehicle environment due to intermittent connectivity. Our proposed work should handle the lost data and it can recovery these data to the requested node.
- *Minimal*: Our proposed work has to reduce any overheads as much as possible. These overheads should not effect to data from applications. Our proposed work also should minimize the number of data retransmission that is a duplication retransmission.
- *Practical*: Our proposed work should operate well in a realistic vehicle environment that there are an intermittent connectivity issue and a scalable issue.

1.2 Scope and Assumption

The scope of this dissertation is limited to the following:

- This dissertation considers the vehicular ad hoc networks that 1) there is no infrastructure 2) there is no centralize server.
- The energy consumption is not concerned in this dissertation.
- This dissertation proposes a forwarder selection algorithm and a reliable broadcast protocol for vehicular ad hoc network.
- The proposed algorithm and the proposed protocol do not require any geographical knowledge or any global information.
- The proposed algorithm and the proposed protocol work distributed systems. Every node has its self-decision.

- The proposed algorithm and the proposed protocol use only information from 1-hop neighbors.
- The proposed protocol is designed for small data broadcasting not streaming.
- The proposed protocol supports for applications that do not sensitive to delay such as navigators, advertisement and entertainments.
- The proposed algorithm and the proposed protocol are evaluated on Network Simulator (NS3) with realistic mobility traces from Simulation of Urban Mobility (SUMO)

Additionally, we assume the following:

- There is no interference from obstacle, such as building and there is no a selfish node in networks.
- All vehicles are equipped with wireless devices.
- All packets will be broadcast until they are delivered to all nodes or they expire.

1.3 Summary of Contributions

The main contribution of this dissertation is a new non-geographical knowledge data dissemination on vehicular ad hoc networks. The proposed work consists of a forwarder selection algorithm and a reliable broadcasting protocol. There are the interesting properties as following explanations. First, this work does not require any geographical knowledge for any operations so it can avoid a privacy issue and it can tolerant inaccurate local data. Second, this work is a self-decision algorithm. Each node can operate with its own decision that relies on its own information. A centralize leader and global information are not needed. This can provide fast and reliable data dissemination to a distributed system. Finally, this work is designed to be practical and scalable. It can be implemented and support for a real devise in a real scenarios. An evaluation was done on the realistic simulation with maximum 2706 nodes in a scenario.

1.4 Dissertation Organization

The rest of the dissertation is organized as follows. The next chapter describes background to understand the problem and the tradition solution. This chapter also includes the literature review that contributes to this dissertation. Chapter 3 presents our proposed reliable broadcasting protocol that is explained in three main mechanisms; a node selection mechanism, a waiting timeout mechanism and a beacon mechanism. In Chapter 4, we evaluated performance of our proposed work in each main mechanisms and the complete protocol. Finally, Chapter 5 concludes the dissertation and discussion for further research.

CHAPTER 2
Background, Related Work and Motivation

2.1 Background

2.1.1 Wireless Ad Hoc Network

Wireless networks are widely used in daily life but devices in these wireless networks need some centralized infrastructure to be intermediate nodes between each device. The well-known networks are wireless local area networks and cellular networks which both of them respectively need access point devices and base stations to deliver data between client devices. So client devices in these networks can directly communicate to their 1-hop infrastructure. On the other hands, wireless ad hoc network is a multi-hop communication. Each device can act as a client node or an intermediate node. An advantage is nodes in wireless ad hoc networks do not require any infrastructures for their communication. Therefore, this type of networks can operate in the place that cannot deploy any infrastructure or it can operate in some disaster situations that existing infrastructures are destroyed [21].

Wireless ad hoc network is a distributed system. Each node has a self-decision for data transmission. This leads to some of problems that do not occur in regular wireless networks. One of the most well-known problems is a hidden terminal problem [22] that is the primary cause of lost data in wireless ad hoc networks. An example of hidden terminal situation is shown in Figure 2.1. There are A, B and C in the area. Let A and C would like to transfer data to B but they are not in the transmission range of each other. So if A and C concurrently transfer data to B, the transferred will be collided and B will lose the received data. In order to solve this problem, IEEE802.11 [23] uses request to send (RTS) and clear to send (CTS) before nodes transfer any data. A node that would like to send out the data will broadcast RTS message to its destination. If its destination responses with CTS message, it will start to transfer the data. Unless its destination responses with CTS message, it will set backoff interval and it will broadcast RTS message again when its backoff interval expires. The RTS/CTS mechanism is deployed in MAC layer. In MAC layer, nodes also send an acknowledgement to response to its sender but this mechanism can ensure the reliability only within 1-hop. So data dissemination in wireless ad hoc networks does require a mechanism in higher layer than MAC to achieve the reliability.

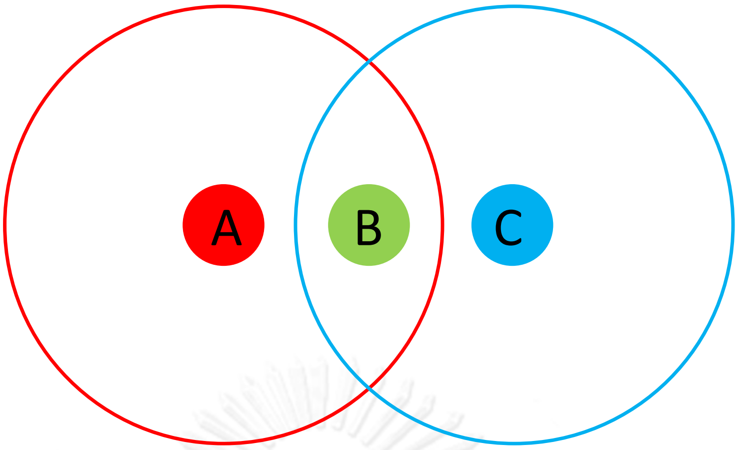


Figure 2.1 A Hidden terminal problem.

Vehicular Ad hoc Network (VANET) is an outgrowth of wireless ad hoc network. It shares the same advantages as other wireless ad hoc networks that nodes can directly communicate with each other. This leads VANET to attract many researchers as a new way of communication for vehicles. In the past few years, several projects have been proposed, such as CarTalk (Thailand), CVIS (European Commission) [24] and IntelliDrive (US) [25]. This type of communication can cooperate with satellite, cellular or other short range communications as illustrated in Figure 2.2. There two types of communication can be categorized: Vehicle-to-Vehicle communication (V2V) and Vehicle-to-Infrastructure communication (V2I). These communications support many applications and services that are introduced in Intelligent Transportation System (ITS). Some of applications and services are design for a safe and convenient driving, such as a collision avoidance system, a navigator, a traffic management system and an electronic toll collection, but the others have different purposes, e.g., commercial and entertainment. An example of these applications and services can be shown in Figure 2.3. Therefore, an efficient data dissemination is a key success for such applications.

Designing data dissemination mechanism on vehicular ad hoc network is challenging because a vehicle has a unique mobility characteristic. A node in general wireless ad hoc networks can be represented by a random waypoint model but a realistic mobility trace for vehicular environment need to be a specific model. The model for vehicles has to consider a road structure, a traffic light, a type of vehicle and a driver’s behavior. As a result, previous works that have been proposed for data dissemination in general wireless ad hoc networks cannot work well on vehicular environment. The important issues that cause the performance degradation of previous general works are a broadcast storm issue [2] and a long-time disconnection issue [3]. Moreover, VANET has its own standard for a physical layer and a medium control access layer that should be concerned.

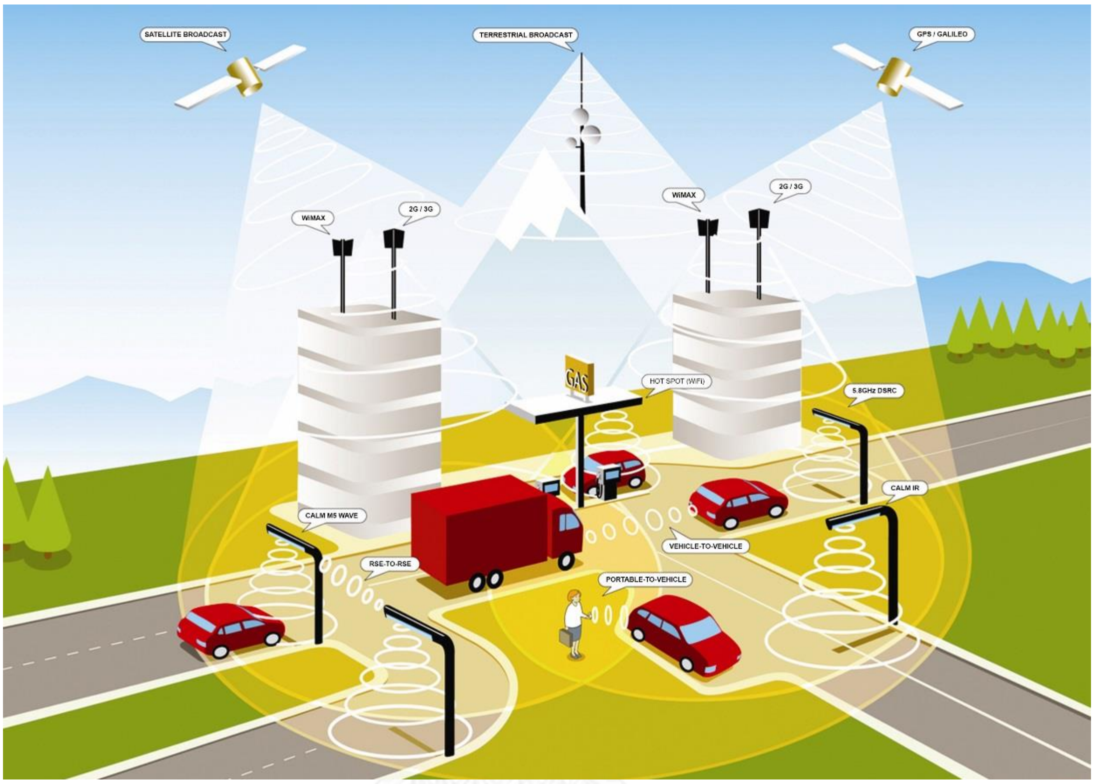


Figure 2.2 Vehicular networks [24].

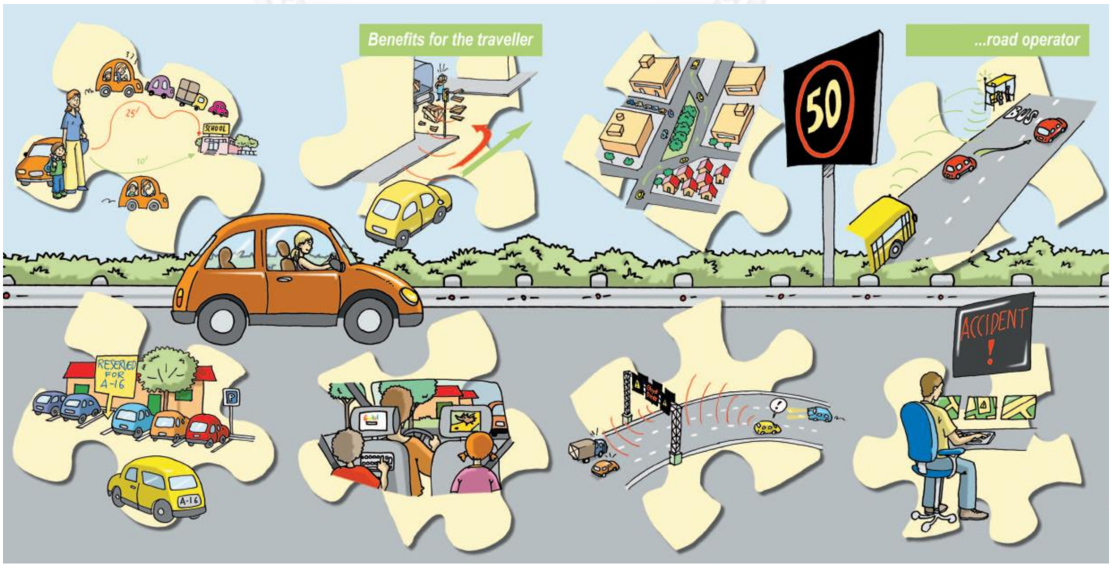


Figure 2.3 Applications and services in intelligent transportation system [24].

2.1.2 IEEE1609 WAVE and IEEE802.11p

IEEE1609 (WAVE: Wireless Access in Vehicular Environments) and IEEE802.11p are standards for wireless communication between vehicles. IEEE802.11p consists of standard for medium access control layer and physical layer while IEEE1609 WAVE relates to the higher layer as shown in Fig.2.4.

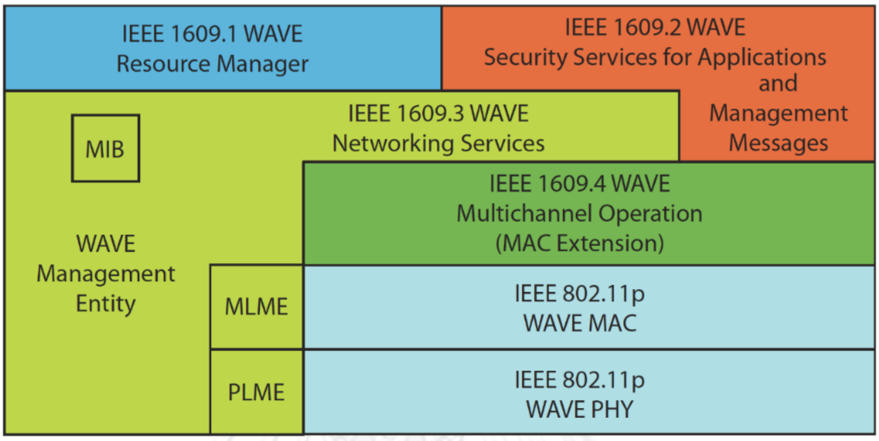


Figure 2.4 IEEE1609 (Wave) and IEEE802.11p structures [26].

A. IEEE1609 (WAVE: Wireless Access in Vehicular Environments) [27]

There are four subclass of IEEE1609. Each subclass manages for each module as following list.

- IEEE 1609.1: Resource Management
- IEEE 1609.2: Security Services for Application and Management Messages
- IEEE 1609.3: Networking Service and WAVE Management Entity. This subclass also has extension management for MAC and PHY that are MAC Layer Management Entity (MLME) and PHY Layer Management Entity (PLME).
- IEEE 1609.4: WAVE Multichannel Operation (MAC Extension). This subclass is a standard that connects to IEEE802.11p. It can separate data to channels, such as Control Channel for control data or Service Channel for application data. Service channels can be categorized to Non-Safety, Traffic Efficiency and Critical Safety as illustrated in Fig.2.5. Figure 2.6 shows the cooperation between IEEE1609.4 and IEEE802.11p MAC.

Figure 2.5 Multi-channel operations for vehicular networks in IEEE1609 (Wave) [26].

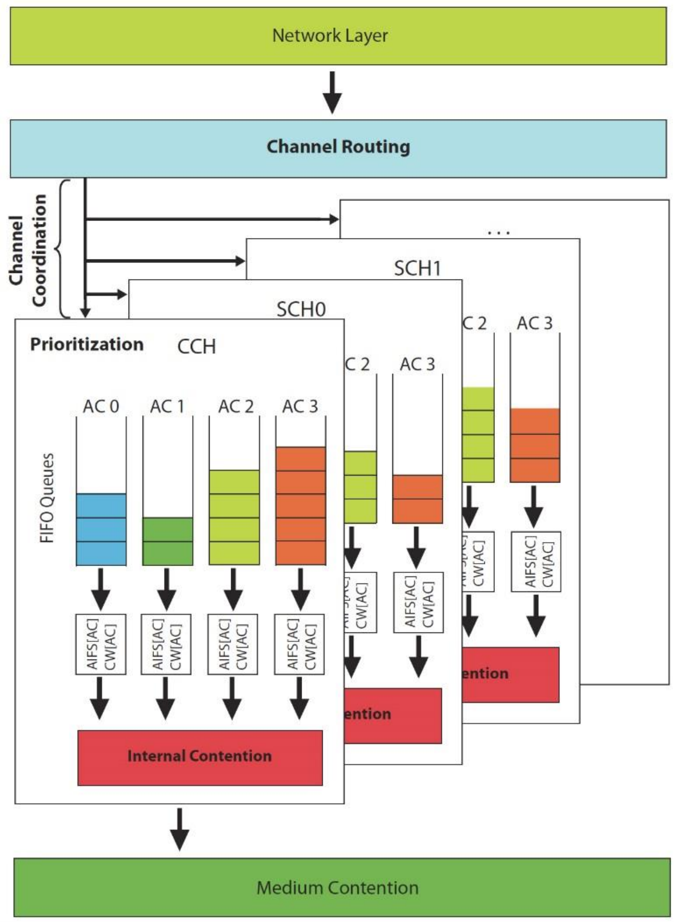


Figure 2.6 Relationship between multi-channel in IEEE1609 (Wave) and access categories in IEEE 802.11p [26].

B. IEEE802.11p (DSRC: Dedicated Short Range Communications) [28]

Frequency ranges of DSRC are authorized by Federal Communication Commission (FCC) in 1999 for vehicular networks. The frequency is 5.9 GHz with bandwidth 75 MHz. In 2000, DSRC is in the control of IEEE and the standard is called IEEE802.11p.

There are 4 types of MAC access priority that are AC0 - AC3 (Access Categories). The highest priority is AC3. This mechanism is called an Enhanced Distributed Coordination Function (EDCA). It is also used in IEEE802.11e. Each type of priority has its own access time and its own back off interval as illustrated in Fig.2.7. The comparison of each IEEE 802.11 standard can be concluded in Table 2-1.

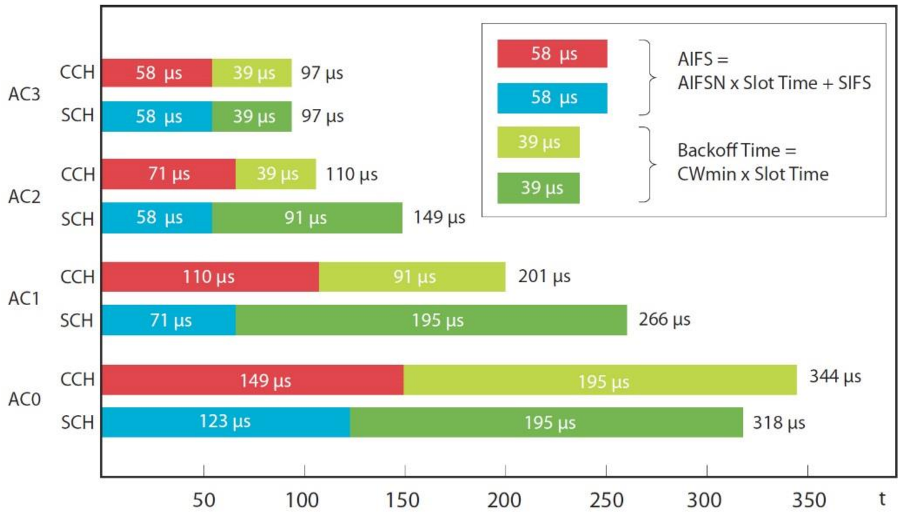


Figure 2.7 Access categories for MAC access priority in IEEE 802.11p [26].

Table 2.1 *Comparison of IEEE 802.11 standards.*

IEEE	Release[23]	Frequency (GHz)	Bandwidth (MHz)	Data Rate (Mbps)	Modulation	Outdoor Distance (m.)
802.11a	Sep. 1999	5	20	54	OFDM	120
802.11b	Sep. 1999	2.4	20	11	DSSS	140
802.11g	Sep. 2003	2.4	20	54	OFDM,DSSS	140
802.11n	Oct. 2009	2.4/5	20/40	72.2/150	OFDM	250
802.11p	Nov. 2010	5.9	10	6-27	OFDM (doubling 802.11a parameter)	Up to 1000

2.1.3 Connected Dominating Set

A "Dominating Set" (D) can be defined in graph theory. For a graph $G = (V, E)$, D is a subset of V. Every vertex that is not in D has to be adjacent at least one member of D. So a "Connected Dominating Set" means every member of D has to be adjacent to at least one member within D too. Then the possible answers of connected dominating set of G are subset of the possible answers of regular dominating set. Figure 2.8 shows the examples of regular minimum dominating set and connected dominating set.

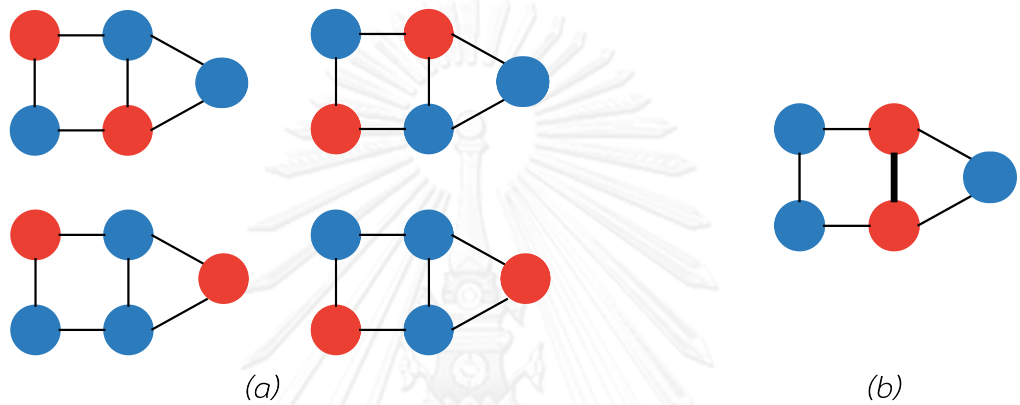


Figure 2.8 Examples of dominating set (a) Minimum dominating set and (b) Minimum connected dominating set.

The dominating set problem is a classical NP-Complete decision problem [10]. It believes that there is no efficient algorithm that can find the minimum dominating set for a given graph. The example of exact algorithms for minimum dominating set of an n-vertex graph can be list as shown in Table 2.2. For an approximation algorithm, a greedy algorithm can find an answer with an approximation factor $1+log|V|$ of minimum dominating set. Raz and Safra [29] also show that no algorithm can achieve an approximation factor better than $clog|V|$ for some $c > 0$ unless $P=NP$. But all of these algorithms require global knowledge of topologies. Therefore, these algorithm cannot be used in this research. The heuristic algorithms that use only local knowledge are discussed in related works.

Table 2.2 *Computation complexity of different algorithms.*

Algorithm	Computational complexity of n-vertex graph
Inspecting by all vertex subset	$O(2^n n)$
Fomin et al. [30]	$O(1.7159^n)$
Fomin et al. [31]	$O(1.5137^n)$
Van Rooji et al. [32]	$O(1.5048^n)$

2.1.4 Bloom Filter

Burton Howard Bloom has introduced a bloom filter since 1970 [16]. Bloom filter is designed for reducing memory usage and disk access in limited resource system. It is applied to many applications in computer science [33] including VANET [34-36] An empty Bloom filter contains a bit array that all bits set to '*0*'. When an element is added, it will be fed into multiple hash functions. These functions mark '*1*' bits in positions of bit array to represent the element. For example, as illustrated in Fig. 2.9 when '*a*' is added, a value from hash functions will set '*1*' bits at position *(1,7)* to represent '*a*'. To check whether an element is a member of a set, the element is fed to the same hash functions. The results will be compared with the existing filter. If all bits in the result have the same '*1*' bit positions, this element probably is a member of this set as '*a*' in Fig. 2.9. Otherwise, this element obviously is not a member of this bloom filter as '*d*'.

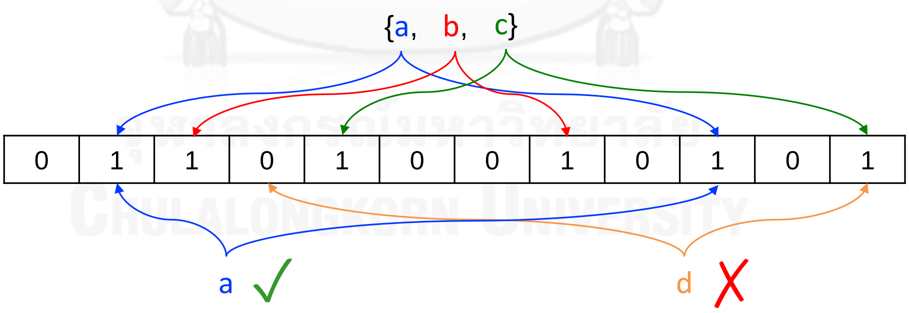


Figure 2.9 An example of Bloom filter operations.

As previously mentioned, the occurrence of false positive error depends on bloom filter size and number of elements in set. Let m is the number of bits in a bloom filter array, n is the number of elements in a set and k is a number of hash functions. The probability of false position error, p can be derived as shown in equation (2.1) [33] so to minimize the rate of false positive k can be calculated by equation (2.2). If we know the number of elements and acceptable rate of false positive error, we can theoretically calculate the number of hash functions and the number of bits in an array.

$$k = \frac{m}{n} ln2 \tag{2.1}$$

$$p = (1 - (1 - \frac{1}{m})^{kn})^{k} \tag{2.2}$$

Another property that is unlike other hash table is union and intersection of bloom filter that has the same size can be respectively implemented with bitwise OR and AND operation. A union is a lossless operation but an intersection can loss some information that increase the rate of positive errors. This property can reduce complexity of computation in some protocol algorithms.

As mentioned earlier, the first property is that the size of a bloom filter does not depend on the number of elements and the size of elements. This property helps create a fixed-size beacon that contains variable size data structures. Another property of bloom filters is set operations of bloom filters can be implemented by bitwise operations that can reduce the complexity of algorithm that will be discussed later. Consequently, a bloom filter can be an essential solution for an efficient beacon in VANET.

2.2 Related Work

In this section, we have reviewed the related work in literatures. The first subsection is about the works on reliable broadcasting protocol on VANET. These works proposed forwarder selection algorithms and broadcasting mechanisms. Another subsection considers previous solutions for overhead reduction and collision from beacon.

2.2.1 Reliable Broadcasting Protocol on Vehicular Ad Hoc Networks

Simple flooding is a tradition 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 intermittent connectivity in VANET but all nodes still rebroadcast packets as 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 literatures. 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 based on their self-decision algorithm into two groups. The first group makes 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). They use position or direction of nodes to make decision. The example of protocols are PGB [37], EAEP [5], POCA [7] and DV-Cast [8].

Another group of protocols makes decision by node properties. A node in these protocols makes decision by comparing its properties to its neighbors. This group is focused in our work because protocols in this group can avoid using the geographical knowledge that causes privacy issue. These protocols use the density information (number of 1 hop neighbors) and the topology information such as a list of 2-hop neighbors and relationship between neighbor nodes. The interesting protocols in these groups are DECA [9] and APBSM [6].

Each type of protocols is illustrated in Fig.2.10 and the mentioned protocols are described as following.

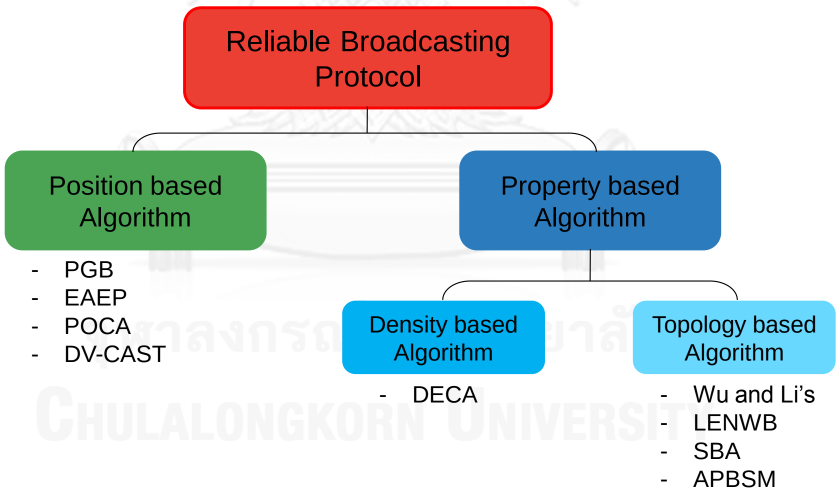


Figure 2.10 Different types of reliable broadcasting protocols.

PGB [37] (Preferred Group Broadcast) is a broadcasting mechanism in CAR protocol. PGB is designed to avoid collision in dense area, such as traffic light area. PBG can solve a collision problem that can be found in RREQ mechanism of AODV. PGB calculate a waiting timeout for each node. A node with the shortest timeout will rebroadcast the packet. Nodes at the edge of broadcasting circulate have shorter waiting timeout than nodes that are closer to 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 the higher probability value than other nodes. However, probability function can cause redundant retransmission in the same area. EAEP also need to know broadcasting direction so EAEP is an appropriate solution for highway environments. EAEP outperforms Simple Flooding in terms of efficiency and overhead but it takes about 30 seconds to deliver data to most of vehicle in highway topology. Moreover, it cannot support for long-tine disconnected environments that can be found in highway scenarios.

POCA [7] (Position-Aware Broadcasting Protocol) uses the geographical knowledge to select the next rebroadcast node. A node with furthest distance to the source node will be selected. The source node piggybacks the selected node's identifier to the broadcasting packet. The selected node will immediately rebroadcast once it receive the packet. This mechanism avoids the delay from waiting timeout. POCA also attaches acknowledge messages to its beacon so it can recover missed messages in the long-time disconnected scenarios. POCA is the fastest protocol that uses GPS information. It also provide very low overhead.

DV-Cast [8] (Distributed Vehicular Broadcast Protocol) is deigned to disseminate data to warn most of vehicles when an accident is happened. DV-Cast uses the broadcast suppression mechanism that is a probability function. This function relies on the distance to a source/precursor node. A node that is the furthest node to source node, have 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 to forward the packets while it is running away from source node. DV-Cast also periodically broadcast a beacon message but it is used only for neighbor discovery so DV-Cast has the same problem with EAEP. It cannot handle the long-time disconnected scenarios.

DECA [9] (Density-Aware Broadcasting Protocol) relies on only the density information. A source node makes a decision by selecting its neighbor with 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 beacon to reduce overhead in dense areas. DECA provide very high speed of data dissemination while it produces the lowest overhead. But DECA cannot perform well in high complexity scenarios, such as a Manhattan grid topology or a realistic topology.

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 neighbor to construct CDS. The CDS is calculated by Stojmenovic's algorithm [14] which is extended from Wu and Li's algorithm. Stojmenovic's algorithm is a self-decision algorithm. Nodes eliminate themselves from CDS members. If they do not complete all conditions of algorithm, they will not be CDS members. Nodes in CDS are preferred nodes for rebroadcast. These nodes have shorter waiting timeout than other nodes. APBSM can perform well in both highway and urban scenarios. As reported in [6], it outperforms DV-Cast in terms of reliability and overhead.

The algorithms that use topology information can also be found in broadcasting protocol for regular mobile ad hoc networks. Although these protocols are not specifically designed for vehicular environment but these algorithms are interesting because they try to construct CDS members without any geographical knowledge. Some of algorithms can be extended and used for data dissemination in vehicular ad hoc networks.

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 is a set of node A's neighbors and N_{NB} is 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 is 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 is a set of node A's neighbors, N_B is a set of node B 's neighbors and N_C is 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. The computation complexity of internode condition, intergateway condition and gateway condition respectively are $O(n^2)$, $O(n^2)$ and $O(n^3)$.

LENWB [13] (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 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 [12] has the similar elimination algorithm as found in LENBW. Upon receiving the broadcast 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.

Stojmenovic's algorithm [14] uses combination of self-decision CDS forming from Wu and Li's algorithm and rebroadcast node elimination in SBA. According to Wu and Li's algorithm, Stojmenovic's algorithm uses geographical knowledge instead of 2-hop neighbor list. Nodes still can check whether it can complete all three conditions or not. 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 as same as in SBA. This algorithm has been extended and used in APBSM.

These protocols are based on topology properties. They use 1-hop neighbor list or 2-hop neighbor to select the CDS members. The advantage is these algorithms do not any geographical knowledge but they are designed for general mobile ad-hoc networks that may not be efficient in vehicular environment.

2.2.2 Beacon Mechanism for Protocols on Vehicular Ad Hoc Networks

VANET is a distributed system, a protocol needs a solution for neighbor discovery and local information sharing. The only way is to broadcast a hello message or a beacon. All of necessary data for the protocol are attached to the beacon. Although a beacon is a small packet that is periodically broadcast to maintain accuracy, too much data can cause a bulky beacon. This leads to a contention problem due to limited resources in wireless networks. The European Telecommunications Standards Institute (ETSI) standards the beaconing rate at 1-10 beacons per second depending on applications [38]. Several solutions have also been proposed to reduce a number of beacons [17-19, 37]. These works appropriately adjust the beacon rate following environments, vehicle density, neighbor change rates, and speed of vehicles, wireless channel conditions, communication reliability, and delay. The detail of each work is described below.

Connectivity Aware Routing (CAR) [37] and our previous work, Linear Adaptive Interval (LIA) [17] dynamically adapt a beacon interval to vehicle density. The vehicle density is the number of 1-hop neighbor. The interval is adjust with linear equation that lengthen the beacon interval when a node is in a dense area to reduce probability of wireless collision. Thaina et al. [19] use a linear regression and a k-nearest neighbor classifier to appropriately adapt the beacon interval with vehicle density information.

Adaptive Traffic Beacon (ATB) [18] is only an algorithm that use both vehicle density and networks condition, such as wireless channel conditions, communication reliability, and delay.

The others try to prevent the collision of beacon broadcasting by desynchronization method [20] because the collision increases the overhead from data retransmission and it also affects the accuracy of information. The desynchronization method is a self-organizing time scheduling. Each node listens to other's transmission interval then it try to shift its transmission time to an empty slot without changing beacon interval. This work can significantly reduce collision appearance from beacon while all nodes maintain the same beacon interval.

Although these works can reduce a number of beacons, the information in a beacon depends on a type of protocol and a purpose of protocol. These works cannot solve the bulky beacon problem. This is the motivation of our proposed work.

2.3 Motivation

Most of previous protocols require the position or GPS for their operations, which can violate to privacy issue because nodes have to advertise their position to others. Some of them do not support the intermittent connectivity problem due to lacking of acknowledgement messages. Moreover, the efficient solutions need high complexity of computation and high overhead to maintain accurate information. Therefore, we propose a simple algorithm that can operate without any geographical knowledge. The algorithm uses density and topology information. Some properties of graph theory and interesting properties of Bloom filter are applied to simplify our proposed algorithm and to gain its performance near to ideal value. This also provides a fixed size beacon that can significantly reduce overhead. Our proposed protocol is designed based on practical usage to perform well in the real world so our performance evaluation has been done in realistic mobility traces on a real road structure.

Table 2.3 *Comparison of reliable broadcasting protocol for vehicular ad hoc networks.*

	PGB[37]	EAEP[5]	DV-CAST[8]	APBSM[6]	DECA[9]	POCA[7]	NoG+DTA
Forwarder Node Selection Factor	Edge of Broadcasting Circular	Edge of Broadcasting Circular	Edge of Broadcasting Circular	Connected Dominating Set (CDS)	Density	Edge of Broadcasting Circular	Density and Connected Dominating Set (CDS)
Information for Selection Algorithm	Signal Strength (RSSI)	Position (GPS)	Direction and Position (GPS)	Position (GPS)	Density	Position (GPS)	Density and 2-hop Neighbor List
Forwarder Node Selection Mechanism	Waiting Timeout	Waiting Timeout and Probability Function	Waiting Timeout and Probability Function	Waiting Timeout	Source and Precursor Node Selection	Source and Precursor Node Selection	Waiting Timeout
Rebroadcasting Delay	Waiting Timeout and Propagation Delay	Waiting Timeout and Propagation Delay	Waiting Timeout and Propagation Delay	Waiting Timeout and Propagation Delay	Propagation Delay	Propagation Delay	Waiting Timeout and Propagation Delay
Computation Complexity	O(1)	O(1)	O(1)	$O(n^3)$	O(n)	O(n)	O(n)
Intermittent Connectivity Support	No	No	No	Yes	Yes	Yes	Yes

Table 2.3 *Comparison of reliable broadcasting protocol for vehicular ad hoc networks. (Continued)*

	PGB[37]	EAEP[5]	DV-CAST[8]	APBSM[6]	DECA[9]	POCA[7]	NoG+DTA
Information for Waiting Timeout Calculation	Signal Strength (RSSI)	Distance Between A Node and Source/Precursor Node	Distance Between A Node and Source/Precursor Node	Number of 1-hop Neighbors (Reversed Function)	Number of 1-hop Neighbors (Reversed Function)	Distance Between A Node and Source/Precursor Node	Number of 1-hop Neighbors (Directed Function)
Waiting Timeout Usage	Every Rebroadcasting	Every Rebroadcasting	Every Rebroadcasting	Every Rebroadcasting	Intermittent Connectivity Recovery	Intermittent Connectivity Recovery	Every Rebroadcasting
Beacon Interval	Adaptive Beacon Interval	-	Every 1 s.	Every 0.5 s.	Adaptive Beacon Interval (1.5-7 s.)	Adaptive Beacon Interval (1.5-7 s.)	Adaptive Beacon Interval (1.5-7 s.)
Information Inside Beacon	GPS	-	GPS	GPS and Acknowledge Message	No. of 1-hop Neighbors and Acknowledge Message	GPS and Acknowledge Message	No. of 1-hop Neighbors, 1-hop Neighbor List and Acknowledge Message
Beacon Size	Constant	-	Constant	Variable	Variable	Variable	Constant

CHAPTER 3

Non-Geographical Knowledge Broadcasting Protocol

A node's movement changes frequently and rapidly so beacon messages have to be frequently broadcast to provide accurate geographical knowledge for position based protocols. This can cause the broadcast storm problem from beacon transmission. The information from an equipment such as a GPS device also does not provide accurate data due to GPS drift. Moreover, broadcasting location information that can be tracked by unknown people that can be concerned as privacy violation [39-41]. 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. We also propose a new reliable broadcasting protocol to support the mentioned algorithm. The protocol uses a store-and-forward technique to handle intermittent connectivity. A beacon mechanism is employed to help nodes to discovery their neighbors and to recovery missing packets.

In this chapter, we discuss about protocol mechanism and details of three main modules: forwarder node selection algorithm, waiting timeout mechanism and beacon mechanism so each component can be illustrated in Fig. 3.1.

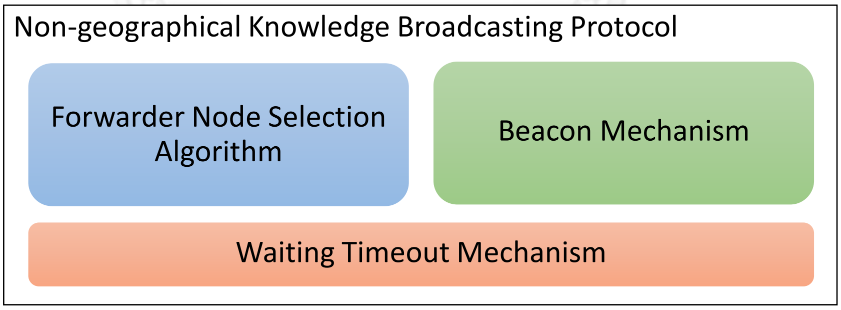


Figure 3.1 Main modules of non-geographical knowledge broadcasting protocol.

3.1 Motivations and Overview

Non-geographical Knowledge Broadcasting Protocol (NoG) consists of three main modules.

(1) *Forwarder node selection algorithm* is a major factor of protocol efficiency. This module selects the next rebroadcast node. If a protocol can maximize a number

of receive in each rebroadcasting, the protocol will be close to ideal retransmission overhead. NoG can support any forwarder node selections that do not require geographical information. In this dissertation, NoG uses our Density based and Topology based algorithm (DTA) to select CDS members that are preferred forwarder nodes.

(2) *Waiting timeout mechanism* is normally used for collision avoidance in a distributed system but waiting timeout mechanism can increase delay to overall system. It needs to be carefully designed and it is not just a random time module because tradeoff between collision probability and addition delay should be considered.

(3) *Beacon mechanism* helps nodes to exchange their local information and it helps nodes to detect the missing packet. Although a beacon is a small packet that is periodically broadcast to maintain accuracy but too much information or too often broadcasting leads to a contention problem due to limited resources in wireless networks. Then the beacon message should be compact and the broadcast rate should be suitable with each vehicle environment.

NoG 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 3.2 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 *C* 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* or *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.

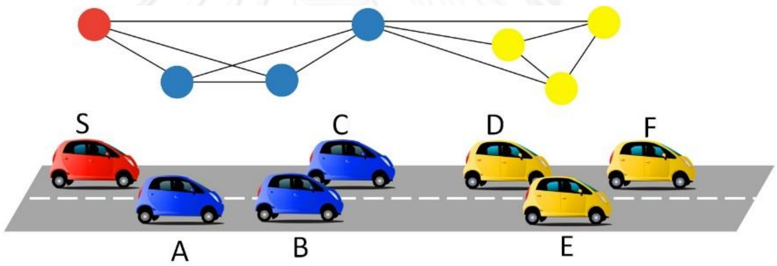


Figure 3.2 A normal broadcasting scenario.

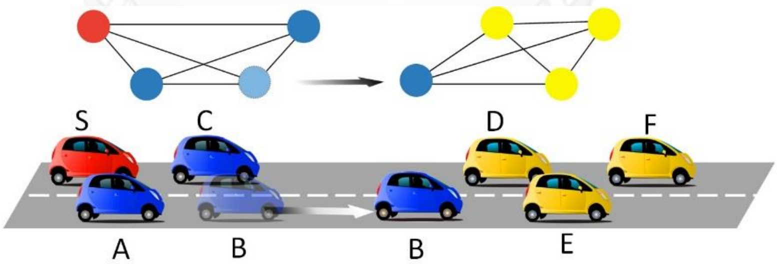


Figure 3.3 An intermittent connectivity scenario.

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 Fig. 3.3. Nodes *A*, *B*, and *C* already received the broadcasting packet from *S*. When *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.

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. Figure 3.4 describes the pseudocode of NoG protocol. The details of main modules also are explained here.

3.2 Forwarder Node Selection Algorithm

An interesting characteristic of vehicle environment is that vehicles always form groups. The vehicle environment is a non-uniform 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, our forwarder node selection algorithm 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.

```
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);
```


Figure 3.4 NoG pseudocode

However, only nodes with the highest density cannot cover all nodes in high density and complex scenarios. The number of covered node results decreased when density of scenarios is increased as shown in Fig. 3.5. This is because a number of nodes in CDS is not enough to cover in low density areas which locate between high density areas as shown in Fig. 3.6. We also compare the CDS members from the density algorithm and the exact CDS algorithm. Figure 3.7 shows that a ratio of CDS members, which are selected from both the density algorithm and the exact CDS algorithm decrease when density is increased. Note that the results in Fig. 3.7 are simulated only in highway scenarios due to computation time of the exact CDS algorithm in urban scenarios. Figure 3.8 shows an example of a complex scenario that consists of low density areas and high density areas. If red nodes are the highest density nodes that will be forwarder nodes in each area, there will be no forwarder nodes to cover blue areas (low density areas). In order to extend the coverage area, other properties need to be considered.

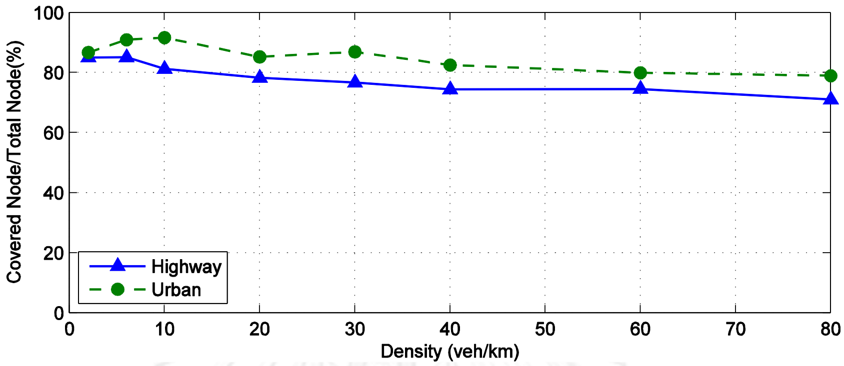


Figure 3.5 Percentages of Covered nodes to total nodes in scenarios.

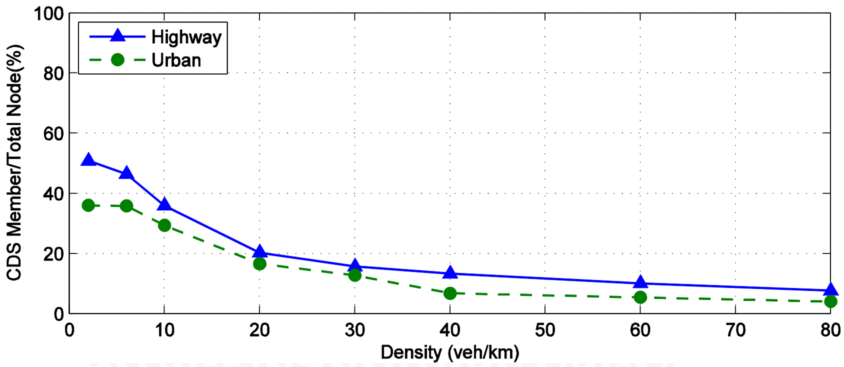


Figure 3.6 Percentages of CDS members to total nodes in scenarios.

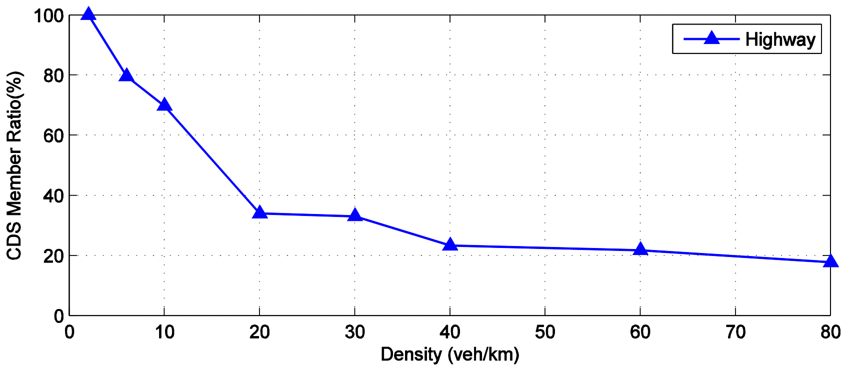


Figure 3.7 Percentages of CDS nodes from a density algorithm that are CDS members from an exact CDS algorithm.

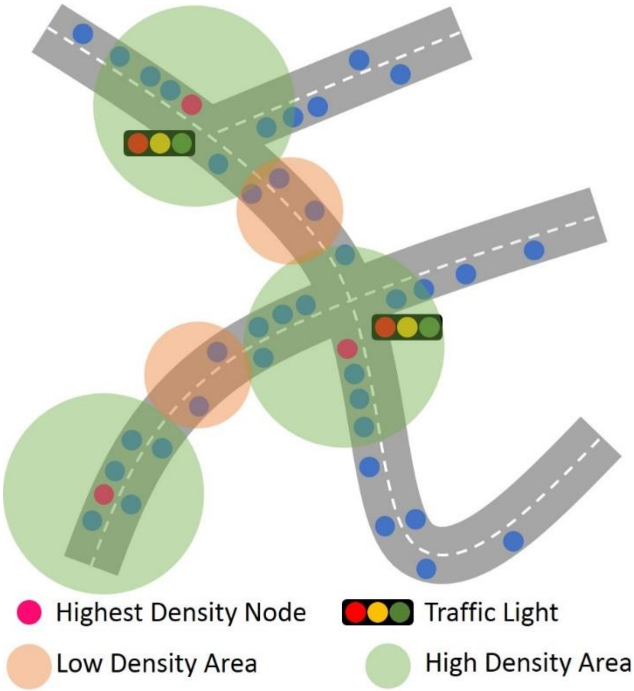


Figure 3.8 An example of a complex scenario.

We create a topology simulator that can generate relationship between nodes in each scenario. The simulator can import a realistic mobility trace from Simulation of Urban Mobility (SUMO) [42] and Traffic and Network Simulation Environment (TraNS) [43]. The simulator can set wireless parameters such as transmission range. It will generate node movements and analyze coverage results and a number of CDS nodes. A flowchart of simulator can be presented in Fig. 3.9. Moreover, we use this simulator with the exact algorithm that inspects all vertex subsets (neighbor node relationship) in each scenario to determine interesting properties for our algorithm.

The interesting properties that we have found are shown as following explanations.

- *Density Condition*: The exact algorithm selects a node to be a CDS member because it has higher number of 1-hop neighbors than the other nodes. As mentioned above, this property is already used in our forwarder node selection algorithm. The density information can be easily retrieved from other neighbors and nodes can simply compute with $O(n)$ complexity so this property is the most efficient factor. However, the number of CDS member is not enough to cover all nodes in complex road structures.

- *Topology or k-Common Neighbor Condition*: This property considers a node has neighbor that is not covered by other k neighbors and these k neighbors are common neighbor to each other. This property can help algorithm to increase coverage results at border of high density area in Fig. 3.8.

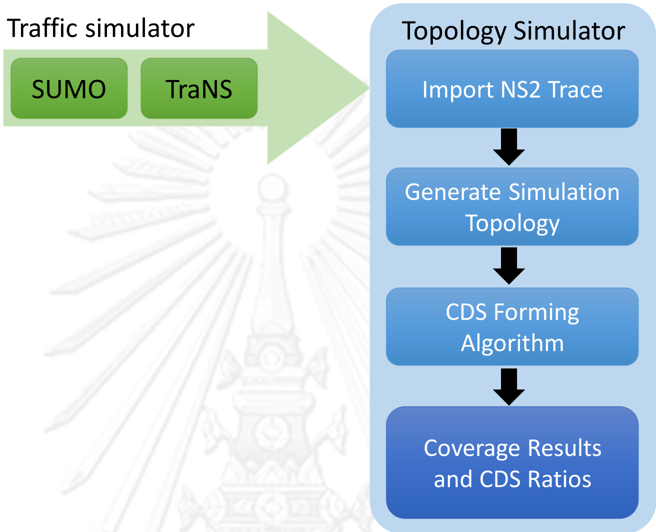


Figure 3.9 A flowchart of our topology simulator.

This condition is an important factor 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 [44], but the maximum transmission range of 802.11p is up to 1000 meters [28]. 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 3.10. 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 CDS member 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 k-common neighbor is an important condition for CDS member selection.

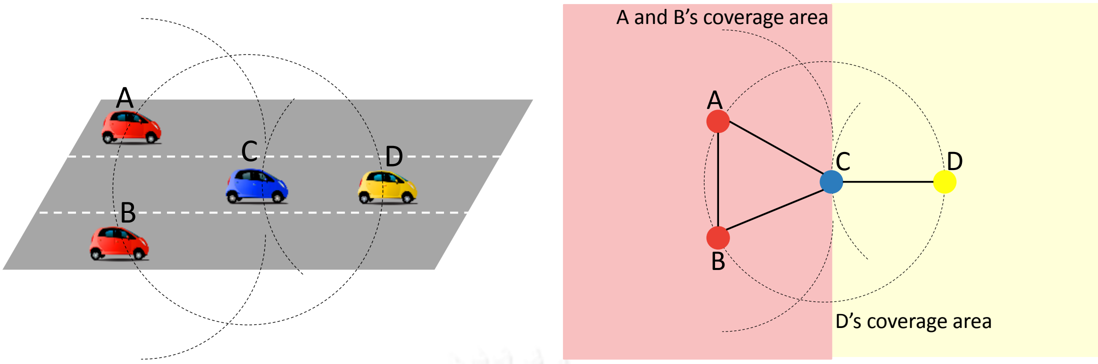
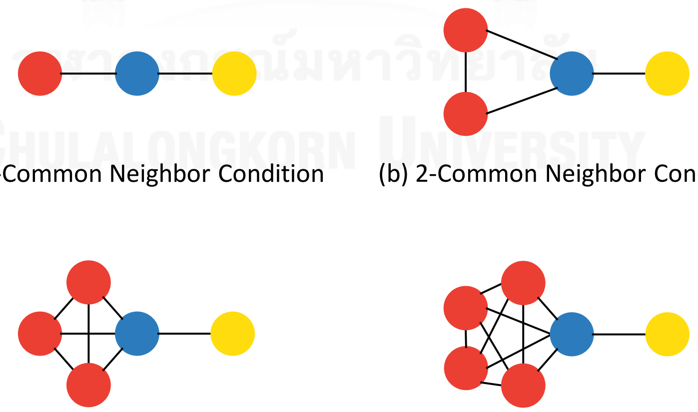


Figure 3.10 An example of 2-Common Neighbor Condition.

If we consider *k-Common Neighbor Condition*, we can illustrate each *k-Common Neighbor Condition* as shown in Figure 3.11 and Table 3.1. Let k is a number of common neighbors, N_{CDS} is a set of neighbors of a node that considers its CDS state and $N_1, N_2, \ldots, N_k$ respectively are sets of neighbors of Node_1, Node_2, ..., Node_k which these nodes are members of N_{CDS} and are common neighbor nodes of each other. Each condition complexity increases when a number of common neighbors (k) increases. The computation complexity also relies on types of information. If this algorithm uses a list of neighbors, a node has to look up to its neighbors' list so it has one additional list-search per one common neighbor. On the other hand, the algorithm that know position of nodes can directly compute neighbors' relationship with their distance but this leads to inaccurate information because it does not know the real communication or real link among neighbors.



(a) 1-Common Neighbor Condition (b) 2-Common Neighbor Condition

(c) 3-Common Neighbor Condition (d) 4-Common Neighbor Condition

Figure 3.11 k-common neighbor condition.

Table 3.4 Summary of k-common neighbor condition

k	Set Condition ($\boldsymbol{x \in N_{CDS}}$)	Computation Complexity	
		List of neighbors	Position (GPS)
k=1	$\exists x \notin N_1$ when $Node_1 \in N_{CDS}$	$O(n^3)$	$O(n^2)$
k=2	$\exists x \notin N_1 \cap N_2$ when $Node_1 \in N_{CDS} \cap N_2$ and $Node_2 \in N_{CDS} \cap N_1$	$O(n^5)$	$O(n^3)$
k=3	$\exists x \notin N_1 \cap N_2 \cap N_3$ when $Node_1 \in N_{CDS} \cap N_2 \cap N_3$, $Node_2 \in N_{CDS} \cap N_1 \cap N_3$ and $Node_3 \in N_{CDS} \cap N_1 \cap N_2$	$O(n^7)$	$O(n^4)$
k=4	$\exists x \notin N_1 \cap N_2 \cap N_3 \cap N_4$ when $Node_1 \in N_{CDS} \cap N_2 \cap N_3 \cap N_4$, $Node_2 \in N_{CDS} \cap N_1 \cap N_3 \cap N_4$, $Node_3 \in N_{CDS} \cap N_1 \cap N_2 \cap N_4$ and $Node_4 \in N_{CDS} \cap N_1 \cap N_2 \cap N_3$	$O(n^9)$	$O(n^5)$

In this dissertation, we use a list of 1-hop neighbors for computation. Moreover, we apply Bloom filter technique that can help us to reduce lookup time to be within $O(n)$ so overall computation complexity of our algorithm is less than the algorithms that use position information. Moreover, this technique can reduce an error issue from inaccurate link state.

In order to use a suitable number of common neighbors, we compare CDS member from each k-common neighbor condition with the exact CDS algorithm. More k value will provide more significant member because the algorithm can accurately separate nodes at edge of broadcasting group and nodes at middle of broadcasting group. In this dissertation, we use *k* that is equal to 2. A reason is when *k* is more than 2, the computation complexity is higher than $O(n^3)$ which cannot be efficiently reduce algorithm to $O(n)$ and false positive error rates from Bloom filter are high.

As mentioned earlier, density is the most efficient factor for CDS member forming in a simple road structure or a simple connection and it also can easily be computed while 2-common neighbor condition is an effective property that can select a right CDS member in a complex road structure or a complex connection. Therefore, we propose a new CDS heuristic 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. This helps DTA to be an appropriate algorithm for vehicular environment that has such a dynamic topology.

There are two conditions for checking CDS state in DTA. 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 Fig. 3.12.

```
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 (2-common neighbor 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;
}}}
```

Figure 3.12 DTA procedure.

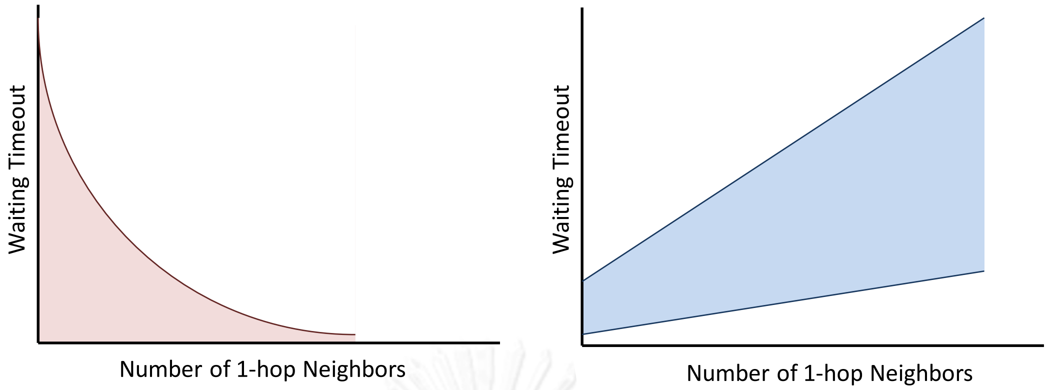
3.3 Waiting Timeout Mechanism

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, a node with the shortest waiting timeout will rebroadcast the missing packet to its neighbors. These two events are explained in Pseudocode of NoG in Fig. 3.4.

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 a situation, protocols should use the number of 1-hop neighbors to be directed variation of waiting timeout function. This waiting timeout function 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 equation (3.1). τ 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. β is a constant value used for expanding the range between minimum waiting timeout and maximum waiting timeout then the best β 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. 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, 2τ, 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 (β). β is used for expanding the range between minimum term and maximum term. The waiting timeout value of each function can be illustrated in Fig. 3.13.

$$W(n) = Random[n\tau, (2 + n\beta)\tau] \tag{3.1}$$



(a) Inversed function (b) Directed function

Figure 3.13 Waiting timeout function.

3.4 Beacon Mechanism

3.4.1 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 8 bytes for each received packet. The beacon structure can be illustrated in Fig 3.14. Although this beacon structure is quite small, the list structure can cause unpredictable behavior. For an example, a node in dense areas that there are many nodes and congest communication. The size of beacon for this node can be extremely large due to the number of neighbors and the number of received packet. In order to solve this issue, we studied the other data structures that can represent data inside set with fixed size value.

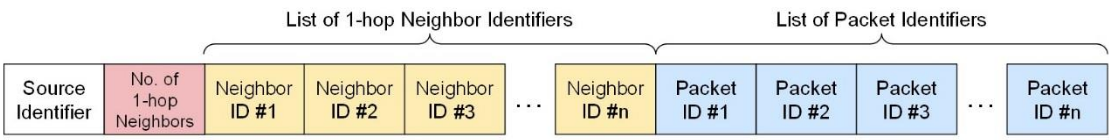


Figure 3.14 Beacon structure.

As mentioned in Chapter 2, Bloom filter has two important properties. The first one is that the size of a bloom filter does not depend on the number of elements inside and the size of elements. This property helps create fixed-size beacons that contain the variable size data structure. Another property of a bloom filter is set operations of bloom filters can be implemented by bitwise operations that can reduce the complexity of algorithm. These properties are unique and cannot be found on other hash function. Consequently, a bloom filter can be an essential solution for a beacon in VANET.

A. Fixed-Size Beacon

As shown in beacon structures, it can be noticed that a list obviously is a variable size data structure taking most of the space in beacon. A Bloom filter has a preferable property because its size is constant even though the number of elements is changed or the size of element increases. In order to fully utilize the space in a beacon, we use a bloom filter to represent all elements inside the lists. The elements are fed into the bloom filter hash functions and a bit array of bloom filter is added to beacon instead of a whole list of the elements. We replace a bloom filter to each list in a beacon as shown in Fig. 3.15. Moreover, an important property of hash that any types of element can be fed with the same hash functions, as a result we can reduce the beacon size by adding elements from all lists in NoG's beacon into a single bloom filter as shown in Fig. 3.16. This indicates that a bloom filter is a solution to create the fixed-size beacon. However, to reduce the size of bloom filters that reflects the size of beacon is a tradeoff between the size and the rate of false positive errors. Moreover, there is a disadvantage when a number of neighbors or a number of received packets are not large enough, a beacon with bloom filter is larger than ordinary one. These issues are discussed in Chapter 4.

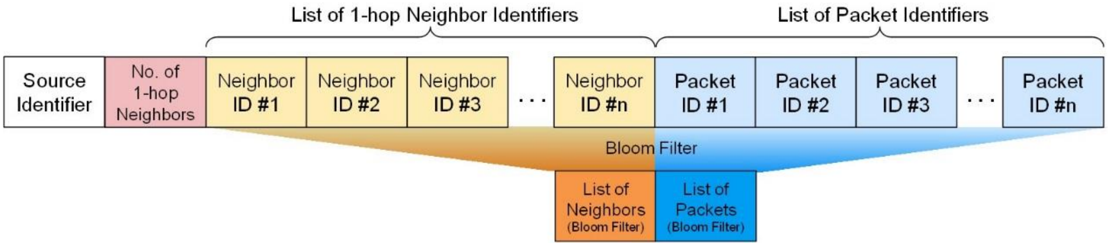


Figure 3.15 Two Bloom filters represent a list of 1-hop neighbors and a list of packet identifiers.

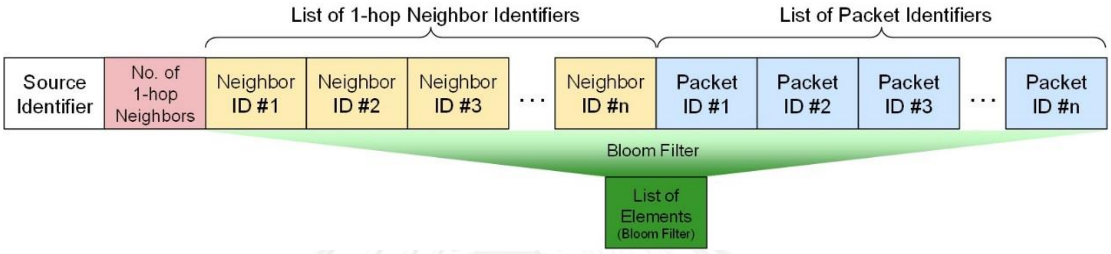
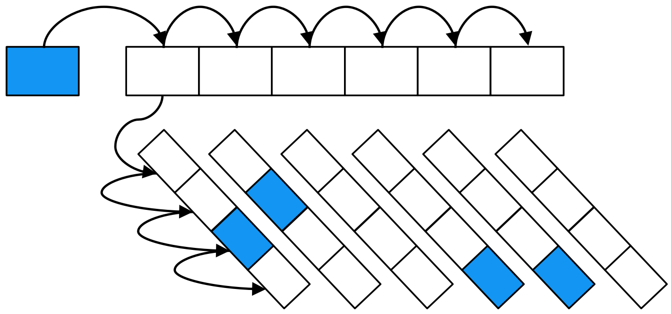


Figure 3.16 One Bloom filter represent both a list of 1-hop neighbors and a list of packet identifiers.

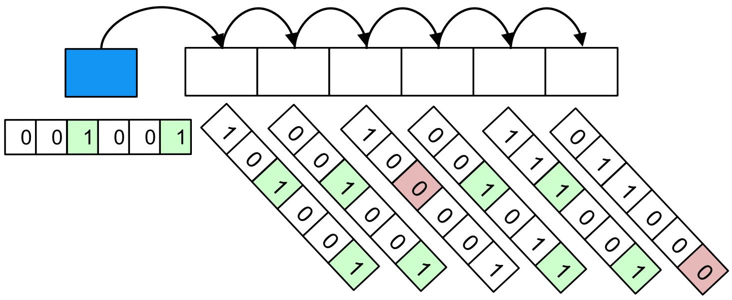
B. Reducing Algorithm Complexity

A property of bloom filter unlike the other hashes is the union and intersection operations of bloom filters can be respectively implemented with bitwise OR and AND operation. As a result of this property, many filters can be reduced to a single filter before checking the membership of an element. For an example, we would like to know that a node is a neighbor of all of other neighbors or not, this can be represent as following set condition.

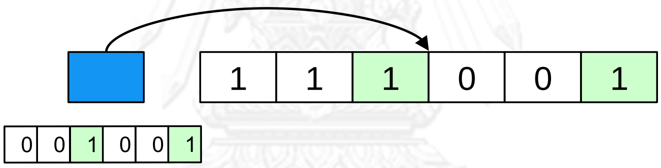
This question normally can be answered by looking up a 1-hop neighbor list of all of its neighbors as shown in Fig.3.17(a). Then this algorithm will spend $O(n^2)$, where n is a number of neighbors. On the other hand, if each neighbor uses a bloom filter to represent its 1-hop neighbors, this question can be answered in $O(n)$ as shown in Fig.3.17(b). The reason is a Bloom filter can represent members inside a list and it can answer a membership state within only $O(1)$. Furthermore, when all bloom filters of all neighbors are already prepared as a single bloom filter with union operation, this single Bloom filter can represent all of neighbors of a node's neighbors. Then the question can be answered in only $O(1)$ as shown in Fig.3.17(c).



(a) Searching from a list of 1-hop neighbor lists.



(b) Searching from Bloom filters that represent 1-hop neighbor lists.



(c) Searching from a union bloom filter that represents all 2-hop neighbors.

Figure 3.17 Computation complexity of each data structure.

This property of bloom filter can be applied to other conditions. Let node *A* is considered its CDS state with 2-Common Neighbor Condition. The 2-Common Neighbor Condition considers a neighbor that is not covered by a pair of *A*'s connected neighbors. We separate the 2-Common Neighbor Condition into two phases. In the first phase, Bloom filters of a pair of *A*'s connected neighbors can be reduced into a single bloom filter by union operation, then this bloom filter represents neighbor lists of these connected neighbors. This operation has no loss but this operation has to be done upon a node receiving a new beacon from its neighbor to reduce the complexity of connected neighbor searching. In the second phase, the union result from each pair can be reduced into one bloom filter by intersection operation. A flowchart of reduced algorithm can be shown in Fig. 3.18 and an example of reduction phase can be shown in Fig. 3.19.

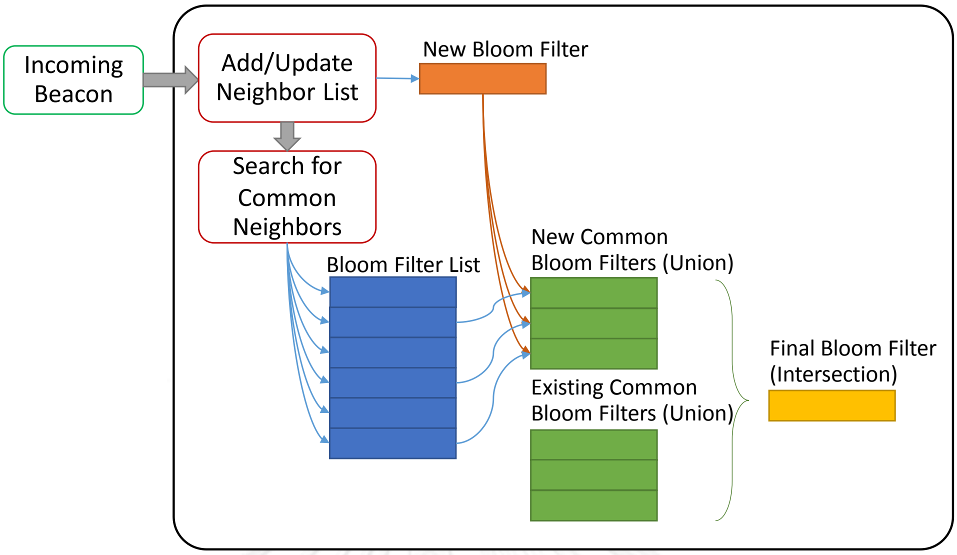


Figure 3.18 A flowchart of reduced algorithm for 2-common neighbor condition.

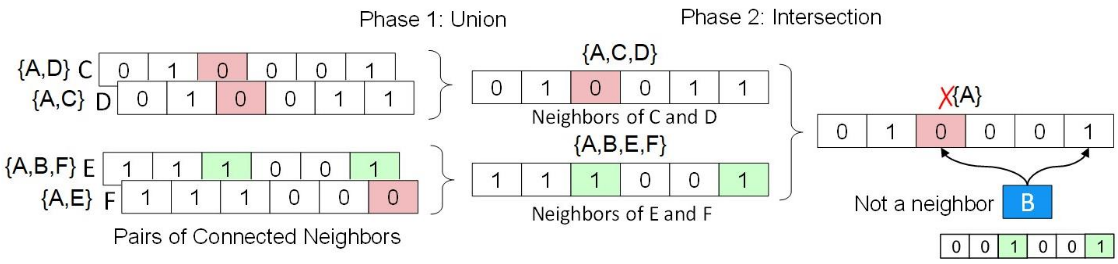


Figure 3.19 Reduced algorithm of 2-common neighbor condition.

For an example, let node *B* be a neighbor of *A*. *B*'s hash value can be represent as "001001". This means that *B* is only a neighbor of *A* and E. *C*, *D* and F are not *B*'s neighbor. In order to consider 2-common neighbor condition, *A* has to have at least a neighbor that is not covered by a pair of its common neighbors. *C-D* and *E-F* are pairs of common neighbors of *A* so we use union operation to combine Bloom filters of *C-D* and *E-F* together. A union Bloom filter will represent all of neighbor of these common neighbor nodes. *B*'s hash value still is included in *E-F*'s union Bloom filter while it is excluded in *C-D*'s union Bloom filter. In a next step, we combine all union Bloom filters of 2-common neighbors into one Bloom filter by intersection operation. An intersection operation can help us to collect all '0' from every Bloom filter so if any pairs of 2-common neighbors do not cover some neighbors, this information will be included in the combined Bloom filter. We can notice that a red

'0' bit of *C-D*'s Bloom filter will be kept until the last Bloom filter while a green '1' bit of B's hash value can be lost because we focus on a pair of common neighbor that do not cover the other neighbors. As a result, we can answer that B is not covered by a pair of *A*'s 2-common neighbor which are *C* and *D* with comparison complexity only in $O(n)$. In this case, *A* is a CDS member.

However, this complexity reduction has an intersection, which can cause a false position error. This will be discussed in Chapter 4.

3.4.2 Beacon Interval Calculation

There is the fact that high frequent beaconing can cause the broadcast storm problem in dense area whereas low frequent beaconing can decrease protocols' performance in sparse area. In the reality, density of vehicles has related to speed of vehicles [45]. So vehicles in dense area usually move slower than vehicles in sparse area. A traffic topology in dense area will change less frequently than a traffic topology in sparse area. Consequently, high frequent beaconing (meaning to short beacon interval) is needed in sparse area but it is unnecessary in dense area. Beacon overhead for each constant interval can be illustrated in Fig. 3.20. More frequency of beacon broadcasting causes more beacon overhead. This leads to higher retransmission overhead in higher density scenarios due to a collision issue as shown in Fig. 3.21. Accordingly, NoG uses adaptive beacon interval as linear function with a limited longest interval to appropriately calculate the beacon interval in each density environment. This can prevent collision occurrences that affect to a number of retransmissions.

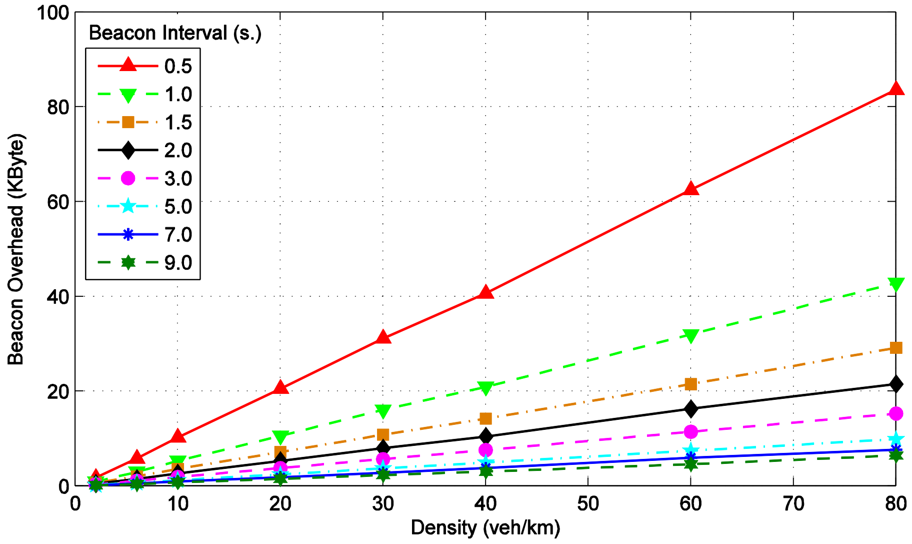


Figure 3.20 Beacon overhead results of constant beacon intervals.

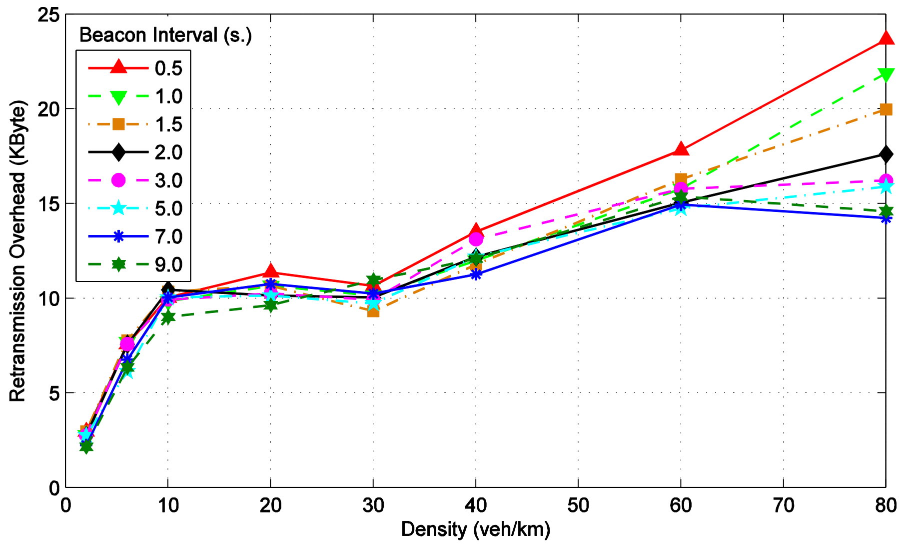


Figure 3.21 Retransmission overhead results of constant beacon intervals.

The algorithm linearly increases the beacon interval based on network density, called Linear Adaptive Interval or LIA [17]. The algorithm can reduce beacon overhead without decreasing the protocol performance. The beacon interval can be illustrated in Fig. 3.22.

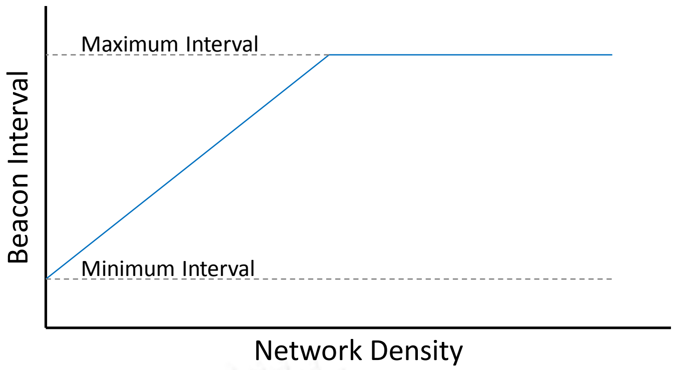


Figure 3.22 Linear adaptive interval calculation.

As mentioned, the beacon interval is linearly increased depending on the network density. The network density ($netDensity$) 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 equation (3.2). The beacon interval ($beaconInv$) calculation is represented by equation (3.3). $minInv$ is a minimum beacon interval. c is a constant value. $maxInv$ is the longest interval that does not affect the performance of protocol.

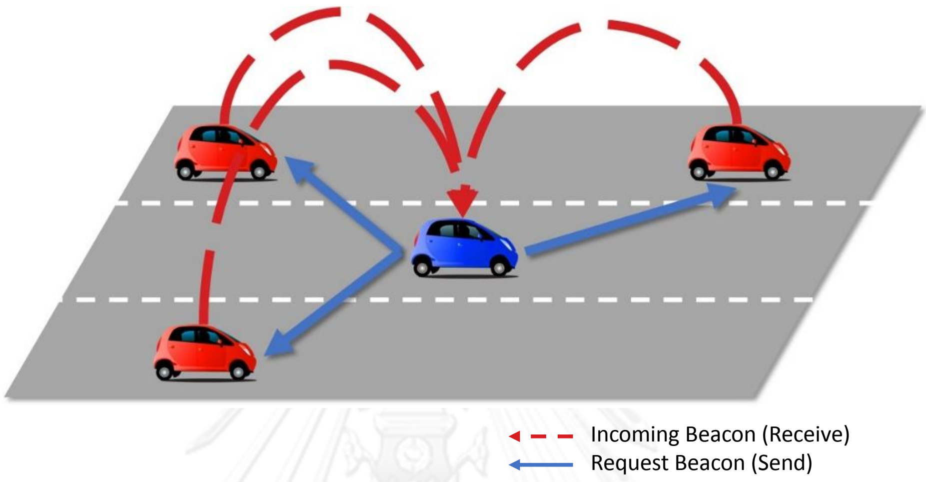
$$netDensity = n + p \tag{3.2}$$

$$beaconInv = \min[minInv + (c \times netDensity), maxInv] \tag{3.3}$$

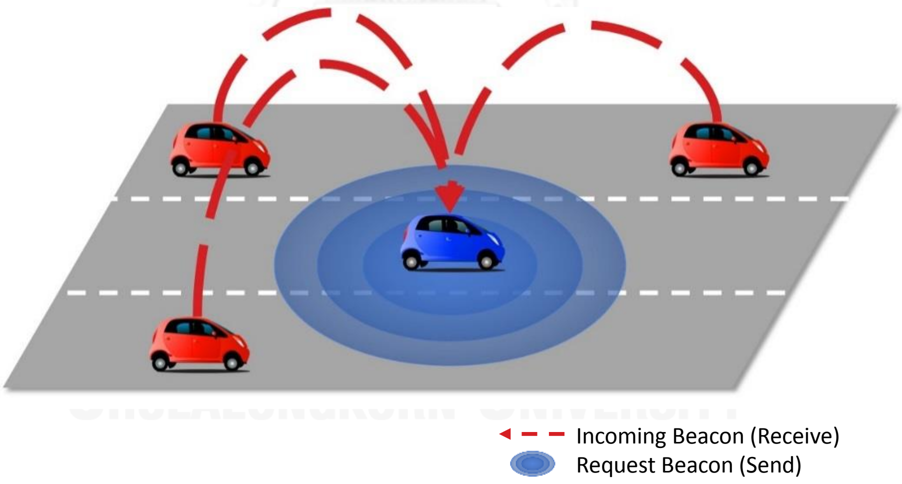
3.4.3 Beacon Mechanism in Missing Message Scenarios

When there are any missing packets, NoG uses acknowledgments in beacon to detect those packets. In the case that nodes detect that their neighbors miss the packets, they will set the waiting timeout. The node with the shortest waiting timeout will rebroadcast the missing packets. In another case that a node finds itself missing a packet by receiving neighbors' beacon, it will send back a beacon immediately to allow its neighbors to know that it misses the packet. So neighbors which have the missing packet can rebroadcast the packet.

Although the mechanism above can help a node to discover and immediately receive the missing packet, it can flood lots of beacons that cause the broadcast storm problem. This is because one beacon from a neighbor will make the node generate one beacon to send back to the neighbor. Therefore, our new beaconing mechanism implemented in NoG limits number of beacons that the node will send back to neighbors when it detects the missing packet as shown in Fig.3.23. This can reduce congestion in the networks. However, limiting number of beacons can reduce speed of data dissemination and decrease overall reliability. In order to find the proper number of beacons, we have simulated our protocol in different number of beacons to send back and see its performance metrics. The parameter tuning will be discussed in chapter 4.



(a) A node send a request beacon per an incoming beacon.



(b) A node send only a request beacon within one beacon interval.

Figure 3.23 Beacon mechanism in missing message scenarios.

CHAPTER 4
Performance Evaluation

This chapter consists of three sections. Section 4.1 describes simulation configurations including types of data, propagation loss models and simulation scenarios. Section 4.2 focuses on parameter setting of each main module and each module evaluation. The last Section 4.3 is the performance evaluation of our proposed protocol in realistic scenarios comparing with other previous work.

4.1 Simulation Configuration

4.1.1 Type of Data and Payload Size

We use three types of data that possibly be disseminated in vehicular environment. Each type of data affects to a number of frames that has to be broadcast. IEEE 802.11 standard specifies the maximum payload size of IEEE 802.11 frame is 2304 bytes [23] but IEEE 1609.4 standard specify the maximum size is 1400 bytes [27]. Then, in this dissertation, the maximum payload size of each frame is 1400 bytes. Table 4.1 shows details of each data type.

- *Message*: This type of data represents a small data size that can be broadcast within one frame. A message is used for updating information on navigator system, traffic information or communication. The size of message is 512 bytes. Most of our simulation use this type of data because the simulation time is much shorter that other types of data and it is the most famous data size in broadcasting in vehicular network research field.
- *Picture*: This type of data represents a small size of picture that can be transferred over vehicular ad hoc networks. The picture can be appeared on the navigator system, traffic monitoring or accident warning system. The resolution of picture is 800x640 pixels with JPEG compression so the size of picture is about 50 Kbytes.
- *Sound*: This type of data represents a short length of telephone quality sound. The sound quality is 8 Kbps bitrate [46] and the length is 15 seconds. Total file size is 120 Kbytes. This type of data can show a limitation of protocol according to number of frames that are generated.

Table 4.1 Details of each data type.

Type of data	Total size (Kbytes)	Number of frame
Message	0.5	1
Picture	50	36
Sound	120	86

4.1.2 Propagation loss model

A propagation loss model affects to quality of transmission or transmission success rate. In this dissertation, we use two types of propagation loss models in Network Simulator 3 (NS-3) [47]: range propagation loss model and Nakagami propagation loss model (Rayliegh fading model). Each propagation loss model configuration is concluded in Table 4.2.

- *Range propagation loss model*: This propagation loss model depends only on the distance (range) between transmitter and receiver. When the receiver is beyond the maximum distance, the received signal power is -1000 dBm. The range propagation loss model can easily setup for simulation and simulation results obviously represent behavior of protocol. As a result, we use this propagation loss model for our parameter setup. We use the maximum range at 250 meters because it represent the realistic transmission range [48] and it also the most famous range using in the vehicular network research. Figure 4.1 shows the transmission success rate of range propagation loss model.
- *Nakagami propagation loss model (Rayliegh fading model)*: As reported in [48], this is the most realistic propagation loss model for vehicular network. The propagation loss model can represent path loss in real vehicular environment that has obstacles including tree, building and vehicles. The suggestion m value for Nakagami propagation loss model is 1. This value also is the same configuration for Rayliegh fading model that concerns Doppler Effect. According to IEEE 802.11p standard, the transmission range is upto 1000 meters with transmission success rate 80% at 250 meters. Figure 4.2 shows the transmission success rate of range propagation loss model.

Table 4.2 Configuration of each propagation loss model.

Propagation loss model	Configuration	Transmission range (meters)
Range propagation loss model	MaxRange = 250	250
Nakagami propagation loss model	m0 = 1, m1 = 1, m2 = 1,	Upto 1000 Transmission success rate 80% at 250 meters

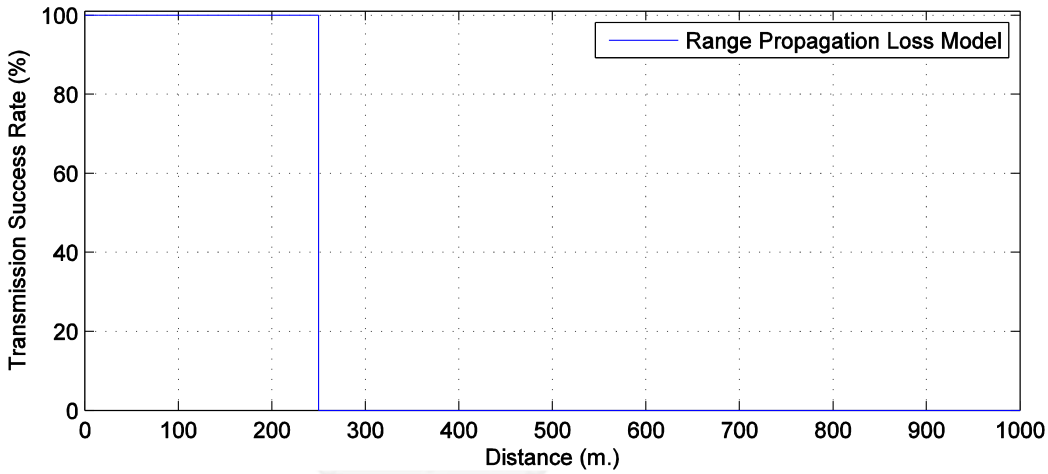


Figure 4.1 Transmission success rate of range propagation loss model.

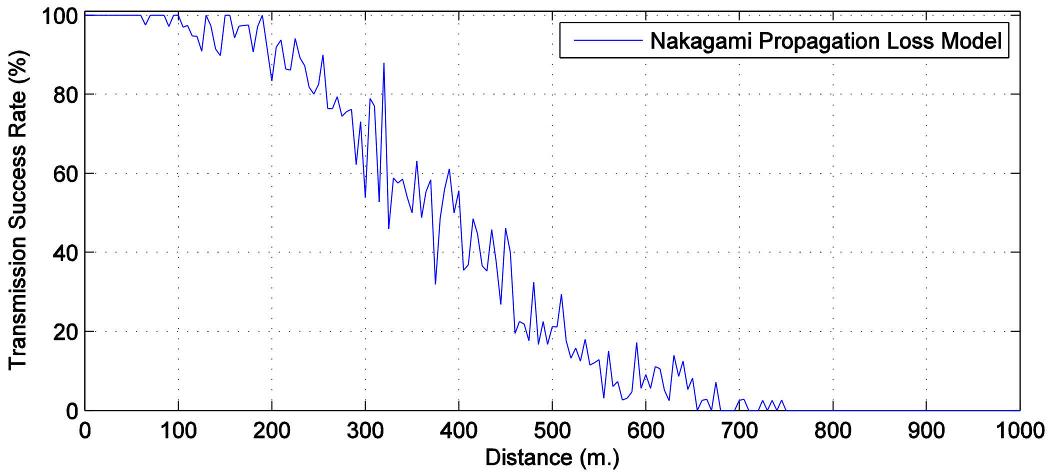


Figure 4.2 Transmission success rate of Nakagami propagation loss model.

4.1.3 Simulation Scenario

If a protocol is well designed, it should operate with the same performance not depending to road structure. In this dissertation, we use two simple road structures and one real map road structure. Details of each road scenario are in Table 4.3. A mobility trace of vehicle are generated from a traffic simulation, called Simulation of Urban Mobility (SUMO) [42]. This traffic simulator provides the realistic vehicle mobility in microscopic level so the behavior of driver are different depending on road, traffic density and type of vehicle. Then, we convert the mobility trace (XML) from SUMO to be in Network Simulator 2 (NS-2) [49] format via Traffic and Network Simulation Environment (TraNS) [43]. A flowchart of creating mobility is shown in Fig. 4.3.

Table 4.3 Details of each road scenario.

Road scenario	Road structure	Vehicle density (veh./km.)	Number of vehicles
Highway scenario	- 4-kilometer length straight road - two lanes for each direction	2 10 20 30 40 60 80	8 40 80 120 160 240 320
Urban scenario	- 2x2 kilometers Manhattan grid pattern - one lanes for each direction	2 10 20 30 40 60 80	24 120 240 360 480 720 960
Real map scenario	- real road structure in Siam area and Sukhumvit area, Bangkok, Thailand - including real traffic lights, number of lanes and one-direction road	2 10 30 60 80	69 345 1035 2070 2760

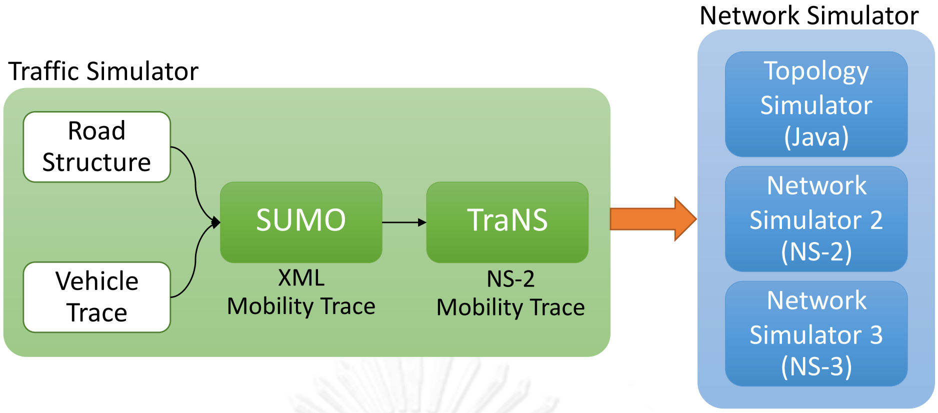


Figure 4.3 A flowchart of creating mobility trace.

All of mobility traces consist of two vehicle maximum speeds; 50 km/h and 80 km/h. A ratio of both vehicle maximum speed is 50:50 and these vehicles are randomly placed in scenarios.

Simple road structure: The simple road structure consists of two type of scenarios. The first one is a highway road scenario that is a 4-kilometer length straight road with two lanes for each direction. The second one is an urban road scenario that is a 2x2 kilometers Manhattan grid pattern with one lane per direction. Both of scenarios can be illustrated in Fig. 4.4. In order to know behavior of vehicles in each scenario, we analyze size of vehicle groups. Figures 4.5 and 4.6 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 non-structure the same as the realistic long distance highway road. On the other hand, vehicles are non-uniform distributed in urban scenarios. There are many several sizes of group in each scenario. Therefore, both simple road structures and mobility traces can represent the realistic environment of vehicles in both highway areas and urban areas. We use these scenarios for parameter setup and preliminary simulation.

Real map road structure: This road structure bases on the downtown road structure of Siam area and Sukhumvit area, Bangkok, Thailand as depicted in Fig. 4.7(a). The scenario includes real traffic lights, number of lanes and one-direction road used nowadays. The vehicle distribution is non-uniform as same as in real traffic scenarios as shown in Fig. 4.7(b). Total road length is 34.5 kilometers with maximum 2760 vehicles.

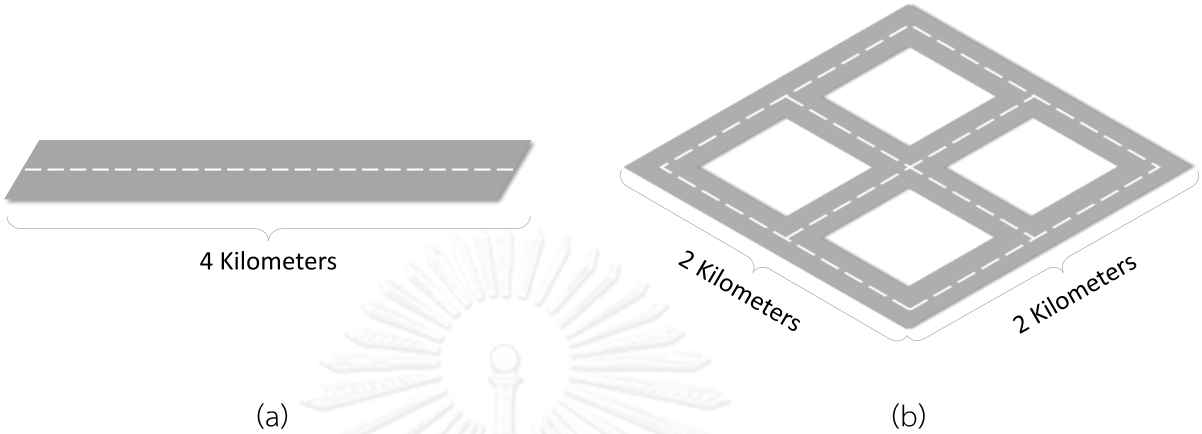


Figure 4.4 Simulation scenarios (a) a highway scenario and (b) an urban scenario.

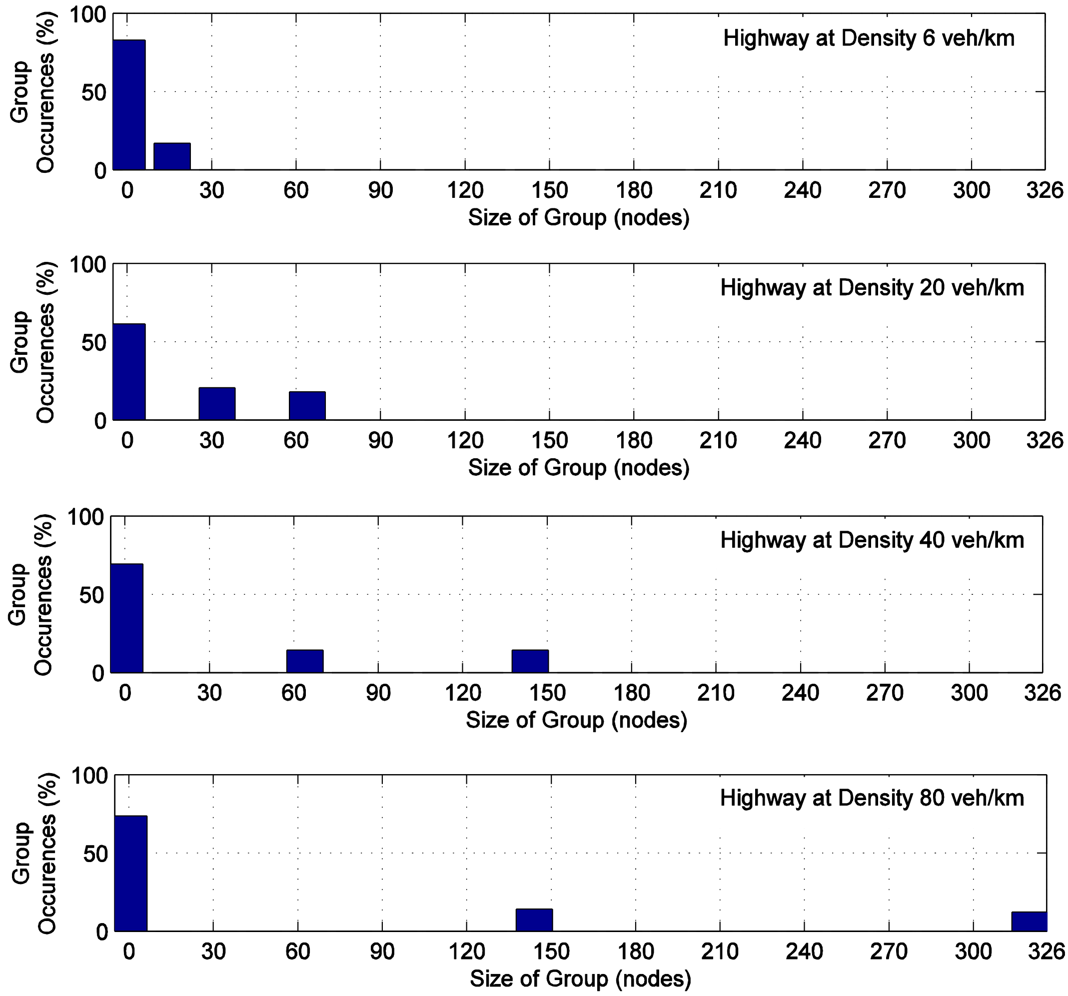


Figure 4.5 Occurrences of each size of group in highway scenarios.

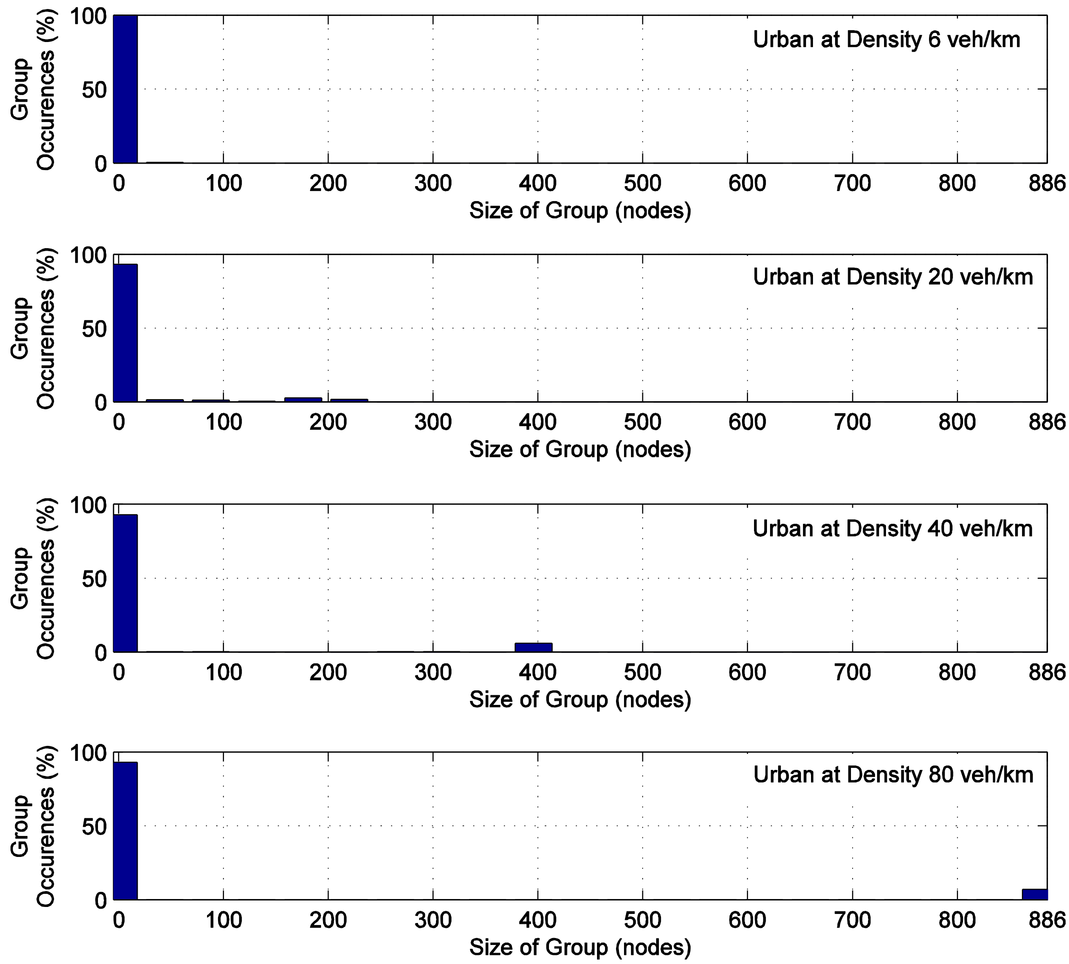


Figure 4.6 Occurrences of each size of groups in urban scenarios.

(a) (b)

Figure 4.7 A real map simulation scenario (a) a real map and (b) a simulation capture.

4.2 Module Level Performance Evaluation

In order to get appropriate parameter configuration and performance of each module, the three main modules of NoG are separately evaluated in this subsection.

4.2.1 Forwarder Node Selection Algorithm

We use our topology simulator to evaluate our proposed DTA in term of coverage and CDS member ratio. 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. 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. In this subsection, simulation configurations are described in Table 4.4.

Table 4.4 *Simulation configurations.*

Simulator	Topology Simulator
Simulation time	200 s.
No. of simulation	100
Road scenario - Vehicle density (veh./km.)	Highway – 2, 10, 20, 30, 40, 60 and 80 Urban – 2, 10, 20, 30, 40, 60 and 80
Propagation loss model	Range propagation loss model with 250-meter transmission range
Compared Algorithm	• *Ideal:* An exact algorithm that consider all of nodes relationship in scenarios. (Only in highway scenarios, due to high computation complexity) • *Density based algorithm (DEN)*: Only nodes with the highest number of 1-hop neighbors are members of CDS. This algorithm represents the density based algorithm. • *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. • *DEN + 1-Common Neighbor*: Members of CDS consist of nodes with the highest number of 1-hop neighbors and nodes that can pass the 1-common neighbor condition.

Table 4.4 *Simulation configurations. (Continued)*

Compared Algorithm	• ***DTA*****:** The algorithm that we proposed. DTA is a density and topology based with 2-common neighbor condition. Members of CDS consist of nodes with the highest number of 1-hop neighbors and nodes that can pass the 2-common neighbor condition.
Metrics	• ***Covered node****:* is measured as a percentage of the number of nodes that are covered by members of CDS to total nodes in the group. • ***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.

Coverage Results*:* Covered node results in highway scenarios are shown in Fig. 4.8 and urban scenario results are shown in Fig. 4.9. 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. So the number of members in CDS from DEN is not enough to cover all nodes in the groups as mentioned in Chapter 3. 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 then 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 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 1-common neighbor and DTA. Both of them provide the highest coverage results in the simulation excluding the results from the exact algorithm. The exact algorithm always provides 100% coverage while DTA can construct CDS members with almost 100% coverage results.

Ratio of CDS Member: The results are shown in Fig. 4.10 and Fig. 4.11 for highway scenarios and urban scenarios, respectively. 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. The exact algorithm is the lowest baseline results for all algorithms.

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 10% 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+1-common neighbor has the highest ratio results. This means that 1-common neighbor condition cannot efficiently prune nodes in vehicular environment.

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 20% 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 on scenarios. This can maximize the coverage results while minimizing a number of CDS members. However, the results still are about 2.5 times of the ideal results.

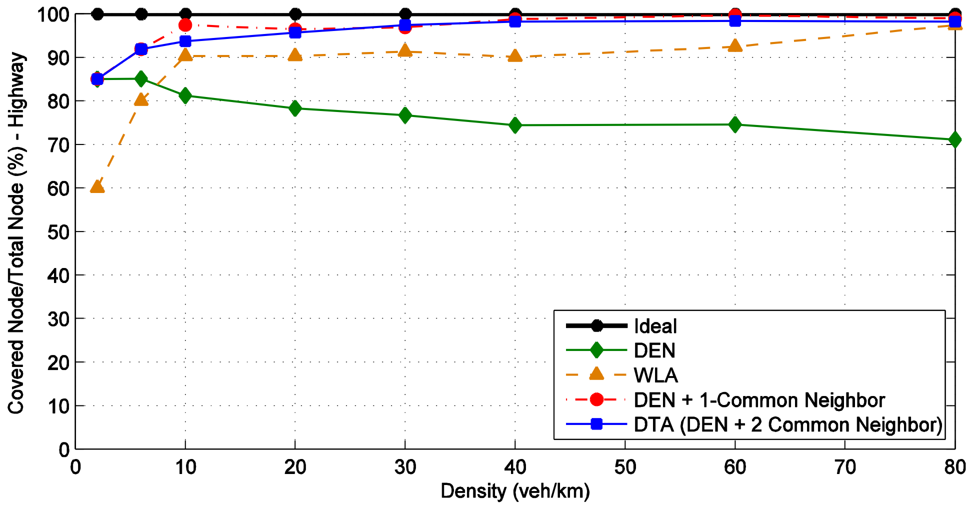


Figure 4.8 Covered node results in highway scenarios.

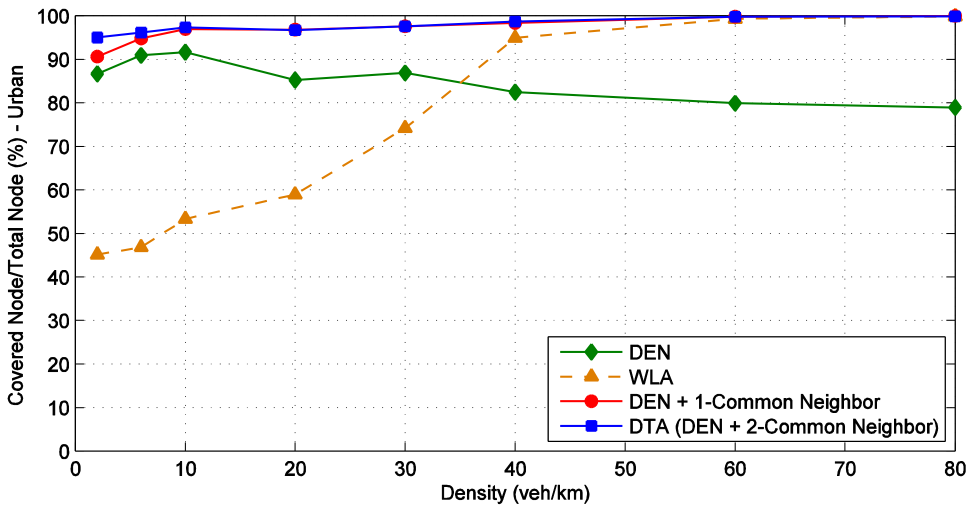


Figure 4.9 Covered node results in urban scenarios.

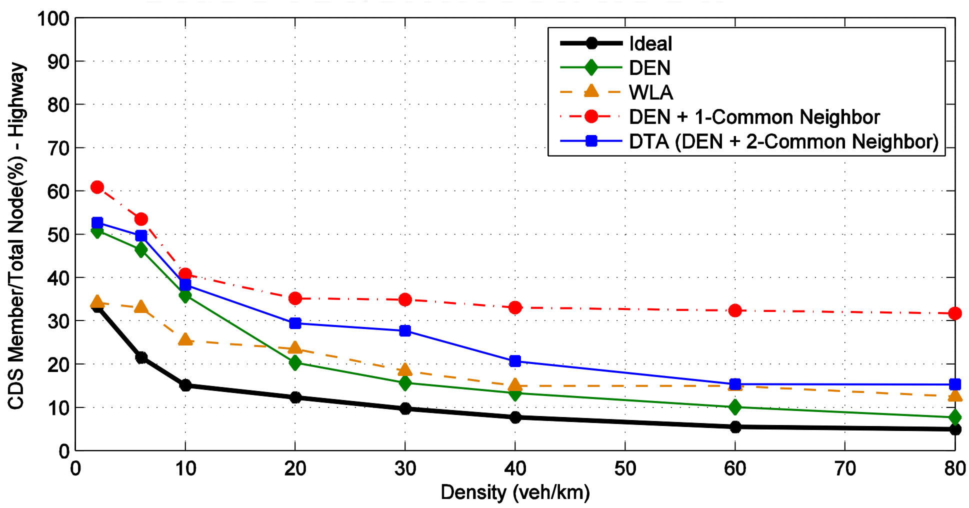


Figure 4.10 Ratio of CDS member results in highway scenarios.

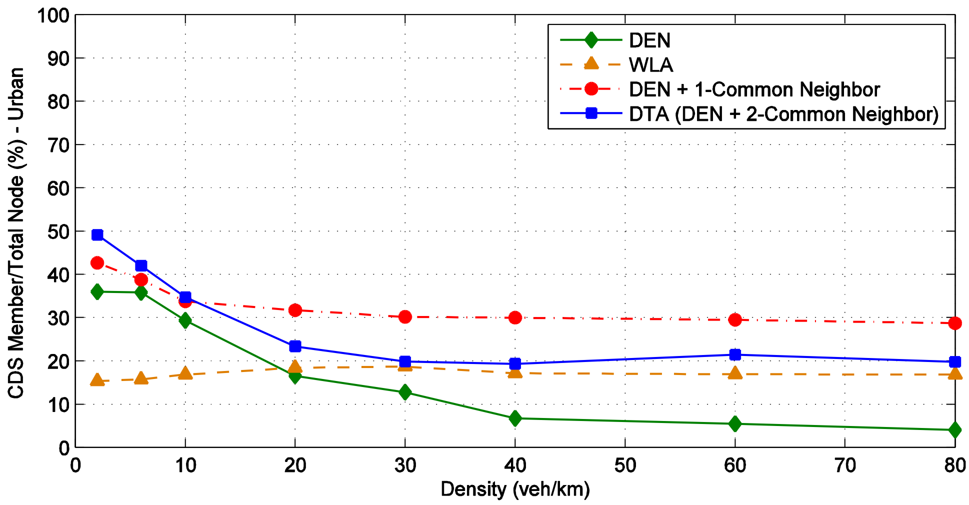


Figure 4.11 Ratio of CDS member results in urban scenarios.

4.2.2 Waiting Timeout Mechanism

As mentioned in Chapter 3, the efficient β value can significantly reduce collision occurrences in dense areas while increasing only a little delay. In order to select the β, we performed a simulation. The simulation setup is show in Table 4.5. The highway scenario is used in this simulation. We evaluated the performance of NoG using the directed function with varied β (1–5).

Delivery ratio: From the results in Fig. 4.12, the inversed function has a problem that cannot deliver packet to most of nodes in high density scenarios or even the lower ones due to the contention problem. On the other hand, the directed function can outperform the inversed function in every β value because this function can avoid collision occurrences.

Total retransmission overhead: The overhead results can be illustrated in Fig. 4.13. The overhead results of the inversed function are extremely high, according to number of collisions. This leads to lots of retransmission. The directed functions provide less retransmission overhead than the inversed function excepting β at 1.

3, 4 and 5 are the most efficient value of β**.** They have the highest delivery ratio and the lowest retransmission overhead. However, 3 is the best value because it introduces the lowest additional delay among these value. According to equation (3.1) in Section 3.3, the maximum waiting timeout depends on β value so the best β value should significantly reduce collision occurrences while increasing only a little delay. Therefore, 3 will be used as β value for NoG in the rest of our evaluation.

Table 4.5 Simulation configurations.

Simulator	Network Simulator 3 (NS-3.16)
Simulation time	200 s.
No. of simulation	20
Road scenario - Vehicle density (veh./km.)	Real map – 2, 10, 30, 60 and 80
Data Type	Message: 512 bytes with ten sources Lifetime: 50 s.
Propagation loss model	Range propagation loss model with 250-meter transmission range
Compared waiting timeout calculation	● *NoG using inversed function*: The famous function that used in most of research in VANET. ● *Nog using directed function with varied β (1-5)*: Our proposed function that solve a contention problem in high density areas. Different values of β are evaluated for parameter tuning.
Metrics	● *Delivery ratio*: is measured as a percentage of the number of nodes that received packets to total nodes in scenarios. ● *Total retransmission overhead*: is measured as bandwidth consumption from packet retransmission then normalized with delivery ratio. This can represent the overhead that protocol need to consume for increasing a number of received nodes.

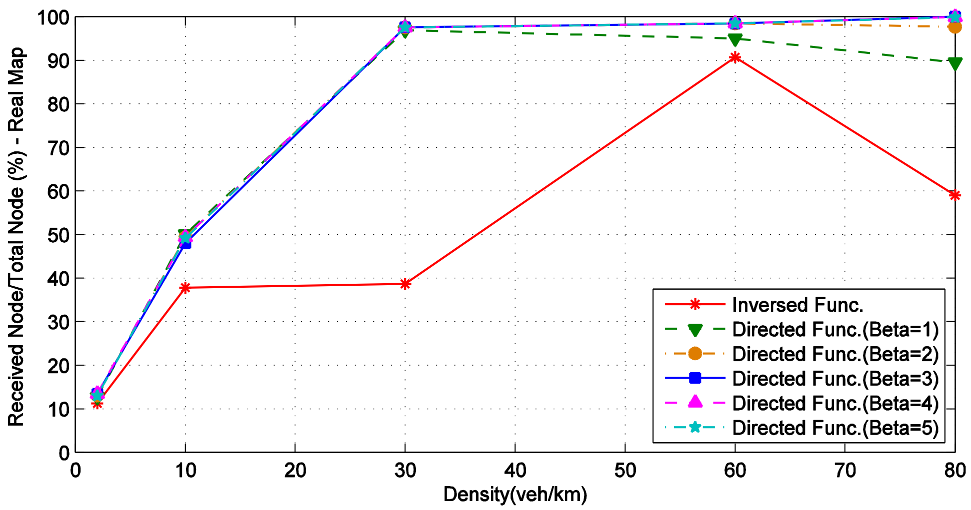


Figure 4.12 Delivery ratio results of different waiting timeout functions.

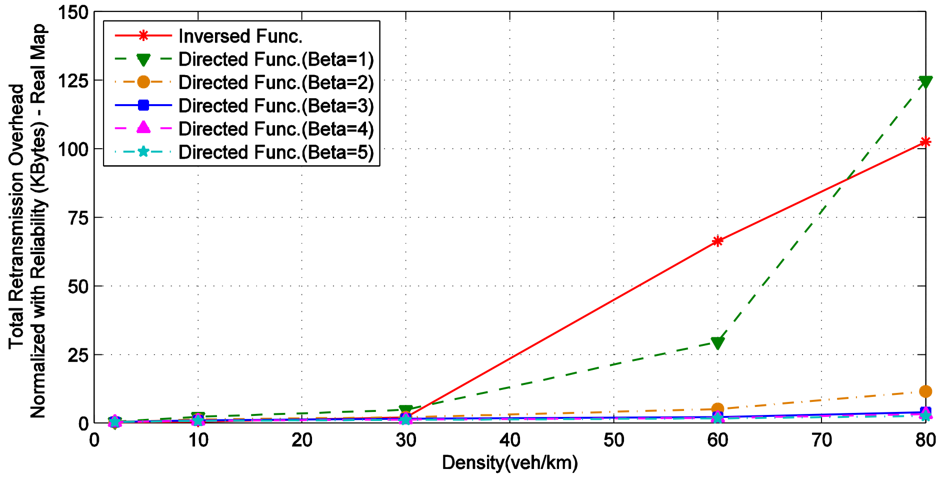


Figure 4.13 Total retransmission overhead results of different waiting timeout functions.

4.2.3 Beacon Mechanism

A. Bloom filter on beacon

In this subsection, we consider an effect on the forwarder node selection algorithm or DTA due to its false positive error rate. First, we analyze a number of neighbor nodes that can be found in real map scenarios. Then, we use this number to calculate Bloom filter bit array size and a number of hash functions that provide an acceptable false positive error. In order to know a number of neighbor nodes in each density scenarios, we use topology simulation to analyze real map scenarios. Figure 4.14 shows the cumulative graph of number of neighbor nodes in each density. As a result, the maximum number of neighbors is 252, but 90% of nodes has less than 142 neighbors. Therefore, we use n equal to 142 elements to find a number of hash functions and a number of bits in an array according to equation (2.2) in Section 2.1.4. Table 4.6 shows the false positive error rates from 5% to 25% and Bloom filter configurations.

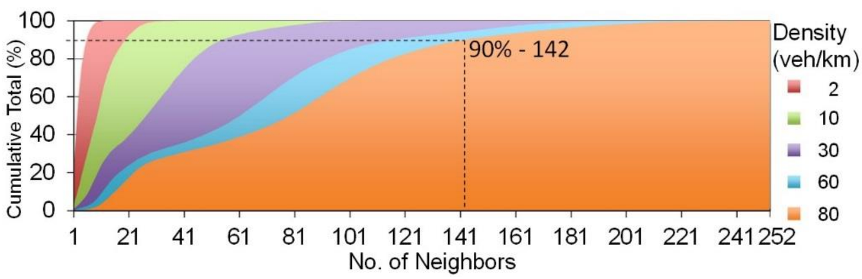


Figure 4.14 Cumulative graph of the number of neighbors in each density scenario.

Table 4.6 Bloom filter configuration

False positive error rate	Number of hash functions	Size of bit array
5%	4	886
10%	3	681
15%	3	561
20%	2	476
25%	2	410

False Positive Errors vs. Number of Elements: All configurations are tested with 10-250 elements. There are two types of elements: a node identifier and a packet identifier. The node identifier is an IPv4 address and the packet identifier is an IPv4 address of source node with a 32-bit unique identifier. After the elements are added, the Bloom filter is randomly queried with elements that are and are not in the set. The false positive error results are shown in Fig.4.15 and the results are close to the theoretical results that can be calculated by (2.2) in Section 2.1.4.

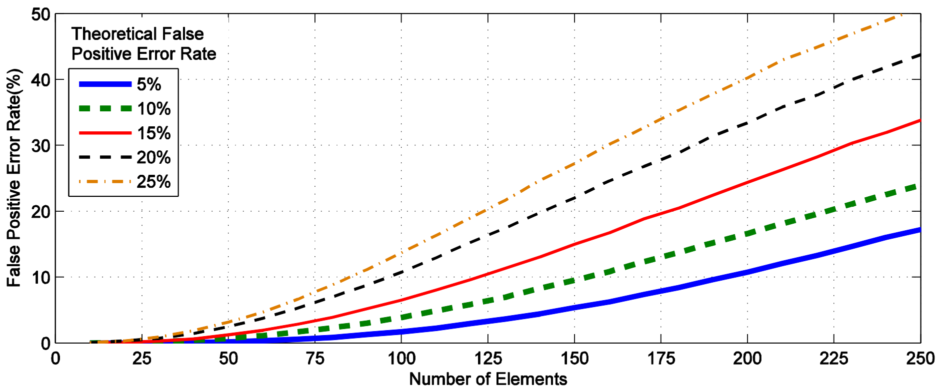


Figure 4.15 Practical false position error rate and theoretical false positive error rate.

False Positive Errors vs. Set Operations: As previously mentioned in Section 3.4.1B, we use a Bloom filter to reduce the computation complexity of 2-common neighbor condition in DTA. This simulation uses the error rates with different configurations (10%, 15% and 20%). The other parameter configurations are shown in Table 4.7. As shown in Fig.4.16, the original algorithm (Original) with a Bloom filter beacon has less than 7% false positive for all density scenarios and all configurations. In order to reduce the complexity of 2-common neighbor condition computation, Bloom filter information is lost by an intersection operation. The results show that the reduced algorithm is sensitive to the rate of false positive error, which is related to the number of neighbors. Therefore, the reduced algorithm is a tradeoff between the size of Bloom filter and the loss of information. Then, we use a 15% false positive error rate configuration to evaluate NoG protocol in term of beacon overhead. Because the 15% false positive error rate configuration provides the maximum error rate 15% as theoretical computation and its size is significantly less that 10% false positive error rate configuration.

Beacon overhead: Beacon overhead results are illustrated in Fig. 4.17. A fixed size beacon structure decreases the beacon overhead upto 80% or an average reduction is 67% from all densities. A beacon with Bloom filter structure helps protocol to save unnecessary bandwidth consumption on limited resource environments like VANET. A disadvantage of this solution is the Bloom filter array causes additional overhead when the elements; a number of 1-hop neighbors or a number of received packets are less than 14 or 8, respectively so this can increase beacon overhead in sparse areas. However, the additional overhead in sparse areas does not decrease the performance.

Table 4.7 Simulation configurations.

Simulator	Network Simulator 3 (NS-3.16)
Simulation time	200 s.
No. of simulation	20
Road scenario - Vehicle density (veh./km.)	Real map – 2, 10, 30, 60 and 80
Data Type	Message: 512 bytes with ten sources Lifetime: 50 s.
Propagation loss model	Range propagation loss model with 250-meter transmission range
Compared algorithm	• ***Original DTA algorithm****:* The original DTA algorithm consists of density based algorithm and 2-common neighbor condition. • ***Reduces algorithm:*** The reduced DTA algorithm consists of the original density based algorithm and the reduced 2-common neighbor condition computation.
Metrics	• ***False positive error rate:*** is measured as percentage of number of incorrect DTA computations to total times of computation • ***Beacon overhead****:* is measured as total bandwidth consumption for beacon transmission from all nodes in scenarios.

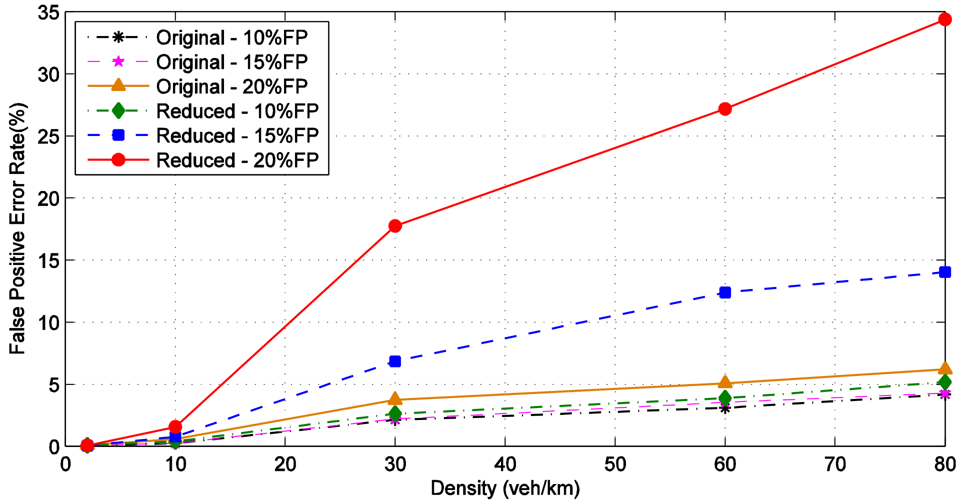


Figure 4.16 False position error rates in real map scenarios.

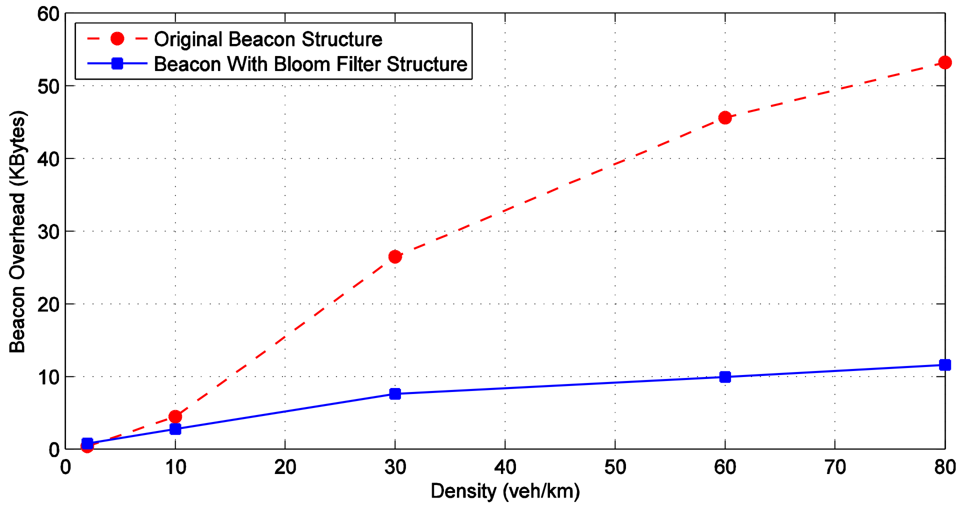


Figure 4.17 Beacon overhead results of different beacon structures.

B. Adaptive 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 for each density scenario, we performed lots of simulation. The beacon interval is varied from 0.5 to 9 seconds in different density scenarios (2–80 veh/km). We use highway scenarios in this simulation. The other parameters are set as shown in Table 4.8.

Table 4.8 *Simulation configurations.*

Simulator	Network Simulator 2 (NS-2.34)
Simulation time	50 s.
No. of simulation	20
Road scenario - Vehicle density (veh./km.)	Highway – 2, 6, 10, 20, 30, 40, 60 and 80
Data Type	Message: 512 bytes with one source Lifetime: 10 s.
Propagation loss model	Range propagation loss model with 250-meter transmission range
Compared Beacon Interval (s./beacon)	0.5, 1.0, 1.5, 2.0, 3.0, 5.0, 7.0 and 9.0
Metrics	• *Retransmission overhead:* is measured as total bandwidth consumption for packet retransmission from all nodes in scenarios. • *Beacon overhead:* is measured as total bandwidth consumption for beacon transmission from all nodes in scenarios. • *Data dissemination speed:* is measured as (4.1), 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. $$y(t) = \frac{\sum_{i=0}^{t} r_i}{n} \times 100 \quad (4.1)$$

In order to select a beacon interval for each density, we observe the data dissemination speed as shown in Fig. 4.20-4.27. The beacon intervals are selected from the first group of beacon intervals that has of the fastest data dissemination. Then, we consider the beacon overhead in Fig. 4.18 and retransmission overhead in Fig. 4.19 to looking for an appropriate interval. The beacon intervals that are selected for each density are shown in Table 4.9.

From simulation results, we observed that 1.5 seconds are the beacon interval that provides the fastest data dissemination 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 equation (3.3) in Section 4.4.2, c is equal to 0.2, $minInv$ is 1.5, and $maxInv$ is 7.

Table 4.9 Suitable beacon interval for each vehicle density.

Vehicle density (veh./km.)	Beacon interval (s.)
2 – 10	3
20 – 40	4
60 – 80	7

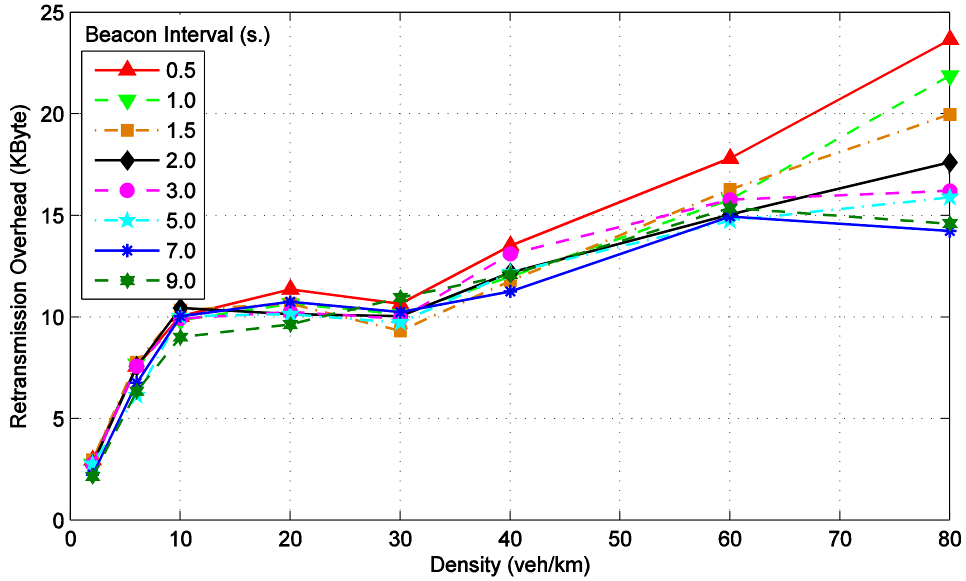


Figure 4.18 Retransmission overhead results of different beacon interval.

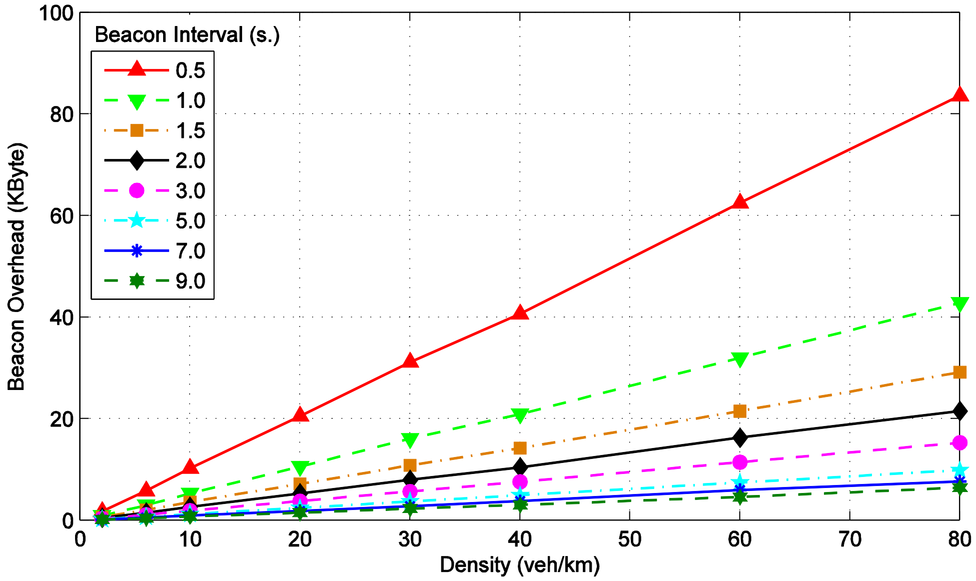


Figure 4.19 Beacon overhead results of different beacon interval.

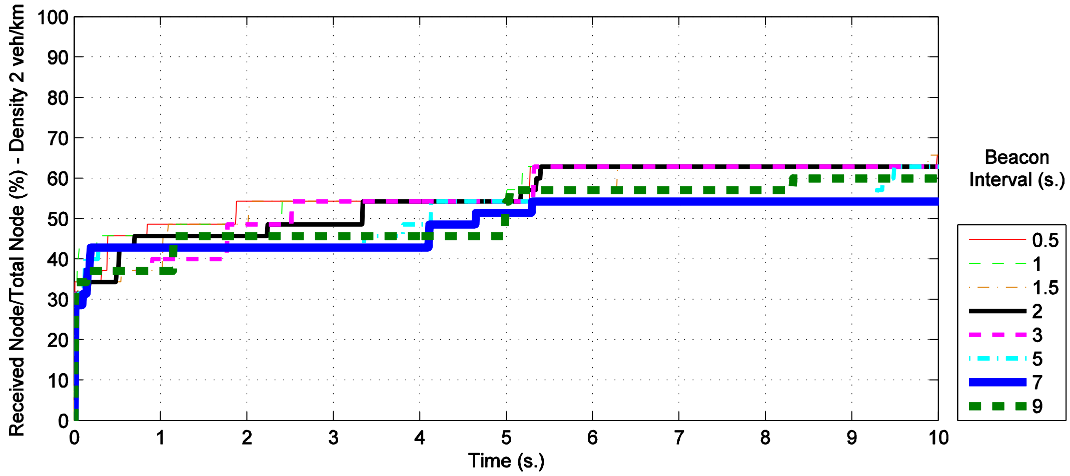


Figure 4.20 Data dissemination speed results at density 2 veh/km.

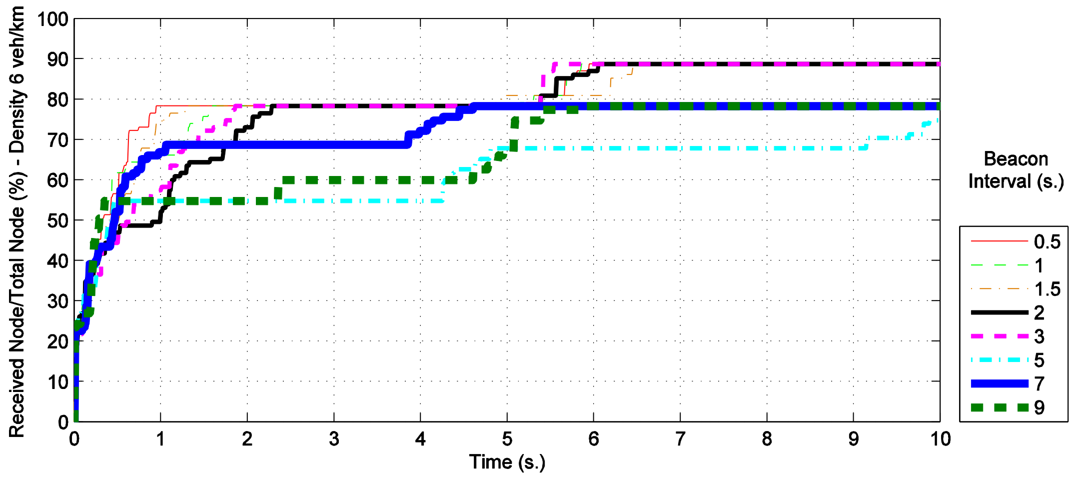


Figure 4.21 Data dissemination speed results at density 6 veh/km.

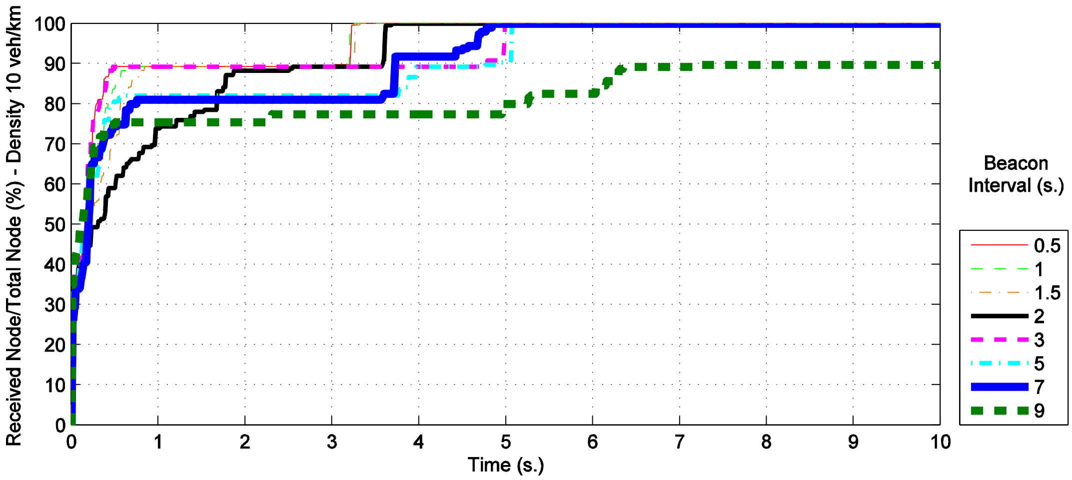


Figure 4.22 Data dissemination speed results at density 10 veh/km.

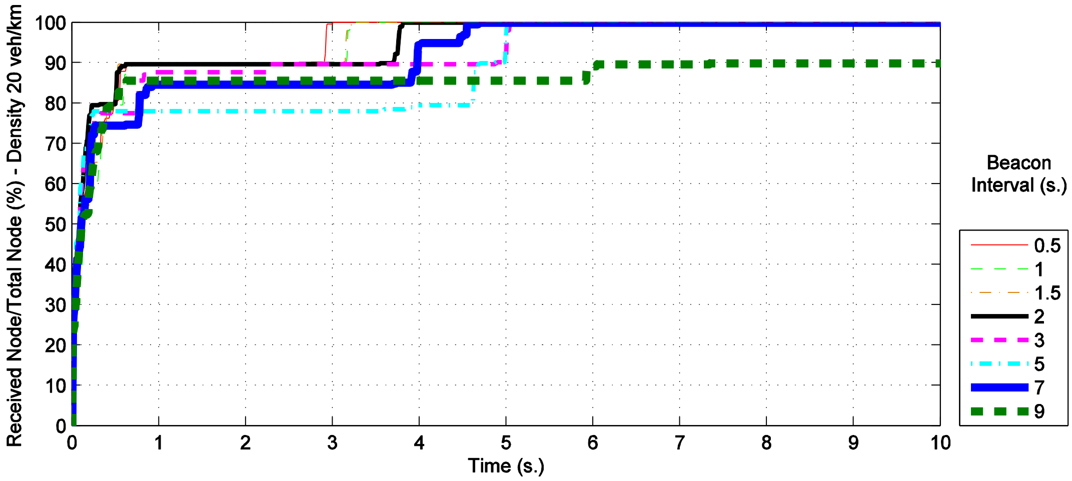


Figure 4.23 Data dissemination speed results at density 20 veh/km.

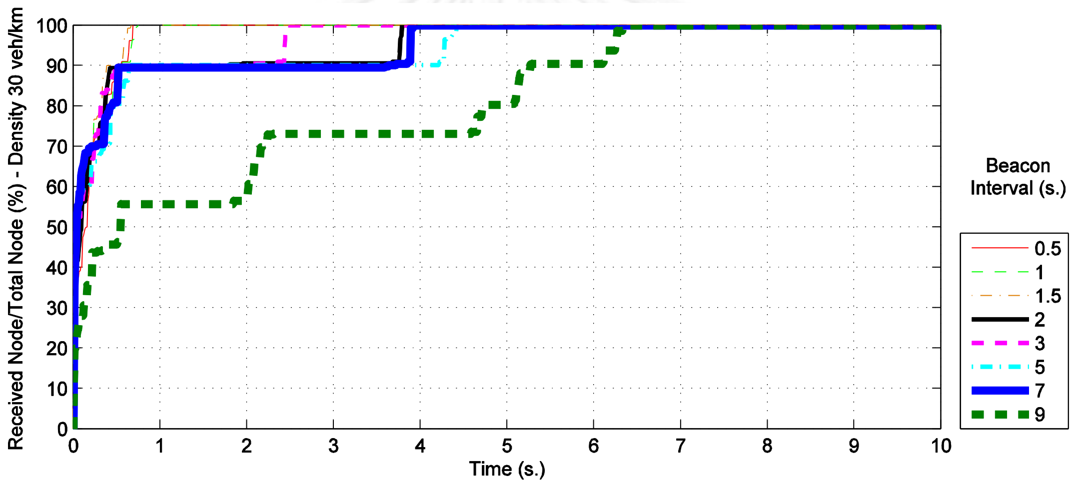


Figure 4.24 Data dissemination speed results at density 30 veh/km.

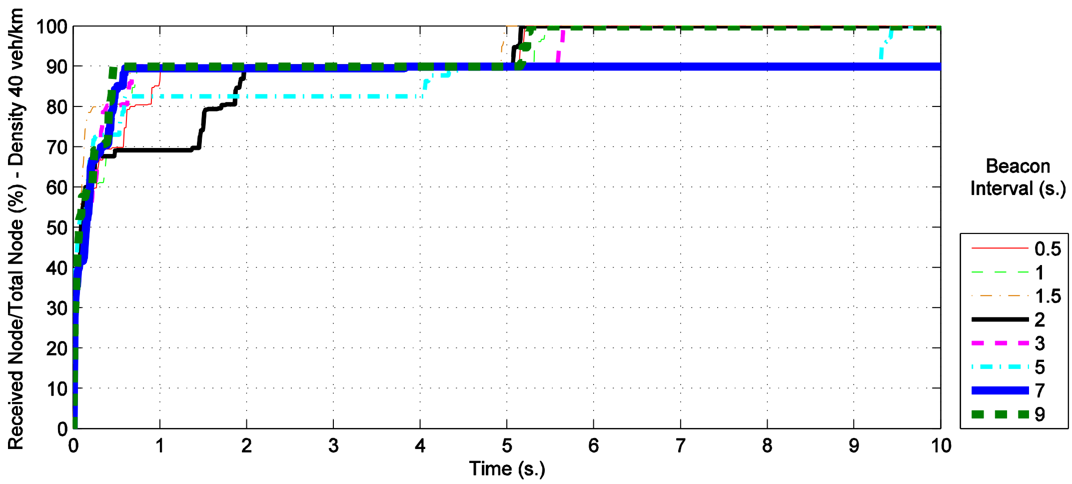


Figure 4.25 Data dissemination speed results at density 40 veh/km.

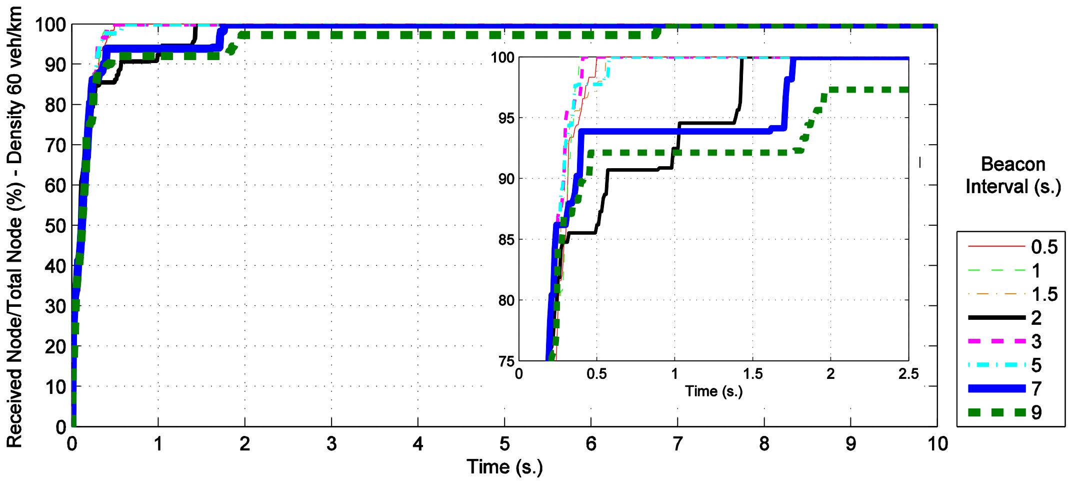


Figure 4.26 Data dissemination speed results at density 60 veh/km.

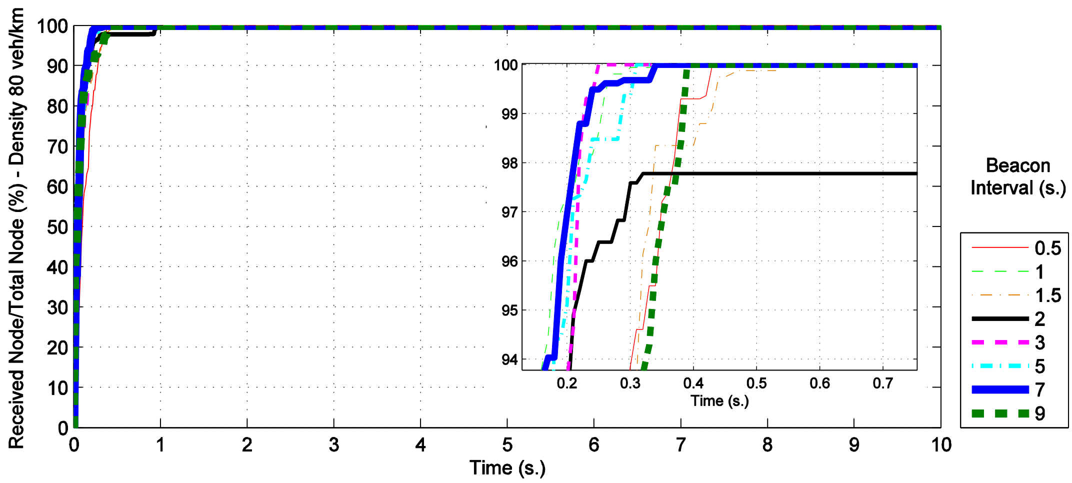


Figure 4.27 Data dissemination speed results at density 80 veh/km.

We use the adaptive beacon interval configuration above to evaluate on both highway scenarios and urban scenarios. The other parameter configurations are in Table 4.10.

Beacon overhead: Beacon overhead results for highway scenarios and urban scenarios are shown in Fig. 4.28. For highway scenarios, LIA reduces beacon overhead upto 81% or an average reduction is 68% and for urban scenarios LIA reduces beacon overhead upto 81% or an average reduction is 72%. Therefore, LIA significantly eliminates unnecessary beacon overhead in every scenario without decreasing protocol performance.

Table 4.10 Simulation configurations.

Simulator	Network Simulator 2 (NS-2.34)
Simulation time	50 s.
No. of simulation	20
Road scenario - Vehicle density (veh./km.)	Highway – 2, 6, 10, 20, 30, 40, 60 and 80 Urban – 2, 6, 10, 20, 30, 40, 60 and 80
Data Type	Message: 512 bytes with one source Lifetime: 10 s.
Propagation loss model	Range propagation loss model with 250-meter transmission range
Compared beacon interval algorithm	● *Nog with constant interval 1 s.* ● *Nog with LIA (1.5-7 s.)*
Metrics	***Beacon overhead:*** is measured as total bandwidth consumption for beacon transmission from all nodes in scenarios.

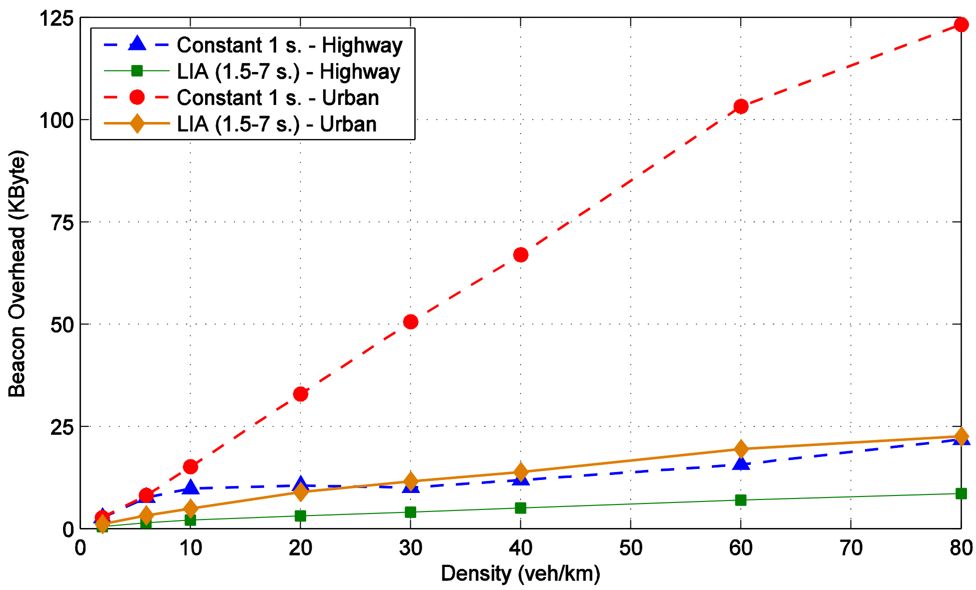


Figure 4.28 Beacon overhead results of highway scenarios and urban scenarios.

C. Missing Message Mechanism

Our new beaconing mechanism implemented in NoG limits a number of beacons that the node will send back to its neighbors when it detects the missing message. This can reduce congestion in the networks. However, limiting number of beacons can reduce speed of data dissemination and decrease overall reliability. To find the proper number of beacons, we have simulated our protocol in different number of beacons to send back and see its performance metrics. This simulation is setup as shown in Table 4.11.

Table 4.11 *Simulation configurations.*

Simulator	Network Simulator 3 (NS-3.16)
Simulation time	200 s.
No. of simulation	20
Road scenario - Vehicle density (veh./km.)	Real map – 2, 10, 30, 60 and 80
Data Type	Message: 512 bytes with ten sources Lifetime: 50 s.
Propagation loss model	Range propagation loss model with 250-meter transmission range
Compared number of limit beacons	Non-beacon, 1, 5, 10, 20, Unlimited beacon
Metrics	• *Beacon overhead:* is measured as total bandwidth consumption for beacon transmission from all nodes in scenarios. • *Data dissemination speed:* is measured as (4.1), 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. $$\mathrm{y(t)} = \frac{\sum_{i=0}^{t} r_i}{n} \times 100 \qquad (4.1)$$

The results of the simulation are shown in Fig. 4.29-4.34. As can be seen, unlimited beacon causes extremely high overhead that affects to speed of data dissemination. These beacon transmissions also increase number of collision occurrences. The non-beacon configuration has the least overhead but it cannot provide fast data dissemination in low density areas. On the other hand, the 1-beacon configuration offers the second least overhead while providing high data dissemination speed comparable to other numbers of beacons in all density scenarios. In consequence, the 1-beacon configuration is the most efficient setting, and hence it is used in NoG.

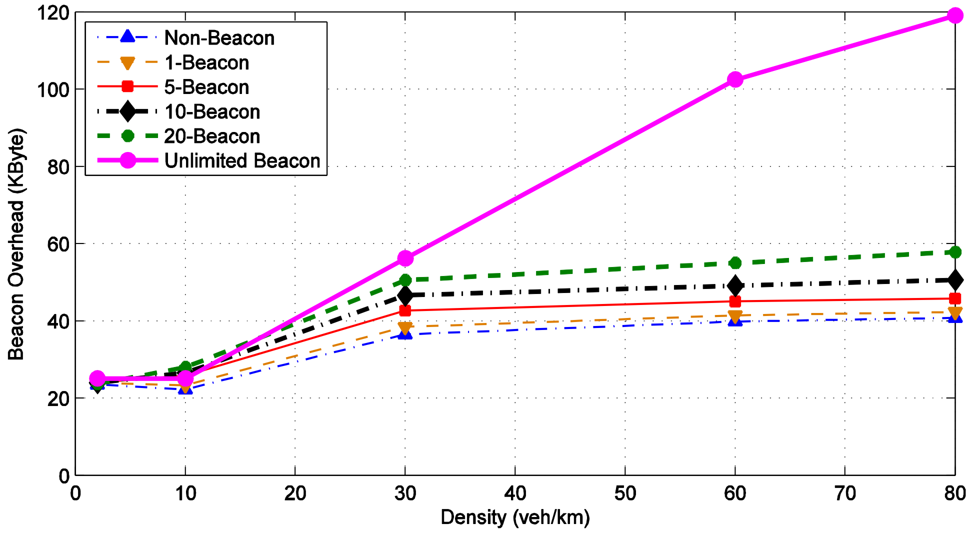


Figure 4.29 Beacon overhead results of a different number of beacons.

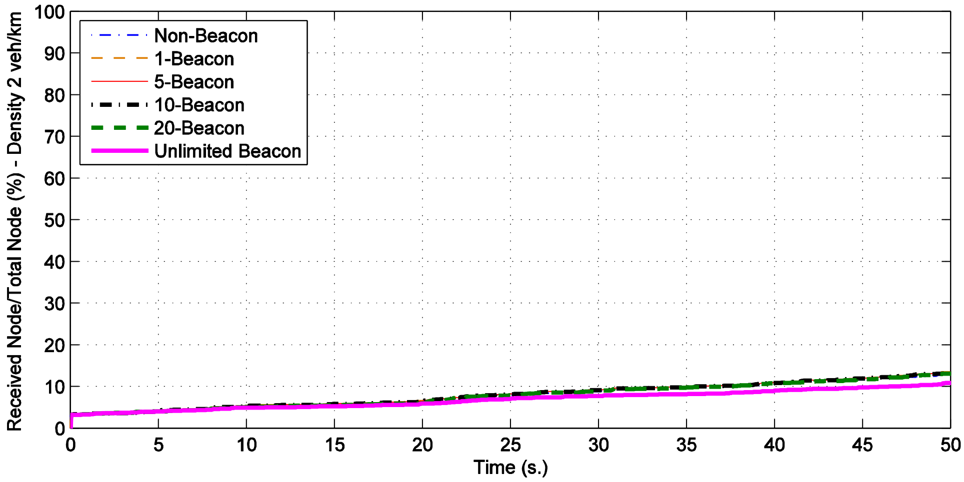


Figure 4.30 Data dissemination speed results at density 2 veh/km.

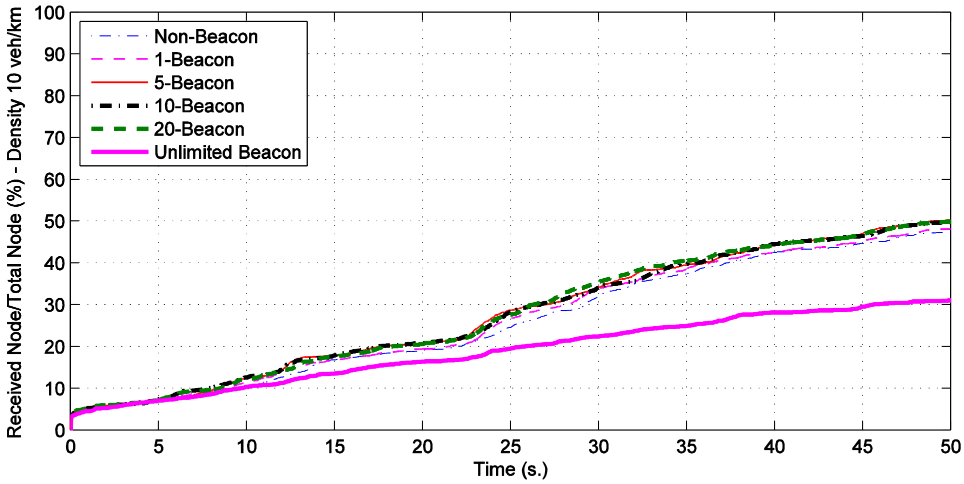


Figure 4.31 Data dissemination speed results at density 10 veh/km.

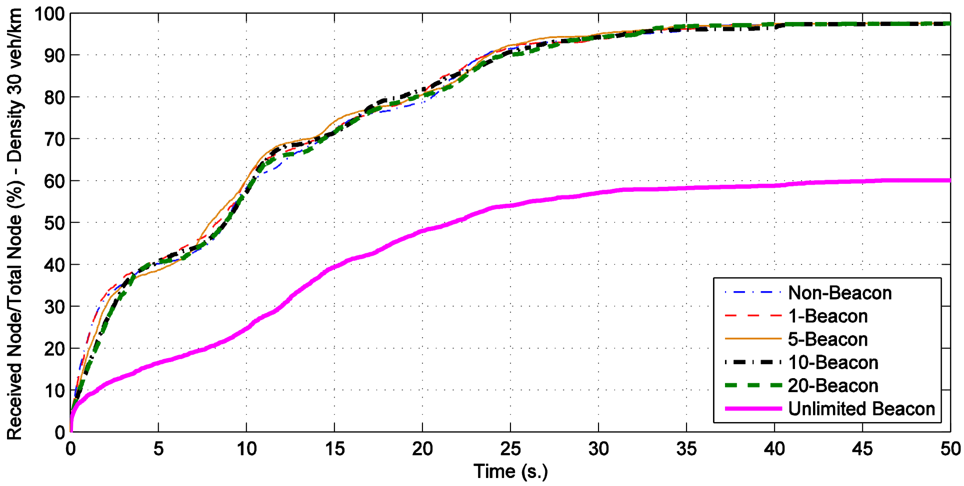


Figure 4.32 Data dissemination speed results at density 30 veh/km.

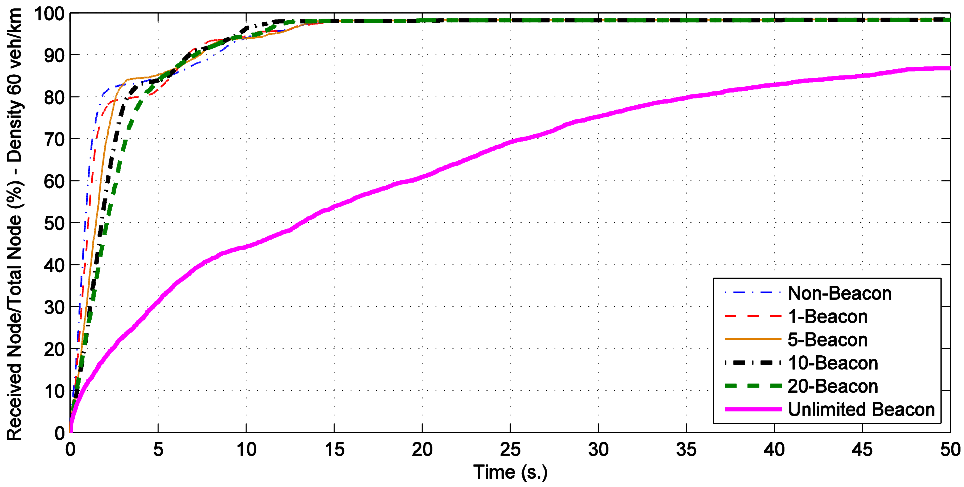


Figure 4.33 Data dissemination speed results at density 60 veh/km.

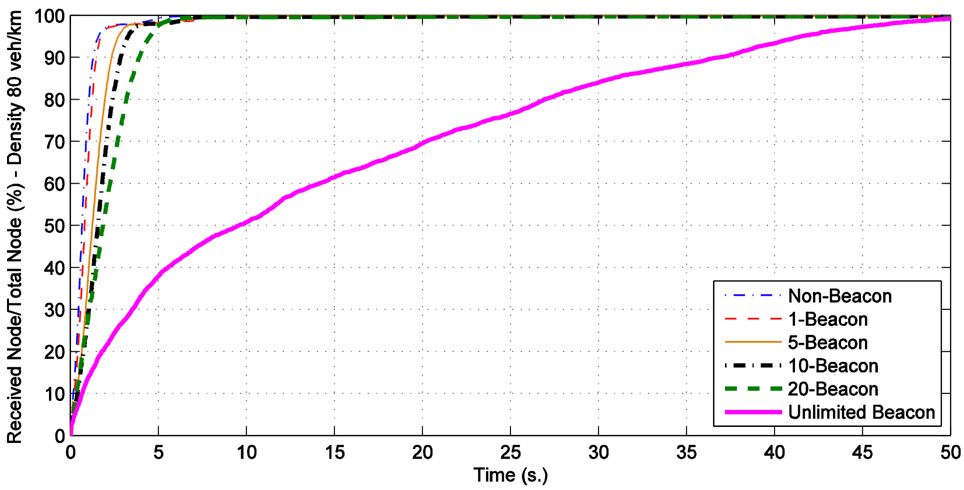


Figure 4.34 Data dissemination speed results at density 80 veh/km.

4.3 Protocol Performance Evaluation

In this subsection, we evaluate our purpose protocol and other previous work in term of performance and efficiency. First, we compare each protocol in simple scenarios, which are highway traces and urban traces. Then, we simulate these protocols in real map scenarios to show their practical usability. Finally, our proposed protocol is evaluated in extremely high data transfer for its scalability test.

4.3.1 Performance Evaluation in Simple Scenarios

All simulation settings are shown in Table 4.12 including protocol parameters and metrics that we are interested.

Table 4.12 Simulation configurations.

<table>
<tr><td>Simulator</td><td>Network Simulator 3 (NS-3.16)</td></tr>
<tr><td>Simulation time</td><td>50 s.</td></tr>
<tr><td>No. of simulation</td><td>20</td></tr>
<tr><td>Road scenario - Vehicle density (veh./km.)</td><td>Highway – 2, 6, 10, 20, 30, 40, 60 and 80
Urban – 2, 6, 10, 20, 30, 40, 60 and 80</td></tr>
<tr><td>Data Type</td><td>Message: 512 bytes with ten sources
Lifetime: 10 s.</td></tr>
<tr><td>Propagation loss model</td><td>Nakagami propagation loss model (m=1)
with transmission success rate 80% at 250 meters</td></tr>
<tr><td>Protocol setup</td><td>Simple Flooding: Simple flooding represents the simplest protocol with the fastest speed of data dissemination.

EAEP: EAEP represents the epidemic protocol, which uses probabilistic approach and counter-based approach. It operates without beaconing. All parameters are setup as same as in [5].
<table><tr><td>Retransmission interval</td><td>0-2 s.</td></tr></table>
DECA: DECA represent the protocol, which uses only density information and avoids using wait timeout mechanism. It is proposed in our previous work.
<table><tr><td>Beacon interval</td><td>Adaptive Interval (1.5 - 7 s.)
LIA: c=0.2, minInv=1.5, maxInv=7</td></tr><tr><td>Beacon size</td><td>5 bytes + 8 bytes for each message acknowledgement</td></tr></table>
</td></tr>
</table>

Table 4.2 *Simulation configurations. (Continued)*

<table>
<tr><td>Protocol setup</td><td>

APBSM: APBSM represents the protocol, which uses store and forward technique and uses waiting timeout to alleviate the broadcast storm problem. It is chosen as it is shown to provide the highest reliability among the existing protocols we found in the literature. All other parameters are setup as same as in [6].

<table>
<tr><td>Beacon interval</td><td>Constant interval (0.5 s.)</td></tr>
<tr><td>Beacon size</td><td>21 bytes + 8 bytes for each message acknowledgement</td></tr>
</table>

NoG+DTA: NoG is our proposed protocol in this dissertation that operates with our DTA algorithm that also proposed in this dissertation.

<table>
<tr><td>Beacon interval</td><td>Adaptive Interval (1.5 - 7 s.)
LIA: c=0.2, minInv=1.5, maxInv=7</td></tr>
<tr><td>Beacon size</td><td>5 bytes + 70 bytes for Bloom filter bit array</td></tr>
<tr><td>Number of limited responded beacon</td><td>1</td></tr>
<tr><td>Waiting timeout calculation</td><td>β=3</td></tr>
</table>

</td></tr>
<tr><td>Metrics</td><td>

- ***Delivery ratio:*** is measured as a percentage of the number of nodes that received packets to total nodes in scenarios.
- ***Retransmission overhead:*** is measured as total bandwidth consumption for packet retransmission from all nodes in scenarios.
- ***Beacon overhead:*** is measured as total bandwidth consumption for beacon transmission from all nodes in scenarios.
- ***Data dissemination speed:*** is measured as (4.1), 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.

$$y(t) = \frac{\sum_{i=0}^{t} r_i}{n} \times 100 \quad (4.1)$$

- ***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.

</td></tr>
</table>

Delivery ratio: As shown in Fig. 4.35, in low density highway scenarios (less than 10 veh/km), Simple flooding and EAEP provide almost the same delivery ratio; that is, about 40% at 2 veh/km and 71% at 6 veh/km. DECA, APBSM and NoG+Dta achieve higher delivery ratios (56% at 2 veh/km and 78% at 6 veh/km). Simple flooding and EAEP cannot perform well because vehicles that carry the message rebroadcast it before meeting other vehicles. This leads the retransmission to become useless. DECA, APBSM and NoG+DTA have mechanisms to hold the message and rebroadcast it when new neighbors are found. Acknowledgement in beacon message is the key to help the protocols to discover new neighbors, which have not received the message.

In medium density scenarios (between 10 to 60 veh/km), simple flooding and EAEP achieve 80~100%, whereas DECA, APBSM can reach 100% for all scenarios. In medium density scenarios, there are still partitions between groups of vehicles. Simple flooding cannot handle the partitions as usual. But we can see that EAEP can perform better that simple flooding. This is because the waiting time inserted before retransmission is long enough to let nodes carry the message to other groups. However, EAEP still cannot reach 100% in some scenarios where the waiting timeout is not long enough.

In high density scenarios (more than 60 veh/km), all protocols provide 100% delivery ratio because all nodes are connected.

When compare the delivery ratio results of urban scenarios (Fig. 4.36) to that of highway scenarios (Fig. 4.37), in overall the delivery ratio of urban scenarios is lower. This is because vehicle routes in urban scenario are more complex than those in highway scenario in that intersections exist and traffic jam causes more partitions. However, we can see that NoG+DTA can outperform the other protocols. In more complex scenarios, NoG+DTA uses the efficient nodes for its rebroadcasting and the efficient mechanism for missing message retransmission helps NoG to provide the highest reliability.

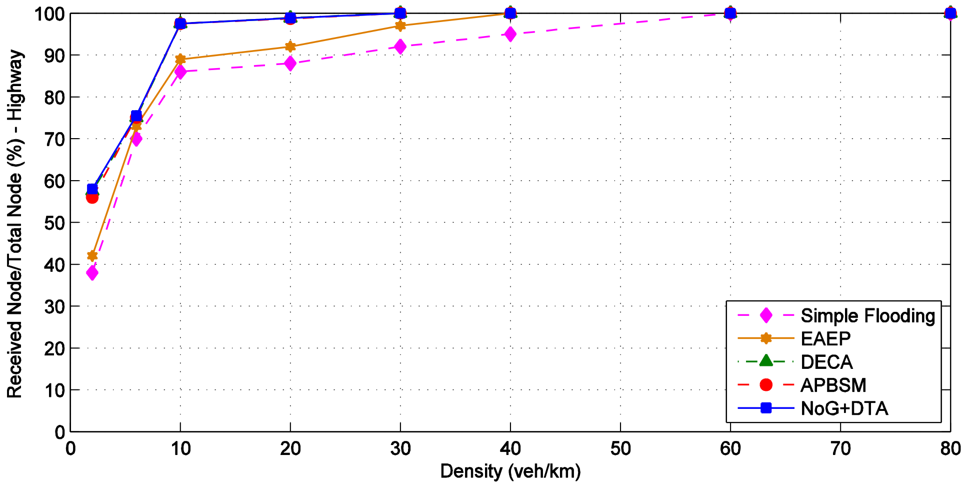


Figure 4.35 Delivery ratio results in highway scenarios.

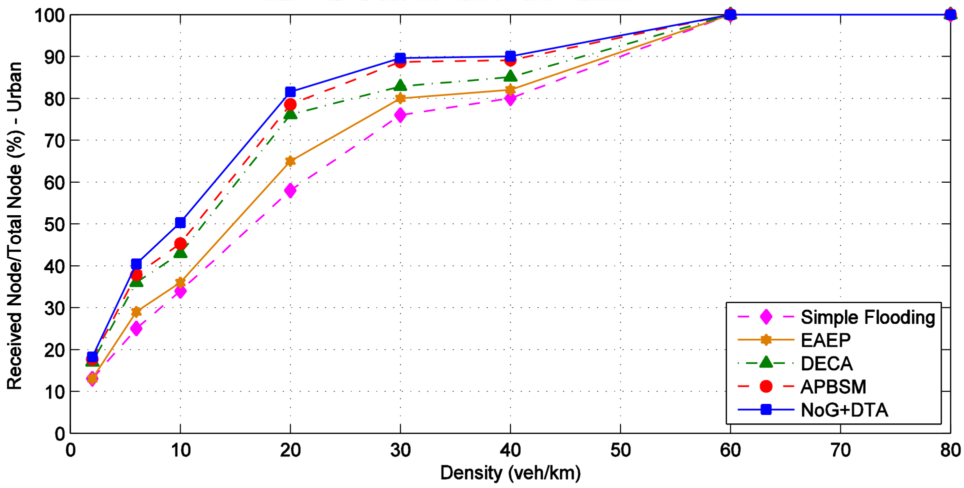


Figure 4.36 Delivery ratio results in urban scenarios.

Retransmission overhead: As shown in Fig. 4.37 for highway scenarios, NoG+DTA and DECA achieve much lower retransmission overhead than others for all cases. This is because NoG+DTA uses a directed function to calculate its waiting timeout to avoid collision and DECA also has the rebroadcast mechanism that lets the precursor to select the next forwarder node so both of protocols have collision occurrences and redundant retransmissions less than others. Simple flooding generates a lot of transmissions, as it is blind flooding. EAEP uses its probability function that does not efficient enough to operate in long time disconnected scenarios so most of nodes participate in rebroadcast mechanism. APBSM also generates high number of transmissions closely to simple flooding and EAEP. A reason is from APBSM's waiting timeout function. APBSM uses the inversed function to calculate a waiting timeout depending on number of neighbors but vehicles running in the same road section mostly have the same number of neighbors. As a result, those vehicles set the same waiting timeout for next retransmission and broadcast at the same time. This causes more than one retransmission in a particular area.

The retransmission overhead results in urban scenarios (Fig.4.38) are almost the same trend as results in highway scenarios. Interesting results are in low density scenarios which DECA, APBSM and NoG+DTA have higher overhead than simple flooding and EAEP due to their missing message recovery. However these protocols have higher delivery ratio than simple flooding and EAEP. Another interesting results are in high density scenarios, NoG+DTA has lower retransmission overhead than DECA because it has the more efficient forward node in more complex scenarios as it was designed. Therefore, only density information is not enough for selecting forwarder nodes in high complex scenarios.

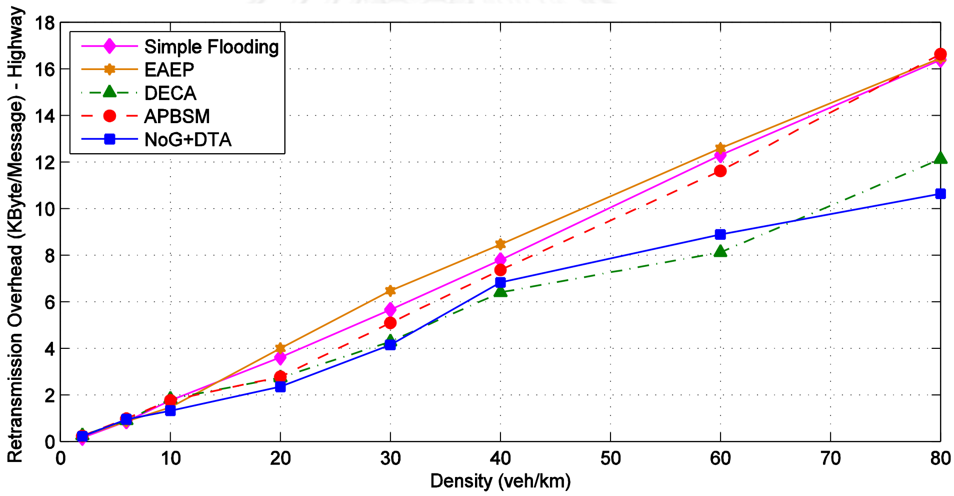


Figure 4.37 Retransmission overhead results in highway scenarios.

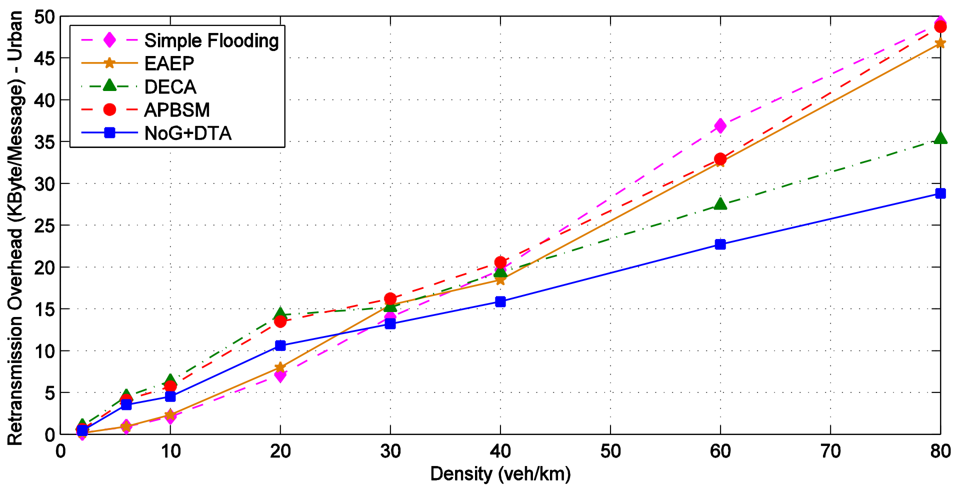


Figure 4.38 Retransmission overhead results in urban scenarios.

Beacon overhead: Figure 4.39 and figure 4.40 show beacon overhead in highway scenarios and urban scenarios, respectively. DECA consumes the least beacon bandwidth for both scenarios because it needs only density information and it uses an adaptive beacon interval to reduce unnecessary overhead. On the other hand, APBSM use a constant beacon interval for its retransmission mechanism. APBSM also require GPS information to construct CDA members so its beacon extremely high. Thus, it has the highest beacon overhead for all scenarios.

NoG+DTA has the biggest size of initial beacon because it requires a list of 1-hop neighbors for CDS member computation but we use Bloom filter bit array instead of list structures. This helps NoG+DTA to have a fixed size beacon structure. As a result, NoG+DTA can reduce its overhead in complex scenarios. We can notice that NoG+DTA increases less beacon overhead than others in higher density and more complex scenarios. A disadvantage is when a number of neighbors or a number of received packets are not large enough, a beacon with bloom filter is larger than ordinary one. Moreover, we also apply the adaptive beacon interval to NoG+DTA to reduce traffic contention in high density scenarios.

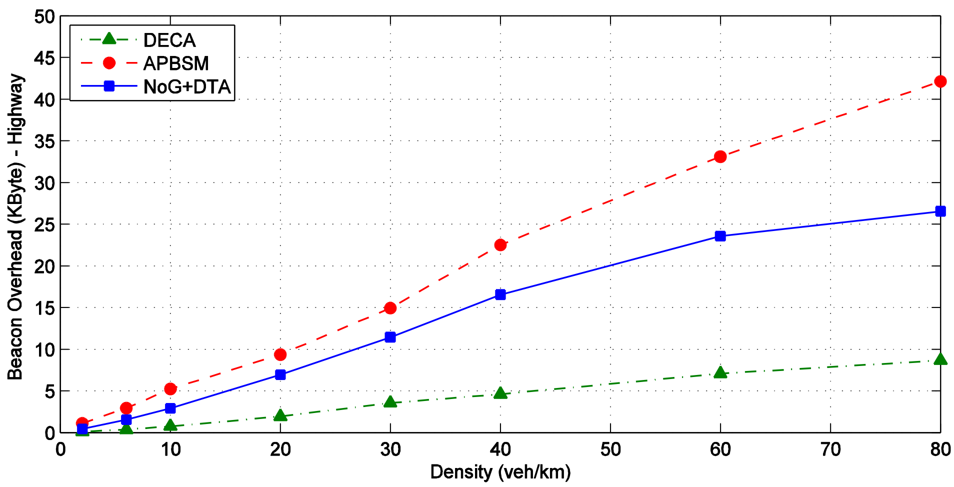


Figure 4.39 Beacon overhead results in highway scenarios.

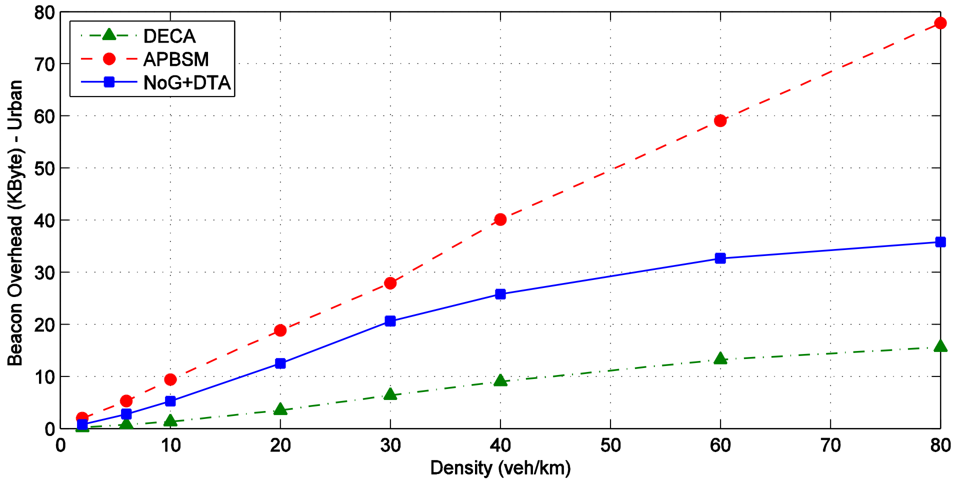


Figure 4.40 Retransmission overhead results in urban scenarios.

Data dissemination speed: Figure 4.41-4.48 are shown simulation results of 2, 6, 10, 20, 30, 40, 60 and 80 veh/km in highway scenarios. For all cases, simple flooding is the fastest protocol to disseminate data. However, simple flooding has the lowest delivery ratio results. Simple flooding stops increasing its received nodes when there is a network partition. EAEP is the slowest protocol because it just stores the message for a while and forwards with some probability value but at the end, EAEP can deliver data to more number of nodes than simple flooding. APBSM disseminates data very fast but still not the fastest due to its waiting time that causes lots of retransmissions happening like flooding behavior. So in higher density case, where there are many nodes in a particular area, AckPBSM will be slower due to its collision problem. DECA also has very fast data dissemination speed because nodes in DECA immediately broadcast an incoming message when they receive a new packet but it uses only density information that sometimes leads to select inefficient forwarder nodes that affect to additional delay for waiting timeout. In all scenarios, it can be seen that NoG+DTA outperforms EAEP, DECA and APBSM by providing the fastest speed of data dissemination excluding simple flooding that does not provide high delivery ratio but NoG+DTA has the highest number of received nodes at the end of simulation.

Figure 4.49-4.56 are shown simulation results of 2, 6, 10, 20, 30, 40, 60 and 80 veh/km in urban scenarios. All of results mostly are as same as the highway results but for urban scenarios, data dissemination speed of DECA is slower than APBSM because it has lower retransmission overhead, which decreases a probability that a new node will receive a new packet. However, APBSM is still slower than NOG+DTA due to its waiting timeout. A reason that we have found is the retransmissions of APBSM are from CDS members but these CDS members have the same waiting timeout. Accordingly, lots of collisions decrease APBSM performance.

Data dissemination speed results in highway scenarios.

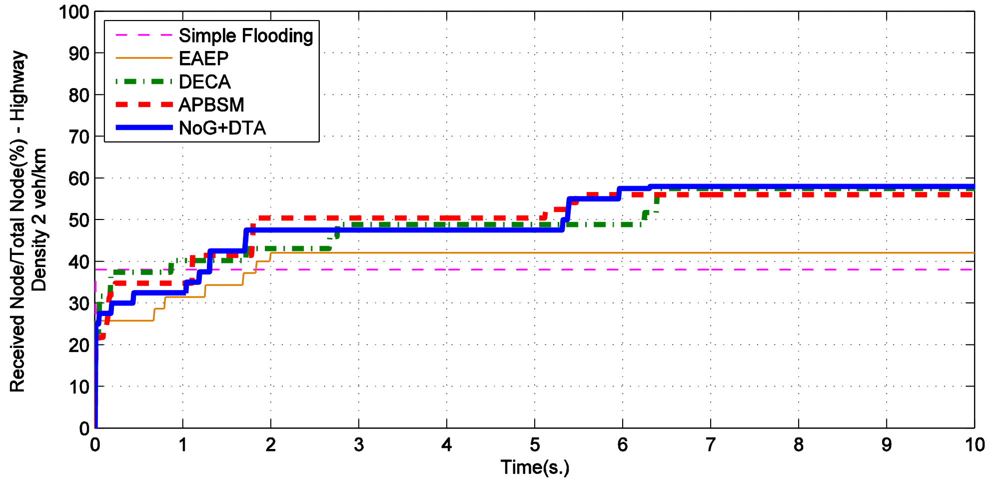


Figure 4.41 Data dissemination speed results at density 2 veh/km.

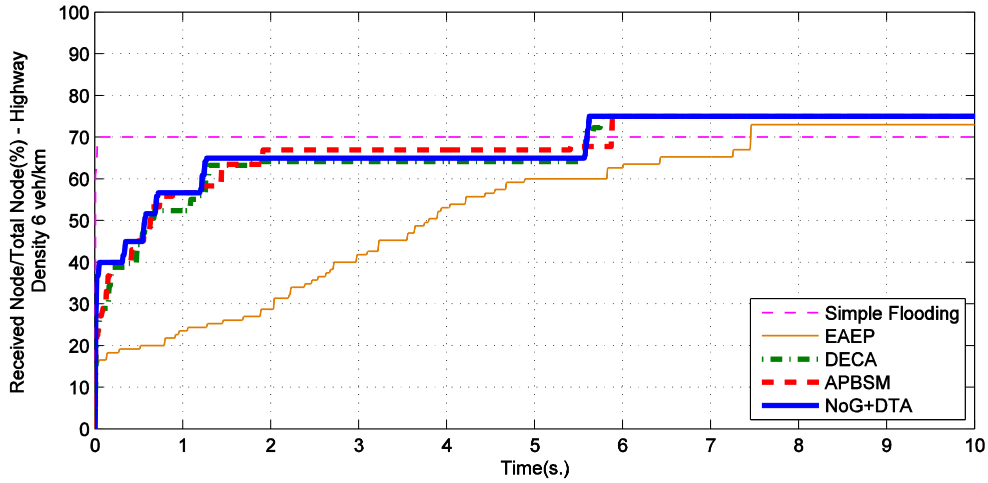


Figure 4.42 Data dissemination speed results at density 6 veh/km.

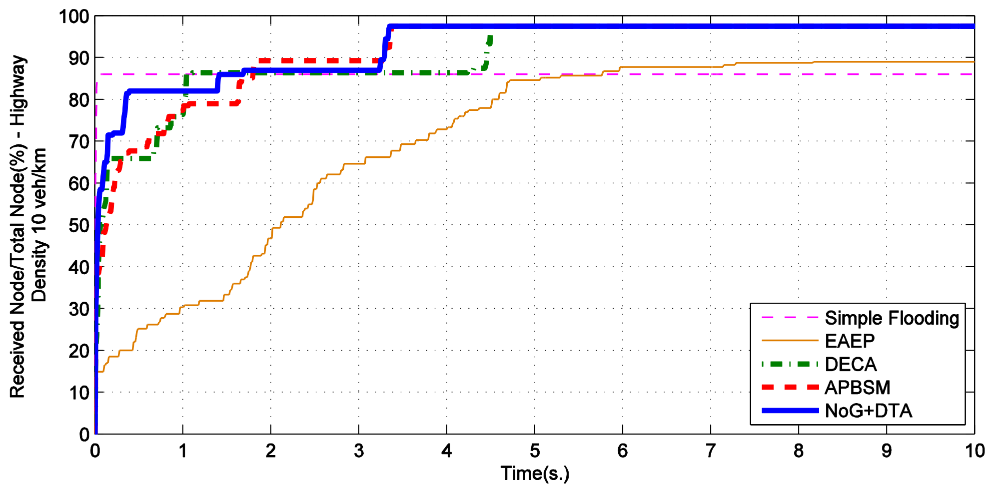


Figure 4.43 Data dissemination speed results at density 10 veh/km.

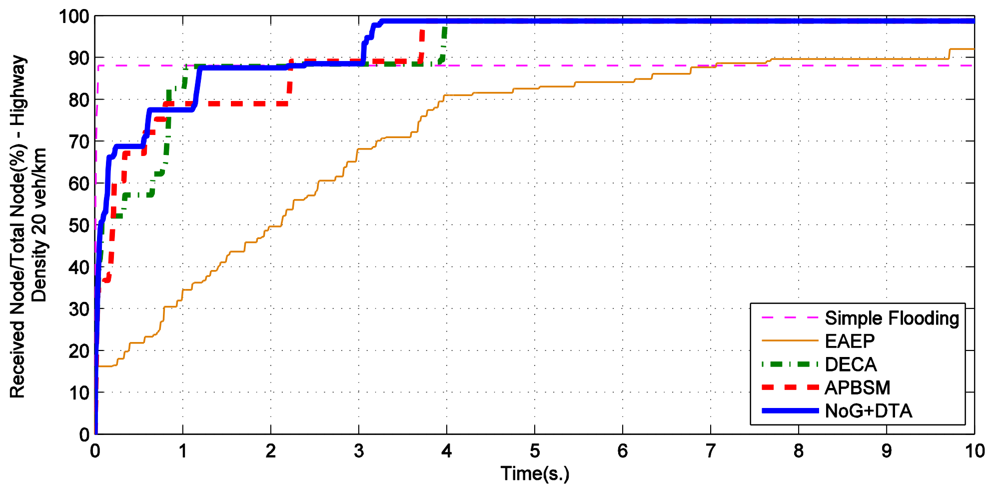


Figure 4.44 Data dissemination speed results at density 20 veh/km.

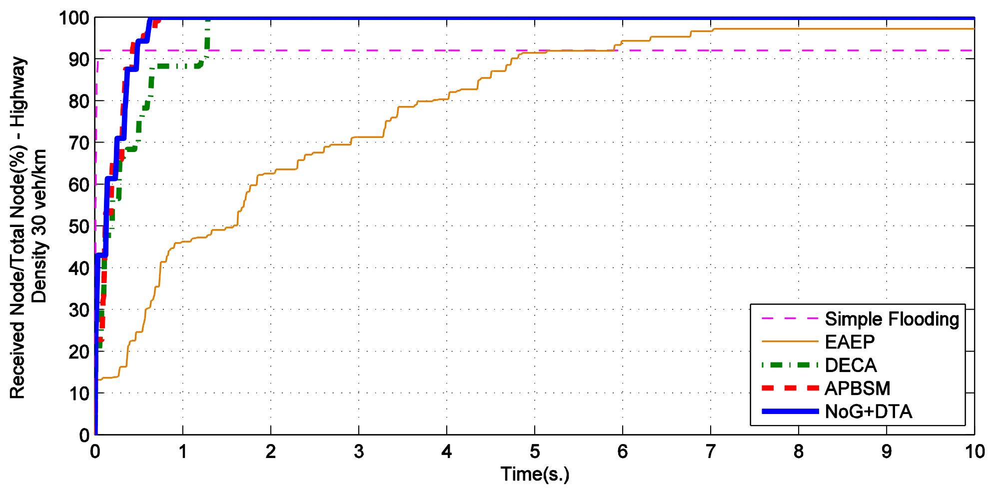


Figure 4.45 Data dissemination speed results at density 30 veh/km.

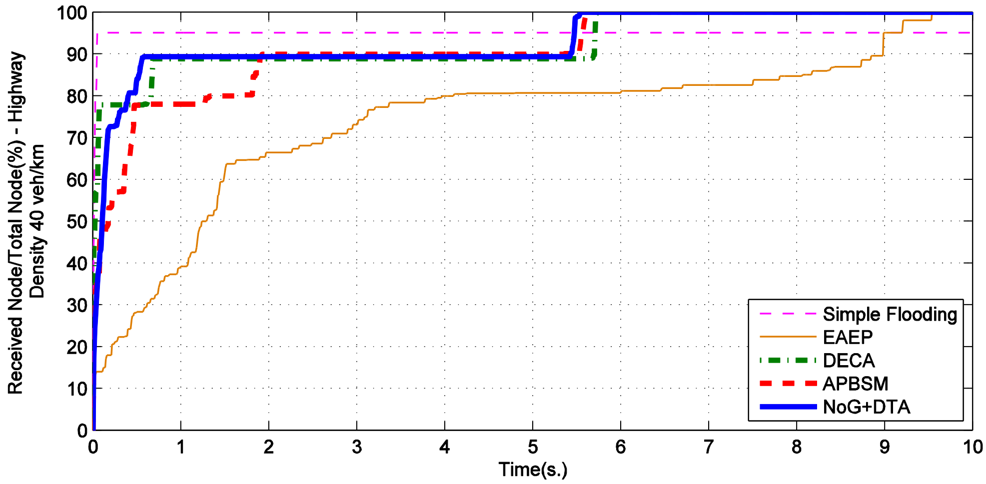


Figure 4.46 Data dissemination speed results at density 40 veh/km.

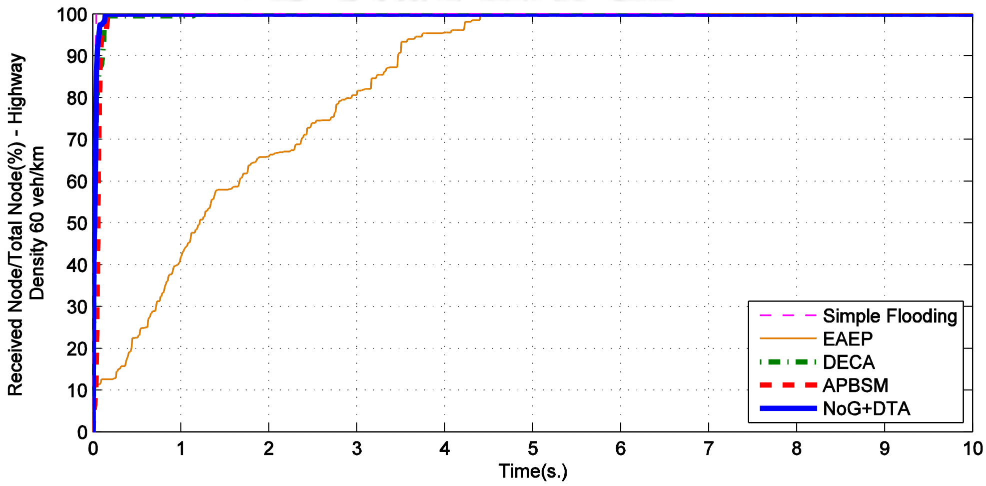


Figure 4.47 Data dissemination speed results at density 60 veh/km.

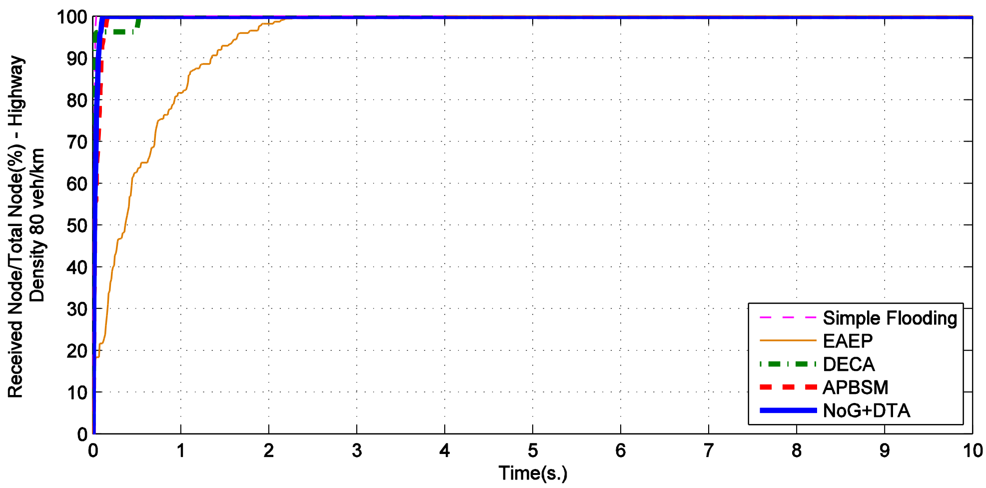


Figure 4.48 Data dissemination speed results at density 80 veh/km.

Data dissemination speed results in urban scenarios.

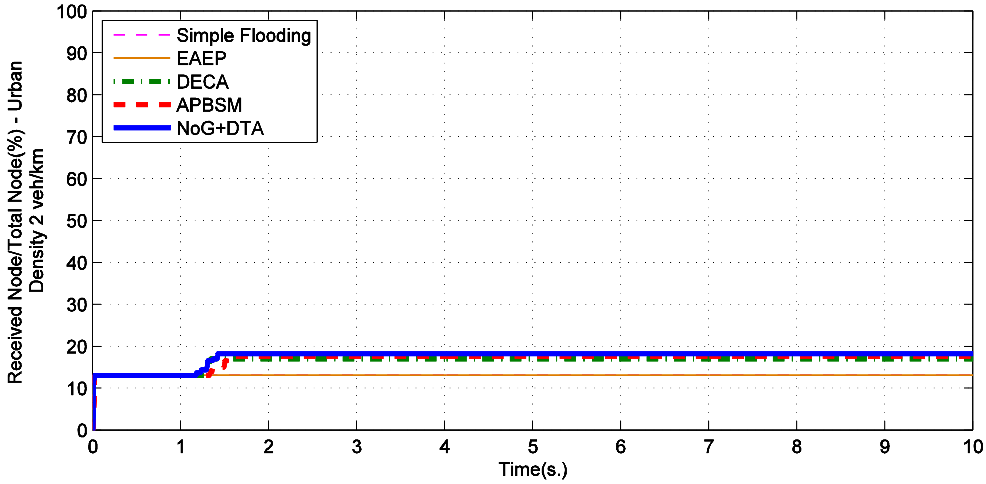


Figure 4.49 Data dissemination speed results at density 2 veh/km.

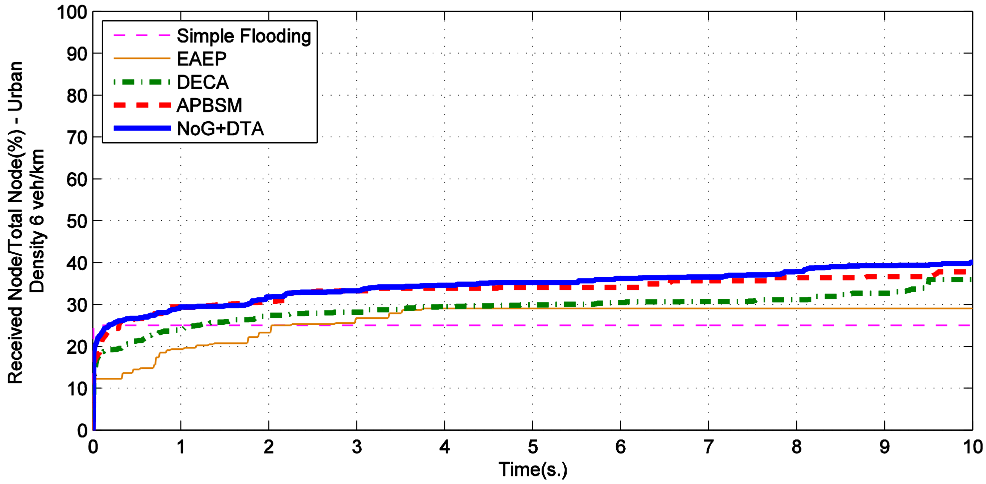


Figure 4.50 Data dissemination speed results at density 6 veh/km.

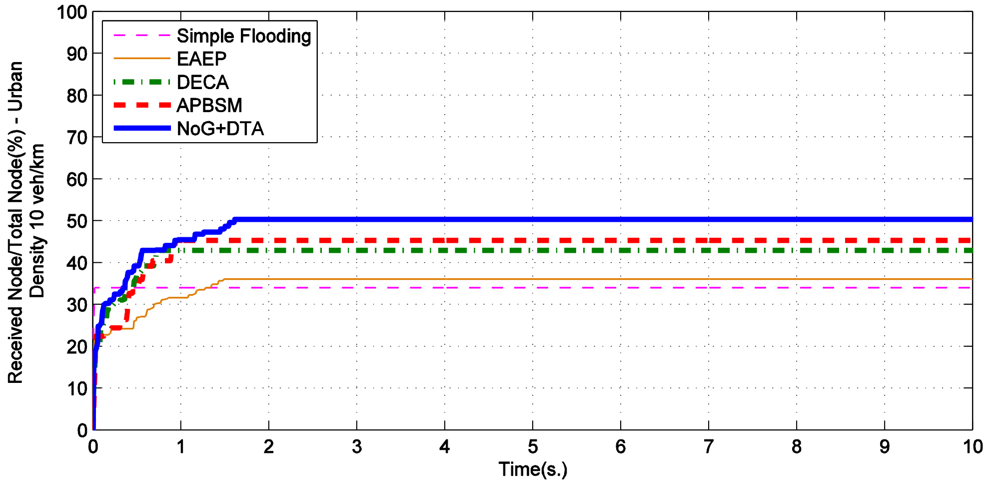


Figure 4.51 Data dissemination speed results at density 10 veh/km.

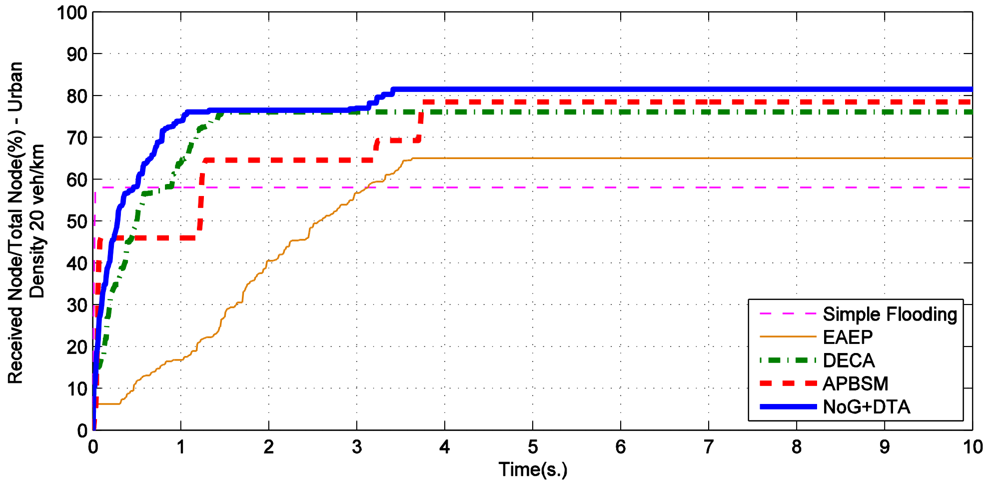


Figure 4.52 Data dissemination speed results at density 20 veh/km.

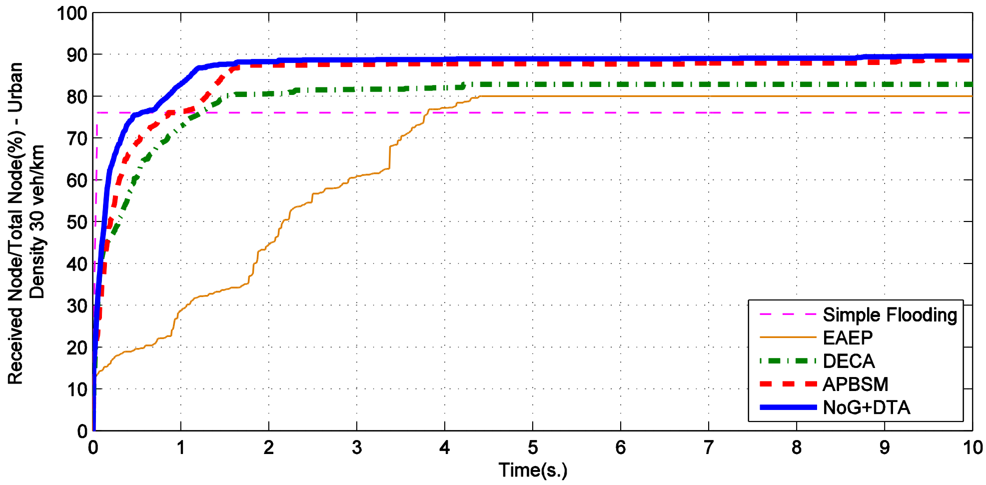


Figure 4.53 Data dissemination speed results at density 30 veh/km.

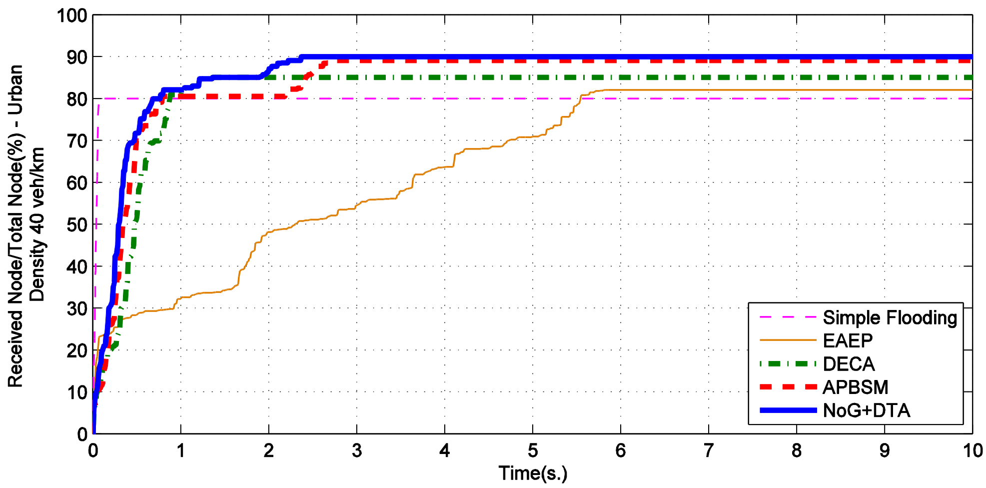


Figure 4.54 Data dissemination speed results at density 40 veh/km.

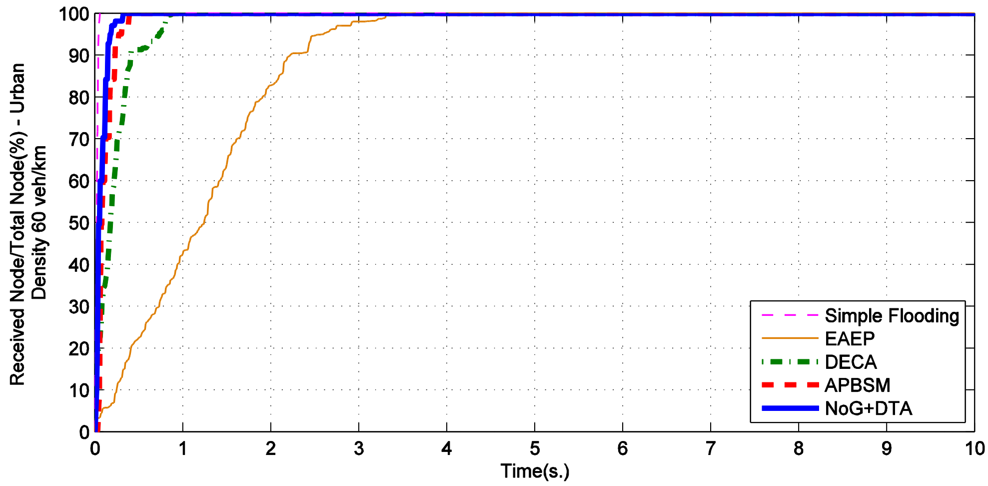


Figure 4.55 Data dissemination speed results at density 60 veh/km.

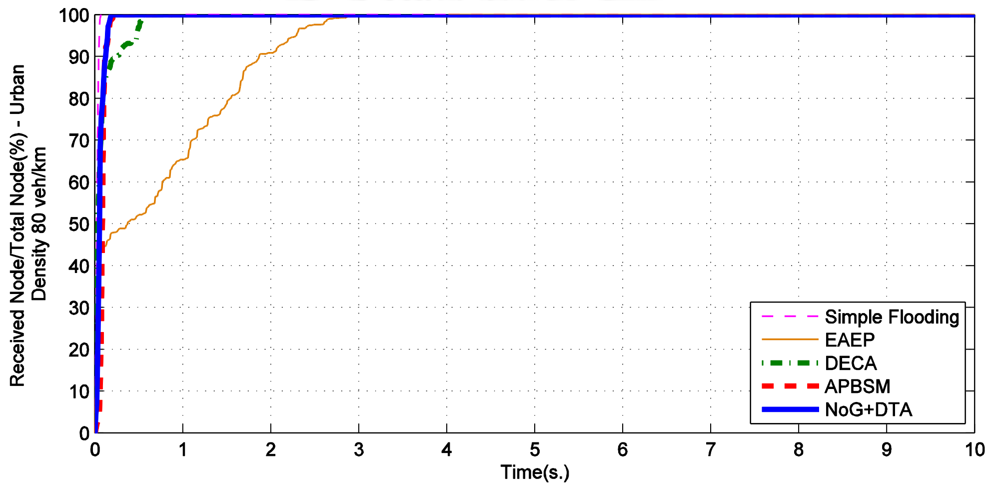


Figure 4.56 Data dissemination speed results at density 80 veh/km.

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 and NoG+DTA is a CDS member.

The results are shown in Fig. 4.57 and Fig 4.58 for highway scenarios and urban scenarios, respectively. For density based algorithms, DECA have its best results in highway scenarios because the algorithms can operate well in simple scenarios. DECA can perform in highway scenarios better than urban scenarios. A reason is that DECA selects the next rebroadcast node from source or precursor so the selected nodes that already received the packet from the others then this node will not rebroadcast as selected node again. For topology based algorithms, the results of APBSM is the same trend with coverage results in Fig. 4.8 and Fig 4.9. 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.

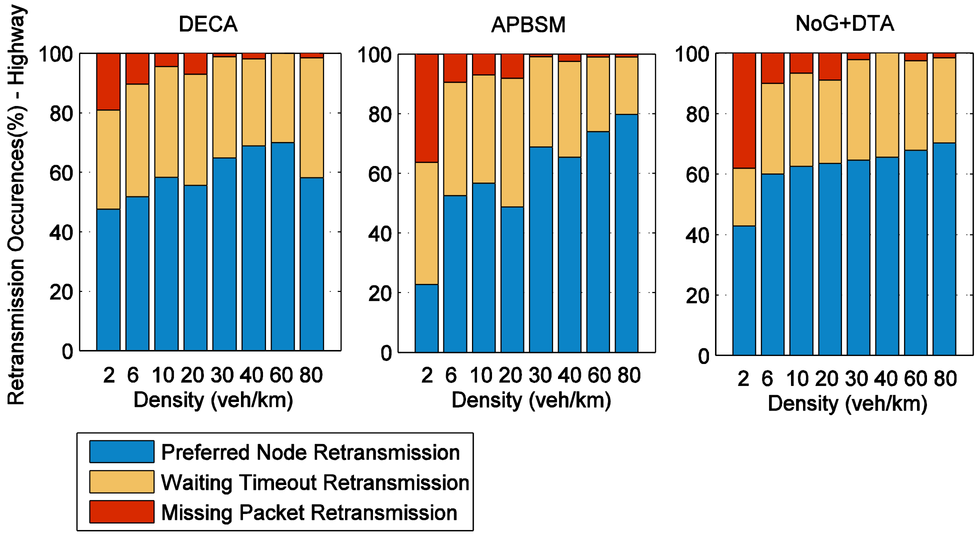


Figure 4.57 Source of retransmission results in highway scenarios.

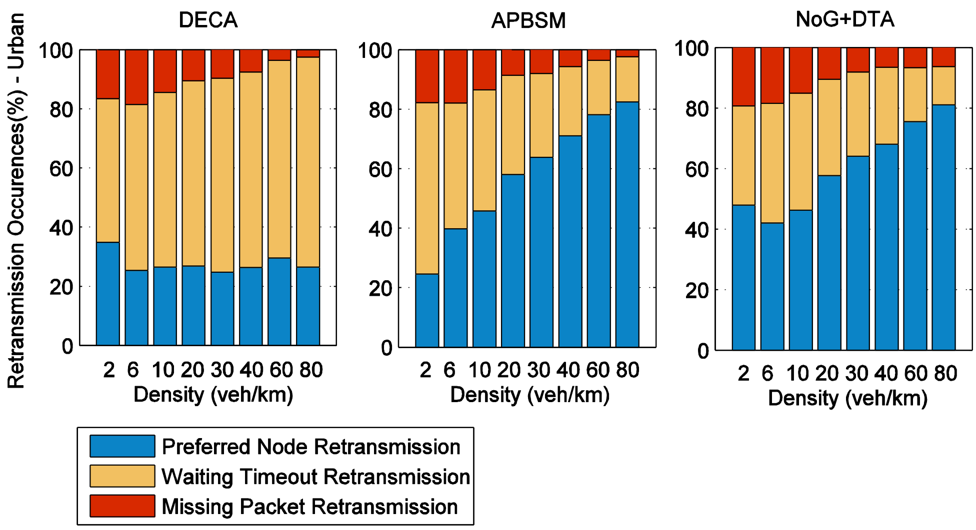


Figure 4.58 Source of retransmission results in urban scenarios.

4.3.2 Performance Evaluation in Real Map Scenarios

The real map scenario is used for realistic simulation that can reflect the performance of protocol in the real world. This real map scenario includes all real traffic environments, such as traffic lights and one-direction roads. This scenario also has an extremely high number of nodes with maximum 2760 nodes in a scenario. The other parameters are setup as shown in Table 4.13.

Table 4.13 *Simulation Configurations.*

<table>
<tr><td>Simulator</td><td>Network Simulator 3 (NS-3.16)</td></tr>
<tr><td>Simulation time</td><td>200 s.</td></tr>
<tr><td>No. of simulation</td><td>20</td></tr>
<tr><td>Road scenario - Vehicle density (veh./km.)</td><td>Real map – 2, 10, 30, 60 and 80</td></tr>
<tr><td>Data Type</td><td>Message: 512 bytes with ten sources
Lifetime: 50 s.</td></tr>
<tr><td>Propagation loss model</td><td>Nakagami propagation loss model (m=1)
with transmission success rate 80% at 250 meters</td></tr>
<tr><td>Protocol setup</td><td>Simple Flooding: Simple flooding represents the simplest protocol with the fastest speed of data dissemination.

EAEP: EAEP represents the epidemic protocol, which uses probabilistic approach and counter-based approach. It operates without beaconing. We use 10 seconds for its waiting timeout instead of 2 seconds in [5] to handle more intermittent connectivity in the real map scenario and 10 seconds provide the highest reliability in the simulations. All other parameters are setup as same as in [5].
<table>
<tr><td>Retransmission interval</td><td>0-10 s.</td></tr>
</table>
DECA: DECA represent the protocol, which uses only density information and avoids using wait timeout mechanism. It is proposed in our previous work.
<table>
<tr><td>Beacon interval</td><td>Adaptive Interval (1.5 - 7 s.)
LIA: c=0.2, minInv=1.5, maxInv=7</td></tr>
<tr><td>Beacon size</td><td>5 bytes + 8 bytes for each message acknowledgement</td></tr>
</table>
</td></tr>
</table>

Table 4.13 Simulation configuration. (Continued)

<table>
<tr><td>Protocol setup</td><td>
APBSM: APBSM represents the protocol, which uses store and forward technique and uses waiting timeout to alleviate the broadcast storm problem. It is chosen as it is shown to provide the highest reliability among the existing protocols we found in the literature. All other parameters are setup as same as in [6].
<table>
<tr><td>Beacon interval</td><td>Constant interval (0.5 s.)</td></tr>
<tr><td>Beacon size</td><td>21 bytes + 8 bytes for each message acknowledgement</td></tr>
</table>
NoG+DTA: NoG is our proposed protocol in this dissertation that operates with our DTA algorithm that also proposed in this dissertation.
<table>
<tr><td>Beacon interval</td><td>Adaptive Interval (1.5 - 7 s.)
LIA: c=0.2, minInv=1.5, maxInv=7</td></tr>
<tr><td>Beacon size</td><td>5 bytes + 70 bytes for Bloom filter bit array</td></tr>
<tr><td>Number of limited responded beacon</td><td>1</td></tr>
<tr><td>Waiting timeout calculation</td><td>β=3</td></tr>
</table>
</td></tr>
<tr><td>Metrics</td><td>

Delivery ratio: is measured as a percentage of the number of nodes that received packets to total nodes in scenarios.
Retransmission overhead: is measured as total bandwidth consumption for packet retransmission from all nodes in scenarios.
Beacon overhead: is measured as total bandwidth consumption for beacon transmission from all nodes in scenarios.
Data dissemination speed: is measured as (4.1), 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.

$$\mathrm{y(t)} = \frac{\sum_{i=0}^{t} r_i}{n} \times 100 \qquad (4.1)$$
</td></tr>
</table>

Delivery ratio: Delivery ratio results are shown in Fig. 4.59. In low density scenarios (less than 10 veh/km), SF and EAEP provide very low reliability because the intermittent connectivity and they lack of acknowledgment information as previously mentioned. DECA, APBSM and NoG+DTA have missing packet recovery so they can handle intermittent connectivity and have very close results.

In medium density scenarios and high density scenarios (more than 10 veh/km), as the density increases, the delivery ratios of SF and EAEP increase. This is because there are less intermittent connectivity occurrences in higher density scenarios. Nevertheless, they have never reached 100% reliability because they lack of acknowledgment information. APBSM provides only 10-11% at density 60-80 veh/km because AckPBSM has extremely high collision caused by the broadcast storm problem from its beacon. DECA also has a low delivery ratio result at density 80 veh/km. DECA also has the broadcast storm problem at density 80 veh/km due to a lot of redundant retransmissions. Those redundant retransmissions are from the inversed waiting timeout function so nodes in the same area have very short range of waiting timeout intervals. NoG+DTA provides almost 100% of delivery ratio at every density scenarios. The efficient mechanisms including our proposed next forwarder algorithm, our new waiting timeout algorithm and our new beacon mechanism can help NoG+DTA to handle in extremely high density with 2760 nodes. DTA can select the efficient node that maximizes the number of received nodes so it helps protocol to reduce the redundant retransmission. The new waiting timeout algorithm can avoid redundant retransmission in dense area and new beacon mechanism can reduce resource consumption from overall traffic. These mechanisms help NoG+DTA to be more scalable than other protocols.

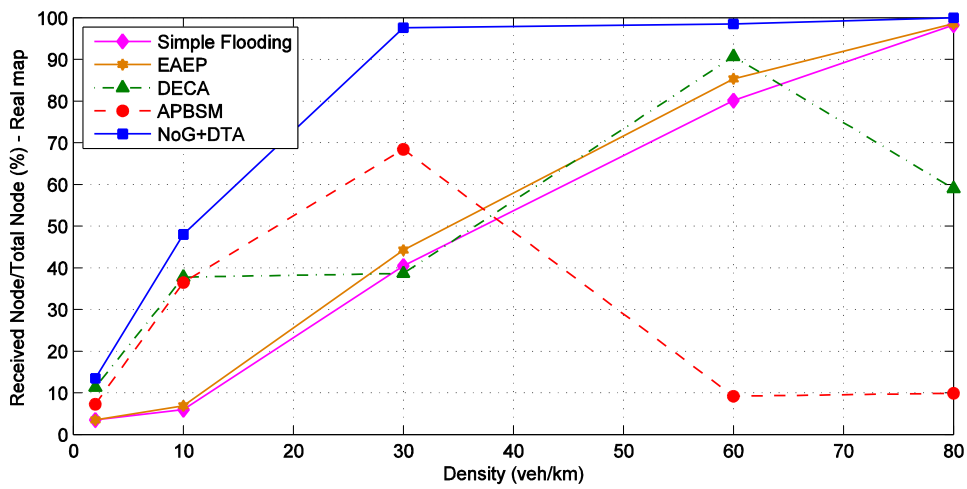


Figure 4.59 Delivery ratio results in real map scenarios.

Retransmission overhead: Retransmission overhead results are shown in Fig. 4.60. Simple flooding has constantly increased their overhead because every node will participate to rebroadcast message if it received the message. EAEP has almost the same retransmission overhead with simple flooding because its rebroadcasting depends on a probability based approach and a counter-based approach. These mechanisms do not work well in high complex scenarios. APBSM has the broadcast storm problem from too frequent beacons due to its precise position requirement so the retransmission mechanism cannot properly work. But these beacons consume about 99% of its total bandwidth usage. As a result, APBSM has very low number of retransmission. This affects to its delivery ratio. DECA also has the broadcast storm problems due to number of retransmissions. In high density, its waiting timeout range from inversed function is too short so it causes lots of redundant retransmissions and most retransmissions are collided. NoG+DTA is designed to solve the mentioned issues that found in other protocols. DTA can select the efficient node that maximizes the number of received nodes that helps protocol to reduce the redundant retransmissions. The new waiting timeout and new beacon mechanism also helps NoG+DTA to be efficient so NoG+DTA increases only a small size of overhead in high density scenarios. DTA also uses the efficient node that

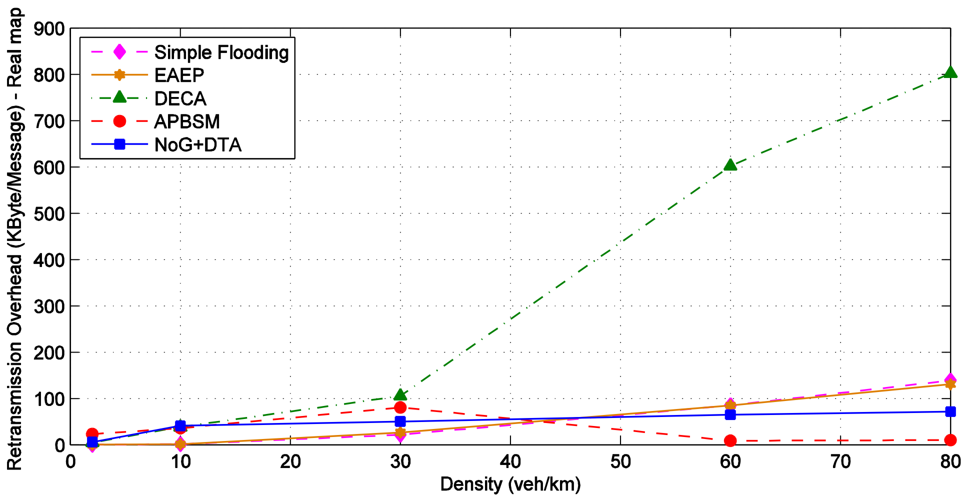


Figure 4.60 Delivery ratio results in real map scenarios.

Beacon overhead: APBSM uses the short constant beacon interval that linearly increases the beacon overhead depending on a number of nodes. When the number of nodes is extremely high, its beacon overhead is seriously high as illustrated in Fig. 4.61. This leads to the broadcast storm problem and decreases protocol performance.

DECA and NoG+DTA have beacon overhead with the same trend as discusses in simple scenarios (Section 4.3.1).

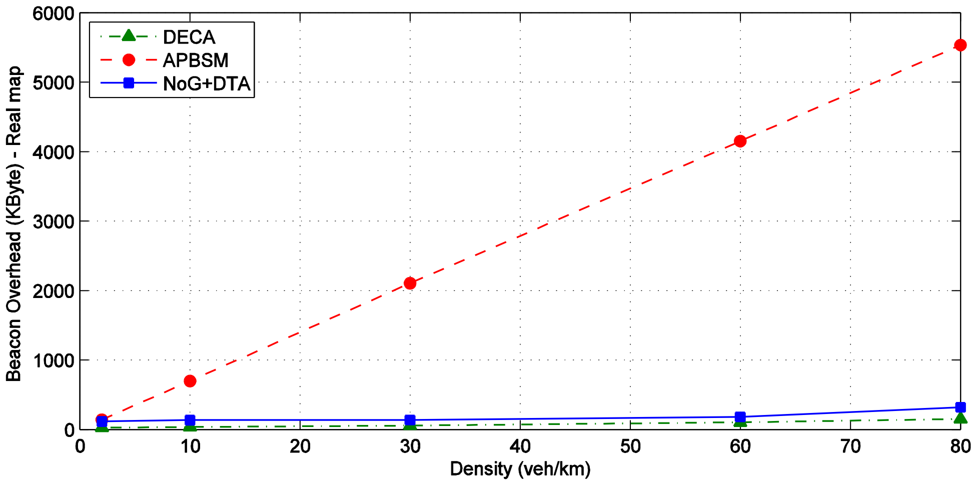


Figure 4.61 Beacon overhead results in real map scenarios.

Data dissemination speed: Figure 4.62-4.66 shows speed of data dissemination at density 10 veh/km, respectively. In low density scenarios (less than 30 veh/km), although simple flooding is the fastest protocol to disseminate data but it cannot deal with intermittent connectivity. EAEP has the lowest speed due to long waiting timeout before rebroadcasting but this helps EAEP to increase more delivery ratio than simple flooding. APBSM, DECA and NoG+DTA do not have much difference of speed of data dissemination. NoG+DTA provides little higher data dissemination speed than DECA and APBSM because NoG+DTA has lower redundant retransmissions according to better forwarder nodes and better mechanism.

In medium density and high density (more than 30 veh/km), simple flooding has the fastest speed of data dissemination in the beginning of simulation but it stops to increase number of received nodes when it has found disconnected network. EAEP has the same behavior as in low density scenarios. APBSM have a broadcast storm problem. Comparing with other protocol it provides extremely low speed of data dissemination and delivery ratio. DECA has the broadcast storm problem either but its speed of data dissemination is still higher than APBSM and EAEP. NoG+DTA has slower speed of data dissemination compared to simple flooding, however, NoG+DTA can reach the same number of received nodes as simple flooding in about 55 milliseconds and it can increase number of received nodes to 80% of total nodes within less than

2.5 seconds for 34.5-kilometer length scenarios. So NoG+DTA can provide speed of data dissemination with the close speed to simple flooding even in intermittent connectivity scenarios.

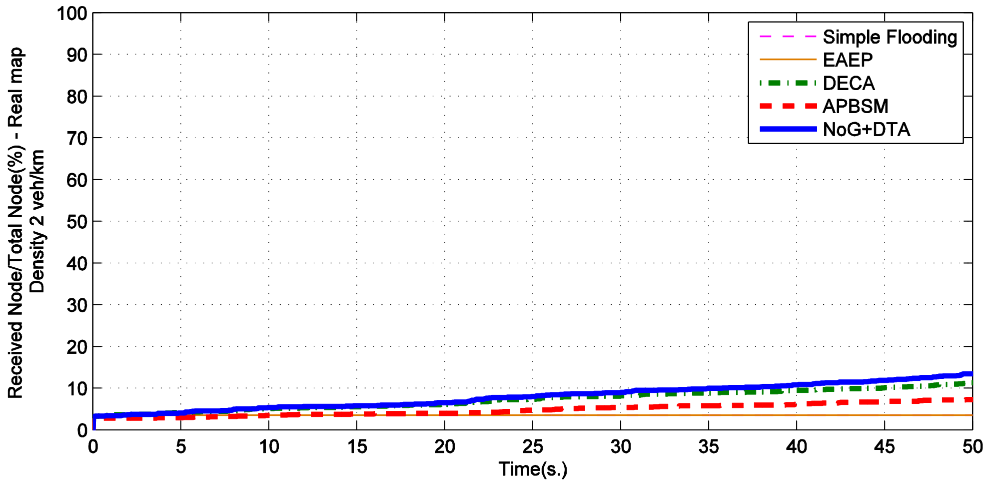


Figure 4.62 Data dissemination speed results at density 2 veh/km.

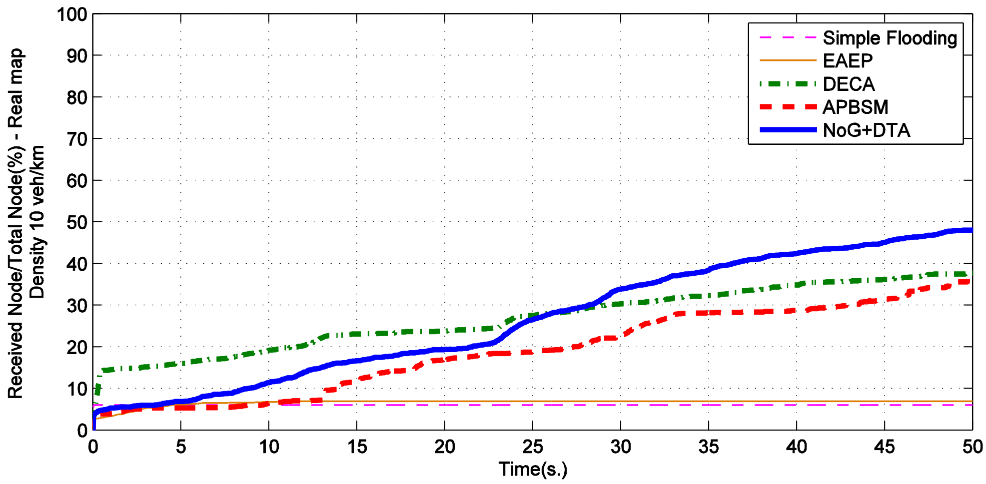


Figure 4.63 Data dissemination speed results at density 10 veh/km.

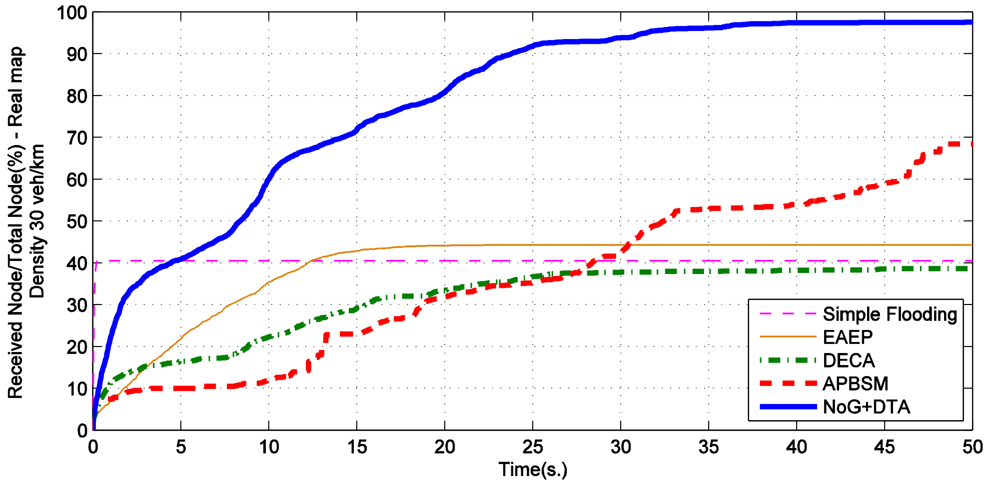


Figure 4.64 Data dissemination speed results at density 30 veh/km.

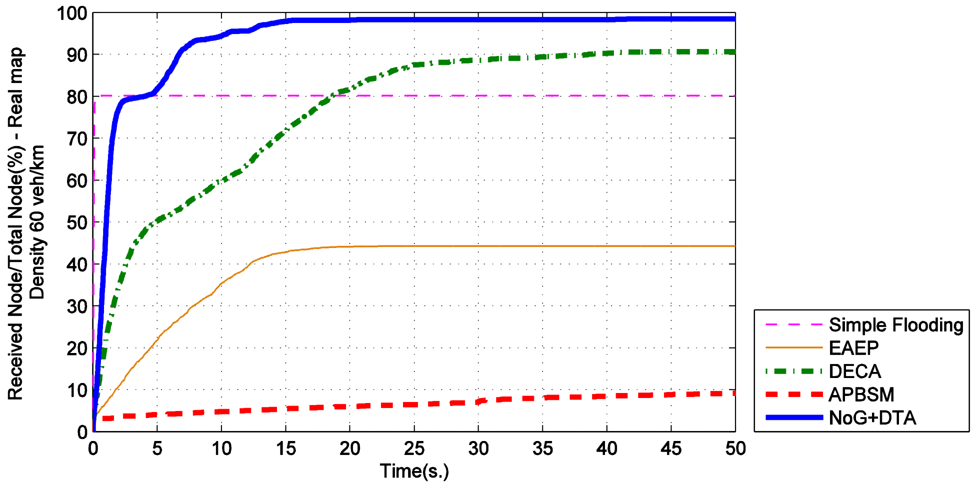


Figure 4.65 Data dissemination speed results at density 60 veh/km.

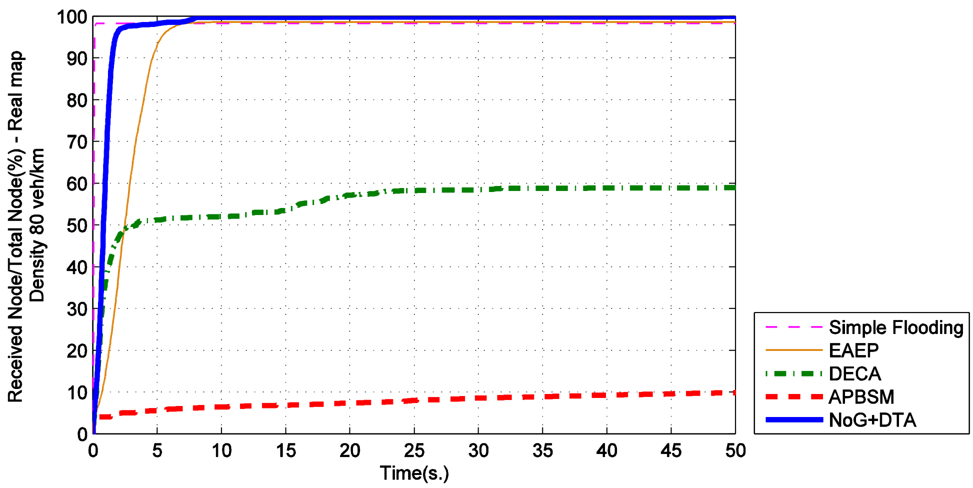


Figure 4.66 Data dissemination speed results at density 80 veh/km.

4.3.3 Multi-Source and Multi-Packet Performance Evaluation

Multi-source and multi-packet simulation can evaluate our proposed protocol in term of scalability. The results tell us the limitation of system that uses our protocol. Configurations of multi-source simulation and configurations of multi-packet simulation are setup as depicted in Table 4.14.

Table 4.14 Simulation Configurations.

<table>
<tr><td>Simulator</td><td>Network Simulator 3 (NS-3.16)</td></tr>
<tr><td>Simulation time</td><td>300 s.</td></tr>
<tr><td>No. of simulation</td><td>20</td></tr>
<tr><td>Road scenario - Vehicle density (veh./km.)</td><td>Real map – 2, 10, 30, 60 and 80</td></tr>
<tr><td>Data Type</td><td>Multi-source:
Message: 512 bytes with 1, 5, 10, 20, 30, 40 and 80 sources
Lifetime: 50 s.
Multi-packet:
Picture: 50 Kbytes with 1 and 2 sources
Lifetime: 100 s.
Sound: 120 Kbytes with 1 and 2 sources
Lifetime: 100 s.</td></tr>
<tr><td>Propagation loss model</td><td>Nakagami propagation loss model (m=1)
with transmission success rate 80% at 250 meters</td></tr>
<tr><td>Protocol setup</td><td>NoG+DTA: NoG is our proposed protocol in this dissertation that operates with our DTA algorithm that also proposed in this dissertation.
<table>
<tr><td>Beacon interval</td><td>Adaptive Interval (1.5 - 7 s.)
LIA: c=0.2, minInv=1.5, maxInv=7</td></tr>
<tr><td>Beacon size</td><td>5 bytes + 70 bytes for Bloom filter bit array</td></tr>
<tr><td>Number of limited responded beacon</td><td>1</td></tr>
<tr><td>Waiting timeout calculation</td><td>β=3</td></tr>
</table></td></tr>
<tr><td>Metrics</td><td>• Delivery ratio: is measured as a percentage of the number of nodes that received completed files to total nodes in scenarios.
• Retransmission overhead: is measured as total bandwidth consumption for packet retransmission from all nodes in scenarios.</td></tr>
</table>

A. Multi-Source Simulation

This simulation represents a scenario that there are many sources in network. We use a broadcasting message for each source. This mean only a packet will be broadcast from one source within each packet lifetime.

Delivery ratio: The results in Fig. 4.67 show the performance of NoG+DTA. NoG+DTA can handle concurrent broadcasting upto 20 sources and the delivery ratio decreases less than 10% in 30 broadcast sources scenarios. However, when a number of sources are increased, the delivery ratio results decrease due to collision issue.

Retransmission overhead: Retransmission overhead results in Fig. 4.68 reflects to number of collision occurrences in each scenario. We can notice that when the number of sources is more than 30, the retransmission overhead significantly increases. The reason is our protocol schedules the broadcasting time or waiting timeout mechanism that are a randomize scheduling. As a result, when there are many nodes that want to broadcast packets and many of packets need to be broadcast, the range of random time is not wide enough to avoid broadcast collision. In order to solve this problem and improve NoG+DTA performance, we can extend the waiting timeout range by changing β value in equation (3.1) Section 3.3 or use time division medium access control that can split the resource into slots and let each node take its turn to rebroadcast its packets. Our protocol can be applied with the existing work [20] [50].

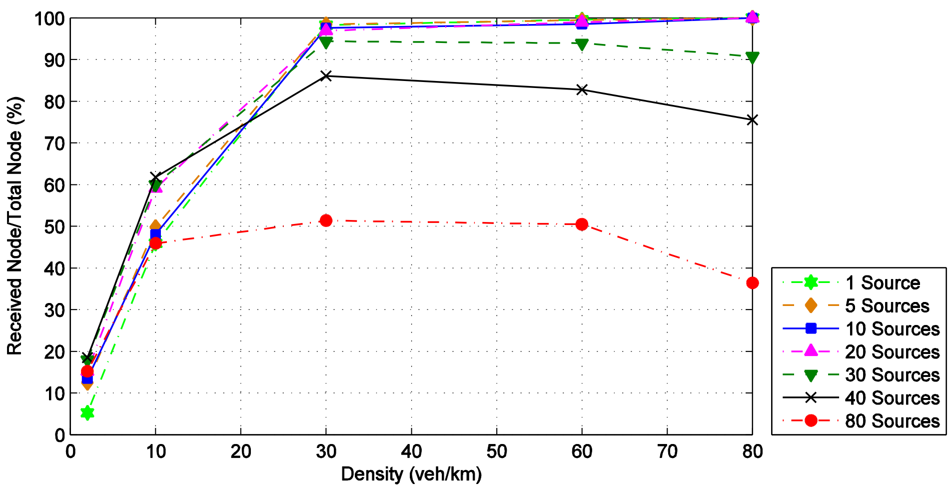


Figure 4.67 Delivery ratio results of multi-source simulation.

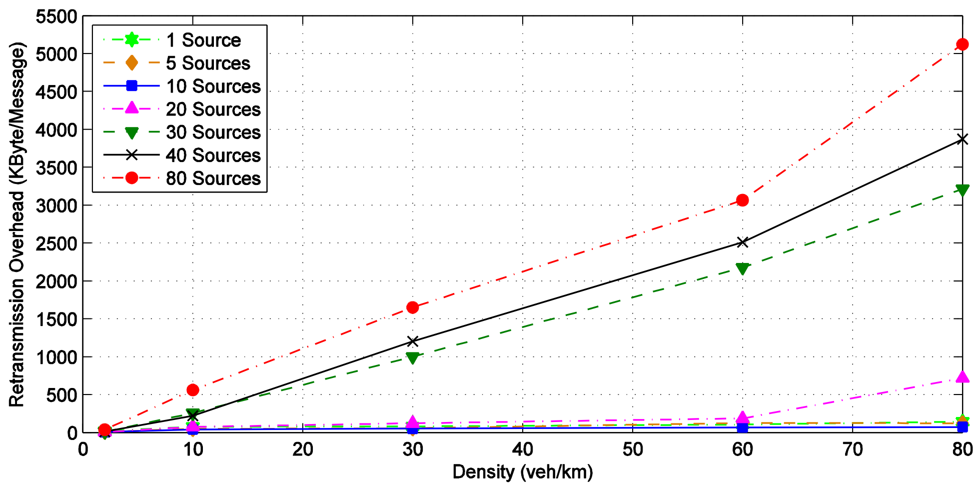


Figure 4.68 Retransmission results of multi-source simulation.

B. Multi-Packet Simulation

This simulation represents a different scenario that there are a few sources broadcast lots of packets in the network. From the previous scenario NoG+DTA can handle upto 20 sources in the scenarios without delivery ratio degradation but to handle lots of packets from a few source is different. This simulation tries to break the limitation of NoG's time scheduling issue that is mentioned above. A picture data type is a medium file with size 50 Kbytes that requires 36 packets for completed file transfer and a sound data type is a medium file with size 120 Kbytes that requires 86 packets for completed file transfer.

Delivery ratio: The delivery ratio results consider only a number of nodes that received completed file not only the exact number of packets. Then, a received node of a picture means a node that received 36 packets with different packet identifiers. From the results in Fig. 4.69, one source of picture broadcasting can deliver data to most of nodes in scenario but never reach 100%. Two sources of picture broadcasting and one source of sound broadcasting have the close delivery result at about 60-70%. Two sources of sound broadcasting scenario, NoG reaches to only 30% of nodes scenarios due to broadcasting storm problem as mention in the last scenario.

Retransmission overhead: The retransmission overhead results in Fig. 4.70 have the same trend with multi source results that when a number of packets increase, a number of retransmissions per packet increase. The random time scheduling issue is a reason for these increasing retransmissions that affect to delivery ratio results. The extended waiting timeout range or the time division medium access can solve this issue in order to support multi-source and multi-packet broadcasting scenarios.

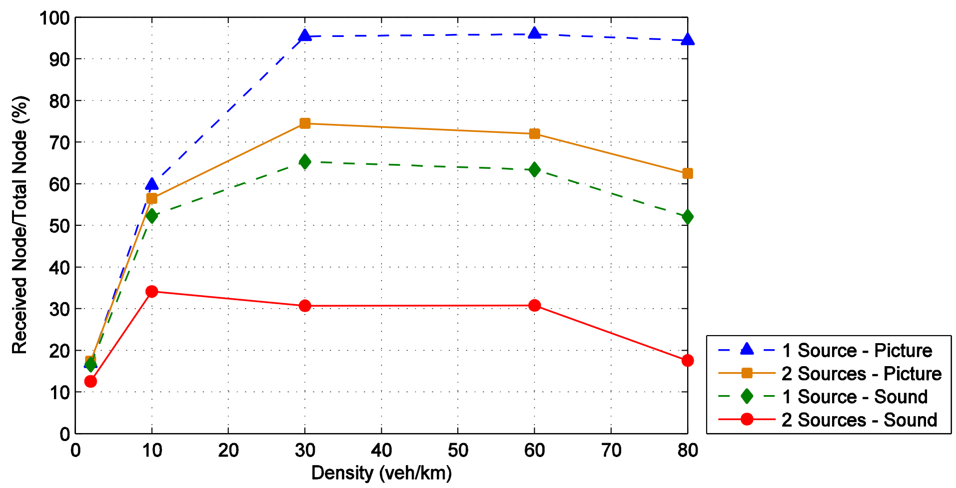


Figure 4.69 Delivery ratio results of multi-packet simulation.

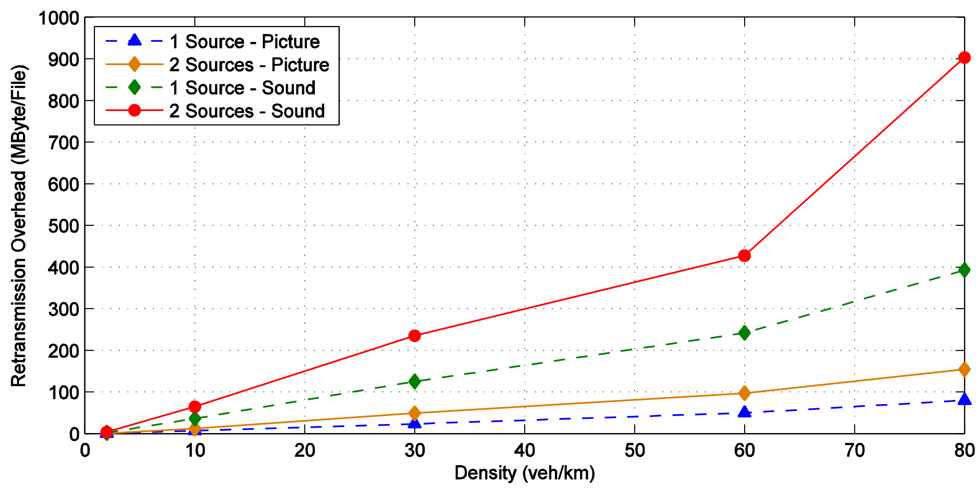


Figure 4.70 Retransmission results of multi-packet simulation.

4.4 Summary

As previously mentioned in Section 3, NoG can be applied with other algorithms. In this case, we can use the waiting timeout mechanism and beacon mechanism that are well designed to avoid most of issues in VANET and applied these modules to the other protocols such as EAEP, DV-Cast and POCA that require geographical knowledge for their operation. A node may accurately know neighbors' position and it may use distance information for each retransmission. The modified protocol will improve their performance in term of retransmission overhead and data dissemination speed. However, position information can violate the privacy issue and it also need a short beacon interval to maintain neighbors' position. Therefore, the improvement may not much different due to inaccurate position issue and inaccurate neighbor's relationship.

NoG protocol and DTA algorithm that do not require any geographical knowledge are evaluated in this section. Each module of NoG operates as designed. DTA also outperform than other existing CDS forming algorithm. Then, the simulations in different of scenarios are done to compare the performance of NoG+DTA with tradition solution and previous work. NoG+DTA provide the highest delivery ratio results with the lowest overhead. The performance of NoG+DTA in the multi-source and multi-packet scenarios shows that NoG+DTA can handle concurrent packet broadcasting upto 20 sources.

CHAPTER 5

Conclusion

5.1 Dissertation Summary

In this dissertation, we focus on data dissemination of vehicular ad hoc network. VANET is a hot research topic and is gaining a lot of interest because of the promising applications for Intelligent Transportation System (ITS) that requires a solution for exchanging their data. We believe that if a protocol can provide fast and reliable data dissemination with low overhead, it will be one of the key successes for such applications.

Non-geographical knowledge protocol (NoG) is proposed to achieve those goals. NoG does not require any position knowledge (GPS) for its operation. This is because position information affects to a protocol design that it need accurate positions which cause more overhead than other information. NoG uses only density information and a list of 1-hop neighbors so it can be tolerant to inaccurate data. NoG consists of three main new design modules. These modules help NoG to reduce overhead and increase the performance of protocol as following explanation.

The next forwarder selection algorithm is the most important module in NoG because a node makes a broadcasting decision through this function. We proposed density based and topology based algorithm (DTA). DTA is a combination of density based algorithm and topology based algorithm. The density based algorithm prefers a node with the highest density to rebroadcast data while the topology based algorithm prefers a node that can complete the 2-common neighbor condition. DTA provides the highest coverage result than the other CDS heuristic algorithms.

The waiting timeout module uses the directed function to avoid collision occurrences. We observe that most of protocols calculate their waiting timeout using an inversed function but this function shortens the range of waiting timeout in dense areas. In order to avoid this issue, the directed function is used in NoG. It reduces retransmission overhead upto 91% in the highest density scenario.

The beacon module is applied with several solutions to reduce beacon overhead as low as possible. NoG uses a linear adaptive beacon interval, called LIA. This algorithm appropriately calculates a beacon interval to suit each density scenario. Then, a beacon with Bloom filter structure is introduced to help NoG having a fixed size beacon. The fixed size beacon supports NoG to reduce beacon size in high density area. Another property of Bloom filter helps DTA to reduce its computation complexity. To the best of our knowledge, NoG is the first protocol that applies the Bloom filter technique for beacon structure. Using LIA and Bloom filter decrease NoG beacon overhead upto 80% of original beacon mechanism.

In order to evaluate performance of NoG+DTA, we simulate our protocol in various scenarios, such as a highway scenario, an urban scenario and a real map scenario. There are many of data types and vary number of sources in the simulation. From the simulation results, NoG+DTA outperform other existing protocols in term of delivery ratio, data dissemination speed and overhead as its design goals. NoG also supports concurrent broadcasting upto 20 sources in our simulations.

5.2 Discussion on Limitations and Future Works

Despite of several benefits, there are limitations that should be mentioned.

DTA algorithm has the highest covered nodes in the scenarios with the lowest number of nodes (CDS members) that used for covering. However, DTA has to use a number of CDS members almost two times more than an exact algorithm that knows global information. We believe that there are the other properties that need to be investigate for algorithm improvement.

The second interesting issue is Beacon structure of beacon need to be tuned for each density scenario to provide the suitable Bloom filter size. If the size of beacon is too large, there will be the additional size of beacon in sparse area that is unnecessary. The size of bloom filter can be reduced with Bloom filter compression but parameters of compression also require fine tuning.

The third issue is from the random waiting timeout scheduling. Although the waiting timeout is calculated from the directed function, when a node has to random its waiting timeout from the calculated range, it could be the same time with other nodes if there are many nodes and there are many packets to rebroadcast. In order to solve this issue, we can change parameters in our waiting timeout function to lengthen the range of waiting timeout. An alternative solution is to use time division medium access control that can split the resource into slot and let each node take its turn to rebroadcast its packets.

5.3 Concluding Remark

We introduce the completely new design of protocol that can efficiently disseminate data in vehicular ad hoc network without any knowledge of geographic or position. We believe that this dissertation can be the beginning of data exchanging solution for further study of a higher layer and higher complex protocol and be a part of success intelligent transportation system.

REFERENCES

1. Sichitiu, M. L. and Kihl, M., *Inter-vehicle communication systems: a survey.* Communications Surveys & Tutorials, IEEE, 2008. **10**(2): p. 88-105.
2. Naumov, V., Baumann, R., and Gross, T., 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. 2006, ACM: Florence, Italy. p. 108-119.
3. Wisitpongphan, N., Fan, Bai, Mudalige, P., and Tonguz, O. K. On the Routing Problem in Disconnected Vehicular Ad-hoc Networks. in INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE. 2007.
4. Vahdat, A. and Becker, D., *Epidemic routing for partially connected ad hoc networks.* Technical Report, 2000(CS-200006).
5. Nekovee, M. and Bogason, B. B. Reliable and Effcient Information Dissemination in Intermittently Connected Vehicular Adhoc Networks. in Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th. 2007.
6. Ros, F. J., Ruiz, P. M., and Stojmenovic, I., Acknowledgment-Based Broadcast Protocol for Reliable and Efficient Data Dissemination in Vehicular Ad Hoc Networks. Mobile Computing, IEEE Transactions on, 2012. **11**(1): p. 33-46.
7. Na Nakorn, K. and Rojviboonchai, K. POCA : Position-Aware Reliable Broadcasting in VANET. in 2nd Asia-Pacific Conference of Information Processing APCIP2010. 2010. Nanchang, China: IEEE.
8. Tonguz, O. K., Wisitpongphan, N., and Fan, Bai, *DV-CAST: A distributed vehicular broadcast protocol for vehicular ad hoc networks.* Wireless Communications, IEEE, 2010. **17**(2): p. 47-57.
9. Na Nakorn, N. and Rojviboonchai, K. DECA: Density-aware reliable broadcasting in vehicular ad hoc networks. in Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON), 2010 International Conference on. 2010.
10. Qayyum, A., Viennot, L., and Laouiti, A., *Multipoint relaying: an efficient technique for flooding in mobile wireless networks.* Research Report, INRIA, 2000. **RR-3898**.
11. Wu, J. and Li, H., On calculating connected dominating set for efficient routing in ad hoc wireless networks, in Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications. 1999, ACM: Seattle, Washington, USA. p. 7-14.

12. Peng, W. and Lu, X., On the reduction of broadcast redundancy in mobile ad hoc networks, in Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing. 2000, IEEE Press: Boston, Massachusetts. p. 129-130.
13. Sucec, J. and Marsic, I., An efficient distributed network-wide broadcast algorithm for mobile ad hoc networks. CAIP Technical Report 248, Rutgers University, 2000.
14. Stojmenovic, I., Seddigh, M., and Zunic, J., *Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks.* Parallel and Distributed Systems, IEEE Transactions on, 2002. **13**(1): p. 14-25.
15. Na Nakorn, K. and Rojviboonchai, K., *DECA-bewa: Density-aware reliable broadcasting protocol in VANETs.* IEICE Transactions on Communication, 2013. **96-B**(5).
16. Bloom, B., Space/Time Tradeoffs in Hash Coding with Allowable Errors. Communications of the ACM, 1970. **13**(7).
17. Na Nakorn, N. and Rojviboonchai, K., Efficient beacon solution for wireless ad-hoc network, in 7th International Joint Conference on Computer Science and Software Engineering (JCSSE'10). 2010, IEEE: Bangkok, Thailand.
18. Sommer, C., Tonguz, O. K., and Dressler, F. Adaptive beaconing for delay-sensitive and congestion-aware traffic information systems. in Vehicular Networking Conference (VNC), 2010 IEEE. 2010.
19. Thaina, C., Na Nakorn, K., and Rojviboonchai, K. A study of adaptive beacon transmission on Vehicular Ad-Hoc Networks. in Communication Technology (ICCT), 2011 IEEE 13th International Conference on. 2011.
20. Settawatcharawanit, T., Choochaisri, S., Intanagonwiwat, C., and Rojviboonchai, K. V-DESYNC: Desynchronization for Beacon Broadcasting on Vehicular Networks. in Vehicular Technology Conference (VTC Spring), 2012 IEEE 75th. 2012.
21. Conti, M. and Giordano, S., *Multihop Ad Hoc Networking: The Reality.* IEEE Communications Magazine, 2007. **45**(4).
22. Siva Ram Murthy, C. and Manoj, B. S., *Ad Hod Wireless Netowrks: Architectures and Protocols*. Vol. 6. 2008, New Jersey, USA: Prentice Hall Professional Technical Reference.
23. IEEE standard for information technology telecommunications and information exchange between systems local and metropolitan area networks specific requirements, *Part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications*. 2007, IEEE: New York, USA.

24. European Commission Information Society and Media. *Cooperative Vehicle-Infrastructure Systems (CVIS)*. 2009 [cited 2014 23 January]; Available from: http://www.cvisproject.org/.
25. Row, S. J., *IntelliDrive: Safer. Smarter. Greener.* Public Roads, 2010. **74**(1): p. 4.
26. Grafling, S., Mahonen, P., and Riihijarvi, J., Performance Evaluation of IEEE 1609 WAVE and IEEE 802.11p for Vehicular Communications, in IEEE 2010 Second International Conference on Ubiquitous and Future Networks (ICUFN). 2010, IEEE: Korea.
27. U.S. Department of Transportation. *IEEE 1609 - Family of Standards for Wireless Access in Vehicular Environments (WAVE)*. 2006 [cited 23 January 2014]; Available from: http://www.standards.its.dot.gov/fact_sheet.asp?f=80.
28. IEEE standard for information technology telecommunications and information exchange between systems local and metropolitan area networks specific requirements, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments*. 2010, IEEE: New York, USA.
29. Raz, R. and Safra, S. A sub-constant error-probability low-degree test, and sub-constant errorprobability PCP characterization of NP. in 29th Annual ACM Symposium on Theory of Computing. 1997. New York, USA: ACM.
30. Fomin, F. V., Grandoni, F., Pyatkin, A. V., and Stepanov, A. A., *Combinatorial bounds via measure and conquer: Bounding minimal dominating sets and applications.* ACM Transactions on Algorithms, 2008. **5**(1).
31. Fomin, F. V., Grandoni, F., Pyatkin, A. V., and Stepanov, A. A., *A measure & conquer approach for the analysis of exact algorithms.* Journal of the ACM, 2009. **56**(5).
32. Van Rooij, J. M. M. J. , Nederlof, J., and Van Dijk, T. C., Inclusion/Exclusion Meets Measure and Conquer: Exact Algorithms for Counting Dominating Sets, in 17th Annual European Symposium on Algorithms. 2009, Springer.
33. Broder, A. and Mitzenmacher, M., *Network Applications of Bloom Filters: A Survey.* Internet Mathematics, 2004. **1**(4).
34. Wewetzer, C., Scheuermann, B., Lubke, A., and Mauve, M. Content registration in VANETs; saving bandwidth through node cooperation. in Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on. 2009.
35. Verma, K., Hasbullah, H., and Kumar, A. An efficient defense method against UDP spoofed flooding traffic of denial of service (DoS) attacks in VANET. in Advance Computing Conference (IACC), 2013 IEEE 3rd International. 2013.

36. Yu, Y., Li, X., Gerla, M., and Sanadidi, M. Y. Scalable VANET content routing using hierarchical bloom filters. in Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International. 2013.
37. Naumov, V. and Gross, T. Connectivity-Aware Routing (CAR) in Vehicular Ad-hoc Networks. in INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE. 2007.
38. ETSI, Intelligent Transport Systems (ITS); Vehicular Communications;Basic Set of Applications; Definitions. 2009, European Telecommunications Standards Institute: TR 102 638.
39. Parno, B. and Perrig, A. Challenges in Securing Vehicular Networks. in The 4th Workshop on Hot Topic in Netwoks (HOTNETS-IV). 2005. College Park, Maryland, USA.
40. Dotzer, F., Privacy issues in vehicular ad hoc networks, in Proceedings of the 5th international conference on Privacy Enhancing Technologies. 2006, Springer-Verlag: Cavtat, Croatia. p. 197-209.
41. Raya, M. and Hubaux, J. P., *Securing vehicular ad hoc networks.* ACM Journal of Computer Security, 2007. **15**(1): p. 39-68.
42. German Aerospace Center, (DLR). *Simulation of Urban Mobility (SUMO)*. 2013 [cited 2014 23 January]; Available from: http://sumo.sourceforge.net.
43. Piorkowski, M., Raya, M., and Hubaux, J. P. *Traffic and Network Simulation Environment (TraNS)*. 2008 [cited 2014 23 January]; Available from: http://trans.epfl.ch.
44. US Department of Transportation Federal Highway Administration, *Chapter 3: The 13 Controlling Criteria*. 2007, New York, USA: US Department of Transportation Federal Highway Administration.
45. Artimy, M., *Local Density Estimation and Dynamic Transmission-Range Assignment in Vehicular Ad Hoc Networks.* Intelligent Transportation Systems, IEEE Transactions on, 2007. **8**(3): p. 400-412.
46. Marshall, Dave. *Implications of Sample Rate and Bit Size*. 2001 [cited 2014 24 January]; Available from: http://www.cs.cf.ac.uk/Dave/Multimedia/node150.html.
47. National Science Foundation. *The Network Simulator (NS-3)*. 2012 [cited 2014 23 January]; Available from: http://www.nsnam.org/.
48. Báguena, M., Calafate, C. T., Cano, J. C. , and Manzoni, P., *Towards realistic vehicular network simulation models*, in *2012 IFIP Wireless Days (WD)*. 2012, Dublin, Ireland.
49. Varadhan, K. *The Network Simulator (NS-2)*. 2011 [cited 2014 23 January 2014]; Available from: http://www.isi.edu/nsnam/ns.

50. Bai, S. and Jung, J., Congestion Awareness Multi-Hop Broadcasting for Safety Message Dissemination in VANET. IEICE Transaction on Communication, 2010. E93-B(21).

APPENDIX

APPENDIX A
Publication

During my Ph.D. study, I have published several papers as follows.

International Journal Publications

1. Kulit Na Nakorn and Kultida Rojviboonchai, “Non-GPS Data Dissemination for VANET”, International Journal of Distributed Sensor Networks, Volume 2014, Article ID 906084, January 2014, page(s):1-17.
2. Kulit Na Nakorn and Kultida Rojviboonchai, “DECA-bewa: Density-Aware Reliable Broadcasting Protocol in VANETs”, IEICE Transaction of Communication, Volume E96-B, Issue 5, May 2013, page(s):1112-1121.

International Conference Publications

1. Kulit Na Nakorn, Yusheng Ji and Kultida Rojviboonchai, “Bloom Filter for Fixed-Size Beacon in VANET”, in Proceeding of the 79th IEEE Vehicular Technology Conference (IEEE VTC), Seoul, Korea, May 18-21, 2014.
2. Kornkanok Khaoampai, Kulit Na Nakorn and Kultida Rojviboonchai, "Low Complexity Floor Localization Algorithm for Mobile Phone",in Proceedings of the 11th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON 2014), Nakhon Ratchasima, Thailand, May 14-17, 2014.
3. Nattavit Kamoltham, Kulit Na Nakorn and Kultida Rojviboonchai, "From NS-2 to NS-3 - Implementation and evaluation",in Proceedings of the 2012 Computing, Communications and Applications Conference (ComComAp), Hong Kong, January 11-13, 2012.
4. Nattavit Kamoltham, Kulit Na Nakorn and Kultida Rojviboonchai, "Improving Reliable Broadcast over Asymmetric VANETs Based on a RSSI-Voting Algorithm", in Proceedings of the 2011 International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS), Chiang Mai, Thailand, December 7-9, 2012.
5. Nathakorn Nophakunvijai, Sukhumarn Archasantisuk, Kulit Na Nakorn and Kultida Rojviboonchai, "DECA on Android: Reliable Broadcasting in Ad Hoc Network on Android Platform",in Proceedings of the 2012 ICT International Senior Project Conference (ISPC2012), Bangkok, Thailand, April 20, 2012.

6. Chayanin Thaina, Kulit Na Nakorn and Kultida Rojviboonchai, "A Study of Adaptive Beacon Transmission on Vehicular Ad-Hoc Networks", in Proceedings of the 13th IEEE International Conference on Communication Technology (IEEE ICCT), Jinan, China, September 25-28, 2011.
7. Wipawee Viriyapongsukit, Kulit Na Nakorn and Kultida Rojviboonchai, "A Novel Packet Dropping Policy for Vehicular Ad-Hoc Networks",in Proceedings of the 13th IEEE International Conference on Communication Technology (IEEE ICCT), Jinan, China, September 25-28, 2011.

VITA

Kulit Na Nakorn was born in Suratthani, Thailand, on August, 1986. He graduated his high school from Suratthani School in 2004. While in high school, he got a scholarship from American Field Service to be an exchanged student in Oregon, United State for 1 year. Then, he received the B.Eng. and M.Eng. Degrees in Computer Engineering from Chulalongkorn University, Bangkok, Thailand in 2008 and 2010, respectively. His Bachelor Degree has been supervised by Asst. Prof. Dr. Krerk Piromsopa. His master and doctorate have been under the supervision of Asst. Prof. Dr. Kultida Rojviboonchai. In 2009, he received a grant for master degree from the Department of Computer Engineering, Chulalongkorn University through the CP CU Academic Excellence Scholarship (Ad-cha-ri-ya-kuen-rang Scholarship). Since 2011, he has received another grant for doctorate from the scholarship of Graduate School, Chulalongkorn University to commemorate the 72nd anniversary of his Majesty King Bhumibala Aduladeja. His field of interest includes various topics in MANETs, VANETs, communications, wireless networks, distributed systems and embedded systems.