Universidad de Deusto

Clustering, Medium Access Control, And Data Aggregation In Vehicular Ad Hoc Networks

by ABOOBEKER SIDHIK KOYAMPARAMBIL MAMMU

A thesis presented to the University of Deusto in fulfillment of the thesis requirement for the degree of Doctor of Philosophy in Computer Science and Telecommunication Engineering

Director NEKANE SAINZ BEDOYA And UNAI HERNÁNDEZ JAYO

UNIVERSIDAD DE DEUSTO

CLUSTERING, MEDIUM ACCESS CONTROL, AND DATA AGGREGATION IN VEHICULAR AD HOC NETWORKS

by ABOOBEKER SIDHIK KOYAMPARAMBIL MAMMU

A thesis presented to the University of Deusto in fulfillment of the thesis requirement for the degree of Doctor of Philosophy in Computer Science and Telecommunication Engineering

Director NEKANE SAINZ BEDOYA And UNAI HERNÁNDEZ JAYO

PhD Student Director

Bilbao, January 2016

Clustering, Medium Access Control, And Data Aggregation In Vehicular Ad Hoc Networks

Autor: Aboobeker Sidhik Koyamparambil Mammu
Director: Nekane Sainz Bedoya And Unai Hernández Jayo

Printed in Bilbao
First Edition, January 2016

Praise be to Allah

This PhD thesis is dedicated to my mother, Kunjumma, and my father, Mammu.

Abstract

Traffic congestion and traffic accidents are some of the serious problems faced by humans in their day to day life. These social problems causes a lot of human inconvenience, traffic congestion for instance reduces the productivity of humans by wasting a lot of time being on the road. To reduce the traffic congestion and road accidents, several traffic efficiency and safety applications can be implemented using the vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) communication. The network consisting of vehicles and road side units (RSUs) are known as vehicular ad hoc networks (VANETs). V2V and V2I communications exchange two different types of messages, periodic and event-driven messages. Periodic messages of a vehicle updates the nearby vehicles about its current position, speed and direction and event-driven messages are transmitted during a hard brake or detection of dangerous road condition.

Most of the traffic safety and efficiency applications are based on the broadcasting of safety or update messages by vehicles or RSUs. This type of transmission is designed to help reduce road accidents and traffic congestions. However, during traffic accidents, there can be a huge number of vehicles in a certain region that can lead to congestion in the channel access. Furthermore, congestion in the channel also arises during the busy hours of the day. The reliability of the safety application depends upon the successful delivery of safety message to its neighbouring vehicles and RSUs with a predictable delay, which is one of the main functions of a deterministic medium access control (MAC) protocol proposed for VANETs.

Another important challenge in VANETs is to achieve scalability. With the increasing size of the road system, the data load eventually congests

the network channel. This challenge can be solved by reducing the data collisions and data load through clustering, MAC and data aggregation. The main aim of this thesis is to design a cluster based MAC protocol that is scalable in high density scenarios, transmit the safety message with high reliability, and deliver the safety message with a predictable delay. Moreover, reduce the channel load during traffic accidents using data aggregation application.

This thesis makes the following research contributions:

- A cluster head (CH) election, cluster formation, and cluster maintenance algorithm that increases the stability of the cluster structure and minimises overall management overhead for the maintenance of the algorithm.
- A hybrid protocol based on contention free and contention based channel access. Time division multiple access (TDMA) slot allocation priority based on different parameters such as future position, and standard score.
- A DA-CMAC protocol that allocates slots based on the direction of movement of vehicles.
- A cluster based data aggregation scheme that aggregates data based on the density of vehicles.

I will show that my CH election protocols, cluster maintenance and direction aware clustering protocols improves the stability of cluster architecture, there by reducing the maintenance overhead caused due to CH re-election, re-clustering, and re-configuration. The CH election protocols presented here leads to an hierarchical efficient network topology. Additionally, I will show that the proposed MAC protocols performs better than the current wireless access for vehicular environment (WAVE) standard in terms of predictability, reliability, and scalability. Moreover, the cluster based data aggregation is proposed to reduce the load in the channel. The protocols are evaluated using computer simulations that are developed in network simulator ns-2,

and the network simulator ns-3. Moreover, the performance of the proposed protocols are compared with the WAVE standard, SBCA and the HCA protocol. Furthermore, the ability of the proposed protocols are demonstrated by detailed delivery delay analysis, including percentage of access collisions, cluster and CH lifetime, precision, and level of aggregation in different densities.

Author's publications from this thesis

1. **Mammu, A.S.K.**, Sharma, A., Hernandez-Jayo, U., and Sainz, N., ”A Novel Cluster-Based Energy Efficient Routing in Wireless Sensor Networks,” *in Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on,* vol., no., pp.41-47, 25-28 March 2013.

2. **Koyamparambil Mammu, A.S.**, Hernandez-Jayo, U., and Sainz, N., ”Cluster-based MAC in VANETs for safety applications,” *in Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on,* vol., no., pp.1424-1429, 22-25 Aug. 2013.

3. **Mammu, A.S.K.**, Hernandez-Jayo, U., Sainz, N., and de la Iglesia, I., ”Cross-Layer Cluster-Based Energy-Efficient Protocol for Wireless Sensor Networks,” *Sensors* 2015, 15, 8314-8336.

4. **Koyamparambil Mammu, A.S.**, Jiru, J., and Hernandez Jayo, U., ”Cluster Based Semantic Data Aggregation in VANETs,” *in Advanced Information Networking and Applications (AINA), 2015 IEEE 29th International Conference on,* vol., no., pp.747-753, 24-27 March 2015.

5. Jiru, J., **Koyamparambil Mammu, A.S.**, and Roscher, K., ”Adaptive decision algorithms for data aggregation in VANETs with defined channel load limits,” *in Intelligent Vehicles Symposium (IV), 2015 IEEE ,* vol., no., pp.1367-1372, June 28 2015-July 1 2015.

6. Unai Hernandez-Jayo, **Aboobeker Sidhik K.M.**, and Idoia De-la-Iglesia (2014). ” Reliable Communication in Cooperative Ad

hoc Networks”, *Contemporary Issues in Wireless Communications,* ISBN: 978-953-51-1732-2, InTech, DOI: 10.5772/59041.

7. Hernandez-Jayo, U., De-la-Iglesia, I., Lacoume, I., **Koyamparambil Mammu, A.S.**, ”Short paper: I-CROSS: Intersection crossing warning application based on V2V communications,” *in Vehicular Networking Conference (VNC), 2013 IEEE*, vol., no., pp.206-209, 16-18 Dec. 2013.
8. **Koyamparambil Mammu, Aboobeker Sidhik.**, Hernandez-Jayo, Unai., Sainz, N., “Direction Aware Cluster-Based multi channel MAC Protocol for Vehicular Ad Hoc Networks, ” *WiMob 2015 on* Oct 19 - 21 2015.
9. **Koyamparambil Mammu, Aboobeker Sidhik.**, Hernandez-Jayo, Unai., Sainz, N., “A Cluster based MAC protocol for improving traffic safety in VANETs” 2015. Accepted to *Wireless Networks*.
10. Hernandez-Jayo, Unai.,**Koyamparambil Mammu, Aboobeker Sidhik.**, Sainz, N., “Deterministic MAC protocol based on clustering for VANETs,” Book Chapter submitted to *Springer*.

Acknowledgements

"In the name of Allah, the most Gracious, the most Merciful"

"My Lord! Increase me in knowledge." (20/114).

I prostrate to Allah thanking him for granting me the strength and perseverance to complete my Ph.D.

This dissertation would not have been possible without the guidance and the help of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this study.

First and foremost, I would like to express my deep appreciation to Dr. Unai Hernandez Jayo and Dr Nekane Sainz Bedoya for their exemplary supervision, great support, and valuable advice throughout my PhD program. I would like to express my deepest gratitude to Dr. Asier Perallos for granting me the scholarship to pursue PhD in University Of Deusto.

I sincerely would like to thank all my colleagues at the Mobility group for the beneficial discussions, and continuous exchange of knowledge.

I would like to thank my ex colleague Inaki, for his very useful guidance during my initial years in University Of Deusto. Sincere thanks go to Prof. Jaafar MH Elimirghani, from the School of Electronic and Electrical Engineering at University Of Leeds, for providing supportive recommendation letters at the time of my application for PhD program at the University of Deusto.

I would like to thank Conchi, Mekki, Salma, Najat and Mohammed Sebbahi, for supporting me throughout my stay in Bilbao. From the bottom of my heart, I thank my parents for their prayers, care, and

unconditional love. There is nothing I can do to thank you enough. They always had confidence in me and offered me encouragement and support in all my endeavors.

Contents

List of figures

List of tables

Acronyms

VANETs	Vehicular ad hoc networks
V2V	Vehicle to Vehicle
V2I	Vehicle to Infrastructure
RSUs	Road Side Units
MAC	Medium Access Control
TDMA	Time Division Multiple Access
CH	Cluster Head
DA-CMAC	Direction Aware Cluster Based Multi-Channel MAC Protocol
WAVE	Wireless Access for Vehicular Environment
HCA	Hierarchical Clustering Algorithm
ITS	Intelligent Transportation Systems
QoS	Quality-of-Service
CMs	Cluster Members
CSMA/CA	Carrier Sense Multiple Access With Collision Avoidance
CCP	Control Channel Period
SCP	Service Channel Period
CBSDA	Cluster Based Semantic Data Aggregation

IDs	Identifiers
CCHI	Control Channel Interval
SCHI	Service Channel Interval
SCHs	Service Channels
CCHs	Control Channels
ETSI	European Telecommunications Standards Institute
IEEE	Institute of Electrical and Electronics Engineers
FCC	Federal Communication Commission
DSRC	Dedicated Short Range Communications
SP	Synchronization Period
GPS	Global Positioning System
GI	Guard Interval
MANETs	Mobile Ad Hoc Networks
CAM	Cooperative Awareness Message
DENM	Decentralized Environment Notification Message
SPAT	Signal Phase And Timing Message
SAM	Service Announcement Message
MMAC	Multi-Channel MAC
QCH	Quasi Cluster Head
QCM	Quasi Cluster Member
RCM	Region-based Clustering Mechanism

MOBIC	Mobility Based Metric for Clustering
RSS	Received Signal Strength
DGMA	Distributed Group Mobility Adaptive
TSD	Total Spatial Dependency
RMAC	Robust Mobility Adaptive Clustering
FCR	Fast Collision Resolution
STDMA	Self-organizing TDMA-based MAC
SDMA	Space Division Multiple Access
DMMAC	Dedicated Multi-channel MAC
ABF	Adaptive Broadcast Frame
TAG	Tiny AGgregation
SOTIS	Self-Organizing Traffic-Information System
MDS	Minimum Dominating Set
GV	Gateway Vehicle
CJ	Cluster-Join Message
BCH	Backup Cluster Head
D-CBM	Deusto Cluster Based MAC
RJR	Request-Join-RSU Message
RJC	Request-Join-Cluster Message
ACK	Acknowledgment message
GHL	Gateway Head List

GTL	Gateway Tail List
CS	Connected Set
US	Undecided State
CBR	Channel Busy Ratio
CL	Cluster Length

1

Introduction

Future era of telematic applications are being led by the maturation of recently deployed intelligent transportation systems (ITS), assisted by the joint effort between the automotive and information communication technology industry. Vehicular Ad-Hoc Networks (VANETs) serve as the important part of ITS, where it utilizes the possibility of cooperative communication between Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I). VANETs can solve social problems (air pollution, traffic jams, and traffic accidents) and lead to an incredible business potential.

VANETs have a wide variety of applications that varies from safety applications such as cooperative collision avoidance to infotainment applications such as gaming application. Traffic safety applications enhances the driving and reduces the possibility of road accidents by providing enough reaction time to a dangerous scenario so that the driver can apply brakes (eco-driving). Some of the examples of safety applications are cooperative collision warning, cooperative intersection collision avoidance, road accident management, video streaming etc. Periodic and event driven messages from safety applications could be used to caution the driver about a possible collision or dangerous driving. Thereby, reducing the number of road collisions and their thus reduce the number of car accidents and their intensity.

The main goal of traffic efficiency applications are achieved by delivering the traffic information from one location to control center with in the predicable and reproducing the data with high precision from all vehicles after data aggregation. Some of the traffic efficiency applications are traffic monitoring, traffic management, platooning etc. Infotainment applications improve the comfort level of passengers travelling in the vehicle. Moreover, these applications can reduce the traffic congestion and improve the road utilization. Traffic information messages or periodic messages from vehicles can be used for improving traffic management, thereby reducing travelling time to the destination and fuel consumption. Lastly, infotainment and marketing services can be made available using this type of communications. For example, a restaurant can advertise its offers to different users in the road based on the current location and direction. Moreover, internet connectivity can be made available to travelers for increasing comfort.

Quality-of-Service (QoS) required for different applications varies from hard real time to soft real time communication requirements. Delay occurring for an application with soft real time might compromise service quality and in an application with hard real time, a delayed response might lead to a tragedy. Therefore, research especially on safety applications for VANETs has been receiving a lot of interest in the last few years. Several research institutes, academics and major players in the automobile industry are analyzing and evaluating the efficiency of V2V and V2I systems (e.g., DSRC, 802.11p. ITS G5A) for safety and non safety applications. The IEEE 1609 family of standards are developed for wireless communication between vehicles on the road. The IEEE 1609 standard is also known as Wireless Access in Vehicular Environments (WAVE) [IEEE 13]. WAVE deal with protocols in upper layers and multi-channel operations [Jiang 08, Uzcategui 09]. The IEEE 1609.4 trial standard [IEEE 13] of WAVE functions on the top of IEEE 802.11p in the MAC layer. IEEE 1609.4 trial standard focuses on multi-channel operations of a DSRC transceiver. Several automobile manufactures, research institutes and universities are involved in many research projects that focuses on creating standards for V2V or V2I communication, frequency band allocations, analyzing performance of different protocols in different layer, and verifying performance of protocols in field operational tests. However, the widespread implementation of new technologies of VANETs poses various technical issues, concerning network architecture, routing, medium access, topology, channel modeling, security, performance, and applications definitions. These issues posses challenges in the implementation of safety and non safety applications in VANETs.

1.1 Motivation

VANETs are intended to reduce traffic accidents in the road using different safety applications. Some of the safety applications are intersection collision warning, cooperative collision avoidance etc. VANETs are also used for several non-safety applications such as traffic information management to gaming applications. However, it has been shown that transmission and access collisions in WAVE increases, when traffic density increases. Moreover, VANETs based on WAVE does not support high reliability, predictability and scalability required for safety applications.

One of the peculiar characteristics of VANETs is the rapidly changing and dense network topology, resulting from the high speed and congestion of vehicles on the road [Dietzel 14]. This dynamic and dense topology causes channel access collisions, channel congestion, and the hidden terminal problem. Clustering the vehicles into smaller groups can make the network appear smaller and more stable in the view of each vehicle [Mammu 13]. Some of the advantages of clustering is to reduce the number of vehicles interacting with each other. Clustering the vehicles based on the different characteristics of VANETs such as similar speeds, and the direction of travelling can reduce the variation of speed between the vehicles in the same cluster and increase the communication link period between members in the cluster. Moreover, this type of clustering will increase the intra-cluster stability. Furthermore, clusters elect a member to be a leader for coordinating the communication between themselves.

Clustering in VANETs requires selecting a stable CH that produces a stable cluster. Dynamic mobility of vehicles affect the stability of the cluster structure. New vehicles joining the cluster and other vehicles leaving the cluster change topology of the cluster. Having a simple clustering scheme in forming and maintaining clusters will save a significant amount of time and increase the efficiency of the channel bandwidth. Since all vehicles in VANETs need to send safety messages with fewer or no transmission and access collisions. The neighbouring vehicles must receive these messages within the time limit and with high reliability. The dynamic topology of VANETs requires a high frequency of periodic messages to keep track of speed, position and direction of neighbouring vehicles. All of this flooding of periodic messages leads to severe channel access collisions and channel congestion, which can be alleviated by a proper MAC and data aggregation protocol [Mammu 13, Jiru 15]. MAC protocol is responsible for the channel access in VANETs. MAC protocol determines vehicles right to access the shared wireless medium. One of the examples of MAC protocol is to use centralized MAC protocol based on the TDMA scheme which distributes time slots to all vehicles that are currently within the transmission radius. In these protocols, a centralized coordinator is required to allocate slots to nearby vehicles.

MAC protocol for safety applications must meet the scalability, reliability, predictability and fairness requirements. Scalability [Casavant 94] is defined as the maximum number of vehicles that can be given a time slot in a frame without suffering a noticeable loss of performance. The predictability of safety message means the delay encountered between generation and delivery of the message, it does not necessarily need to be low to meet the requirements of safety applications. A predictable channel access delay means even in highly dense scenarios, it is able to meet the delay requirements of particular applications. Moreover, vehicles can re schedule and re order messages to meet the predictable delay requirements of different applications.

Reliability is the ability of MAC protocol to organize packet transmissions to reduce transmission collisions and channel interference between neighbouring vehicles. Lastly, fairness means the equal probability for all vehicles in the network to access the medium for transmitting the same data type. Furthermore, if the medium access delay and the interference during medium access vary greatly between transmissions, the MAC layer should try to spread these variations among vehicles as fair as possible.

Safety messages will demand high reliability and low delay, whereas non-vital road and weather information will be tolerant to longer delays. These different data types necessitate QoS provisioning, which can be achieved by a clustered network [Mammu 15]. TDMA is a technique that can be used to assign unique time slots to each vehicle in the cluster. The aim of any slot allocation scheme is to make the process of slot allocation simple and fast. Also, non-safety messages need to be delivered within the predictable delay. In order to deliver these messages, the channel congestion during busy hours of traffic is reduced by aggregating data in certain vehicles or stations and forwarding these data to the control center.

1.2 Hypothesis

Mobility aware clustering in VANETs will improve the scalability of VANETs in highly dense environments. Electing CH based on the relative mobility, and distance between neighbours will increase the CH lifetime. Clusters formed based on the direction of movement will reduce overhead due to cluster maintenance. The performance of CH election, cluster formation, and cluster maintenance algorithm will be measured by the CH lifetime with variable speeds and densities. Cluster based MAC protocol using the TDMA technique will improve reliability, fairness, and predictability of messages in VANETs. Performance of MAC protocol will be analyzed using delivery delay, access collisions and packet delivery rate. Cluster based semantic data aggregation based on the density of vehicles will reduce the data load in the channel. The performance of data aggregation scheme is measured using the precision of different metrics and data aggregation levels in high densities.

1.3 Objectives

The main objective of this work is to design and implement a clustering algorithm that elects a stable CH and form a stable cluster. In order, to achieve a stable cluster structure with minimum cluster reconfiguration, I encounter three important problems. These problems are cluster formation, cluster maintenance, and intra-cluster and inter-cluster communications. In this work, I propose three different CH election algorithms. Two of the CH algorithms are combined with MAC protocol. These two protocols are based on the integration of clusters and scheme for channel access, where CH allocates time slot to all CMs for accessing the channel.

I have proposed three different cluster based MAC protocols to achieve these goals. First protocol is based on contention based (CSMA/CA) and contention free (TDMA) channel access. Second protocol is based on TDMA, where slot allocation is done based on the priority of vehicles. The new slot allocation scheme should be predictable, fairer and reliable. Lastly, DA-CMAC protocol is a multi channel MAC protocol based on TDMA, where the time period is divided into CCP and SCP. In this work, I also propose a dynamic TDMA slot assignment scheme for cluster based VANETs. In this scheme, CH uses local IDs to manage the collision-free intra-cluster communications.

Lastly, this thesis proposes CBSDA scheme that uses an aggregation framework module. The framework uses a tree-based data structure to aggregate and fuse vehicular data. Using this data structure, different levels of data aggregation can be introduced to achieve the objective of adaptive data aggregation. The general idea of these aggregation levels is that the data structure is reduced in size by increasing aggregation levels. The aggregation level is changed dynamically when the density of vehicle changes. The aggregation level increases when the number of vehicle in a cluster increases. This result in smaller data structure, which is then forwarded to the next CH. Moreover, aggregation level increases and decreases adaptively based on the density of vehicles.

1.4 Research contributions

The main goal of this thesis is to develop a cluster based MAC and data aggregation scheme that can support large scale VANETs by ensuring message reliability, predictability, fairness, and scalability for safety and traffic efficiency applications. This is accomplished through three tasks, which are the main contributions of this work:

- Three different cluster formation algorithms are proposed based on direction of movement, position of vehicles and transmission range. Three different CH election algorithms are proposed based on speed deviation, the number of connections, distance between RSUs and vehicles, distance between start of the cluster segment and distance between the neighbours on the road. During the cluster formation process, the cluster members will be grouped based on the direction of travel and assigned local IDs by the CH. The design and implementation of CH election and cluster formation algorithms shows that fewer CH changes occur compared to existing algorithms. This lead to minimum overhead thereby resulting in fewer re-clustering and delivers an efficient hierarchical network topology. The cluster maintenance algorithm proposed handles the topology changes. The proposed algorithm takes advantage of the local IDs that are assigned in my cluster formation algorithm. In cluster maintenance algorithm, different events that can lead to merging, joining and leaving the cluster are considered. Details will be discussed in Chapter 3.

- Three cluster-based MAC protocols to coordinate intra-cluster communications. First proposed protocol is a hybrid protocol (D-CBM CSMA) that is a combination of both contention based and contention free channel access. Secondly, the MAC protocol (D-CBM TDMA) is based on TDMA slot allocation, where the CH allocates slots to CMs based on priority (future position, speed etc). Lastly, multi channel MAC protocol (DA-CMAC) where the period is divided into SCP and CCP. In other words, the time cycle is divided into two different intervals, CCH Interval and SCH Interval as with WAVE. In DA-CMAC, the SCP and CCP are divided into slots and mini slots. The slots are allocated to vehicles based on the local IDs. Details will be discussed in Chapter 4.

- A cluster based semantic data aggregation scheme to reduce the channel congestion. The aggregation increases adaptively when the number of vehicles in the cluster increases. A tree data structure is used for storing the data in each CHs or RSUs. The data structure grows when the vehicle density increases and data structure reduces when the vehicle density decreases. Data fusion is performed to reduce the data load. Three different decision schemes are proposed to increase the precision of data. Details will be discussed in Chapter 5.

1.5 Outline

The rest of the thesis is organized as follows. A literature review of the current standards of IEEE and ETSI for VANETs has been given in Chapter 2. Moreover, Chapter 2 presents the recently proposed protocols for clustering, MAC and data aggregation in VANETs. Chapter 3 introduces the CH election, cluster formation and cluster formation protocols to increase the scalability of VANETs. Chapter 4 illustrates MAC protocols for increasing the reliability and predictability of VANETs for safety applications. A cluster based data aggregation scheme is illustrated in Chapter 5. Chapter 6 presents an evaluation of the proposed algorithms and protocols by using extensive simulations. The results are studied and analyzed carefully. Chapter 7contains the concluding remarks and direction of future research.

2

State of the art

This section provides an overview of various standards for VANETs and previous work related to the field of clustering, MAC layer and data aggregation. Some of these schemes are developed for either VANETs or ad hoc networks. Such schemes are considered in this thesis because of their similarity to VANETs. In the following sub sections, I will outline the characteristics of VANETs, as well as the type of messages in VANETs. Then I will give a background of IEEE standards for MAC protocols for VANETs. I also will explain the clustering approaches and different MAC protocols for VANETs. In the end, I will review some data aggregation schemes for VANETs. .

2.1 VANET fundamentals

VANETs interconnect vehicles using a wireless channel to increase road safety and traffic efficiency. A car in a traffic jam, for instance, could warn approaching vehicles to increase road safety. In contrast, cars leaving the traffic jam could notify approaching vehicles about how much delay to expect. These vehicles could then check for an alternate route which increases traffic efficiency. VANETs pose some unique requirements and challenges to provide a sufficient quality of service.

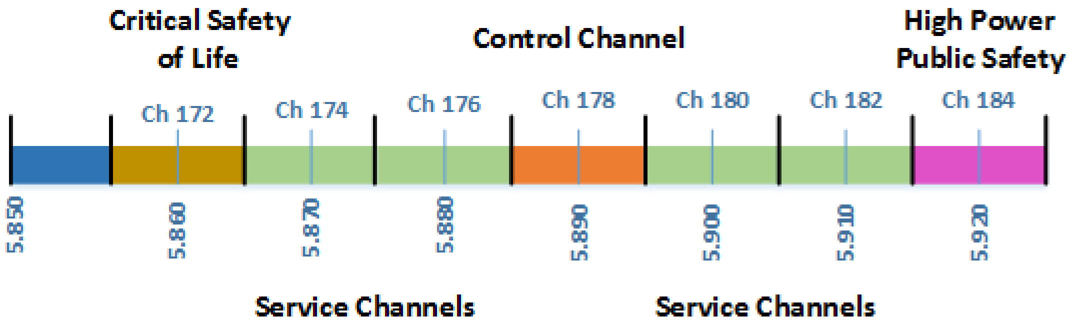


Figure 2.1: IEEE spectrum allocation for VANETs

2.1.1 Architectures

Different architectures for VANETs have been proposed and standardized. Two important of these are the IEEE Standards for VANETs known as the WAVE stack and the European ITS Station Reference Architecture put forward by the ETSI.

2.1.1.1 IEEE 1609 WAVE standards for VANETs

In the US, the FCC has allocated 75 MHz of spectrum at 5.9 GHz for DSRC [IEEE 10] as shown in Figure 2.1, which provides high-speed communication between the vehicles and RSUs. DSRC is divided into 7 channels, each 10 MHz wide, as shown in Figure 2.1. Channel 178 is the CCH, which is used for beacon messages, event-driven emergency messages, and service advertisements. The remaining six SCHs support non-safety applications provided by RSUs. The IEEE has completed the 1609 family of standards for the WAVE standard [IEEE 13] for vehicular communications. WAVE family consist of following standards, as shown in Figure 2.2.

- IEEE 1609.1 defines the various interfaces and services for Resource Manager application of WAVE[IEE 06].
- IEEE 1609.2 presents different formats of secure messages and their processing [IE 06].
- IEEE 1609.3 specifies protocols used for addressing and routing in both transport and network layer. This standard support secure WAVE information exchange [IEEE 07].

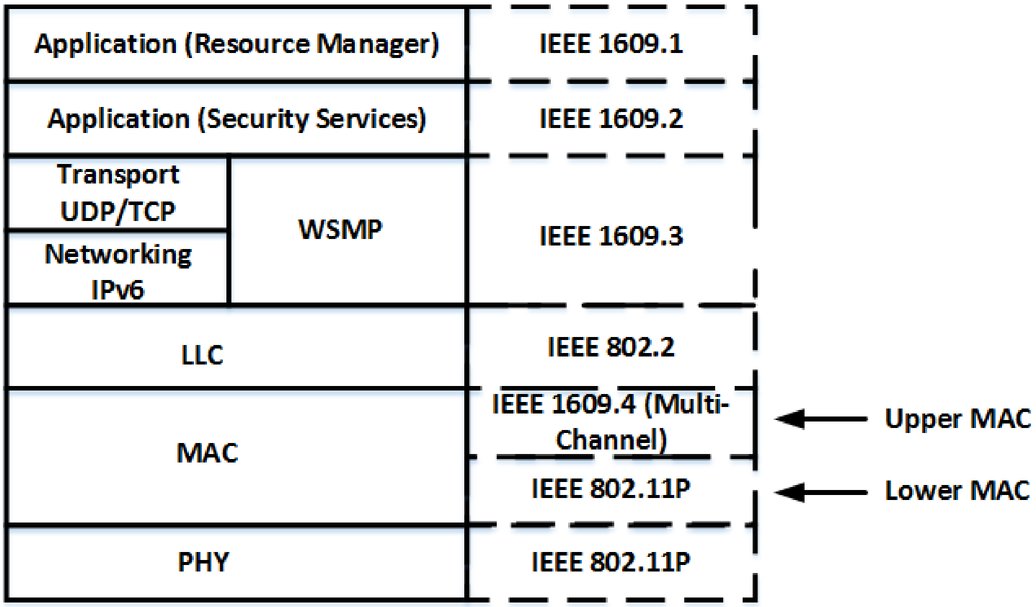


Figure 2.2: IEEE WAVE stack

- IEEE 1609.4 defines MAC and PHY layers [IEEE 06], which are based on IEEE 802.11. The main aim of the thesis is to improve this standard.

The IEEE 1609.4 trial standard [IEEE 06] from family of standards of WAVE operates above the IEEE 802.11p in the MAC layer. Moreover, the main function of the IEEE 1609.4 trail standard is to deal with multi-channel operations of DSRC transceiver. The protocol stack of WAVE is shown in Figure 2.2.

The multi-channel operations have a synchronous period of 100 msec that are divided into two equal interval length of 50 msec for CCHI and SCHI. These are separated each other by a guard interval, as shown in Figure 2.3. During the CCHI, all transceivers listen to updates from neighbours and RSUs or transmit the broadcast messages in the CCH. Similarly, during the SCHI vehicles may listen to the corresponding SCH of their need depending on the services required. The SI of 100 msec is designed based on the latency requirements of traffic safety applications. This enables the vehicles to transmit at least 10 event driven safety updates per second. It also defines a GI at the start of each CCHI and SCHI. The purpose of the GI is to account for the channel switching time. At present, the value of the GI

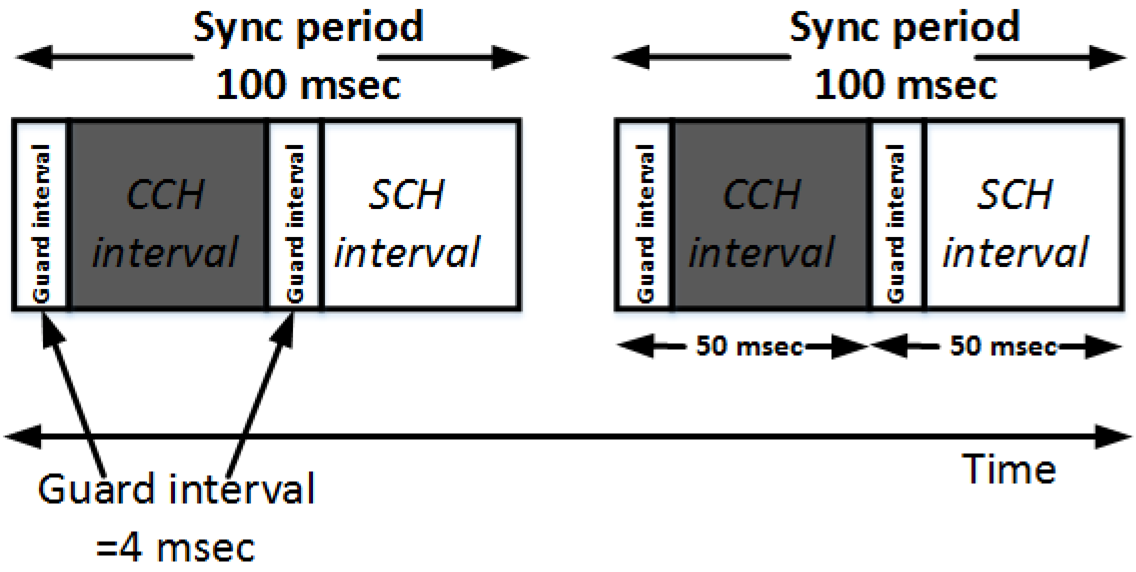


Figure 2.3: Division of time into CCH intervals and SCH intervals

is from 4 to 6 μsec, which is the time overhead for tuning a transceiver from one channel to another channel.

2.1.1.2 **ETSI reference architecture**

The ETSI standardized a communication architecture for ITS as European Norm (EN 302 665). The norm addresses a machine running an ITS module with this architecture as ITS Station. Such a station might be a smart phone, a vehicle, a RSU, or a central control center. An overview of the architecture is shown in Figure 2.4. The ETSI identified three domains for vehicular communication as shown in Figure 2.4. These domains are defined as follows:

- In-vehicle section The in-vehicle section covers the communication inside the vehicle. This communication can be between different on-board systems inside the vehicles or between on-boards systems and different portable devices such as smart phones, or tablets. The ETSI norm defines a personal ITS subsystem that include the use of wearable and portable devices. Moreover, ETSI norm defines a vehicle subsystem that deals with inter-vehicle communication.

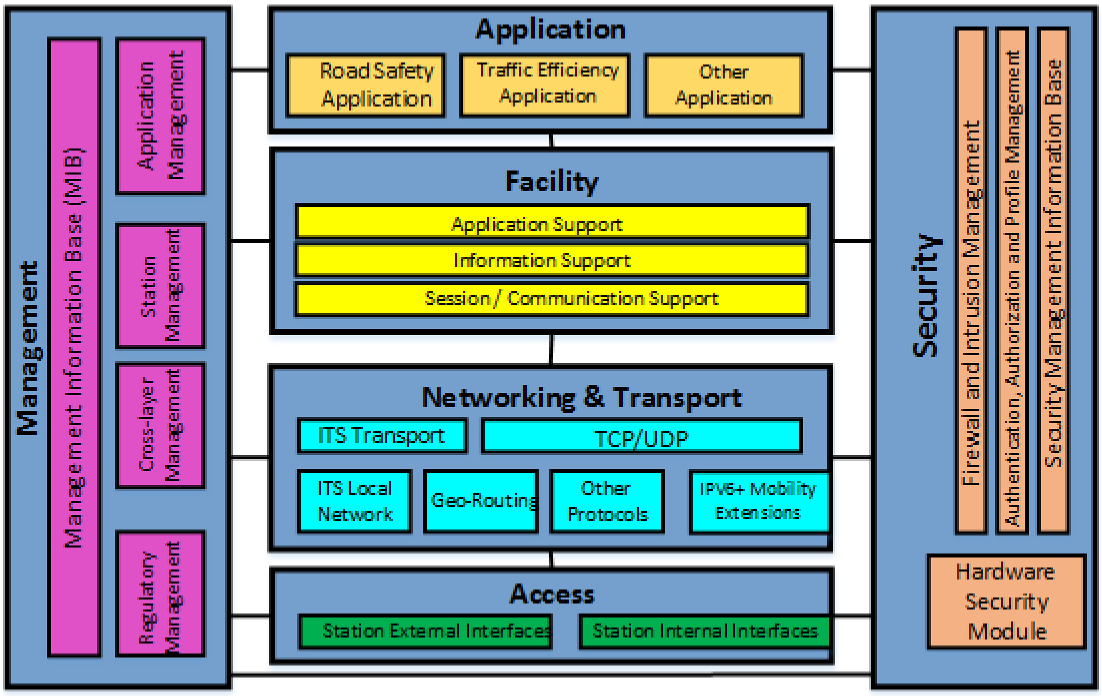


Figure 2.4: European ITS station reference architecture (ETSI)

- Ad hoc section Ad hoc section specifies the V2V and vehicles to environment communication. The environment constitutes of different systems such as RSUs, traffic lights, and traffic sensors.

- Infrastructure section The infrastructure section specifies the advanced functionality like the connection to a certification authority and can provide reliable access to cellular networks. This section is called central subsystem in the ETSI norm.

The architecture can be classified into three parts. First part is a protocol stack like the OSI model. Access layer is the lower layer of the ETSI model. Moreover, above the access layer is the layer for networking and transport which is followed by a facility layer. Topmost layer of the protocol stack is the application layer. The second part of the ETSI model is a component called management entity. This management entity deals with all management tasks for all layers of the protocol stack. Some of the management tasks, for instance are the installation and configuration of applications, congestion control, and advertisements of various services.

The third part of the ETSI model is the security component depicted on the right side of the Figure 2.4. It ensures security to all interfaces of all layers of the protocol stack. Security entity consist of various modules such as encryption, firewall and authentication protocols.

2.1.2 Characteristics of VANETs

The peculiar characteristics of VANETs make their qualitative and quantitative analysis really significant, especially when designing deterministic MAC protocols. VANETs are one of the special class of MANETs, they have different number of peculiar characteristics that makes many protocols designed for MANETs incompatible for VANETs [Blum 04]. Some of the VANETs characteristics that influence the design of a deterministic MAC protocol are:

- Number of vehicles: The vehicle density of a VANET is dynamic. The density may be lower in countryside or larger during busy hour of day in a urban area. MAC protocol should be designed to cope with both the scenarios. The main problem in countryside is communication disconnection due to fewer number of vehicles, while scalability is the main problem in high density areas.

- High mobility of vehicles: Vehicles in a VANET can travel at very high speeds (120km/h), which might lead to short communication period between vehicles. If one vehicle is travelling at a very high speed (120 km/h) and connected to a vehicle that is travelling at a very low speed (40 km/h), the lifetime of the communication link will be short.

- Predictable topology (to some extent): The structure of road is fixed and the movement of vehicles are confined to the road structure. So the movement patterns of VANETs are predicable.

- Critical latency requirement especially in cases of real time traffic safety applications. Hard real time constraints in active traffic safety applications.

- Rapidly changing network topology: Due to high speed of vehicles, the network topology in VANETs changes very rapidly. The MAC protocol should be designed in such a way that it can cope with the changing topology without any disruption.

- Position information: Position messages can be provided by having a GPS receiver on board. Having such position information for communications not only can reduce delivery latency of periodic messages but can increase delivery rate of messages.
- RSU support: Unlike most MANETs, VANETs can take advantage of fixed RSU on the roads. This could improve the performance of centralized MAC protocols.
- Efficiency of battery: The OBUs are powered by battery of vehicles so the energy problem in MANETs is not applicable in VANETs. This enables OBUs to be equipped with vehicles to have better computation resources.
- Various communication environments Urban, sub-Urban etc.

2.1.3 **Different messages in VANETs**

In VANETs, three different types of messages are used for communication between V2V and V2I. These messages are periodic messages, event-driven messages, and informational messages. These messages have different priorities but they share the same bandwidth. Event-driven messages have higher priority compared to other two messages.

- Periodic Messages: These messages are also known as CAMs in ITS-G5 standard. CAMs are generated to inform surrounding vehicles about the host vehicles current status, for example, speed, position, and direction. These messages are used to build up the Neighbourhood Table and posses a generic structure. Moreover, information in CAM messages is important to all neighbouring vehicles of the sender. These CAMs need to be broadcasted frequently to update the neighbour table. However, the frequent updates in a traffic accident area can lead to broadcast storm problem, This might lead to contention, transmission collisions, and inefficient use of the medium.

- Event-driven messages: These messages are also known as DENMs in ITS-G5 standard. DENMs are emergency messages sent to neighbouring vehicles about a possible collision or traffic accident that have been occurred. This type of messages has a very high priority. Many of the active safety applications use these messages to notify the drivers about the unsafe situations. Some of the applications that use these messages are collision avoidance systems, and intersection warning systems. The main parameters of these messages are the reliability and predictability. The transmitter of these messages needs to make sure that the intended vehicles have received them correctly and well in time.

- Informational messages: These messages are non-safety application messages. Some of the informational messages are SPAT and SAM. These messages enhance the driving conditions of driver and increases the passenger comfort. An example of enhancing driving condition is by use of SPAT messages. This type of message enable optimized flow control on traffic light intersections, RSUs frequently broadcast the remaining green light phase for each lane. Another type of message for enhancing passenger comfort is by facilitating internet access to the vehicles. Moreover, this type of messages does not need high priority, but it may require high data rate.

2.2 Different solutions for clustering in VANETs

Clustering in VANETs means dividing the vehicles into small groups based on some common characteristics such as velocity, vehicle destination, travelling direction, vehicle priority. etc. Clusters can avoid flooding in the network and this is the reason why they are so important. The clustering protocols provide following three advantages compared to other protocols [Tomar 13].

- Spatial reuse of network resources.

- Emergence of a virtual backbone.

- Improvement of network stability and scalability from the point of view of a regular CMs.

Protocols	CH election	Cluster Formation	MAC	Scenario	Simulator	Pros	Cons
HCA [Dror 11]	Maximum number of messages received.	4-hops, the maximal distance between a CH and any CM is 2 hops.	TDMA	City	OMNeT++, SUMO	End to End delay is reduced.	Overhead and packet loss is increased due to inter cluster interferences. Does not consider the direction of movement which decreases cluster stability and CH duration.
Zaydoun [Rawashdeh 08]	Nearer to middle of the cluster.	Not Specified	TDMA	City	C++ with graphical interface.	This method leverages contention-free and contention-based MAC to support the different requirements of safety and non safety messages.	Not suitable for high density scenarios and high overhead.
Xi Zhang [Zhang 06]	No reception of a message longer than a particular time units from a CH, then it elects itself as CH.	Received signal strength $>$ threshold , joins the cluster.	TDMA in CMs-CHs, CSMA/CA CHs-CHs	Highway	Simone 2000	Reduces data-congestion and supports QoS for real time delivery of safety messages.	High overhead and complex algorithm, dependence on two radios per vehicle.
CBMAC [Gunter 07]	CH election is based on waiting period of HELLO messages to neighbours.	From undecided state to CM based on reception of one CH messages.	TDMA	City	I-V Communication Based on Traffic Modeling.	Minimises the hidden terminal problem.	Does not select a stable CH during initial CH election.
RCM [Lai 11]	No CH	Vehicles are classified based on geographical area. It is assigned to different channel pools.	TDMA	Highway	A. law el al	Reduced contention by dividing area into different clusters. The throughput is increased.	Not consider specify the scenario. In sparse condition lot of bandwidth and slot remain unused.
TC-MAC [Almalag 12]	Based on lane weight, average distance, maximum number of neighbours, and average distance level. CMs on straight lane elected as CH	Not specified	TDMA	Highway	Ns-3	Improves channel utilization, network scalability, avoids hidden terminal problem, decreases collisions and packet drops.	Cannot be used for safety applications, it is delay intolerant, not evaluated the delay and throughput.
CF-IVC [Kayis 07]	Node is elected as CH by random after relaying one packet to ordinary node.	According to speed. Three different speed groups.	CDMA for Intra, MCS-CDMA for Inter Cluster.	Not simulated	Not simulated	Avoids data collisions.	It neglects any condition that might affect the maximum speed achievable by the vehicle nodes.

Table 2.1: Comparison of cluster based MAC protocols.

Cluster based approaches may be the only viable solutions for supporting high density VANETs [Little 05]. Recently, a lot of research has been focused on improving cluster based VANETs. Most of the proposed papers were on developing cluster based MAC [Bononi 07, Su 07]. The cluster re-establishment, the CH re-election, the CMs joining and leaving frequently are some of the weaknesses in these algorithms.The comparison of recently proposed clustering protocols for VANETs are shown in Table 2.1.

HCA [Dror 11], is a well-known protocol based on clustering. HCA operation is divided into two sections, the formation of cluster and scheduling channel access. The protocol forms hierarchical clusters based on randomized algorithm with a communication radius of at most four hops. Vehicles in the protocol have four states, slave, cluster relay, initial state and CH. The operation of HCA protocol is divided into four parts. They are cluster relay selection, CH selection, synchronization and cluster formation, and cluster maintenance. Moreover, authors analyzed the performance of HCA protocol based on three parameters. The number of elements in the dominating set, time required for forming clusters, and time required for slot allocation to all vehicles inside the cluster. Authors in [Dror 11], do not consider the use of localization systems for identifying the positions, this reduces robustness in VANETs. Moreover, HCA is not suitable for real time applications.

Authors of [Rawashdeh 08] proposed a new cluster based hybrid MAC protocol for VANETs. This protocol is based on the centralized management of clusters. They consider CH to CH communication is not an efficient way to disseminate data. The best way to increase coverage of data transmission is by forwarding to the farthest possible node. The CH maintains two sorted lists. First list consist of all CMs whose location is less than the CH's location sorted in descending order. Second list consist of all CMs whose location is greater than the CH's location sorted in ascending order. In this protocol, CH is always in the middle of the cluster. The main aim of proposed protocol was to limit the number of clusters formed in the network. However, the CH elected need to be always at the middle of the cluster. This is practically impossible in city scenarios.

The authors in [Zhang 06], proposed an analytical model for measuring the end to end delay and packet delivery rate of safety messages. This model investigates the relation between end to end delay and delivery rate in terms of the contention window size. The vehicles within a nearby proximity form a cluster. This scheme is sub divided into three different protocols. The first protocol employs contention based MAC to perform cluster management tasks, such as joining and leaving a cluster and CH election. Second protocol is responsible for the exchange of safety messages and non real-time traffic messages. Third protocol utilizes the MMAC protocol to arbitrate the communications between the CH and CM vehicles within a given cluster. The vehicles in the network can be in four possible states QCH, QCM, CM, and CH. The QCH and QCM provide the system with fault-tolerance. However, the protocol needs an extra hardware for its operation, which will increase the operational cost and reduce the extensive applicability of the protocol.

In paper [Gunter 07], authors proposed a cluster based MAC protocol to reduce the effects of the hidden terminal problem. Vehicles operating on this protocol can take four states. They are Undecided, CM, Gateway and CH states. Initial state of a vehicle in the protocol is called an Undecided state. Vehicles in the Undecided state is not linked to any clusters in the network. An Undecided state vehicle can transform its status to CM, when it receives Hello messages from a CH vehicle. Moreover, CH changes its state to Gateway vehicle, if it receives another HELLO message from a CH vehicle. Furthermore, if two CHs come closer to each other and the distance between each other is less than the transmission range. One of the CH change its status from CH to CM depending upon the weight factor. The weighted factor is composed of connectivity, the mobility and the mean distance to the neighbours. However, the cluster stability decreases with the varying traffic density and degrades its performance in urban scenarios.

RCM [Lai 11] divide the entire network into a set of road segments. The bandwidth is divided into different wireless channel and each road segment is linked with one non-overlapping wireless channel pool. In RCM, there is no central controller for channel access. Low collision chance and high spatial re-usability are two important factors of RCM. Moreover, throughput is increased and channel contention is reduced due to low density of vehicles in each road segment. However, the method in [Lai 11] provides low channel efficiency in case of sparse traffic. The large channel reuse distance can reduce spatial re-usability.

In [Almalag 12], the vehicle with the highest value for a function based on the flow of the majority of traffic and using lane information is elected as CH. However, the changing direction of vehicles affects the stability of CH and creates constant cluster re-configurations.

Authors of [Venkata 11] proposed a cluster based protocol for monitoring the traffic on the road. In this protocol, CH is elected based on the direction of movement and distance. Each vehicle calculates the distance and if the distance is below a threshold distance. Each vehicle is added to the cluster. Moreover, this protocol calculates the number of vehicles coming towards the intersection and transmits this data to the traffic signal. Furthermore, this will help to control the traffic signal and reduce the amount of time spend on the road. CHs will send the message to nearest CH in the direction of destination. However, the density estimation mechanism operation is limited to the vehicles within the cluster.

MOBIC [Basu 01] is an ID based clustering protocol. In MOBIC, the CH is elected based on the minimum relative mobility among one hop neighbours. CH re-election happens, when two CHs are in the transmission radius of each other for a certain period of time. The nodes in the cluster are linked to the CH-ID. CM may join any other CH or it may elect itself as CH. However, higher overhead is created when few nodes move differently compared to other CMs in the cluster.

In [Wang 08], the authors proposed cluster formation technique based on the geographic position of the vehicle. Each cluster will elect a CH based on the priorities obtained from a hash function. The hash function depends upon the trip time of a vehicle. Moreover, the vehicles that travels longer distances are elected as CH. This reduces the CH re-election and increases CH stability. However, the stability of this protocol is not verified in traffic jams and sparse traffic conditions which are very often in day to day life. Furthermore, this is not well suited for city traffic, where there are lot of vehicles joining and leaving cluster rapidly. As new vehicles with longer trip duration reduces the priority of current Ch. This leads to frequent CH re-elections.

A position based clustering is proposed in [Jerbi 07], where cluster are formed based on the geographic position of vehicles. Authors divided the road into different cells and each cell has an anchor point. The closest vehicle to that anchor point is elected as the CH. The cluster re configuration happens, when the CH leaves the cell. However, this protocol fails to address the cluster maintenance and CH re-election due to the high mobility of vehicles.

In [Souza 10], the authors propose a novel beacon based clustering algorithm that organize the network nodes into different clusters. This algorithm introduces a new aggregate local mobility criterion which is used to decide upon cluster reorganization and incorporates a contention method to avoid triggering frequent reorganizations. In this mechanism instead of RSS, the position of the vehicle is used for calculating the weight associated to the vehicles. This algorithm is a beacon based aimed at prolonging the cluster lifetime in VANETs. It displays a better performance in terms of stability as compared to the previous method. However, since the vehicles are highly dynamic in nature the position of the vehicles change very fast and hence may induce computational overhead in calculating the weight associated with the vehicles.

A position based clustering protocol known as DGMA is proposed in [Zhang 08], where clusters are formed based on the mobility information which represent vehicles current position, cluster size, and cluster velocity. Moreover, each node updates its latest position frequently using update messages. If the newest position differs more than a threshold from the previous position it computes the direction (angle w.r.t. the coordinate system) and the speed (magnitude of the movement) and records it. Furthermore, each node calculates the TSD based on these parameters. A higher TSD means the vehicle has a larger set of neighbours with similar speed. DGMA is suited for scenarios where vehicles are travelling in groups. However, DGMA does not take into consideration the destination of the vehicles, and this parameter is very important to prolong cluster lifetime and cluster stability because vehicles with the same destination have the same route and can easily travel in clusters.

A cross layer protocol is proposed in [Salhi 09], where road is divided into different clusters and each cluster elects a CH. This protocol is based on geographical data collection and dissemination technique. In this protocol, each vehicle in the network is linked to a cluster depending upon its current location. However this protocol incurs more overheads for V2V and V2I communication. Thus its performance is affected based on the availability of an infrastructure.

VWCA [Daeinabi 11] is a cluster based protocol, where CH is elected based on weighted parameters. The parameters taken into consideration during CH election are the number of one hop neighbours, the direction of movement, the entropy, and the distrust value. VWCA works with an adaptive allocation of transmission range technique, where HELLO messages and density of traffic around vehicles are used to adaptively adjust the transmission range among them. However, they do not consider the destination of the vehicles as a determinant parameter to arrange clusters. A suitable solution to prolong the cluster lifetime, stability, fairness, avoid congestion and overhead considering the vehicular behavior is essential.

A variation from the position based approach is described in [Goonewardene 09] where a utility based cluster formation technique is used. In the utility function position and velocity close to a threshold, are used as the input parameters. The threshold is computed based on the previous available traffic statistics. A status message is periodically sent by all the neighbouring vehicles. After receiving this information, each vehicle chooses its CH based on the results produced by the utility function.

Goonewardene et al. [Goonewardene 09] have proposed a clustering scheme named RMAC for highly dynamic VANETs. RMAC selects optimal CHs based on relative node mobility metrics of speed, locations, and direction of travel. In [Goonewardene 09], each vehicle entering into the network collects the neighbour vehicles information, assuming precedence to each vehicle and polls each vehicle individually (according to precedence) to check whether it is CH or not and then joins the cluster. Also every vehicle in the network collects 2-hop neighbours information along with 1-hop neighbours information from the CH through periodic polling. RMAC employs a vehicle precedence algorithm to adaptively identify the nearby 1-hop neighbours and select optimal CHs based on relative vehicle mobility, locations of vehicles and direction of travelling. The main limitation of this approach is that the proposed algorithm requires each vehicle to keep sending out update information to neighbours which can introduce lots of communication overhead.

Adaptable Mobility-Aware Clustering Algorithm based on Destination in vehicular networks [Morales 11] takes into account the destination of the vehicles to arrange the clusters and implements an efficient message mechanism to respond in real time and avoid global re-clustering. There might be a problem with knowing the final destination a prior as drivers usually do not use navigation system for known routes. Cluster size is variable according to vehicle density, speed and required minimum bandwidth or QoS where parameters can be predefined or provided on the fly from vehicle sensors and application profiles.

Little and Agarwal [Little 05] describe directional propagation protocol , disseminates traffic and road conditions by using a combination of MANET and delay tolerant networking) methodologies. In [Little 05], the clusters are formed based on mobility metric and the signal power detected at the receiving vehicles on the same directed pathway. The RSS is used as criteria to assign weights to the vehicles and based on this weights the CHs are elected. Through such method this protocol helps in forming stable clusters. However, it does not consider the losses prevalent in the wireless channel. In practical scenario effects of multi-path fading are bound to affect the cluster formation method and thus the stability.

In [Tung 13], authors propose a cluster based protocol for intersection scenario in VANETs. This protocol is proposed to avoid collisions in intersections. Moreover, all vehicles initially entering the cluster region is elected as CH. The CH changes its status to CM, when a CH closer to intersection is discovered. Furthermore, CH remains in same status, until the last CM passes the intersection. In this protocol, intra cluster communication is performed using short range radio or near field communication. Inter cluster communication is performed using cellular networks. However, CH stability is reduced due to distance between vehicle to intersection and due to different directions of vehicles.

A affinity propagation based cluster formation algorithm is proposed by authors in [Shea 09], where cluster stability is increased by minimizing both speed deviations between neighbours and distance between CHs and its CMs. The affinity metric is based on two significant parameters that are shared between neighbouring vehicles. They are responsibility and availability parameters, where responsibility means how good is one vehicle to become exemplar and responsibility means the desire of a vehicle to to become exemplar. Moreover, stability of algorithm is increased by grouping vehicles that are travelling in same direction into one cluster.

In [Maslekar 09], authors proposed a clustering algorithm based on the direction of the vehicles in a given geographic region. This protocol is used to estimate the density of vehicles on a given road. Propagation function is used to decide if a packet is to be forwarded only if a vehicle finds another vehicle in front of it within the radio range and moving along the same direction. In the proposed solution the packet is forwarded to vehicles only if they are moving along the same direction. However, realistic representation of the network is especially important for VANETs.

In [Maslekar 11], a direction based algorithm is proposed for the urban scenario. According to the predefined path and the position of vehicles, clusters are formed before the road intersection. However, this algorithm can only be used in some special situations of VANETs, and does not consider the mobility of vehicles.

Different scenarios of VANETs present challenges for cluster creation and maintenance, there by reducing cluster stability and increasing cluster overhead. The highway, city and intersection scenario requires different characteristics for the election of CHs and for the formation of a cluster. For this reason, ideal clustering protocol should be designed in such a way that the cluster and CH stability is increased. Otherwise, the frequent reconfigurations will lead to higher overhead and degraded performance.

2.3 Different MAC protocols in VANETs

The lastly approved IEEE 802.11p standard for inter-vehicle communication [IEEE 10] defines two message types, periodic status updates and event-triggered warning messages, that are allowed to share the dedicated ITS control channel in the 5.9 GHz ITS frequency band. To access that shared wireless channel the standard employs CSMA/CA.

CSMA/CA is a contention based protocol, it solves the hidden node problem but exposed node problem still needs to be addressed. However, during a traffic accident there will be a huge concentration of vehicles in a particular area, this may lead to a longer contention period and broadcast storm. CSMA/CA cannot define an upper bound for the channel access delay during congestion. The comparison of recently proposed MAC protocols are shown in Table 2.2.

FCR algorithm [Kwon 02] is a contention based protocol proposed to reduce the waiting period of channel access after collision occurrence. The back off timer generated after collision is redistributed based on the active nodes and number of consecutive idle slots. The FCR algorithm uses a smaller contention window for

Protocols	**CFP/CBP**	**Pros**	**Cons**
CSMA/CA [Bianchi 96]	CBP	Little coordination is required, more robust to network changes and has lower overheads.	Low predictability, fairness, and throughput in high density. Slot allocation itself involves collisions and unbounded delay.
STDMA [Yu 13]	CFP	The frames are not synchronized. The number of times a vehicle transmits per frame is determined by its speed.	It is not suited to infotainment applications. A slot is wasted for every vehicle that does not need to transmit.
ADHOC MAC [Borgonovo 04]	CFP	Reliable in multi-hop broadcast service without the hidden terminal problem. Smaller number of relaying nodes than in flooding.	Suffers from throughput reduction due to node mobility
SDMA [Bana 01]	CFP	Robust and instant network initialization. Real-time and accurate address resolution. Highly scalable.	Poor efficiency since most of the cells are empty and suffers the location error problem. Synchronization required.
VeMAC [Omar 11]	CFP	Slots are divided into 3 groups, each group has vehicles moving in same directions. This increases the throughput that was affected by node mobility.	Slot wastage in case of sparse and dynamic traffic.
DMMAC [Lu 10]	CFP/CBP	Hybrid channel access of TDMA and CSMA/CA.	If a node falls out of range, the slot allocated will be unused.
FCR [Kwon 02]	CBP	Reduces transmission failures (due to packet collisions) and the idle slots due to backoff at each contention cycle. Fast backoff procedure.	Low fairness among nodes for channel access.
MAC-SCC [Li 06]	CBP	Reducing the frame collision probability and the bandwidth wasted during backoff.	Require extra hardware for both channels.
VeSOMAC [Yu 07]	CFP	Increased throughput in highways.	Throughput reduced in city when vehicles travel in different directions. Ignoring processing time or propagation delay is unrealistic.
CAH-MAC [Bharati 13]	CFP	Utilizing unreserved time slots for retransmission of a packet which failed to reach the target receiver due to a poor channel condition.	Limited number of vehicles can be handled with strict synchronization and large overhead.

Table 2.2: Comparison of various MAC protocols in VANETs

each station with successful packet transmission and reduces the back-off counter exponentially when a station detects a number of consecutive idle slots.

ADHOC MAC [Borgonovo 04] is a contention free protocol that reduces the hidden terminal problem by giving each vehicle a wide view of transmissions that happens in at least two hops. Additionally, ADHOC MAC is based on dynamic TDMA which is derived from RR-ALOHA for V2V communication. In ADHOC MAC, time slots allocated to various vehicles are grouped together to form virtual frames. In addition, each vehicle selects one slot from a basic channel, where the time slot will be repeated in consecutive frames. Moreover, no frame synchronization is required. However, when a vehicle exits the road there can be unused slots in consecutive frames and this leads to low channel utilization.

VeMAC [Omar 11] is a contention free protocol based on TDMA. VeMAC is a multichannel protocol that allocates time slots in control channel in decentralized way and in service channel in centralized way. Moreover, VeMAC eliminate the hidden terminal problem by allocating disjoint groups of time slots to vehicles travelling in different directions. However, it can't solve the problem of merging collision in the VANETs.

VeSOMAC [Yu 07] is a contention free protocol based on TDMA for V2V communication. VeSOMAC provides predictable delay and high reliability of packets. The slot allocation is based on the vehicles relative location and direction to reduce the delay for delivery of messages. The fast slot allocation is based on a new bitmap based in-band control method for exchanging control information. Moreover, this is well suited for VANETs which are highly dynamic in nature. However, there is no explanation how the newly arrived vehicles get time synchronization, since newly joined vehicle can have worst synchronization between all one hop neighbours. Moreover, the performance can degrade significantly in city scenario where vehicles joins and leaves very frequently.

A STDMA protocol [Yu 13] was proposed for predicable channel access in congestion scenario where number of nodes are more than the number of slots available. STDMA achieve predicable channel access by allocating same slots for nodes that are further away from each other. STDMA is based on distributed way of allocation of time slots. However, if the distance between the two vehicles which share the same time slots are not separated by the threshold distance, this could lead to interference and unbounded delay of packets.

SDMA [Bana 01] is based on discretization method where the entire network is divided in to small region units called segments. Moreover, a mapping function is used to allocate a set of time slots to a particular region and an assignment algorithm based on vehicle current position and region is used to allocate time slot for accessing the wireless channel. However, there will be low bandwidth utilization in sparse traffic.

CDMA [Watanabe 05] is proposed for VANETs due to its robustness against interference and noise. CDMA allocates pseudonoise (PN) for each vehicles in the network. This increases the reliability and scalability of VANETs. However, a lot of issues arise from how to allocate PN codes to different vehicles. Consider an accident scenario where there are large number of vehicles in a particular area. If each vehicle is allocated a unique PN code, the length of these codes will become extremely large, and throughput required for applications such as infotainment may not be attained.

DMMAC protocol [Lu 10] uses adaptive broadcasting for improving the reliability and predictability of messages. The frame is divided into two equal length control and service channel intervals. The control channel interval is further divided into equally-sized contention-free TDMA slots, and a contention-based reservation period. However, the paper do not discuss what happens if a vehicle disconnects from the network for a time period. The slot allocated to that vehicle will be unused, which lead to inefficient utilization of bandwidth. If all vehicles reallocate to the beginning of ABF, then the length of ABF decreases accordingly. Furthermore, the newly joined vehicles would not have enough time slots for transmission. Moreover, the vehicles can also contend for time slots in multiple channel, which could lead to collisions.

MAC-SCC [Brenner 97] is a multi-channel protocol with data and control channel. Each channel is allocated with a network allocation vector. The main purpose of using two channel is pre-schedule next packet transmission in the control channel. Moreover, sending or receiving happens only in one channel at any given time. Furthermore, the bandwidth partitioning between two channels is analyzed using statistical model, which shows 10% for the control channel and 90% for the data channel. However, MAC-SCC requires extra hardware for the operation of two channels, which will increase the hardware cost.

Due to the peculiar characteristics of different scenarios of VANETs MAC layer faces important issues. Some of the main issues are related to mobility, predictability, scalability, bandwidth efficiency, cost, and reliability. Mobility in MAC layer means how flexible it is to support vehicles that arrive and leave the network constantly. Predictability can be linked to delay bound and fairness. Predictability means the messages should arrive the destination within the time bound. Sometimes time bound delay cannot be achieved due to the low fairness for channel access among the vehicles in the network which can lead to less predictable messages. Scalability is need in scenarios such as traffic accidents when the number of vehicles in a region is high which can lead to either channel collisions or number of vehicles are larger than number of slots. Bandwidth efficiency is important for MAC in order to reduce number of unused slots and efficiently use it for retransmissions or by other vehicles in the network. Hardware cost is most important when considering two channel for transmission and reception. The optimal MAC protocol should consider all above parameters for ensuring timely and reliable data delivery of messages.

2.4 Data aggregation

One way to classify different data aggregation techniques is by how data is reduced during the fusion process. This can be done by classifying them into two aggregation schemes known as semantic and syntactic aggregation. A second way is to categorize a scheme by the time when data is aggregated through distinguishing in-node and in-network aggregation. Both categorization concepts are described in the following.

Data aggregation is divided into two different types semantic and syntactic aggregation [Picconi 06]. In syntactic aggregation, the data obtained from different sensors are aggregated under one header which helps to reduce the packet overhead. Moreover, this type of aggregation benefit when the size of the data is comparatively smaller than the packet header. Furthermore, syntactic aggregation is lossless aggregation where only header data is fused and no data from the sensor is fused. On the other hand, semantic aggregation fuse actual sensor data rather than the packet header. For instance, this type of aggregation is beneficial when the sensor data is comparatively larger than the packet header. For example, instead of sending the position of each vehicle, only the average position of all vehicles on a predefined road segment will be send. However, the data precision decreases but ensures data size reduction. In contrast to syntactic aggregation, semantic aggregation schemes often need some knowledge of the collected data to make useful fusion decisions.

The semantic and syntactic aggregations are classified based on how data is reduced in them. In contrast, the in-node and in-network aggregations are classified based on the time and place were data is aggregated. Aggregation performed within the vehicle is know as in-node aggregation. In-node aggregation only fuse single data records that are received by a node. However, in-network aggregation the data is aggregated when it is received, but also on the path from source to the control center node. Moreover, the vehicles can decide to aggregate the recently received data with its own previously received data.

2.4.1 Different data aggregation schemes

This section provides an overview of previous work related to the field of data aggregation. Some of the aggregation schemes presented in this section are developed either for VANETs or WSNs. In this section, tree-based aggregation, cluster-based and structure-free topology approaches will be examined.

A tree-based topology comprises of different number of nodes in different tiers. The tree structure consists of one root node at the top. This root node gathers all information from other nodes down in the tier. The root node is also know as the data sink node. The nodes in different tiers forwards the information collected to the upper nodes in the tier. Ultimately forwarding all information to the root node. During the transfer of information between the tiers, intermediate nodes can aggregate data before forwarding to upper layer, which usually represents the data sink. All other nodes in the tree are data sources, which forward their data towards the sink. During this process, intermediate nodes can cache and aggregate data they receive before forwarding it into the direction of the data sink.

Some of the well known data aggregation approaches based on tree topology are Directed Diffusion [Intanagonwiwat 03] and PEGASIS [Lindsey 02]. In these approaches, the data is aggregated in the lower tier and forwarded to the upper level tier and at the end received by the root node or the data sink. In PEGASIS, the node fuses its own data with the data received from its neighbouring node. The fused data contains the sensor data of node itself and its neighbours. This data is forwarded in the direction of the root node or data sink. In [Graffi 09, Graffi 08], authors proposed a data aggregation scheme based on tree topology for peer to peer system. This aggregation scheme is proposed to collect statistical data of the‘peer to peer system. Moreover, they introduced a management layer in the data aggregation scheme.

In [Madden 02], authors proposed a TAG scheme ad hoc sensors. TAG scheme uses tables to store data and the data aggregated using SQL based queries. TAG operates in two phases. In the initial phase, data sink node sends request for data from all sensors. In the second phase, intermediate nodes in the direction of data sink node selects which data to be fused. After deciding, then intermediate node fuse the data and forwards to the other nodes in the direction of sink node. The main aim of TAG is to reduce the number of data packets in the network.

The quantile digest (q-digest) aggregation scheme [Shrivastava 04] uses a histogram for data representation. This reduces the data size significantly while allowing executing complex functions on the data. The authors argue that storing fused data of basic fusion functions like the average, min, max, and sum only needs constant size. However, more complex operations, such as median, require more data. The q-digest approach provides a space efficient data structure to determine such complex values within a fixed error bound.

Cluster-based aggregation schemes form small groups of nodes know as clusters. Each cluster consist of a CH and CMs. CH may have special responsibility such as collecting data from other CMs. Some of the well known cluster based aggregation schemes are Cougar [Yao 02] and CASCADE [Ibrahim 08]. CH is either chosen randomly or by selecting the node with higher priority among all nodes in a cluster.

CASCADE [Ibrahim 08] is based on cluster topology and using syntactic data aggregation in VANETs. CASCADE uses lossless data aggregation, where all data are placed below single header. The main idea behind this aggregation is that to store only the difference of value to a reference point instead of storing absolute values. Moreover, this reduces the data size because less bits need to be reserved to describe the relative values, in most cases. CASCADE divides the road into different segments of a fixed length. All vehicles within one segment belong to the same cluster with a segment length of 126 meters. Cougar [Yao 02] use their cluster topology for routing. In contrast, CASCADE uses the structure for the data aggregation.

The adaptive aggregation scheme [Chen 07] focus on adjusting the temporal and spacial aggregation degree that they aim to keep at a optimal level. This optimum is chosen so that the quality of service is sufficient for the sink node while no resources are wasted by providing too much information. The authors proposed a two-level aggregation scheme. Firstly, the sensors gather data and aggregate it locally. These sensors can adjust their dissemination frequency to change the level of this temporal aggregation. In regards to the chosen frequency, the sensor nodes forward their aggregated data to a cluster head. Secondly, the cluster-head aggregates the data and forwards it to the data sink. During this step, the cluster-head can influence the spatial aggregation degree by adjusting the aggregation ratio. If insufficient data is gathered, the aggregation ratio is decreased.

Structure-free aggregation schemes require no predefined structure. Jiang et al. [Jiang 10] suggested to maintain statistical data of a density function that reflects the distribution of the gathered data instead of the actual data itself. Lochert et al. [Lochert 08] proposed a hierarchical landmark-based aggregation scheme. Each road-junction is a landmark which is assigned to a certain level. Landmarks of junctions at major roads assemble the highest level. Lower level landmarks stand for smaller streets. The focus of their work lies on travel time distribution. A car driving by a low level landmark can notify it about the current travel time from the previous landmark. The data of multiple lower layer landmarks can then recursively be aggregated to calculate an approximate travel time of two landmarks on a higher layer.

Dietzel et al. [Dietzel 10] proposed a structure-free aggregation framework for VANETs. They argued that the decision to combine two data records is very complex and strict decisions are often impossible due to insufficient information of the consequences of an aggregation. To overcome this problem, they suggested an aggregation scheme based on fuzzy set theory. In their aggregation scheme, input values are firstly translated into natural language. A velocity difference of two aggregation records could then have the value LOW and HIGH. This fuzzy logic reduces the complexity of the decision. A simple decision rule might only aggregate two data records when their difference in velocity and position is LOW. Otherwise, both records are maintained.

The SOTIS scheme [Wischhof 05] introduces an adaptive information dissemination approach for VANETs. In SOTIS, the road is divided into segments of fixed length. All information of vehicles in a segment is aggregated. Therefore, choosing a low segment length yields data with high precision. In contrast, choosing a large segment length leads to more aggregation and less load on the network channel. SOTIS does not specify any particular fusion mechanism. The authors only proposed the usage of a generic function that combines all data of a road segment. Vehicles disseminate their aggregated data periodically. The authors identified two methods to decrease the likeliness of overloading the capacity of the wireless channel. One of these methods is to change the transmission range of the wireless signal, while the second one is to change the dissemination period. SOTIS focuses on the latter. The system adjusts the dissemination period adaptively to prevent overloading the wireless channel. In their experiments, the authors of SOTIS used a starting period of five seconds. This value is then increased or decreased in certain situations.

Lochert et al. [Lochert 07] introduced a probabilistic aggregation scheme for VANETs. They use modified Flajolet-Martin sketches [Flajolet 85] as key data structure for their duplicate insensitive aggregation scheme. These modified sketches allow probabilistic counting, which limits them to representing only sums. Using this approach only counts, sums, and averages can be described for instance and not individual values like the maximum or minimum.

TrafficView [Nadeem 04] is a framework for data dissemination in VANETs. It describes two algorithms for data aggregation and focuses on the decision, which data to aggregate. Two approaches are proposed for this decision. In the first approach, each vehicle in TrafficView divides the road ahead into segments. Then, a ratio is assigned to each of the segments. This ratio describes the importance of a road segment. Choosing a high ratio for a road segment leads to less fusion and higher precision of the data. Thus, decreasing the ratio with increasing distance of the segment to the vehicle results in a more precise view of the close by traffic situation than further away. The second decision approach takes the costs of an aggregation into consideration.

The aggregation strategies described above use very different data structures and ideas for the decision of which data to fuse, how data should be fused, and for dissemination. Different data structures [Madden 02, Lochert 07, Ibrahim 08] have been proposed to store the aggregated data records. In TAG [Madden 02], for instance, a simple table is used to store the data. This is possible because aggregating two data records means fusing all its contained values in TAG. Thus, two rows of the table structure are merged. While TAG proposes to use the absolute values stored in a table, CASCADE [Ibrahim 08] suggests to store only relative values to a fix point. For that reason, nodes are clustered and the center or median of the cluster values is used as a fix point. A different data representation is used in Probabilistic Aggregation [Lochert 07], Quantil Digest [Shrivastava 04] and Parameter Based Aggregation [Jiang 10]. All these schemes store mathematical abstractions instead of storing the data itself. The major disadvantage of these data structures is that no individual element, such as a very slow car, can be identified in the data structure.

During the decision process, data records are identified for fusion. This is a central element of many aggregation schemes and several different approaches have been proposed. One strategy is to define some data groups and fuse all data within this group. Such a group can be defined by splitting the road into segments as suggested in SOTIS [Wischhof 05] and CASCADE [Ibrahim 08], or more freely as proposed in TAG [Madden 02]. In any case, grouping data by one certain metric has a major disadvantage. When grouping by road segments, for instance, important extreme values might be lost by fusing over all elements of the group.

A different strategy for the decision process is used by Quantil Digest [Shrivastava 04] and Probabilistic Aggregation [Lochert 07]. Both methods use mathematical abstractions to represent the data. For that reason, they do not consider a selective fusion. In contrast, they fuse all data into their data structure. This solution accepts an inherent information loss due to the mathematical abstraction. Furthermore they have the problem with the identification of single extreme values as mentioned before.

A third decision strategy is used by TrafficView [Nadeem 04]. Its decision component uses a cost function to identify the two items with least fusion costs. This function takes the distance of the vehicles and the number of vehicles represented by a data record into account. This idea of a cost function could solve the problem of single extreme values, since the cost of fusing such a car should be too high for aggregation. However, TrafficView falls short in considering other metrics in the cost function than the distance of the vehicles and the number of vehicles represented by a data record.

3

Cluster head election, cluster formation, and cluster maintenance

Clustering in VANETs means organizing the vehicles into small groups based on some common characteristics such as vehicle position, speed, and direction etc. However, cluster reconfigurations and CH changes are unavoidable in VANETS due to their dynamic nature. This characteristics of VANETs reduces the stability of cluster structure and increases the overhead due to cluster maintenance. Moreover, most important criterion for any clustering method in VANETs is to form stable clusters with minimum overhead. The stability is defined by long CH duration, and low rate of change of CHs. By clustering the vehicles into groups of minimum relative mobility. In this way, the clustering algorithm can improve the lifetime of the cluster and decrease the number of CH changes and the number of cluster re configurations. Most of the previous clustering techniques are unstable in high mobility environments, because they do not elect a stable CH during initial process of CH election.

In this chapter, I will present three different protocols. First, I will describe in detail the CH election of these protocols. Then cluster formation in these protocols. Lastly, I will describe the cluster maintenance in these protocols.

3.1 Cluster head election for all protocols

In this section, I will present three different types of cluster head election protocols designed to improve stability of the cluster in different scenarios. Stable clustering methods reduce the overhead of re-clustering and lead to an efficient hierarchical network topology. During the creation of VANET clusters, cluster members select one member to be the CH. Fewer CH changes result in a more stable cluster, to achieve this goal, cluster members must select a member that has the potential of longer CH lifetime than other cluster members. Different approaches discussed here aims to select a CH based on the relative speed, number of one hop neighbours, distance to one hop neighbours and distance to RSUs. The proposed protocols are based on the assumption that each vehicle knows its position using GPS.

Initial election of CHs and the total number of CHs in the network affects overall stability of the cluster. In order to improve the stability and optimize the total number of CHs, I model the network using graph theory. Moreover, the optimal number of CHs can be obtained from MDS and the problem of stable cluster formation can be solved using a minimum connected dominating set [Alon 86]. In this thesis, an undirected graph (G) is used to represent the VANETs. Furthermore, I aim to obtain a MDS in G as discussed below, and reduce the number of CH in VANETs. In the first part, I introduce the ideas and definitions of each which will be used throughout the thesis.

- Undirected graph G = (V,E), where V is a set of vehicles that are travelling in the same direction and $E \subseteq V \times V$ is a set of links of vehicles which are in each others communication range or whose distance between each other is less than the cluster radius L.

- A MDS (S) of a graph $G = (V, E)$ is a subset of V ($S \subseteq V$) such as each vehicle in S is in the transmission range of at least one vehicle in V.

- A CH is a member of the MDS. A cluster head organizes and schedules channel access for some members or at least one member in the set V .
- A Gateway Vehicle (GV) is a vehicle that has a direct link between two vehicles in the S.
- A cluster is a group of vehicles that are travelling in the same direction, whose distance between each other is less than or equal to L_r. Each cluster has at least one CH, and can have a set of GVs. A cluster can also be defined as subset of vehicles with the same CH.
- CH_j^l is the CH of the j^{th} cluster travelling in L direction and $CM_{j,v}^l$ is the v^{th} CM of j^{th} cluster travelling in L direction having j^{th} transmission time slot, for $v \in V$. Each vehicle in the cluster is allocated a unique transmission slot.

After initial election of the CH, all vehicles in the network are grouped into different clusters. The details of cluster formation will be discussed in Section 3.2. Moreover, the vehicles in the network take up different roles described in detail below.

- **Undecided state vehicle:**

  Undecided state is the initial role of all vehicles executing the algorithm.

- **Gateway Vehicle :**

  This is an initial state when the vehicle enters the network. A CM changes its state to GV when it receives more than one (**CJ**) message from two or more different CHs. GV has different capabilities as it can have time slots from the RSU in the free period and can transmit the information to RSUs. Information transmitted to RSU will be transmitted to the near by CHs of GV.

- **Cluster Member :**

  A CM is a vehicle that belongs to a particular cluster and it regularly transmits the data in its time slots to its registered CH.

- **Backup Cluster Head:**

  A Backup CH is a CM who has the next priority to be elected as CH, if the current CH loses its status. BCH regularly transmit the data to its CH like other CMs. When the current CH value is below threshold or current CH exit from the highway or cluster etc, a BCH assumes the role of CH.

- **Cluster Head :**

  A vehicle in this mode gathers information regarding speed, direction, lane, and location from its CMs. It allocates transmission slots to its CMs. CHs are responsible for gathering, aggregating and forwarding the data to the RSUs or sink node. Thus, they are responsible for conveying the complete information of its CMs. There are two types of communication for CH, intra and inter cluster communication. Intra cluster communication is between CH and CMs, while inter cluster communication is between CH and its adjacent or registered RSUs.

In this thesis, I have considered three different highway scenarios. Firstly, there are no RSUs along the highway and vehicles are travelling only in one direction. Secondly, I consider that RSUs are deployed along the highway and vehicles are travelling only in one direction. Lastly, I consider RSUs are deployed along the road and vehicles travelling in different directions. The CH election, cluster formation and scheduling depends on the environment. Due to low market penetration of vehicular communication, there can be road without RSUs. The protocol must be flexible enough to work in environment without RSUs.

3.1.1 Cluster head election for CBSDA protocol

In scenarios where there are no RSUs and vehicles travel only in one direction as shown in Figure 3.1. In this scenario, I propose CBSDA protocol. In CBSDA, the road is divided into different segments called clusters. Each cluster elects a CH considering the effect of speed deviation of all CMs, and distance from the start of the cluster during the process of CH election. The start of the cluster is considered here to increase the CH life time in the cluster. Firstly, the CMs listens to all CAM or periodic broadcasts from its one hop neighbours. These broadcast

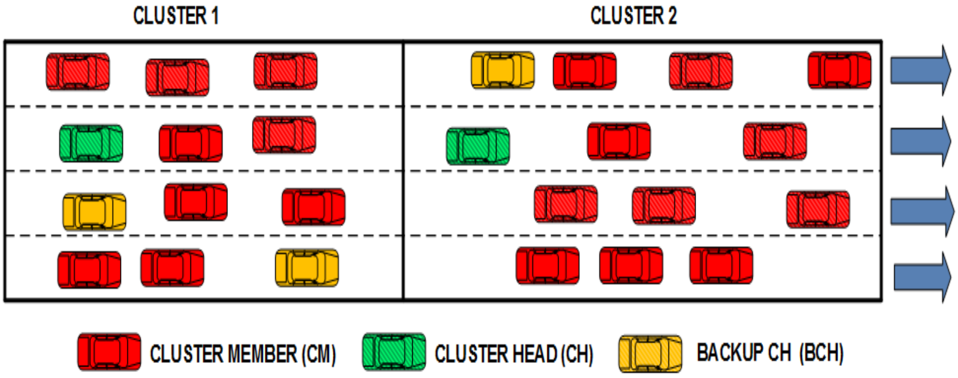


Figure 3.1: Highway scenario of CBSDA protocol

contains information like position, speed, direction of travel etc. Then each vehicle calculates its distance from itself to the start of the cluster using equation 3.1, where $D_{i,bend}$ is distance between vehicle i and the start of the cluster. D_{xbend} and D_{ybend} is x and y axis values of the center of the start of the cluster. D_{xi} and D_{yi} are the x and y axis values of the vehicle.

$$\triangle\ D_{i,bend} = \sqrt{|D_{xbend} - D_{xi}|^2 + |D_{ybend} - D_{yi}|^2} \tag{3.1}$$

Each vehicle calculates the relative speed between itself and its neighbours in the same cluster using equation 3.3, where S_{avg} represents the average speed of all its neighbours in the same cluster. $S_{i,nei}$ represents difference of speed of vehicle i and average speed of vehicle i neighbours. S_n is the speed of the n^{th} neighbour and n is the number of neighbours.

$$S_{avg} = \left(\frac{S_1 + S_2 + S_3 + ...S_n}{n} \right) \tag{3.2}$$

$$\triangle\ S_{i,nei} = |S_i - S_{avg}| \tag{3.3}$$

Each vehicle calculates its chance to become the CH using equation 3.4. In this equation α is the weigh factor, where higher value of α represents more weight to vehicles nearer to start of the cluster. The value of α can be anywhere between 0 and 1. R_{trm} is the maximum length of the cluster and S_{max} is the maximum speed allowed in the road. The vehicle with minimum value of equation 3.4 is elected as CH. The weigh factor is varied to elect the optimal CH that has higher lifetime.

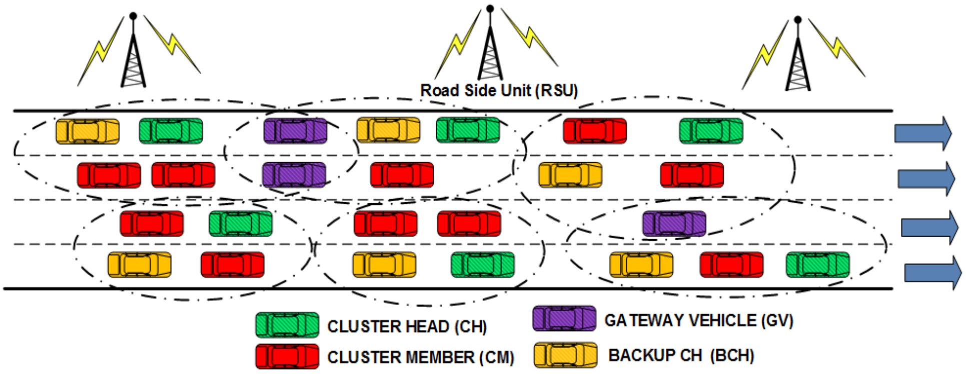


Figure 3.2: Highway scenario of D-CBM protocol

$$F_i = \alpha \times \frac{\triangle D_{i,bend}}{R_{trm}} - (1-\alpha) \times \frac{\triangle S_{i,nei}}{S_{max}} \tag{3.4}$$

3.1.2 Cluster head election for D-CBM protocol

The scenario where RSUs are deployed along the highway and vehicles are travelling only in one direction is shown in Figure 3.2. D-CBM is proposed for this scenario. In D-CBM, initial CHs in the VANET will be elected randomly. Clusters are formed after initial CH election using the minimum distance towards CH. The CH re-election happens when the current CH leaves the cluster and joins the other cluster. In this CH election, all vehicles entering the road are considered as GV and they transmit **RJR** messages to the RSU and register with the nearest RSU database. After waiting for a certain period of time t, RSU elects initial CHs at a random manner from all GVs while keeping a minimum distance between two adjacent CHs (it is considered to be two times the transmission range). Then CH sends **CJ** message to the all neighbouring GVs. The GVs who receive more than one **CJ** message from different CHs in a period of T time units remain in the same state. If a GV receives **CJ** message from a CH, then GV sends a **RJC** message, to the corresponding CH. The message send by GV includes ID of the sender and the corresponding ID of the CH. After receiving **RJC** message from GV, the corresponding CH sends an **ACK** to the GV. Then GV changes its state to CM. This increases the initial cluster stability of the network.

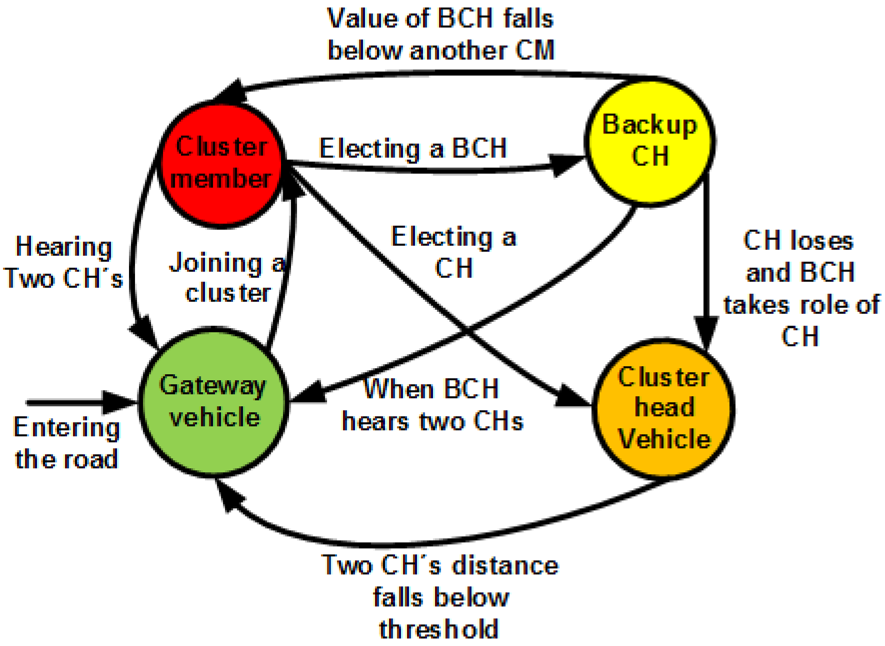


Figure 3.3: State transition of D-CBM protocol

A CM remains part of cluster as long as it periodically receives **CJ** messages from its CH every T time units. Furthermore, CM will change to GV, if it receives more than two consecutive **CJ** messages from different CHs at time $[t, t + 2T]$. Moreover, if CM does not receive **CJ** messages during $[t, t + 3T]$. This means the CM has either got disconnected from the current cluster and it switches to GV. The state transitions between each state of D-CBM is shown in Figure 3.3. A CH sends **CJ** messages to all its neighbours in a periodic interval of T time units. Upon the receipt of **CJ** message the GV sends a **RJC** message to CH, it sends an **ACK** back to the GV and adds the requesting vehicle to its cluster-member list. If CH does not hear from one of its CMs during $[t, t + 2T]$, it removes this CM from its cluster-member list. A CH switches to GV, if its cluster-member list becomes empty. If the distance between two CH is less than dismiss threshold distance D (equal to a transmission range) (i.e., both will receive **CJ** messages from each other), only the CH with fewer members is dismissed to reduce communication overheads. Each vehicle of the dismissed cluster turns to GV and launches a new registration stage to join other clusters. The threshold determines the rate of cluster reconfiguration, and

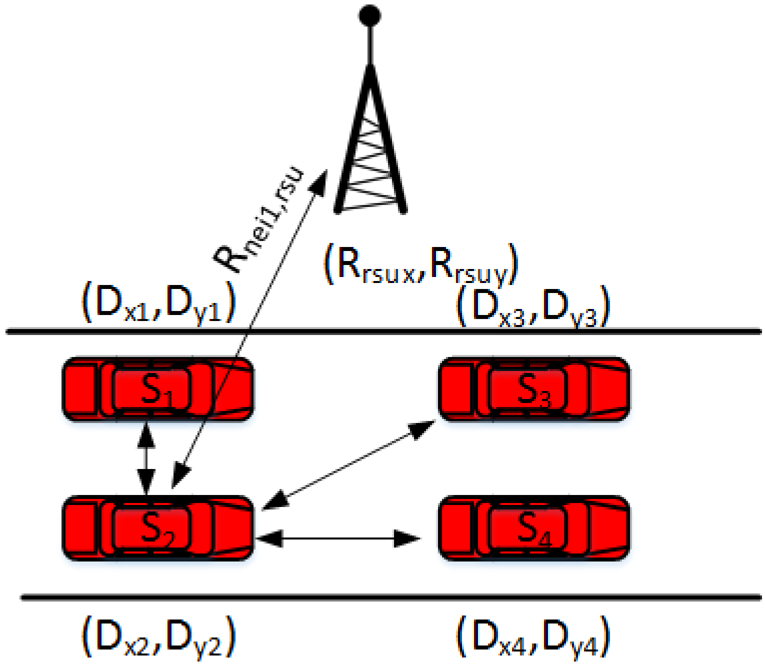


Figure 3.4: Dimensions in a cluster

also, depends on the communication range. It makes no sense when the distance between two CHs is less than the cluster diameter R_{trm}.

$$D_{xavg} = \left(\frac{D_{x1} + D_{x2} + D_{x3} + ...D_{xn}}{n}\right) \tag{3.5}$$

$$D_{yavg} = \left(\frac{D_{y1} + D_{y2} + D_{y3} + ...D_{yn}}{n}\right) \tag{3.6}$$

$$\triangle D_{i,neighbours} = \sqrt{|D_{xi} - D_{xavg}|^2 + |D_{yi} - D_{yavg}|^2} \tag{3.7}$$

After the initial selection of the CHs and formation of clusters, CH generates cluster-ID for the cluster, and periodically selects BCH from its CMs. In order to select BCH, each CM calculates the average values of the x-axis and y-axis of all neighbours using equations (3.5) and (3.6). Then, CM calculate the average distance between itself to its neighbours by using equation (3.7). Moreover, the equation (3.7) means how close are the neighbours to one vehicle. Taking account of this parameter will decrease the packet delay and increase the lifetime of CH. The dimensions represented are shown in Figure 3.4

$$S_{avg} = \left(\frac{S_1 + S_2 + S_3 + ...S_n}{n} \right) \tag{3.8}$$

$$\triangle\, S_{i,neighbours} = |S_i - S_{avg}| \tag{3.9}$$

The CHs elected based only on the distance can lose the connectivity very fast with their CMs if the CH has high or low speed compared to its CMs. In order to avoid electing CHs with high or low speed compared to its CMs , the eligibility of a vehicle should decrease quickly when its speed has big difference from the average speed of the CMs. Thus, a vehicle with large speed deviation is assigned lower priority. First, each CM calculates the average speed of all the neighbours using equation (3.8). Then, the CM calculates the deviation of speed and average speed of its neighbours using equation (3.9).

$$R_{nei,rsu} = \sqrt{|R_{xnei} - R_{xrsu}|^2 + |R_{ynei} - R_{yrsu}|^2} \tag{3.10}$$

$$R_{avgnei} = \left(\frac{R_{nei1,rsu} + R_{nei2,rsu} + ..R_{nein,rsu}}{n} \right) \tag{3.11}$$

$$R_i = \sqrt{|R_{xi} - R_{xrsu}|^2 + |R_{yi} - R_{yrsu}|^2} \tag{3.12}$$

$$\triangle\, R_{i,rnei} = |R_i - R_{avgnei}| \tag{3.13}$$

However, the CHs elected based on the smallest distance and speed deviation between its neighbours will not reduce the overall delay of packets from CM to the RSU. This can be reduced by electing a CH closer to the RSU. The equation (3.10) represents the distance between one of its CM and its registered RSU. Additionally, the equation (3.11) represents the average distance of all its neighbours to the registered RSU and equation (3.12) represents the distance between the current CM to its registered RSU. From the equation (3.13), the relative distance between CM and its all other neighbours registered to RSU is calculated. Additionally by using this parameter, the delivery time during CH to RSU transmission and vice-versa can be decreased. Moreover, to avoid elected CHs losing connectivity with their RSUs very soon, the eligibility of a vehicle should decrease quickly when its distance has big difference from the average distance. Thus, a vehicle with large distance deviation is assigned lower priority. The vehicle with lowest distance to RSU should be elected as CH.

The CM that has the minimum combination of average relative speed between the neighbours in the cluster, relative distance to RSUs in the highway compared with itself and its neighbours in the cluster, and the relative distance between itself and all its neighbours in a cluster, based on the equation (3.15) is selected as BCH. Indeed, when a CH is no longer in the cluster and BCH takes over, the cluster structure does not change but only the vehicle playing the role of CH. This allows a stable cluster architecture, with low overhead, and thus better performance. The CH and BCH are elected based on the equations (3.14) and (3.15).

Finally, the relative speed, the distance to the neighbours and distance to the RSU is combined using weighting factors W_1, W_2, W_3 in the equation (3.15). The values of W_1, W_2, W_3 can be varied according to the requirements and summation of these weights is equal to 1 from equation (3.14).

$$W_1 + W_2 + W_3 = 1 \tag{3.14}$$

$$F_i = \left(W_1 \times \frac{\triangle D_{i,neighbours}}{2R_{trm}}\right) + \left(W_2 \times \frac{\triangle S_{i,neighbours}}{S_{max}}\right) + \left(W_3 \times \frac{\triangle R_{i,rnei}}{R_{max}}\right) \tag{3.15}$$

Finally, all the CHs and CMs calculate the value of equation (3.15). The CH re-configuration happens due to several reasons, it can be when the CH goes above a particular threshold for the value of equation (3.15) (value considered here is 0.7), or CH leaves the cluster by taking a highway exit. If this occurs, then BCH takes up the role of CH. The previous CH can change its state to either CM or GV based on the reason behind CH re-configuration or it will eventually change to GV when it no longer receives **CJ** messages. The new CH will keep the same cluster-ID as the previous CH. Thus, the cluster structure will remain intact (CMs of the cluster do not have to reorganize in new clusters as the case of existing protocols); indeed, no re-clustering is needed and thus no re-clustering overhead is generated. The new CH will also select a new BCH. In addition, this re-election can avoid the re-clustering of vehicles and increases the stability of the cluster.

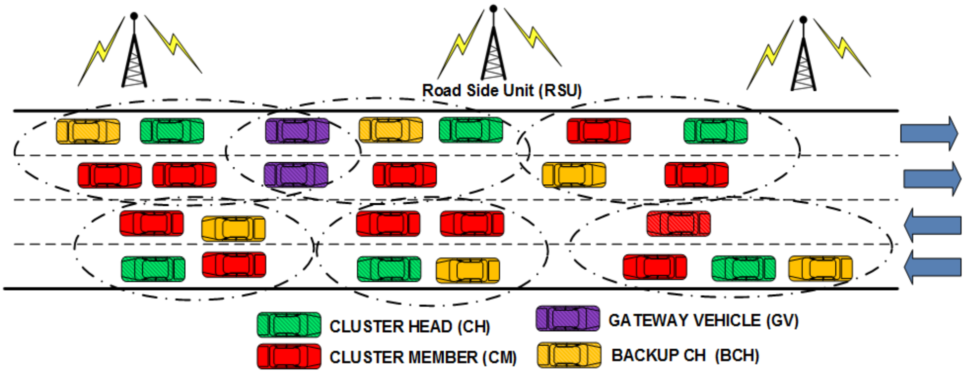


Figure 3.5: Highway scenario of DA-CMAC protocol

3.1.3 Cluster head election for DA-CMAC protocol

In the third scenario, I consider RSUs are deployed along the road and vehicles are travelling in different directions as shown in Figure 3.5. I proposed DA-CMAC protocol for this scenario. In DA-CMAC protocol, the vehicles are clustered based on the direction of movement, because by grouping vehicles travelling in the same direction increases the lifetime of members and reduces the overhead created due to frequent cluster reconfigurations. Vehicles in this DA-CMAC take different roles and each change from one role to other role takes place as shown in Figure 3.6. The states in this figure denote the relevant roles in the network (CH manages scheduling) and lines denote the transition from one state to another state. In this algorithm, I consider there are five different roles for vehicles. They are Undecided state, CH, CM, BCH and GVs. Undecided state is the initial role of all vehicles executing this algorithm. All CMs periodically sent status messages to its CH. The GVs are those vehicles that are linked to more than one CHs. These GVs are used to forward the control information or slot information of one cluster to another cluster. The CH manages and schedules the channel access for members of the cluster (A CH is a member of the minimum dominating set of S) . The CH maintains two lists of GVs, the GHL and the GTL.

The GTL set is a group of all GVs whose position is behind the CHs position sorted in descending order. The GHL list contains all gateway vehicles whose position is greater than the CHs position sorted in ascending order. The CH keeps updating these lists according to the topology change, in order to ease the formation

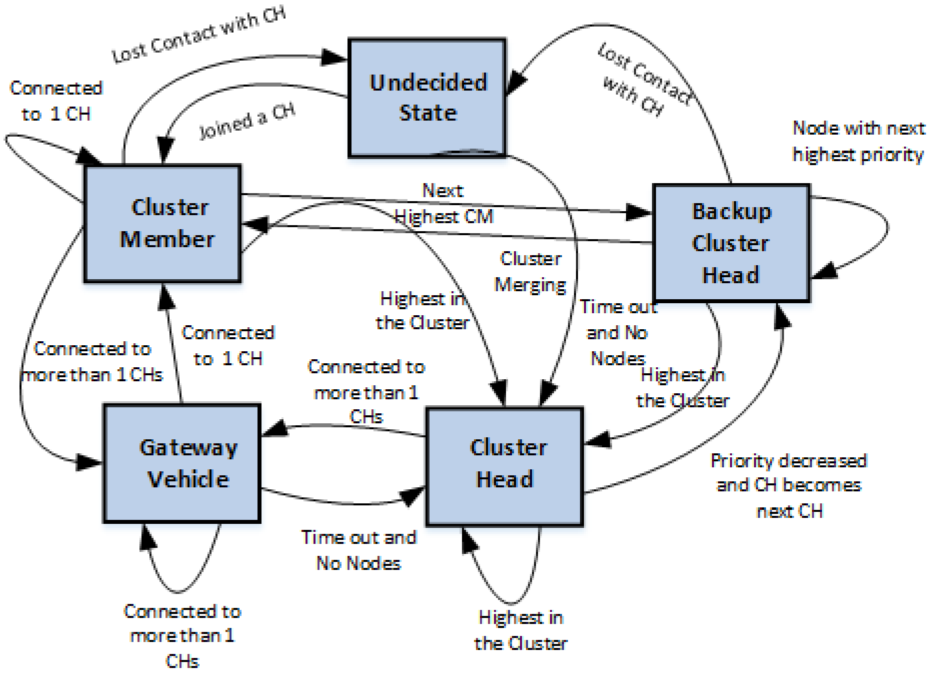


Figure 3.6: State transition of DA-CMAC protocol

of clusters, in which the maximal distance from a CH to any other vehicle in its cluster should be less than or equal to L_r. When the distance between two CHs is detected to be less than or equal to a predetermined threshold, $D(D \leq L_r)$, the backup CH will take the position of CH. The CH will select a backup CH that has the next highest priority factor among all vehicles in the cluster other than itself. To increase the stability of the cluster topology, the elected CH should be as stable as possible.

Each CH will calculate the future positions of all of its CMs after time t_n, based on received speeds as $P(t_n) = P(t_1) + s_1(t_n - t_1)$, where $P(t_1)$ is the position of a vehicle at t_1, t_n is the end of the next frame, and s_1 is the speed of the vehicle advertised at time t_1. If more than 75% of the current CMs of the CH become out of the transmission range at t_n but are still within the backup CHs range, the current cluster will hand the responsibility to the BCH.

I consider different factors while electing the CH. Any vehicle should only consider the parameters of those vehicles that are one hop away from itself. Too many CHs around the same set of vehicles led to no MDS [Alon 86]. The information about neighbours helps in achieving MDS. CH calculates priorities for all registered vehicles based on the status message received earlier. I make various assumptions while designing the equation for calculating the priorities. I make two assumptions, no two vehicles can receive the same priority in its one hop neighbour or cluster. Moreover, no two neighbouring vehicles can have the same instant speed during their travel in the network. This means that the relative speed between two vehicles cannot be zero always. A vehicle becomes a CH only if it satisfies the following conditions.

- The vehicle that has the highest priority among those vehicles that are travelling in the same direction and in its one-hop neighbourhood.
- Most number of connected neighbours. The overall CS β, is the utmost number of vehicles that are one hop neighbour to vehicle i. This is represented as

  $$\beta_i(t) = \sum_n C(i,j,t) \tag{3.16}$$

  where j is a potential one hop neighbouring vehicle. $C(i,j,t)$ is equal to 1 if both i and j are with in the transmission range of each other at time t. Additionally, $C(i,j,t)$ is equal to 0 if both i and j are not in the range of each other at time t. Moreover, I have identified the number of CS between a vehicle and all other neighbours that are travelling in the same direction.
- Vehicle that is closest to its one hop neighbours. Each vehicle calculates the average distance between all neighbours and itself using the equation (3.19). This represents how close are neighbours to one vehicle. Taking account of this parameter will decrease the packet delay and increase the lifetime of CH.

  $$D_{xavg} = \left(\frac{D_{x1} + D_{x2} + D_{x3} + ...D_{xn}}{n}\right) \tag{3.17}$$

  $$D_{yavg} = \left(\frac{D_{y1} + D_{y2} + D_{y3} + ...D_{yn}}{n}\right) \tag{3.18}$$

  $$\triangle D_{i,ne} = \sqrt{|D_{xi} - D_{xavg}|^2 + |D_{yi} - D_{yavg}|^2} \tag{3.19}$$

- Less average speed difference between the neighbours. This condition takes into account the mobility of one hop neighbours. Each vehicle calculates the average difference of speed between one vehicle and all its CMs using the equation (3.21). This parameter is used to avoid elected CHs losing connectivity with their neighbours very soon, the eligibility of a vehicle should decrease quickly when its speed has a big difference from the average speed. Thus, a vehicle with large speed deviation is assigned lower priority.

$$S_{av} = \left(\frac{S_1 + S_2 + S_3 + ...S_n}{n} \right) \tag{3.20}$$

$$\triangle\, S_{i,ne} = |S_i - S_{av}| \tag{3.21}$$

Overall, to avoid elected CHs losing connectivity with their neighbours very soon, the eligibility of a vehicle should decrease quickly when its velocity has a big difference from the average speed, when the distance between each other is large, and the number of connected neighbours is less. Thus, a vehicle with large speed deviation, less number of neighbours and with the largest distance between all its neighbours is assigned lower priority. Many possible solutions can be used to compute the priority of a vehicle while considering the aforementioned criteria. To solve this, I define that the priority of vehicle p_i is given by the equation 3.22.

$$p_i = Hash(P(t_n) \oplus i) \oplus E_i \tag{3.22}$$

A hash function is used to generate a unique priority for vehicle i according to the input of local ID, the future position of the vehicle, and the eligibility function. The eligibility function E_i is defined by equation 3.23.

$$E_i = \beta_i(t) e^{-0.2 \triangle S_{i,ne} \triangle D_{i,ne}} \tag{3.23}$$

Where $\beta_i(t) \in (0, \beta_i^{max})$ is the number of possible connected neighbours at time t and $S_{i,ne} \in [0, 50]$ is the speed deviation. β_i^{max} is the maximal number of possible connected neighbours and $D_{i,ne}$ is speed deviation with its one hop neighbours. The units of $D_{i,ne}$ and $S_{i,ne}$ are meters and miles/hour, respectively.

After the cluster head election, each CH form clusters by listening to the periodic updates of one hop neighbours. The cluster formation in different protocols will be discussed in Section 3.2.

3.2 Cluster formation for all protocols

Cluster formation is an important part of clustering in VANETs. Clusters are formed either based on the transmission range of the CH, direction of movement or road divided into different segments. CMs can be either linked to a cluster-ID or CH-ID. In Section 3.1, I have discussed three different protocols for CH election, once a CH is chosen the cluster is given an ID and CMs are always linked to this cluster-ID. This increases the overall stability of the cluster where CM reconfigurations due to CH change are avoided.

In CBSDA protocol, the road is divided into equal length segments and each segment is assigned with a cluster-ID. Each cluster-ID will have a CH, BCH and CMs. The main idea of road segments was to arrange CHs and CMs based on their position in data aggregation that will be discussed in Section 5. Moreover, this technique benefits in low mobility scenarios such as traffic jam or traffic congestion, but the cluster lifetime decreases in high mobility scenarios.

In D-CBM, the vehicles join the CH who is nearest to them. Each vehicle receives cluster-ID from the CH that is associated to the cluster. Then each CM sends periodic messages to the CH. Moreover, CH selects the BCH based on the priority. The CH will add the BCH in the next periodic message to its neighbours. In D-CBM, I have only considered vehicles travelling in one direction. D-CBM is not suitable for scenarios, where vehicles travel in both direction. Considering an example, if both CH and CM travels in opposite direction, the life time of the CM will be low due to the mobility.

In DA-CMAC, I have considered vehicles travelling in both directions and placement of RSUs along the road. After initial election of CHs in both directions, vehicles join the CH which travels in same direction and nearest to them, thereby increasing the CM lifetime and stability of the cluster. CMs update the corresponding CH-ID in the periodic messages and CH updates the CM lists, allocates corresponding slots and select a BCH from CMs based on the priority.

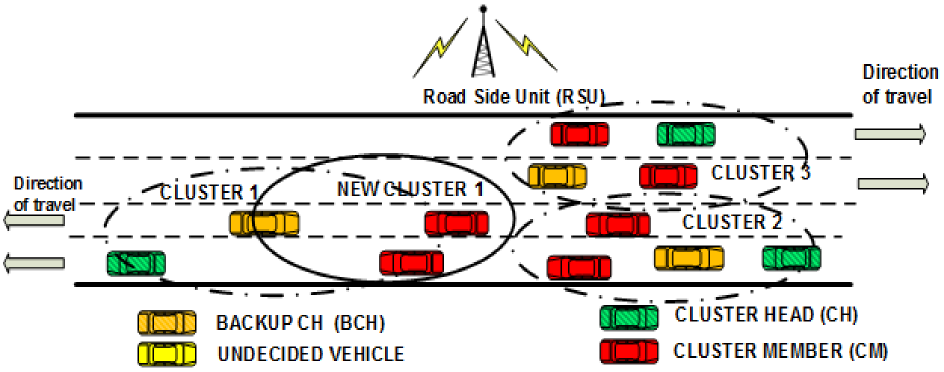


Figure 3.7: CH in the cluster 1 is about to resign. So, it will choose BCH to be CH of new cluster 1

3.3 Cluster maintenance for all protocols

Due to the movement of vehicles, the cluster will not stay the same for long time. The behavior of many vehicles may change the topology of the cluster; for example:

- A cluster head leaving the cluster
- Transformation from undecided state to cluster member
- A cluster member leaving the cluster
- Transformation from cluster member to gateway vehicle
- Merging two clusters

All these changes in the cluster topology will be addressed.

3.3.1 Cluster head leaving the cluster

When a CH is elected using the CH election algorithm, the CH will be locally assigned to ID 1, as the local ID of the cluster in DA-CMAC protocol. If the CH predicts changes in its mobility behavior that might lead to being an unstable CH, it will prepare for giving up its responsibility as a CH. In order to increase the stability of cluster, CH elects a BCH after the formation of the cluster. BCH

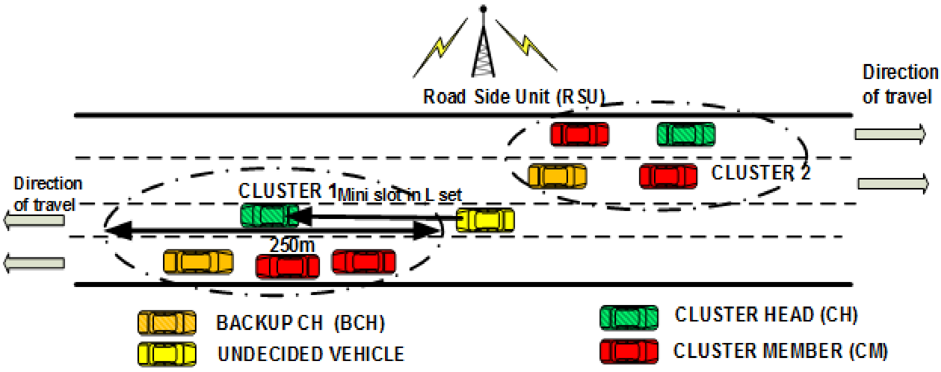


Figure 3.8: A Undecided State (US) vehicle trying to join a single-hop cluster by communicating with the cluster's CH using one of the mini-slot in slot set L. Notice that US is travelling on the same direction as the cluster 1

is the vehicle that has maximum priority among CMs. Each CH will calculate the future positions of all of its CMs after time t_n, based on received speeds as $P(t_n) = P(t_1) + s_1(t_n - t_1)$, where $P(t_1)$ is the position of a vehicle at t_1, t_n is the end of the next frame, and s_1 is the speed of the vehicle advertised at time t_1. If more than 75% of the current CMs of the CH become out of the transmission range at t_n but are still within the BCHs range, the current CH will hand the responsibility to the BCH. From Figure 3.7, we have one-hop cluster where the CH is assigned to a local ID of 1. In this case, the CH is ready to resign. Before this happens, the CH transfer local ID to BCH to be a CH. The reason for that is to have a one-hop cluster even after the current CH leaves. The process will be done by switching the local ID between the current CH, ID 1, and the new CH.

3.3.2 Transformation from undecided state to cluster member

In this section, I will explain the procedures for adding an undecided vehicle to the cluster group as a cluster member. The new vehicle listens to neighbouring CM vehicles update and safety messages. These messages contain the slot and mini slot information of other vehicles inside the cluster. In this section, I explain the procedure for an Undecided State (US) vehicle to join single-hop cluster and two-hop cluster.

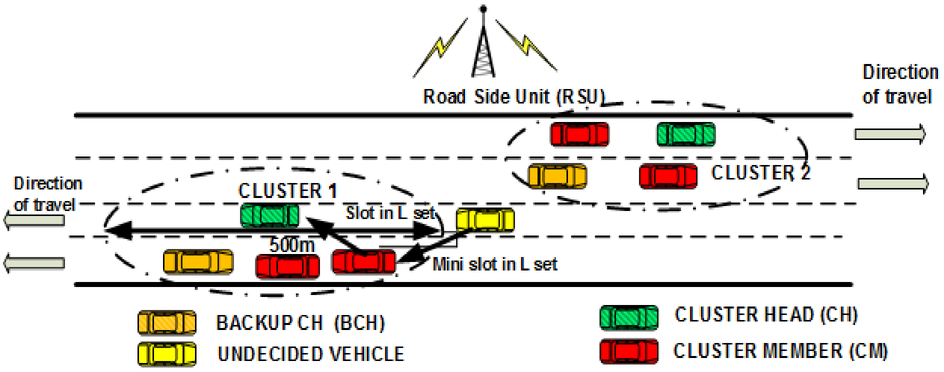


Figure 3.9: A US is trying to join a two-hop cluster by communicating with the CM of cluster 1 using mini slot. Notice that US is travelling on the same direction as the cluster1

A single hop cluster is a group where the radius of the cluster is half of the maximum transmission range of the CH (250m). Figure 3.8 shows how an US vehicle that is travelling in the same direction as the cluster 1 is trying to join the single hop cluster 1. The US can receive update messages from other CMs in the cluster, even it does not come under the transmission range of CH. When US come under the half of transmission range (125m) of CH, it will try to transmit in one of the empty mini slot in the slot set L. Assuming that the mini slot access to CH was successful and CH receives the request from US to join the cluster, if there are empty slots in the CH, it will assign the lowest possible ID and allocate a time slot in the next current frame. If there are no empty slot that means cluster size has reached the maximum number of vehicles. Then CH checks if there are any slot available in the GHL or GTL. If there are available slots, CH allocates an ID and a slot to US as GVs. Moreover, if there are no time slots available in both sets, CH will not allocate any ID or slots. When an US vehicles receives the CH message and comes to know that there is not any slot allocation for itself, US vehicle will again send the request in a mini slot of next frame until it gets a slot in the CH.

The two-hop cluster is a group of vehicles where the radius of the cluster is the maximum transmission range of the CH (250m). Figure 3.9 shows how a US vehicle that is travelling in the same direction as the cluster 1 is trying to join the two hop cluster 1. Assuming that the US vehicle is not within the communication radius of the CH but its in the range of other CMs of the cluster, when the US vehicle receives an update message from one of the CM of cluster 1. It will synchronize itself and try to send the request to join the cluster it will attempt to get the attention of any CMs in the cluster by transmitting in any mini-slot of the set L. Assuming that the mini slot access is successful, which ever CM receives the request from the US vehicle will update the neighbour list before transmitting the update message to all vehicles in the cluster. The CH that detects the neighbour, checks if the position between itself to the neighbour is less than 500m. will inform the CH of the newcomer using the cluster member's mini-slot. When the CH receives a newcomer notification from a cluster member, the CH will start to look for an available local ID to assign to the newcomer and then send it back to the cluster member that discovered the newcomer. Once the cluster member receives the available ID from the CH, it will inform the newcomer in the same way as in the single-hop cluster. All the communications between the cluster member and the CH for assigning the local ID to the newcomer are done using their own mini-slots on the CCH.

3.3.3 Cluster member leaving the cluster

When a CM predicts mobility changes using $P^1(t_n) = P(t_1) + s_1(t_n - t_1)$, where $P^1(t_1)$ is the position of CM^1 at t_1, t_n is the end of the next frame, and s_1 is the speed of the CM^1 at time t_1, let the position of CH at t_n be $P^{CH}(t_n)$. If $|P^1(t_n) - P^{CH}(t_n)| > 250$m, the position of CM is out of range which might lead it to leave the cluster. A cluster member will broadcast this information to the entire cluster using its own time slot in CCH. If a CM fails to do so, the CH will make the CMs local ID available after a certain period of time (2T). If a US vehicle wants to join the cluster, the CH can assign the US vehicle one of the available local IDs. Also, the CH will send an update table of the CMs, GVs, BCH and their local IDs to all vehicles in the cluster.

3.3.4 Transformation from cluster member to gateway vehicle

In my scheme, the cluster is considered to be a single-hop. So, because of the dynamics of vehicles on the road, the CMs of two single-hop clusters may come

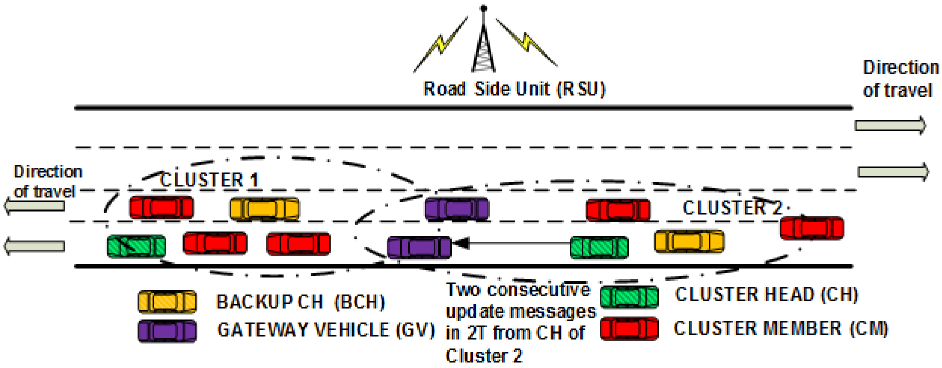


Figure 3.10: Two CMs of cluster 1 transform to GVs, after receiving two consecutive update messages from CH of cluster 2

into range of CH of each others cluster. When a CM are in range of two different CHs that are travelling in the same direction, the CM transforms its status to GV. If it receives more than two consecutive update messages from the other CH, as shown in Figure 3.10. The cluster members that are located at the head and tail of each cluster will be the vehicles that get involved in the CM to GV process. When these vehicles detect the present of the other CH, they will start the transformation process. The transformation process will be performed by vehicles advertising themselves to other CH in the different clusters by sending an update message that includes their IDs and their CHs during their slot. If the total number of GVs in both clusters is less than or equal to the maximum number of the GV size in DA-CMAC, the transformation process will continue. This procedure detecting outsiders will be performed periodically, e.g. 1 second. Once the messages are exchanged between the CMs of different clusters to a different CH, each one of them will try to communicate with the other based on each one's slot of the original cluster. During this communication, the CH will inform their CMs and every CM will inform its CH about the transformation process. Then CH of the cluster 1 will make the ID available for new US vehicles.

3.3.5 Merging two clusters

Two clusters merge each other and become single one-hop cluster. When the number of CMs of a cluster is less than the number of GVs of the cluster. The cluster

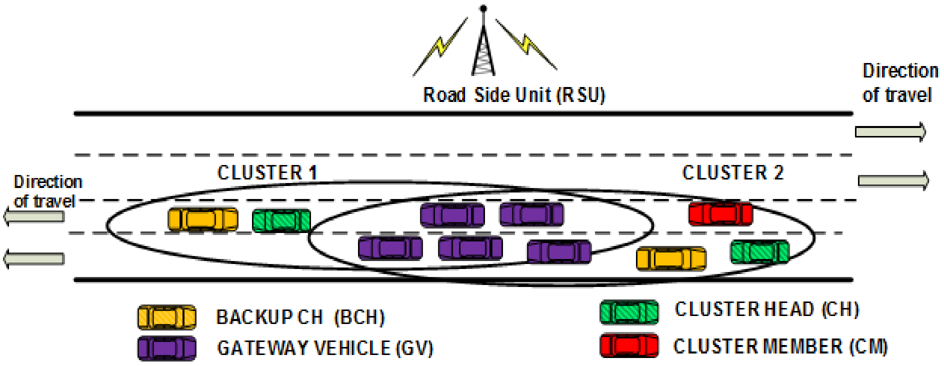


Figure 3.11: Two cluster 1 and cluster 2 before merging, where number of CMs of cluster 2 is higher than cluster 1. After merging CH of cluster 2 take charge of the merged cluster

with lower CMs will merge with the other cluster. The CH in the larger cluster before merging will be the CH of the newly merged cluster. Then, the old CH in the small cluster will resign by releasing the ID 1 and CH of the large cluster retaining the ID 1. Once the merged cluster is formed, the CMs, and BCHs of the old large will have the same IDs, while the CMs of the old small cluster will receive new IDs. The GVs IDs will be listed available and , these GVs gets available CM IDs for newly merged cluster. The new IDs for CMs of the small cluster will be allocated from list of available IDs by the CH. If the clusters are of the same size, the CH of the approaching cluster will be the CH of the new cluster. If the new large cluster seemed to be stable for a long time, not splitting into the two old clusters, every vehicle will calculate the priority value and send it along with its update message to elect a new CH.

3.4 Summary

I have presented three different protocols for CH election based on the traffic characteristics of vehicles on the road. The CH elections in proposed protocols are based on

- The relative mobility between one hop neighbour.
- Number of connected neighbours.

- Distance between itself and its neighbours.
- Distance between itself and neighbouring RSUs.

The cluster formation in these protocols is based on the direction of the travel. All these considerations will improve the duration of communication period between vehicles. The vehicles in these protocols takes up different roles such as, Undecided state, BCH, CH, GVs and CMs. The existence of BCHs will reduce the overhead by avoiding entire cluster reconfiguration when the CH loses its connectivity. The GVs are those vehicles that are in the communication range of two clusters, it will reduce the number collisions due to merging of two clusters. I have also presented a cluster maintenance algorithm to improve the stability of the cluster topology. With the availability of local IDs, the management overhead is reduced when the topology of the cluster changes. I have tested all the proposed protocols using network simulator (ns-2, ns-3). An evaluation of these protocols will be presented in Chapter 6.

4

MAC protocols for VANETs

This chapter presents MAC protocols for different VANET scenarios. The protocols discussed here are both single channel and multichannel protocols based on contention free and contention based communication. The protocols developed here supports vehicles travelling in both directions and eliminating the hidden terminal problem, in order to successfully deliver both periodic and event driven traffic safety messages. The protocol reduces collisions occurring due to vehicular mobility on the channel by assigning different sets of time slots to vehicles travelling in opposite direction and to road side units.

I will start by describing the transmission phase of the D-CBM protocol. In D-CBM protocol, I will describe the single channel and multi channel protocol for VANETs using contention free and contention based communications. Secondly, I will introduce the transmission phase of the DA-CMAC protocol. DA-CMAC protocol reduces collisions occurring due to vehicular mobility on the control channel by assigning disjoint sets of time slots to vehicles moving in opposite directions, gateway vehicles and to road side units. I will start by describing the TDMA slot assignment, and then I will describe the intra-cluster and inter communications. In this protocol, unlike WAVE, all vehicles are able to tune to the CCH or the SCHs during the time cycle. It is designed to allow vehicles to send and receive non-safety messages without any impact on the reliability of sending and receiving safety messages even if the traffic density is high.

4.0.1 Motivation

After comprehensive survey of different well known protocols proposed for VANETs, I find the following issues that should be considered while designing MAC protocols for vehicular communication. The MAC protocol designed should be able to give fairness in channel access to all vehicles in the network. Moreover, not only should it provide fairness but also ensure predicable channel access in case of safety messages. Furthermore, MAC protocol should ensure reliable transmission during traffic congestions.

As VANETs are ad hoc in nature then each vehicle should have enough knowledge of its neighbours. The MAC protocol employed should be fault tolerant and cope with highly dynamic topology of VANETs. Any topological change should deal without any delay in channel access. For example, if a CM linked to a cluster hears packet from another neighbouring cluster, this CM should get a common time slot in both clusters as soon as possible.

As VANETs struggle from congestion during rush hours of traffic or during road accidents, it would result in congestion on the channel. The MAC protocol designed should be able to assign slots to all vehicles in its one hop neighbours which are travelling in both directions. The non safety messages can be transmitted using service channels, some of vehicle might require more than one slots in SCHs. Moreover, protocol should be designed to cater more than one slots for a vehicle in SCHs. Those needs urged us to design a Multichannel MAC for VANETs able to address those issues as well as to achieve better MAC performance with respect to existing MAC protocols.

4.0.2 Requirements for MAC

When designing the MAC protocol for VANETs, there are several important factors that have to be taken into account. Traffic safety and non safety applications have diverse communication requirements, especially regarding reliability and delay.

- Vehicle density of VANETs depends upon scenarios for example a traffic accident and a lot of vehicles pile up in the road. The MAC protocol should scalable enough to allow channel access for the all vehicles in the network.
- Traffic safety applications are real time communication systems, implying demands on predictable delay for delivery of messages. Therefore, access delay must be bounded, so any messages, especially the safety ones, can access the channel within predictable delay. The worst case channel access delay is essential and should not exceed the message deadline. According to these requirements, the MAC protocol must be predictable when the density of VANETs increases.
- Reliability is coupled with the error probability of packets. Successful communication of the VANETs not only requires a predictable MAC protocol to access the the channel, but also depends on packet delivery rate. Non safety messages are tolerant, but safety messages are not, as high priority messages need to have 100% delivery rate. The MAC protocol has to achieve high delivery ratio for both safety and non safety applications.

Since none of the existing MAC protocols reviewed in Section 2.3 are scalable, reliable and predictable for safety applications, none of them address all of the MAC protocol requirements. Specifically, none of them try to address fairness issues at the same time. Fairness is important when an accident occurs, so the RSU can receive complete information from all vehicles in the network and monitor the region of accident. Hence, I have designed the protocols that satisfies all of these design criteria.

4.0.3 Synchronization

In this thesis, synchronization between vehicles are achieved using the one pulse per second (1PPS) signal provided by any GPS receiver. The start of every GPS second is aligned with the rising edge of 1PPS signal. The accuracy using GPS is within 100 ns even for inexpensive GPS receivers. CMs, and CHs synchronise the time using 1PPS signal. By using this signal any CM or CH can identify the current time and slot that is being used within the frame for CCH or SCH transmission.

However, if their is any temporary loss of GPS signal, the synchronization between CMs and CH can still be maintained within a certain accuracy for a time period which depends on the stability of the GPS receiver's local oscillator at each CM or CH. If the GPS signal is lost in a certain region for a long period (longer than a specified threshold), a distributed synchronization scheme, such as the one presented in [Kutzner 03], should be used until the GPS signal is recovered. However, details of such synchronization schemes are out of scope of this thesis.

4.1 D-CBM protocol

The objective of the D-CBM protocol is to achieve stability, predictability, and reliability (less packet losses, timely delivery of data, and thus better packet delivery) based on the cluster formation discussed in Section 3.1. In this section, I consider that a stable cluster structure is attained from Section 3.1. After achieving stability in the cluster structure, the CHs and RSUs allocate time slots to associated CMs and CHs. Two types of communication happens between vehicles and RSUs, which are RSU to CH (vice-versa) and CH to CM (vice-versa) communication. In case of CH to RSU communication, I propose two approaches based on contention free and hybrid. D-CBM based on hybrid approach employs two transceivers for communicating between CH to RSU and CH to CM in different channels is shown in the Figure 4.1. D-CBM based on contention free approach uses single transceiver for both CH to RSU and CH to CM communications.

D-CBM based on hybrid approach is based on the combination of both contention based and contention free protocols. In hybrid approach, the communication between CH to RSU is based on CSMA/CA communication using one transceiver and communication between CM to CH is based on contention free protocol using another transceiver. In CSMA/CA the CH waits for the channel to be free to communicate with RSU. Moreover, if the channel is busy the CH back-off for a random period and access the channel after that period. Contention free protocol is based on TDMA approach, where the frame is divided into fixed number of time slots. The vehicles access the channel in the allocated time slot for communicating with other CMs, CH and RSUs. The slots for CMs and incoming Undecided vehicles are given time slot by its own CH. In this protocol, I consider three types of methods

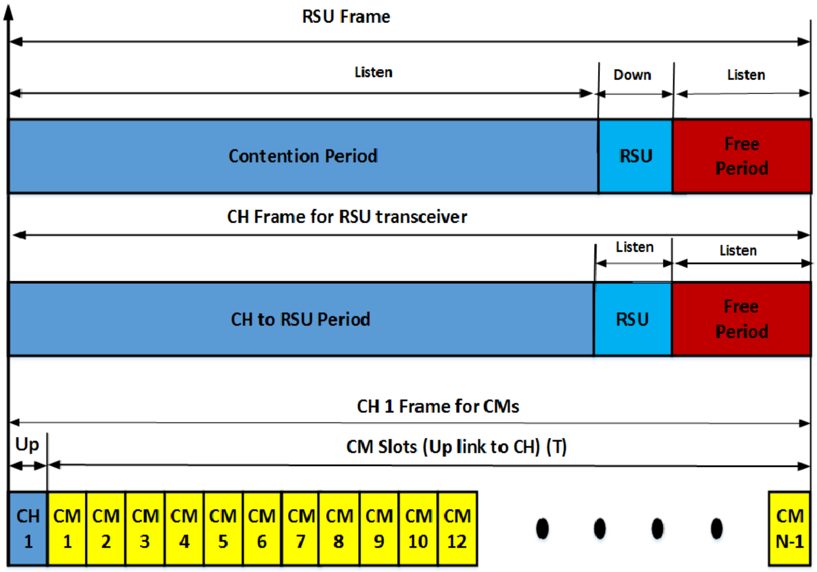


Figure 4.1: Contention based in RSU and TDMA in CH

for slot allocation to CMs. They are random selection, selection based on adaptive standard score, and lastly based on the future position between CH and its CMs. The D-CBM based on hybrid approach using contention based CSMA/CA and contention free TDMA based on random slot allocation are discussed in detail in the following sections. Moreover, different slot allocation methods for contention free approach using single transceiver are also discussed.

4.1.1 D-CBM based on hybrid approach

Hybrid approach is based on the combination of contention based approach for CH to RSU and contention free approach for CH to CM communications is shown in the Figure 4.1. In Hybrid approach, I consider two transceivers are installed in the CHs. These CHs are capable of communicating simultaneously with CMs and RSUs at the same time. In this case, the CH can communicate simultaneously with CMs and RSUs at the same time. The CH to RSU communication is based on CSMA/CA protocol as shown in the Figure 4.2. In CSMA/CA approach, contention period is considered for CH transmission. Moreover, only CH that needs to send messages to RSU will only access the channel. Additionally, CSMA/CA provides bandwidth efficiency. However, the disadvantage is that there can be two

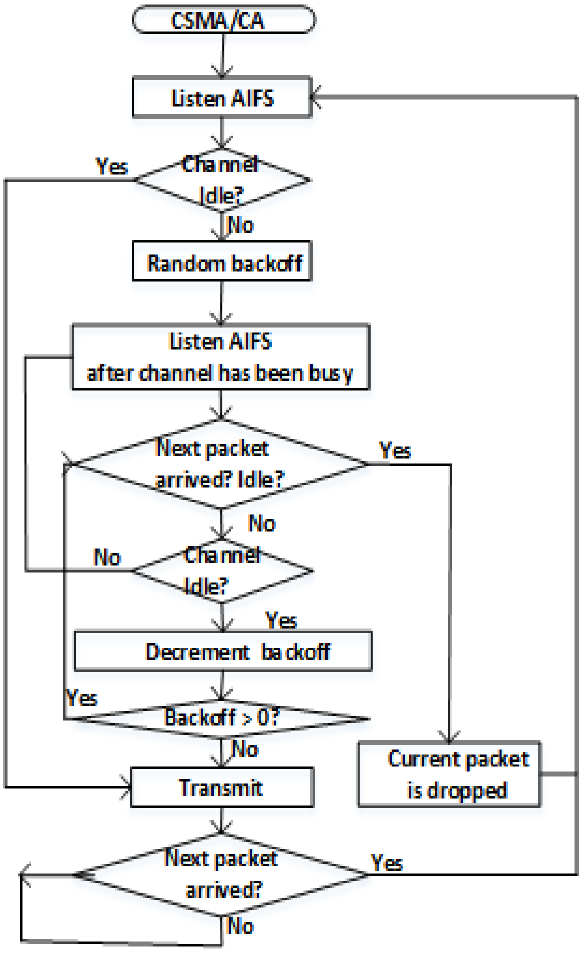


Figure 4.2: CSMA/CA procedure according to 802.11p

CHs trying to access the channel at the same time and leading to collision. The frame of RSU in the contention based channel is divided into three sections, they are contention period for CH access, RSU broadcast or down link period and free period. The CH access the channel in contention period using CSMA/CA protocol.

The CSMA/CA operation for transmitting periodic messages from CH to RSU based on IEEE 802.11p, is shown in Figure 4.2. In CSMA/CA, the cluster member or CH starts by listening to the channel activity during an AIFS amount of time. Moreover, the default parameter settings for the different queues in 802.11p are shown in Table 4.1. After listening to the AIFS time, if the channel is sensed free, the message is sent. After that, the CH checks if a new message from the upper layers is ready to be send, and when there is one, it performs the same action to sent the new message. If during AIFS, the channel is busy or becomes busy, then the CH gets a random back-off value, generated from an exponential distribution, by multiplying the integer from [0..CW] with the slot time 13 μs obtaining 0, 13, 26 or 39 μs. This value will be decreasing every time the node waits for an AIFS and senses the channel free.

When the back-off value gets to 0, then the message can be sent. While the CH is getting its back-off value decreased, it keeps on checking constantly if a new message was generated in the upper layers and is ready to be sent. When that happens, the old message is dropped, and the CH starts again with the whole sending protocol. The CH performs this operation for sending the message to RSU, while CH uses other transceiver for sending and receiving messages from the CMs using TDMA. TDMA is used to reduce the number of collisions by simultaneous transmissions in the same cluster, transmissions are only allowed on assigned slots by the CH. The RSU frame for hybrid approach is shown in the Figure 4.1. The RSU frame consists of contention period and contention free period, where during contention period the CHs access the channel for sending messages to RSUs. The contention free period in RSUs are divided into two sections, they are RSU down-link and free period. In the RSU down-link, the RSUs forward any information received from neighbouring RSUs to CHs regarding any road accidents or traffic congestion etc. The free period is sub divided into a number of mini slots. The vehicles that wish to register with RSUs can use these mini slots in the free period.

The CH has two frames, one for the transceiver that is communicating with the RSU and other for communicating with its CMs. The CH communicate with the CMs using TDMA for accessing the channel. CH frame for TDMA is shown in Figure 4.1. CH frame is divided into three sections, they are CH slot, CM slot, and free period. Moreover, CH is always assigned with first slot in the frame for sending slot information to all CMs. Furthermore, CMs access the channel according to the given time slot. CMs are allocated slots by CHs randomly. Free period consists of a number of mini slots for newly joined vehicles.

	Queue 1	Queue 2	Queue 3	Queue 4
Priority	Highest	→	→	Lowest
AIFS	58 μs	58 μs	71 μs	123 μs
CWmin	3	7	15	15
CWmax	511	1023	1023	1023

Table 4.1: Default parameter setting in 802.11p for the EDCA mechanism

4.1.2 D-CBM based on contention free approach

In this approach, only one transceiver is used for both CH to RSU and CH to CM communication. The RSU frame for CH to RSU communication using TDMA approach is shown in Figure 4.3. The RSU frame is divided into three sets of slots, they are sets of slots for CHs, RSU down-link, and free period. In this approach, RSU randomly allocates time slots to all CHs that are registered with RSU. CH sends packet in its allocated time slot. Then RSU allocate a slot for itself in the frame for sending information to the CHs and CMs. Furthermore, a mini slot from free period is allocated to different vehicles that are not linked to any of the clusters in the network.

CH frame is also divided into five sections as shown in Figure 4.3. In the first section, the first time slot in all frame is reserved for itself for transmitting the slot information to the CMs. Information about assignment of slots is carried by a **CJ** message and includes the suggested scheduling scheme. Periodic messages are messages required to acknowledge reception of **CJ** messages. CH takes the initiative to start managing the allocation of time slot(s) to its CMs, i.e. becoming the master vehicle(s). In this approach, I consider three types of slot allocation by CHs. They are random, velocity based and future position based slot allocation methods. These methods are discussed in detail in following sections. The network will stabilize once all CMs get their time slot in CH frame and all CHs get time slot in RSU frame. Third section is the RSU down link period, where RSU forward the information collected from other RSUs to its CHs. Fourth, CH will transmit the information received from RSU to all CMs. Lastly, free period is divided into number of mini slots for vehicles to join the cluster.

In this approach, I consider the frame length of CH and RSU is considered to be 100ms. Moreover, time delay for a message reaching the RSU from CM is represented by the equation (4.1). As mentioned earlier, the CH frame is divided into various sections which are depicted in the equation (4.2).

Considering the equation (4.3), we can derive different number of possible slots in the CH frame for various packet sizes of CM in the cluster. Respectively, the maximum number of CM slots in the CH frame is obtained from the equation (4.5). The parameters of the equations are explained in the Table (4.2).

Parameter	**Explanation**
$\triangle D_{i,neighbours}$	Average distance between all neighbours and itself
$\triangle S_{i,neighbours}$	Average difference of speed between one vehicle and all its CMs
$\triangle R_{i,rnei}$	Relative distance between one vehicle and its all other CMs to registered RSU
W_1, W_2, W_3	Weighting factors
R_{max}	Maximum transmission range of RSU
R_{trm}	Maximum transmission range of OBU
S_{max}	Maximum speed in the highway
$T_{delay,RSU}$	Time delay for a message to reach the RSU from CM
CH_t	Length of the CH frame in ms
$P_{ch\to rsu}$	Size of the CH to RSU packet
$P_{cm\to ch}$	Size of the CM to CH packet
$P_{rsu\to ch}$	Size of the RSU to CH packet
$P_{ch\to cm}$	Size of the CH to CM packet
R	Data rate in bits/sec
d_{ifs}	Length of inter frame space

Table 4.2: Parameters of CH formation and slot length

$$T_{delay,RSU} = T_{CM2CH} + T_{RSU2CH} \tag{4.1}$$

$$CH_t = \left(\frac{P_{ch\to rsu}}{R}\right)+\left(n \times \frac{P_{cm\to ch}}{R}\right)+\left(\frac{P_{rsu\to ch}}{R}\right)+\left(\frac{P_{ch\to cm}}{R}\right)+((n+3)\times d_{ifs}) \tag{4.2}$$

$$P_{ch\to rsu} = P_{rsu\to ch} = P_{ch\to cm} = P_{crs} \tag{4.3}$$

$$CH_t = \left(3 \times \frac{P_{crs}}{R}\right) + (3 \times d_{ifs}) + \left(n \times \left(\left(\frac{P_{cm\to ch}}{R}\right) + d_{ifs}\right)\right) \tag{4.4}$$

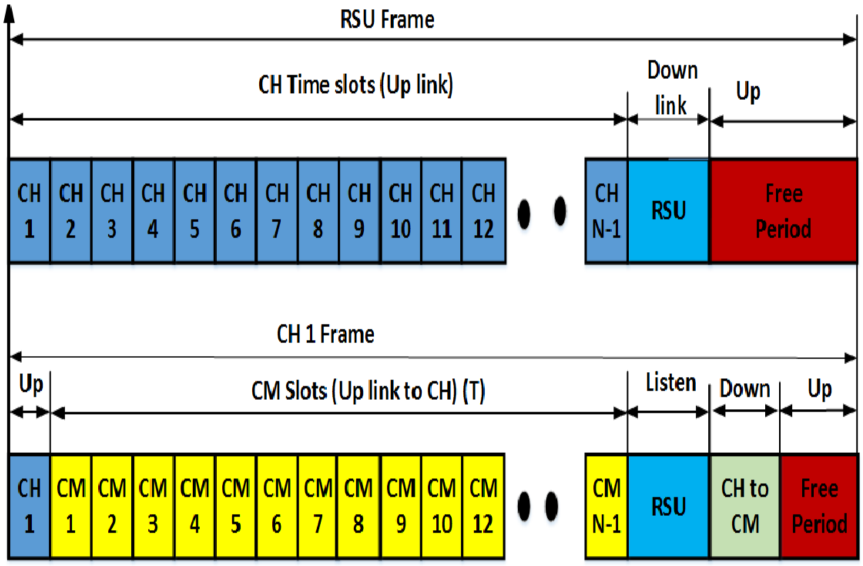


Figure 4.3: TDMA in both RSU and CH Frames

$$CH_t = \left(\frac{3P_{crs} + nP_{cm\rightarrow ch}}{R}\right) + ((n+3)d_{ifs}) \tag{4.5}$$

4.1.2.1 Random allocation of slots

In this method, CH allocates slots to CMs randomly. There is no parameter for assigning slots to CMs. After receiving the time slot, the CM start transmitting the packets to their CH in the respective time slot allocated to them. CH frame based on random allocation is shown in Figure 4.3.

4.1.2.2 Slot allocation based on adaptive standard score

This is an enhanced version of earlier slot allocation, where the CMs are selected randomly and assigned slots in each frame. In this enhanced version, the CMs are allocated time slots based on the adaptive standard score is shown in Figure 4.4. The adaptive standard score is based on the average speed of the cluster and adaptive standard deviation. In this protocol, the CMs are grouped based on the lowest, medium and highest adaptive standard score from speed of the vehicle. This type of slot allocation can improve the reliability of MAC in highly mobile scenarios.

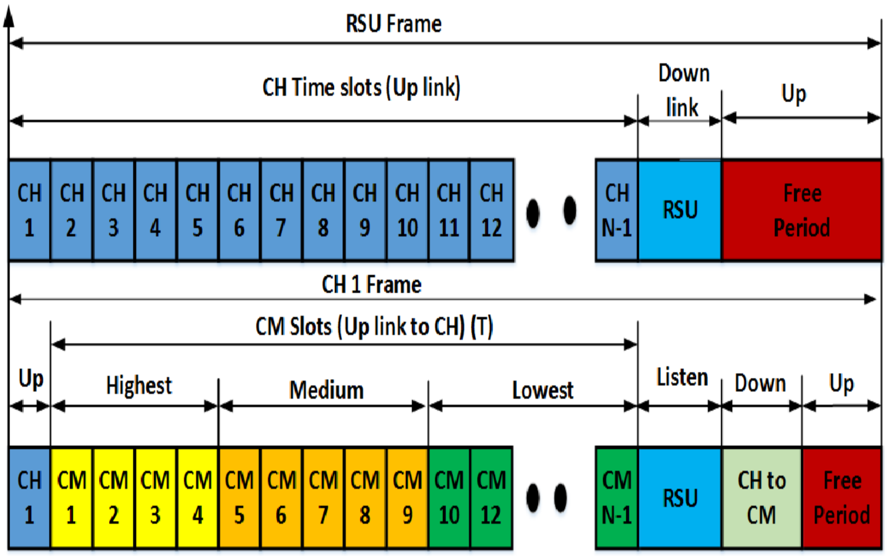


Figure 4.4: Adaptive standard score based slot allocation in CH Frame

For example, the vehicles moving with highest speed deviation have very less time in terms of connectivity with the cluster. These vehicles are given more priority in slot allocation in the start of the CH frame.

$$S_{avg} = \left(\frac{S_1 + S_2 + S_3 + ...S_n}{n} \right) \tag{4.6}$$

$$S_i^{sc} = \frac{S_i - S_{avg}}{\sigma_t} \tag{4.7}$$

The CM slot set is divided into three sets to that is associated with three levels of speed deviation: High, Medium, and Low. The CH calculates the average speed of the CMs by using equation (4.6). Then CH finds the adaptive standard score using the adaptive standard deviation of speed for all cluster and deviation from average speed of cluster for each CMs based on the speed reported in the previous frame. In my adaptive standard score scheme, the value of the speed standard deviation is based on the current speed of vehicles rather than a set value. Then CH allocates CMs based on adaptive standard score into three sets highest, medium, and lowest, as shown in Table 4.3.

The value of S_h and S_l are determined according to the speed interval of the vehicles and the speed deviation in the real traffic scenarios. When there is a high

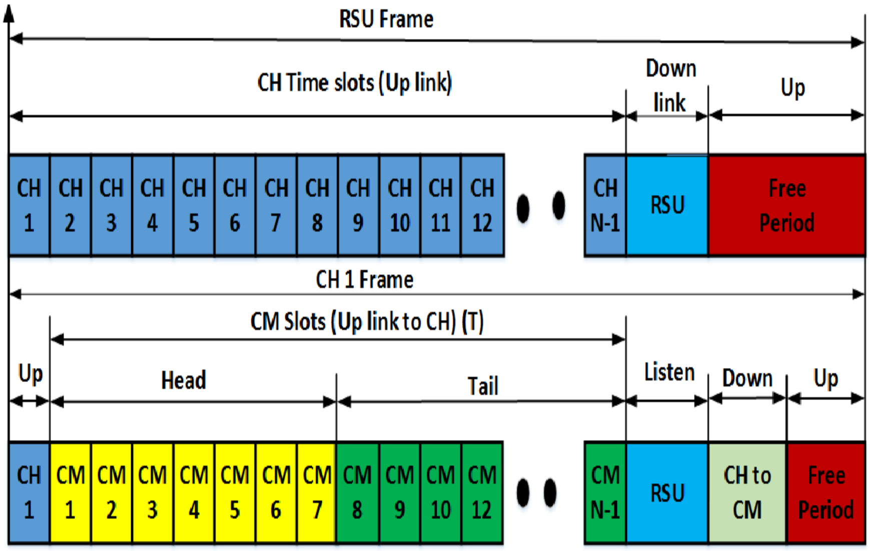


Figure 4.5: Future position based slot allocation

density traffic condition, the value of S_l will be decreased as the speed between vehicles may be not that much, but in highway scenario, the value of S_h will be much more higher. While in the unbalanced traffic scenarios, it is possible that some vehicles cannot acquire a time slot for a long time as it has a larger number of competitors. Therefore the number of time slots associated with speed interval requires to be determined according to the real traffic scenario.

Adaptive standard score	$S_i^{sc} > S_h$	$S_h \geq S_i^{sc} \geq S_l$	$S_l > S_i^{sc}$
Speed level	Highest	Medium	Lowest

Table 4.3: The different sets of slots based on adaptive standard score

4.1.2.3 Slot allocation based on the future position

To increase the reliability of packets those have been affected due to mobility of vehicles, the CH should be able to predict the future position of its CMs based on the speed and current position received in the previous frame. Each CH may have received the periodic updates of vehicles in different times $t_1, t_2, t_3, ..t_n$. The CH will calculate the future position of each CMs till the end of the next frame (t_m).

The position of CM_1 was received at t_1, the CH calculates the future position of CM_1 at t_m by equation (4.8)

$$x_1(t_m) = x_1(t_1) + S_1(t_m - t_1) \tag{4.8}$$

$$P = x_{cm}(t_m) - x_{ch}(t_m) \tag{4.9}$$

The CH will calculate the expected positions of all of its CMs till the end of the next frame (t_m). The CH slots are divided into two sets tail and head as shown in Figure 4.5. Then CH will calculate the distance between itself to all CMs in the end of next frame based on the equation (4.9). The slot will be allocated based on the values of P. If the P value is positive, a slot will be allocated in the Head set and if the P value is negative a slot will be allocated for the CM in the Tail set. The slot allocation in the Head set is based on the descending order for the values of P. The CM with highest value of P will be allocated first slot in the Head set and lowest positive value will be allocated in last slot of Head set. The CM with lowest value of P will be allocated first slot in Tail set and highest negative P value will be allocated last slot in the Tail set. The Table 4.4 represents a slot set allocation based on the future position. This slot allocation will improve the reliability of messages due to the changes in speed and position.

Future position	$x_{ch}(t_m) > x_{cm}(t_m)$	$x_{ch}(t_m) \leq x_{cm}(t_m)$
Slot sets	Tail	Head

Table 4.4: The different sets of slots based on adaptive standard score

In this approach, slot allocation is carried out in CH, where newly joined vehicles contend with each other for acquiring the time slot in the CH frame. Basically, a new vehicle that wants a time slot in CH frame, try to transmit a sample packet in the contention period. If a vehicle wins the contention (i.e., it transmits successfully), it waits for the next frame to receive the time slot. A complete collision-free schedule is achieved as soon as all newly joined vehicles in the previous frame in the cluster have acquired their own slot. The quickest way to achieve a collision-free schedule is to allow all newly joined vehicles to contend for any free time slot in the CH frame. Thereafter, newly joined vehicles can communicate with CH in a collision free manner.

4.1.3 Slot allocation for newly joined vehicles

In this section, I discuss the slot allocation of newly joined vehicles and impact of number of slots on the cluster size. My analysis is split into two cases, first case I consider the cluster size is equal or less than the number of slots in the CH frame and in the other case, the cluster size is greater than the number of slots in the frame. Moreover, these cases are analyzed using Markov chain. In this section, I derive a Discrete-Time Markov-Chain (DTMC) model of the slot allocation, where number of slots in CM section of CH frame versus the cluster sizes.

Without losing in generality, we can assume that the reporting period T, is fixed and common to all CM in the network. I also assume that the reporting period is divided into $(N + 3)$ slots. CMs send one data packet per period to CH. Hence, each CM requires one dedicated slot per reporting period T. Within the CH frame, N slots will be allocated according to characteristics of a registering undecided vehicles for transmitting a packet to its CH. It is assumed that, in the system, all packets have the same size and all time slots also have the same length except the slot of CH and RSU transmission slot. Undecided vehicles arrived in the previous frame will be allocated slots in the current frame of CH.

To simplify the analysis, throughout I will make the following assumptions.

- All incoming undecided vehicles start competing for time slot in the frame. This maximises the contention and, hence, I model a worst-case condition.
- A CM is considered no longer a member or left the cluster. When two consecutive packets are not received by the corresponding CH.

4.1.3.1 Cluster size $\leq$ N

Consider a simple model of a slot allocation where, $A_{(n)}$ is the number of Undecided vehicles that joins the cluster in the n^{th} frame. This is depicted in the Figure 4.6. $A_{(n)}$ is a random variable that follows the Binomial distribution with mean $(N\alpha)$. More specifically, during each frame an Undecided vehicle joins the cluster with probability α and $(1 - \alpha)$ is the probability that no Undecided vehicle joins the cluster in the frame. Where $P_{(a,j)}$ is the probability of j Undecided vehicles joins the cluster in the nth frame.

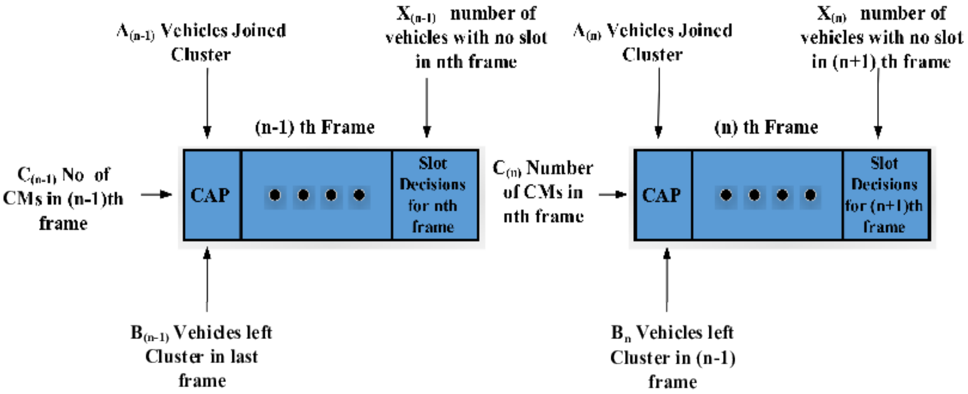


Figure 4.6: Frame specification

$$P_{(a,j)} = \begin{cases} \binom{N}{j} \alpha^j (1-\alpha)^{N-j} & 0 \le j \le N, \\ 0 & j \ge N, \end{cases} \tag{4.10}$$

Parameter	Explanation
T	Reporting period to all CMs in the network.
N	Number of slots in the reporting period.
M	Maximum number of CHs in one RSU frame.
$A_{(n)}$	Number of Undecided vehicles that joins the cluster during the (n)th frame.
$B_{(n)}$	Number of CMs that left the cluster during the $(n-1)$th frame.
$X_{(n-1)}$	Number of CMs in the $(n-1)$th frame that have not received any slot in the (n)th frame and waiting for time slot at the $(n+1)$th frame.

Table 4.5: Parameters

$B_{(n)}$ are number of CMs that have left the cluster in the previous frame, follows Binomial distribution with mean $(N\beta)$. Further, during each frame a CM leaves the cluster with probability (β) and CM that does not leave the cluster is with probability $(1-\beta)$. Where $P_{(b,k)}$ is the probability of k CMs left the cluster in the previous frame.

$$P_{(b,k)} = \begin{cases} \binom{N}{k} \beta^k (1-\beta)^{N-k} & 0 \le k \le N, \\ 0 & k > N, \end{cases} \tag{4.11}$$

Assume that $A_{(n)}$ and $B_{(n)}$ are independent processes. $k < r$, r is the number of CMs present in the cluster in $(n-1)$th frame. Let $X_{(n)}$ be the number of CMs

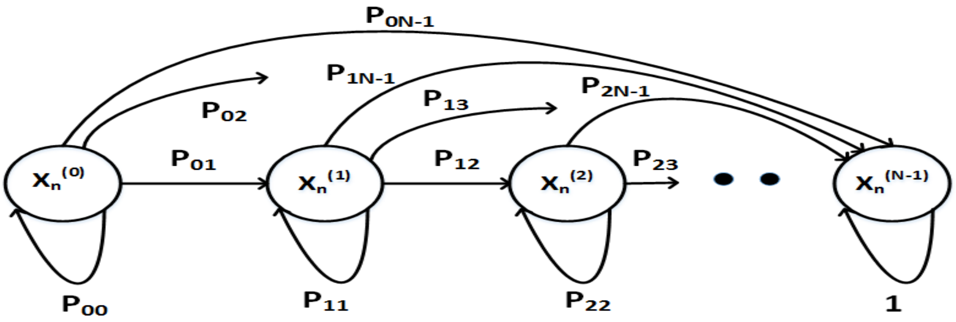


Figure 4.7: State of the nth frame when the cluster size less than or equal to the number of available slots

allocated with a time slot in the nth frame : $X_{(n)}$ can take values up to N. N is the maximum number of slots in one frame. Then $X_{(n)}$ is a Markov chain and evolves according to the equation (4.12).

$$X_{(n+1)} = min\left\{X_{(n)} + A_{(n)} - B_{(n)}, N\right\} \tag{4.12}$$

The number of slots allocated evolve together with the distributions for arrival $A_{(n)}$ and departure $B_{(n)}$ from the cluster leads to a transition probability which is defined by (4.13).

$$P_{i,j} = P(X_{(n+1)} = j|X_{(n)} = i), \quad i, j = 0, 1, 2, ... \tag{4.13}$$

We easily see that if $j > 0$ then

$$\begin{aligned} P_{i,j} = P(A_{(n)} - B_{(n)} = j - i) &= \sum_{k=0}^{\infty} P\left(B_{(n)} = k, A_{(n)} - B_{(n)} = j - i\right) \\ &= \sum_{k=0}^{\infty} P\left(B_{(n)} = k, A_{(n)} = k + j - i\right) \\ &= \sum_{k=0}^{\infty} P\left(B_{(n)} = k\right) P\left(A_{(n)} = k + j - i\right) \\ &= \sum_{k=0}^{\infty} P_{(b,k)} P_{(a,(j+k-i))} \end{aligned} \tag{4.14}$$

The Markov chain model of the ideal slot allocation where the total number of slots is equal to the number of CMs inside the cluster plus the number of Undecided

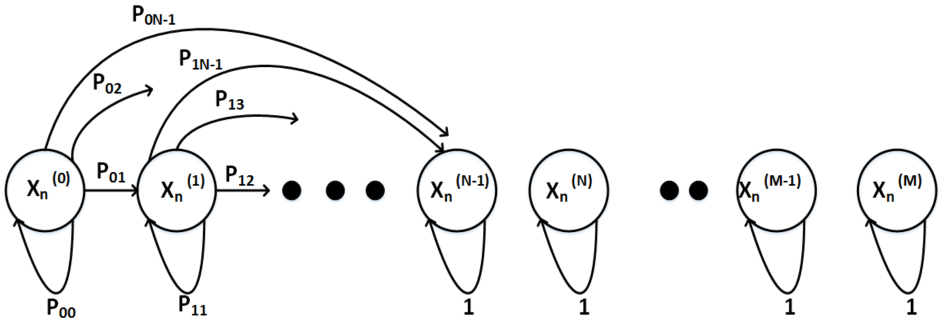


Figure 4.8: State of the nth frame when the cluster size greater than number of available slots

vehicle arrivals and difference of the number of CMs left the cluster in previous frame. In Figure 4.8, the Markov chain is used to describe the slot status of the CH in the frame. Where the circles denote the state of the Markov chain (the number of CMs allocated time slot) and the arcs denote the possible transitions occurring.

4.1.3.2 **Cluster size $>$ N**

Let $A_{(n)}$ be the number of Undecided vehicles that joins the cluster during the nth frame. $B_{(n)}$ is the number of CMs that left the cluster during the $(n-1)$th frame. Let $X_{(n-1)}$ be the number of CMs in the $(n-1)$th frame that has not received any slot in the nth frame and waiting for time slot at the $(n+1)$th frame. Now, if $X_{(n-1)} = 0$, then there are no CMs waiting for slot in the $(n+1)$ frame. All the Undecided vehicles that joins the cluster during that nth frame, namely $A_{(n)}$, will get time slot in the $(n+1)$th frame unless $(A_{(n)} + X_{(n-1)}) > (K + B_{(n)})$, in this case the number of available slots K plus the number of CMs that had a time slot in previous frame and left the cluster during the $(n-1)$th frame is less than the combination of number of Undecided vehicles that had joined the cluster during the nth frame and number of CMs in the $(n-1)$th frame that have not received any slot in the nth frame. Hence $X_{(n)} = (A_{(n)} + X_{(n-1)}) - (K + B_{(n)})$. K is zero, when the number of unused slots in previous frame is zero. If $X_{(n)} > 0$, are the CMs that does not have time slot in the $(n+1)$th frame and $A_{(n+1)}$ are Undecided vehicles who joins the cluster in that frame, subject to capacity limitations.

Assume that undecided vehicles arrival $A_{(n)}, n \geq 0$ and CMs leaving the cluster $B(n), n \geq 0$ is a sequence of identically distributed (i.i.d) Poisson random variables with probability mass function (pmf)

$$P\left(A_{(n)} = k_1\right) = \frac{e^{-\lambda_1}\lambda_1^{k_1}}{k_1!}, k_1 = 0, 1, 2, 3....\infty \tag{4.15}$$

$$P\left(B_{(n)} = k_2\right) = \frac{e^{-\lambda_2}\lambda_2^{k_2}}{k_2!}, k_2 = 0, 1, 2, 3....\infty \tag{4.16}$$

$$P(A_{(n)} = k_1) = a_{(k_1)}, k_1 \geq 0, \tag{4.17}$$

$$P(B_{(n)} = k_2) = b_{(k_2)}, k_2 \geq 0, \tag{4.18}$$

Under this assumption, $X_{(n)}, n \geq 0$ is a DTMC on state space $0, 1, 2, ...(M)$. In this model, I consider M as the maximum possible number of vehicles in the cluster in a frame. In constructing the initial Markov models, I consider the possibilities of state transitions with Undecided vehicle arrival in the previous frame, and CMs leaving the cluster. The transition probabilities can be computed as follows. For $0 \leq j < M$,

$$\begin{aligned} P(X_{(n)} = j | X_{(n-1)} = 0) &= P((A_{(n)} - B_{(n)}) = j | X_{(n-1)} = 0) \\ &= P(A_{(n)} - B_{(n)} = j) \\ &= \left\{Z = a_{(n)} - b_{(m)}\right\}, |n - m| = j \end{aligned} \tag{4.19}$$

The number of CMs without slot in the nth frame which is derived with Skellam distribution. (Z) is the difference of the discrete probability distribution $k_1 - k_2$, using two Poisson distributions with different expected values λ_1 and λ_2.

$$\begin{aligned} P(X_{(n)} = j | X_{(n-1)} = 0) &= e^{-(\lambda_1+\lambda_2)} \left(\frac{\lambda_1}{\lambda_2}\right)^{\left(\frac{j}{2}\right)} I_j\left(2\sqrt{\lambda_1\lambda_2}\right) \\ &= z_j \end{aligned} \tag{4.20}$$

where $I_j(Z)$ is the modified Bessel function of the first kind.

$$\begin{aligned} P(X_{(n)} = M | X_{(n-1)} = 0) &= P((A_{(n)} - B_{(n)}) = M | X_{n-1} = 0) \\ &= P((A_{(n)} - B_{(n)} \geq= M) \\ &= \sum_{j=M}^{\infty} e^{-(\lambda_1+\lambda_2)} \left(\frac{\lambda_1}{\lambda_2}\right)^{\left(\frac{k}{2}\right)} I_k\left(2\sqrt{\lambda_1\lambda_2}\right) \\ &= \sum_{k=M}^{\infty} z_k \end{aligned} \tag{4.21}$$

Similarly, for $0 \leq i \leq M$ and $i \leq j < M$,

$$\begin{aligned} P(X_{(n)} = j|X_{(n-1)} = i) &= P((X_{(n-1)} + A_{(n)} - B_{(n)}) = j|X_{(n-1)} = i) \\ &= P(A_{(n)} - B_{(n)} = j - i) \\ &= \left\{Z = a_{(n)} - b_{(m)}\right\}, |n - m| = j - i \\ &= e^{-(\lambda_1+\lambda_2)} \left(\frac{\lambda_1}{\lambda_2}\right)^{\left(\frac{j-i}{2}\right)} I_{j-i}\left(2\sqrt{\lambda_1\lambda_2}\right) \\ &= z_{j-i} \end{aligned} \tag{4.22}$$

Finally, for $1 \leq i \leq M$,

$$\begin{aligned} P(X_{(n)} = M|X_{(n-1)} = i) &= P(X_{(n-1)} + A_{(n)} - B_{(n)} = M|X_{(n-1)} = i) \\ &= P(A_{(n)} - B_{(n)} \geq M - i) \\ &= \sum_{j=M-i}^{\infty} e^{-(\lambda_1+\lambda_2)} \left(\frac{\lambda_1}{\lambda_2}\right)^{\left(\frac{k}{2}\right)} I_k\left(2\sqrt{\lambda_1\lambda_2}\right) \\ &= \sum_{k=M-i}^{\infty} z_k \end{aligned} \tag{4.23}$$

Combining all these cases and using the notation,

$$s_j = \sum_{k=j}^{\infty} z_k \tag{4.24}$$

The distribution of X_n for a fixed $n \geq 0$. In other words, we are interested in $P(X_n = j)$ for all $j \in s$ and $n \geq 0$. We have

$$\begin{aligned} P(X_n = j) &= \sum_{i=1}^{2N} P(X_n = j|X_0 = i)P(X_0 = i) \\ &= \sum_{i=1}^{M} z_i P(X_n = j|X_0 = i) \end{aligned} \tag{4.25}$$

we get the transition probability matrix

$$P = \begin{pmatrix} z_0 & z_1 & \cdots & z_{(M-1)} & s_{(M)} \\ z_0 & z_1 & \cdots & z_{(M-1)} & s_{(M)} \\ 0 & z_0 & \cdots & z_{(M-2)} & s_{(M-1)} \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & \cdots & z_0 & s_1 \end{pmatrix} \tag{4.26}$$

4.2 DA-CMAC

In this section, I introduce the transmission phase of the DA-CMAC protocol that was discussed in Section 3.1. In DA-CMAC, I assume that every vehicle in the

VANETs is equipped with a single transceiver based on the IEEE 1609 interface. The protocol is designed in such a way that the IEEE 1609 interfaces demodulate one channel at a time. This means, even though the IEEE 1609 interface has 7 channels, it cant use more than one channel at the same time. But, with one IEEE 1609 interface installed, the protocol must be designed to challenge the fact that IEEE 1609 interfaces demodulate one channel at a time. This means, even though the IEEE 1609 interface has 7 channels, it cannot use more than one channel at the same time.

In this protocol, I consider there are c service channels (SCHs) numbered from 0 to $c-1$ and one CCH. Similar to IEEE 1609.4 [IEEE 13] protocol, I introduce the so called system cycle, which is divided into CCP and SCP sub periods and repeat every 100 msec. Each CCP period is a frame of CCH channel. CH utilizes SCP period and takes over the responsibility of intra-cluster management. This task includes: assigning time slots to all CMs, receiving newly arriving vehicles (undecided) processing and disseminating all received messages and advertisements. The proposed protocol assumes a single system cycle that is shared between the SCH and CCH.

In the considered VANETs, all nodes are synchronized based on the procedure discussed in Section 4.0.3. The carrier sensing range of each vehicle is assumed to be the same as the communication range. The transmission range of each vehicle is supposed to be the same and the radio link is assumed to be symmetric. The road segment has balanced traffic on two opposite directions.

As shown in Figure 4.9, the CCP consists of slots for CMs, GVs and CH itself. The time slots in each frame are evenly divided into three sets L, R, U associated with the vehicles going on the left, right, RSU respectively. The L disjoint set is further divided into three sets CMs, GFL, GTL. Each time slot can be owned by only one vehicle in CCP. At the beginning of each cycle, all vehicles switch to CCH channel. Each system cycle starts with a frame sent by the CH. Each vehicle is required to include how the channel is perceived using a field called Slot Information (SI), in order to realize the slot reservations. DA-CMAC extends the SI to contain the speed, direction and mode of vehicles rather than just the status of each slot. When a vehicle needs to reserve a time slot, it needs to sense the channel

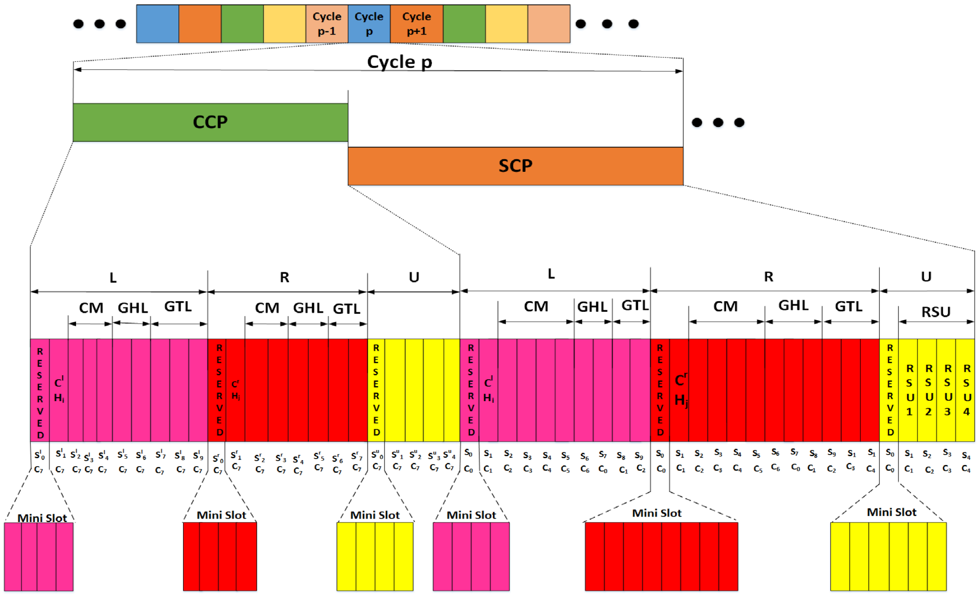


Figure 4.9: DA-CMAC Frames

for a continuous frame interval to determine the available time slots set it is allowed to reserve.

Each CCH interval contains $(N + 3)$ slots numbered from 0 to $N + 2$. $N = N^l + N^r + N^{rsu}$ is the sum of number of slots for vehicles travelling in the left direction $(N^l = N^l_{CM} + N^l_{CH} + N^l_{GHL} + N^l_{GTL})$, number of slots for vehicles travelling in the right direction $(N^r = N^r_{CM} + N^r_{CH} + N^r_{GHL} + N^r_{GTL})$ and road side units in the opposite directions $(N^{rsu} = N^l_{RSU} + N^r_{RSU})$. The total number of vehicles N may change dynamically, and the CH is responsible for updating N and for informing all vehicles in the cluster of the new value of N.

The CH^l_i is the cluster head of i^{th} cluster travelling in the left direction. CH^l_i is like a master vehicle that is allocated with a local-ID 1, which has the responsibility of allocating time slots to CMs and GVs of the cluster. Local ID 0 is not used. CH^l_i transmit the SI in the first time slot of the L set and as mentioned earlier SI contains all time slots for CMs and GVs. All CMs and GVs listen to the SI and transmit in the time slot allocated. SCP period contains $(N+3)$ number of slots from 0 to $N+2$ with $\lfloor \frac{N}{c} \rfloor$ channel cycles in each SCP. All slots are the same size, and the slot size τ is known to all vehicles in the cluster. The CMs, GVs and CH travelling in one direction takes the local-ID equal to the slot number of the previously allocated slot in the CCP.

Time slots in the SCP are used by GVs for inter cluster communication and mini slot in SCP are used by newly arriving vehicles for cluster management (cluster joining). In addition, CCP and SCP are divided into equal size time slots. Moreover, first time slot of all sets L, R, U in both CCP and SCP interval are partitioned into c mini slots. Mini slots on the CCP and SCP are exploited by newly arriving vehicles and RSUs to disseminate status, and safety messages. It is supposed that vehicles ideally remain on the CCH until they are granted a time slot in the CCP period. It is assumed that vehicles are aware of the frame and slot boundaries.

In each time frame, t is used by the vehicle or RSU to specify the channel and the channel cycle according to the following rules, where $0 < t < \left(\left\lfloor \frac{N}{c} \right\rfloor + 1\right) \times c$

- Use channel $t \ mod \ c$ during channel cycle $\left\lfloor \frac{t}{c} \right\rfloor$

The basic idea is that in each logical frame, while idle, vehicle or RSU t listens to channel $t \ mod \ c$ in channel cycle $\left\lfloor \frac{t}{c} \right\rfloor$ and sets the corresponding byte in the CCH in order for other vehicles to be aware. Notice that the Integer Division Theorem guarantees that if $t \neq m$ then either:

- $\left\lfloor \frac{t}{c} \right\rfloor \neq \left\lfloor \frac{m}{c} \right\rfloor$
- $t \ mod \ c \neq m \ mod \ c$

This confirms that no two vehicles own the same channel in same channel cycle. For an illustration, let N = 54 and c = 6. Consider the number of vehicles moving in L and R direction N^l = 20, N^r = 30. The number of RSUs registered in both direction is N^{rsu} = 4. The local-ID for vehicles travelling in L direction is from 1 to 20, for vehicles in R direction is from 1 to 30, and for RSUs is from 1 to 4. As shown in Table 4.6, vehicle travelling in right direction with local ID 15 owns channel (15 mod 6) = 3 during channel cycle $\left\lfloor \frac{15}{6} \right\rfloor$ = 2.

In the Table 4.6, the slots for vehicles travelling in L and R direction are given magenta and red color, the slots for RSUs in both direction are given yellow color. We note that for any given $N = N^l + N^r + N^{rsu}$, the number of unused slots in each SCP period is given by

Channel / Cycle	**0**	**1**	**2**	**3**	**0**	**1**	**2**	**3**	**4**	**0**
5	5	11	17	**Unused**	5	11	17	23	**30**	**Unused**
4	4	10	16	**Unused**	4	10	16	22	29	**4**
3	3	9	15	**Unused**	3	9	**15**	21	28	3
2	2	8	14	**20**	2	8	14	20	27	2
1	1	7	13	19	1	7	13	19	26	1
0	**Reserved**	6	12	18	**Reserved**	6	12	18	24	**Reserved**

Table 4.6: Logical frames in DA-CMAC for N = 54 and c = 6

$$\begin{aligned} (\left\lfloor \frac{N^l}{c} \right\rfloor + 1) \times c - 1 - N^l &= \left\lfloor \frac{N^l}{c} \right\rfloor \times c + c - 1 - N^l \\ &= N^l + c - 1 - (N^l \, mod \, c) - N^l \\ &= c - 1 - (N^l \, mod \, c), \quad (N^l \, mod \, c \neq 0) \end{aligned} \tag{4.27}$$

For $N^l \, mod \, c = 0$, the number of unused slots is 0. By using equation (4.27), we can calculate the number of unused slots in set L, R and U. The number of unused slots in set L from equation (4.27) is 6 -1- (20 mod 6) = 5 - 2 = 3. Moreover, the number of slots in set R for N^r = 30. The value of $30 \, mod \, 6 = 0$, so the number of unused slot is zero. Lastly, for N^{rsu} = 4 is 6-1- (4 mod 6) = 5-4 = 1. The total number of unused slots is 4. These unused slots in channel cycles will be put to work in various ways that depend on the specific clustering regimen under investigation.

4.2.1 Transition from undecided state to cluster member

Those vehicles in the undecided state in both direction listen to the CCH channel c always during the initial approach in order to attain a slot in the CCP and SCP cycle. The protocol reduces transmission collisions when a vehicle in the undecided state try to join the nearest cluster. This scenario can occur when vehicles joining a highway from non highway road. The transmission collision causing due to vehicles travelling in both directions can be avoided by assigning disjoint sets of time slots to vehicles travelling in opposite direction and cluster. Now, suppose US vehicle is just entered the highway from a section road with no V2I or V2V communication and needs to join the cluster and acquire a time slot as a CM or GV.

An access collision happens when two or more US vehicles within the transmission range of each other tries to access the same available mini-time slot. In DA-CMAC, the slot 0 of all sets is reserved and these slots are divided into k mini slots. US vehicle x is travelling in L direction that wish to join the cluster i and require a slot in the L sets of the CCP and SCP. Given N_x^{cm}, N_x^{gv} , and N_x^{rsu} are the set of occupied slots of CM, GV and RSU in the direction of L. $N_x = N_x^{cm} \cup N_x^{gv} \cup N_x^{rsu}$. x will listen to atleast one SCP and CCP cycle and transmit in one of the mini slot in the L set. By listening to the CCH channel 7 for N successive time slots (not necessarily in the same frame), vehicle x can determine set N_x and the time slot(s) used by either GVs or CHs in N_x. Given N_x, vehicle x determines the set of available time slots, A_x, (to be discussed) and then attempts to access the mini slot and ask for any time slot in A_x, say time slot k. If x receives a slot in the SI frame of the corresponding CH, then the slot access of the x is a success. After the transmission of SI by the corresponding CH, all the other CMs (eg; w) add x to its CM list and to the occupied slot list N_w and record the global-ID used by vehicle x to access time slot k, denoted by ID_x^k. Moreover, if x does not receive a slot in SI packet then an access collision have occurred and x need to access the mini-slot again in the next CCP cycle. Once vehicle x acquires a time slot, it keeps using the same slot in all subsequent frames unless a merging collision or transition to CH or GV occur.

4.2.2 Transition from cluster member to gateway vehicle

In a highway, vehicles travel at different speeds, lanes, and move rapidly and two clusters may share an overlapping area for a certain time. When two clusters overlap together, I assume one vehicle is in between two clusters. If CM vehicle receives SI from more than one CH, then it will change its mode from CM to Gateway Vehicle (GV) mode. From Figure 4.10, the vehicle d is member of cluster y initially. Then d receives SI from CH vehicle x without having a slot for itself in the information. d adds x to its list of CHs and add the other CMs of the newly received cluster with all its members to its two hop neighbour list to inform its one hop neighbours to release if any of the slots that are occupied by one hop neighbours. Then compare the available time slots in $A_{gv}(x) \cap A_{gv}(y)$. It se-

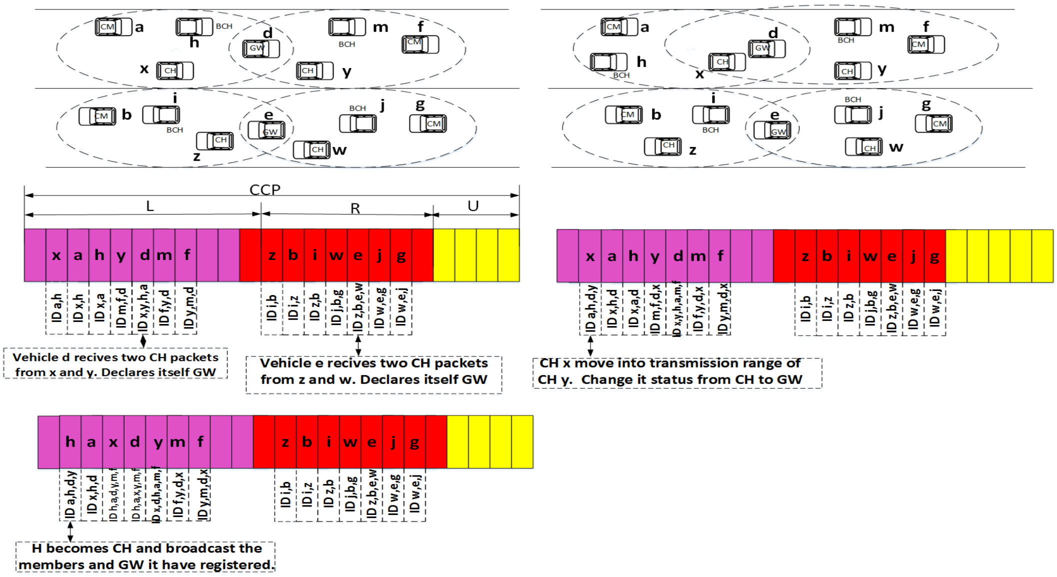


Figure 4.10: Gateway vehicle and time slot

lects a common slot based on the synchronization and an ID is generated by itself based on the slot number. Another scenario can occur , when x receives a message from d without inclusion of CH x. Then CH x identifies a common free slot in $A_{gv}(x) \cap A_{gv}(y)$ and selects a slot and allocate a local-ID with respect to the slot number. If they cannot find a common free slot among $A_{gv}(x) \cap A_{gv}(y)$ then it tries to find a common slot among $(A_{cm}(x) \cup A_{gv}(x)) \cap A_{gv}(y)$. Moreover, if it cannot find a slot among those slots, it trie to find a slot among $(A_{cm}(x) \cup A_{gv}(x)) \cap (A_{gv}(y) \cup A_{cm}(y))$. If it cannot find a common slot again, it trie to find a slot among $(A_{cm}(x) \cup A_{gv}(x) \cup A_{rsu}(x)) \cap (A_{gv}(y) \cup A_{cm}(y))$ if not then it tries to find from $(A_{cm}(x) \cup A_{gv}(x) \cup A_{rsu}(x)) \cap (A_{gv}(y) \cup A_{cm}(y) \cup A_{rsu}(y))$.

4.2.3 Transition from cluster head to cluster member

When CH vehicle x moves to transmission range of neighbouring CH y, the vehicles x or y may receive the SI packet first. The CH with lower number of connected members loses the CH status and elect the BCH as next CH of the cluster. Firstly, if x receives the SI from y first and compares the number of connected neighbours is higher than y, then x continues the role as CH of the cluster and transmits the SI. When y receives the SI information from x then y compares the number of connected neighbours with x. If y have less connected neighbours then y release its slot and allocate the BCH to its slot in the next cluster cycle and takes a slot in the GHL or GTL based on its position compared with newly elected CH. Secondly, if both x and y are having equal number of connected neighbours then the CH will keep the status if the average speed of the neighbours is less than the other. Moreover the vehicle that loses the CH status will assign BCH to it status.

4.3 Summary

In this chapter, I presented three different cluster based MAC protocols for VANETs. First protocol, D-CBM protocol based on combination of contention free and contention based channel access. Secondly, D-CBM based on TDMA, where CHs assign time slots to CMs based on its priority (future position and speed). Lastly, I presented DA-CMAC protocol based on WAVE standard. This thesis work is based on guaranteeing that vehicles receive safety messages with in the deadline. I also changed the concept of having two intervals by having vehicles listening to the control channel and the service channels during the same time cycle. This scheme should be easy and fast to maintain. The evaluation of D-CBM and DA-CMAC will be covered in Chapter 6.

5

Cluster based semantic data aggregation

In recent years, the number of road fatalities in Europe has been decreased by half from 54,900 in 2001 to 25,900 in 2013 [Union 13]. Moreover, the number of fatalities can be further reduced by broadcasting warning messages from traffic safety applications. Furthermore, traffic congestion is an important problem for road transport and a main challenge for transport policy at all levels. The cost of road congestion in Europe is estimated to be over 110 billion euros a year [Panayotis 12]. This congestion can be reduced by using vehicular communication, by notifying a vehicle about the traffic density of the roads ahead so that it can choose an alternate route to reduce the travel duration. This improves overall traffic efficiency of road transport. Traffic accidents and traffic congestion can result in a large amount of data broadcasted in a small region, which may result in dropping of packets due to congestion in the channel. Moreover, this affects the throughput and thereby reduces the reliability of traffic safety applications. In order to reduce the amount of data, I proposed various clustering techniques in Section 3 for increasing scalability of VANETs. In this chapter we can further reduce the channel congestion by using data aggregation technique.

Data aggregation aims to limit the amount of data forwarded between different vehicles and RSUs in VANETs. Data aggregation describes the process of combining data items of different sources [Krishnamachari 02, Rajagopalan 06]. Such sources could be the vehicles, or RSUs in a VANET. RSUs or vehicles might receive and aggregate the vehicular data before it is forwarded to control center or control RSU. The aggregation process can have multiple objectives like eliminating repetitions, to increase or decrease the period of status messages, and to minimise the data load. These objectives can be achieved by discarding duplicate data items and fusing the remaining ones.

In this thesis, the channel congestion is controlled using the density of vehicles registered to each RSU or CH and the number of vehicles a data message represents. Here, I consider two scenarios where stationary RSU collects the data and forward it hop-by-hop, using other RSUs, towards control unit that analyzes the data. In first scenario, RSUs communicates each other wirelessly rather than by a backbone wired network. Second scenario, the data aggregation application is installed in all vehicles. The CH aggregates the data and forwards to control center. Thus immediate forwarding of the received data by each RSU or CH may congest the network channel. A more efficient dissemination approach is to aggregate the information into one data structure first and then disseminate the aggregated data. However, even with this method, the data load grows monotonically towards the control RSU. To break this monotonicity, the size of the data structure can be reduced during the aggregation process by data fusion which combines two data records into one. This thesis proposes a modified version of data aggregation framework [Jiru 14]. In this thesis, I use a tree-based data structure to adaptively aggregate and fuse vehicular data depending on the density of vehicles registered with RSU or CH. In this work, cluster based semantic data aggregation scheme is introduced.

In this chapter, I introduce motivation that lead to data aggregation, semantic data aggregation, and the framework for adaptive data aggregation. Firstly, a conceptional overview of the aggregation framework. This is followed by explanation of all components of the data aggregation framework. Then after, the introduction of the data structure used in the aggregation framework and cluster based semantic data aggregation scheme is explained.

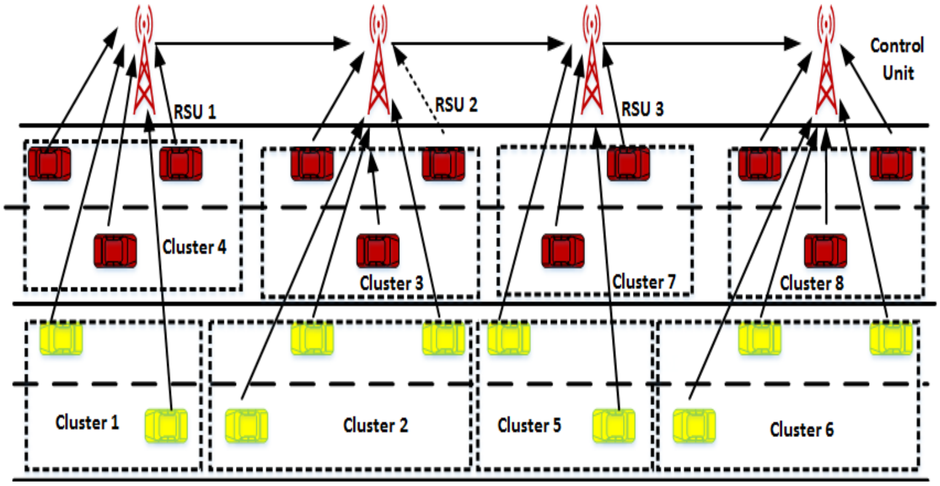


Figure 5.1: Schematic illustration of scenario 1

5.1 Motivation

VANETs generally broadcast different informations such as location, direction of movement, and velocity of vehicles using update or periodic messages. The road safety and efficiency applications can be improved by analyzing centrally the periodic messages transmitted by the vehicles in the road. An application build for analyzing the periodic messages received from the neighbours can improve the reliability of safety messages. Moreover, the applications should rely on its own data and data messages it has received from neighbours. Furthermore, application need to analyse data of vehicles out of reach of the CH or RSU. In order to obtain these data, the vehicles in the network need to forward the data in the direction of the node were application is installed. This forwarding process is executed by using other RSUs or CH to ensure connectivity even under low traffic conditions.

Vehicles on the road broadcasts CAMs with a frequency of 2 Hz, which lies within the range defined in the European norm of the cooperative awareness basic service [V0.0.4 12]. The CAM consists of information about location or position, direction of travel, speed of the vehicles, for instance. RSUs or CHs collects the CAMs broadcasted by neighbouring vehicles on all lanes and forward the data towards the control center hop-by-hop. As forwarding the data towards the destination, RSUs consider only forwarding the data to the next RSU or CH in the direction of control center and ignore all other vehicles. The road is divided into

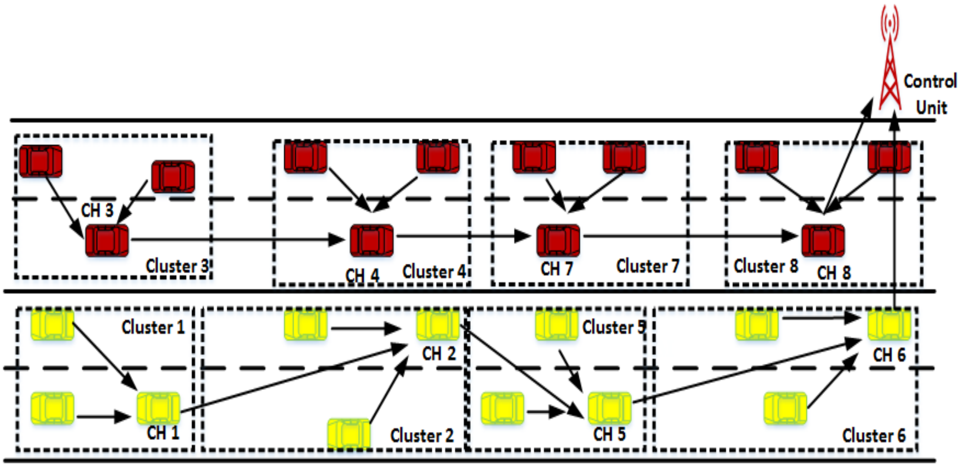


Figure 5.2: Schematic illustration of scenario 2

different road segments and each road segment is assigned to exactly one RSU or CH.

Thus, when a CAM is received by multiple RSUs or CHs, only the RSU or CH assigned to the specific road segment of the vehicle forwards the data of the CAM. Moreover, there are different cluster for different direction, this separates the fusion of two vehicles that are travelling in opposite direction. A schematic illustration of both use cases with infrastructure and without infrastructure are depicted in Figures 5.1 and 5.2. In this thesis, I consider both infrastructure based and infrastructure-free system, where in infrastructure based system, the existence of RSUs is required that act as infrastructure to relay vehicular data. Moreover, it shows from Figure 5.2 that data flow of information from each vehicles travelling in opposite direction are forwarded to near by RSU and RSUs forward these data towards the control unit. On other hand, in infrastructure-free system vehicles aggregate data themselves and forward it to other vehicles which find the data useful. Furthermore, data aging is not relevant, in this use case. In infrastructure free, data is forwarded to other CHs with a predefined control center. From Figure 5.2, the data flow of information from the CHs indicated in yellow and red that are travelling in opposite directions are forwarded towards the control unit.

5.2 Semantic data aggregation

Semantic data aggregation has been chosen due to its high potential of data size reduction as argued in Section 2.4. Semantic aggregation allows to adjust the level of aggregation easily by defining the number of data records that are fused. In semantic data aggregation, actual sensor data are fused and aggregated. Syntax aggregation, in contrast, is concerned with combining data records using the same header. The ability to introduce different levels of aggregation to create a density aware aggregation scheme is therefore more restrained. In this thesis, I consider combination of in-network and in-node semantic data aggregation rather than only one type. Moreover, in-network aggregation looks similar to in-node aggregation due to similarity of data handled by all RSUs and CHs. Additionally, the only difference in semantic in-network data aggregation with in-node aggregation is that two aggregated data is merged in in-network while two single data records are combined in in-node aggregation. Furthermore, the channel load will not reduce significantly while using only in-node aggregation, because data received from neighbouring RSU or CH could not be combined. Therefore, the aggregation scheme developed in this thesis uses both, in-node and in-network aggregation.

5.3 Aggregation framework module

In this thesis, I modify different modules developed based on a modular architecture for aggregation framework [Jiru 14]. Their model, however, does not specify the cluster based data structure, density based control, and decision schemes that are used in this thesis. The aggregation framework shown in Figure 5.3 provides a foundation to design cluster based semantic data aggregation scheme. Each of them has a well defined interface that allows to interchange different implementations of a module easily.

Aggregation framework proposed can be divided into five phases: the decision, fusion, aggregation tree, adaptive control and dissemination phase [Dietzel 10, Dietzel 09, Dietzel 11]. In this thesis, I adopt the definition of different components of the aggregation framework [Dietzel 10, Dietzel 09, Dietzel 11] and extends these components by a process, which is executed in parallel, called aggregation

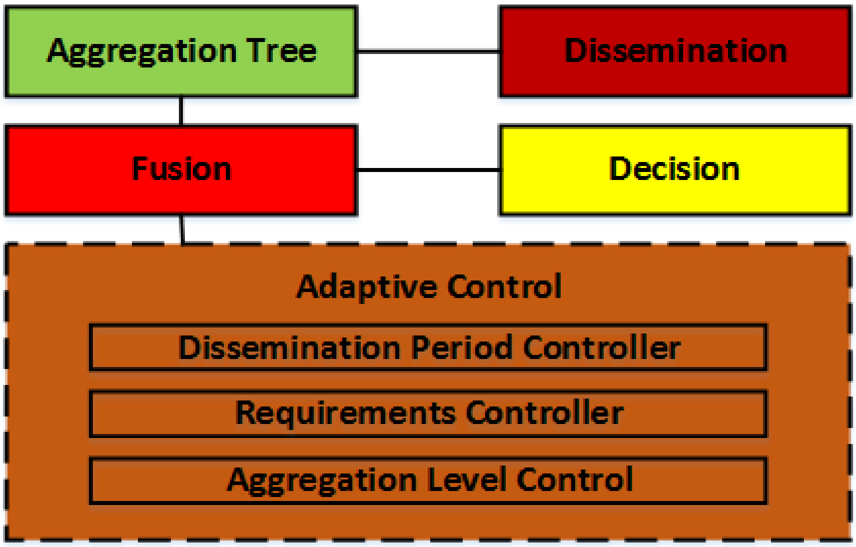


Figure 5.3: Aggregation framework

level control. While decision, fusion, aggregation tree, and dissemination are executed sequentially, the adaptive control runs in parallel as indicated in Figure 5.3. Each phase of the aggregation process is represented by a single component. Each of them has a well defined interface that allows to interchange different implementations of a component easily. Moreover, the Figure 5.3 shows an iterative process of decision, fusion, dissemination, aggregation tree and adaptive control. The fusion phase can access the aggregation level control to obtain information of how to fuse the data and how much data should be fused. This way, the aggregated data is validated by the fusion process. During the dissemination phase, the data structure is then forwarded to the next RSU or CH in the direction of the control unit.

When a RSU or CH receives vehicular information the data is stored in a data structure. The decision component chooses the most similar data records for fusion to achieve high data precision. The fusion component fuses the data which has been selected by the decision component. The data dissemination component defines when and how data is disseminated by a RSU or CH to the next RSU or CH in the direction of the control center. The adaptive control is responsible for the reliable delivery and the end-to-end delay. It monitors the density of vehicles and controls it with the adaptive aggregation scheme maintaining a target numbers of CMs and minimizing the packet loss. In addition, the aggregation tree module is implemented as a cluster based tree data structure.

The implementation of each component defines the properties of an aggregation scheme. The modular architecture provides interchangeability of different implementations. This way, an event-based dissemination component can easily be exchanged by a periodic dissemination component, for instance. Following in this section, each of the five modules of the aggregation framework and its purpose will be introduced in detail.

5.3.1 Aggregation tree component

When an RSU or CH receives CAM messages from different vehicles inside the cluster. The RSU or CH stores vehicular data such as position, velocity, skid resistance and combine the similar data received from neighbouring CH or RSU. Data structure that stores these information in RSU or CH before aggregation and after aggregation is very important. The data structure proposed should provide the following functionality:

- **Data heterogeneity**

  The data structure proposed must have the ability to store different types of vehicular information such as speed, position , direction etc. This information may contain various types of information with different types of values such as floating point, integer, boolean, and text format.

- **Data aggregation**

  The data structure proposed must be capable of merging data of different vehicles in the network into single data structure. Moreover, data structure must allow hierarchical data aggregation. This means that, the data structure must be able to merge single data records with already aggregated data from the neighbouring RSU or CHs.

- **Data fusion**

  Data fusion is a most important component of data aggregation and plays an important part in reducing the size of the data structure. The data structure

Position	Speed	Direction	Skid resistance
100	70	1	6
120	80	1	40
130	90	0	50

Table 5.1: Table as a data structure example

> proposed must support data fusion of sensor data. A data structure must be capable of reducing the size of the structure when required by fusing the multiple data record it holds. For the purpose of this thesis, it is essential that the level of how much data is fused is variable. This allows to implement an adaptive aggregation scheme.

In this thesis, I consider different data structures that meets the above mentioned functionalities. For example, a table is the simplest form of data structure to keep data records. TAG scheme [Madden 02] uses this type of data structure to store data records. In table data structure format, the information of each vehicle in the cluster is stored in one row. The columns represent different values of parameters of the vehicle such as position, direction speed and skid resistance. Table 5.1 represents an example of such a data structure filled with the data records of three different vehicles of the cluster.

Each column is assigned a different data type, which satisfies the data heterogeneity requirement. Using a table for data representation, moreover, allows simple hierarchical aggregation by concatenating the rows of both tables. Also, the fusion operation can simply be implemented by combining the information of two rows in the table. This operation fuses two data records into one. This fusion mechanism, however, only allows to fuse entire rows. This might not be desirable if the application requires more accurate data of a certain metric while other metrics can be fused freely. Simple tables do not support a selective fusion that combines only some metrics of two data records. Thus, single values with a big difference might get lost during fusion. Fusing first and second row of Table 5.1 with an average function, for instance, introduces a high imprecision. The results make it impossible to determine that one vehicle have low skid resistance and might pose a security risk because the application only receives the average skid resistance of

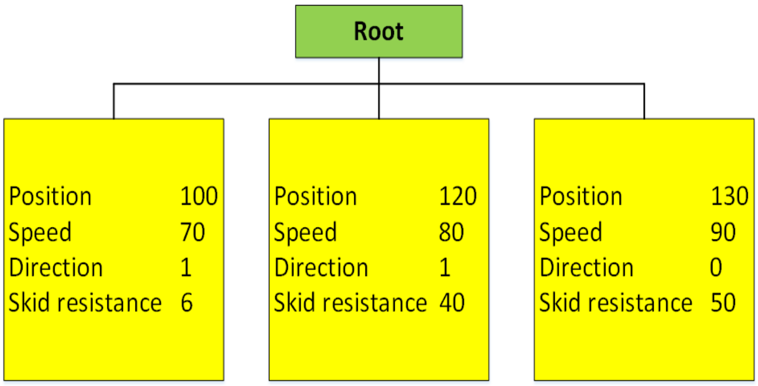


Figure 5.4: Tree representation of Table 5.1

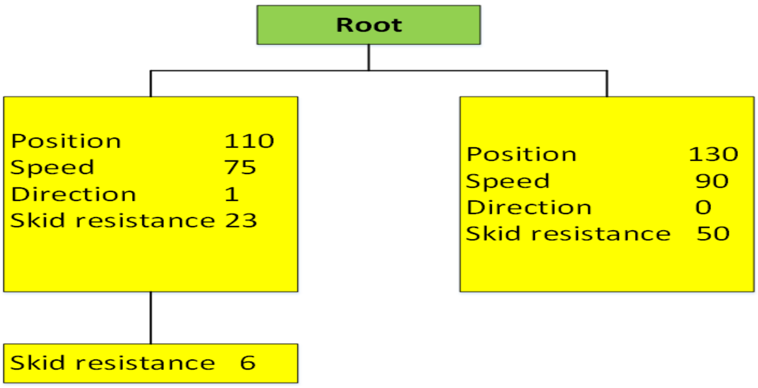


Figure 5.5: Tree representation of Table 5.1 using extrema value aggregation

23. A different way from representing data in a table are mathematical abstractions [Lochert 07, Shrivastava 04, Jiang 10] as introduced before. These, however, fall short describing individual values, which might be required by the application. Therefore, these data structures are not considered in this work.

A more flexible data structure than tables and mathematical abstractions is a tree. It allows to mimic a table representation and extends tables by allowing to combine single metrics. A tree representation of Table 5.1 is shown in Figure 5.4.

Using the same example as before where two data records should be combined, the tree provides a number of options to handle this request. Two of these are depicted in Figures 5.5 and 5.6. Both trees in the figure use an additional tree level to maintain the skid resistance information. Figure 5.5 shows a tree which stores the metric of most difference to the average. In this way, extreme values such as a

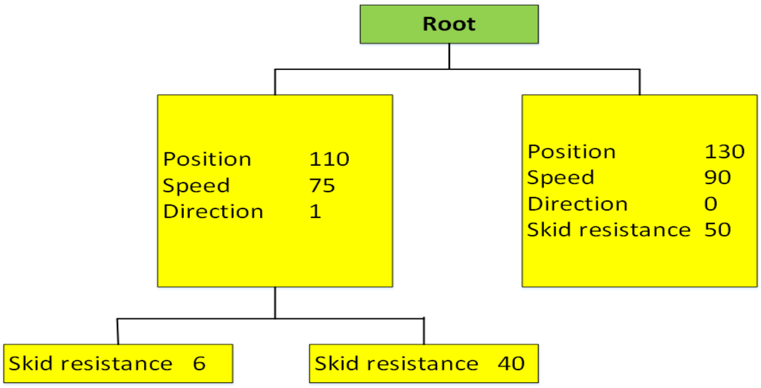


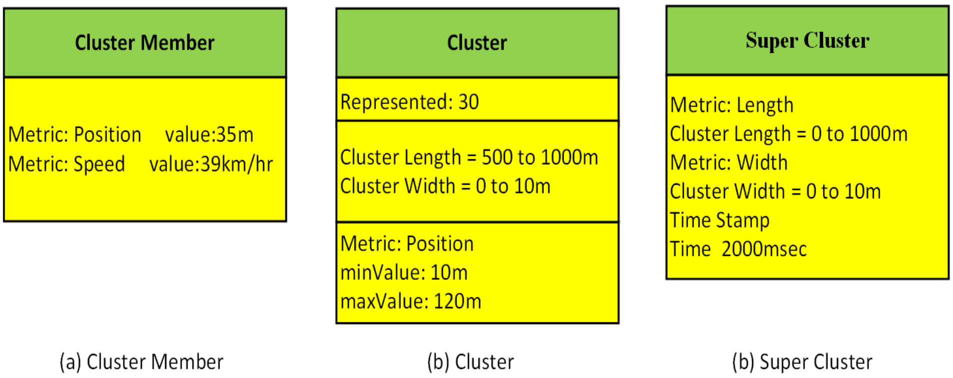
Figure 5.6: Tree representation of Table 5.1 using two level aggregation

very low skid resistance can be preserved. A different approach is used in Figure 5.6, where the individual skid resistance of each vehicle is kept at the leaf of the tree while all other data is averaged.

Adding new data records or other tree structures to an existing tree can easily be done by adding the children of the root nodes of both trees to the root of the new tree. This allows an efficient implementation of in-node and in-network aggregation. The level of aggregation is also easy to adjust using the tree as data structure. In general, the fewer nodes the tree contains, the less capacity it needs. Transferring the data of a small tree to the next RSU does therefore not congest the network channel as much as transferring a large tree. Thus, the aggregation level can be adjusted by restraining the number of nodes in the tree. A tree is therefore used as key data structure in this thesis and will be described in more detail in what follows in this section.

5.3.1.1 **Node types**

To aggregate data, three useful node types for an aggregation tree have been identified. The cluster member or member node, cluster and the super cluster will be introduced in this section. A cluster member or member node, illustrated in Figure 5.7(a), contains the vehicular data. It can represent a full data record of a vehicle. Furthermore, it can contain a partial representation of a data record that does not include all metrics of the vehicle. All previous examples of tree structures contain



(a) Cluster Member (b) Cluster (b) Super Cluster

Figure 5.7: Aggregation tree nodes

only member nodes. A cluster member node only representing a single data record, for instance, represents one vehicle.

When this cluster member is fused with a cluster member node, the number of represented vehicles of both member is summed. The second node type is the cluster node. Its children have values that fit into the cluster length specified by the cluster node. The cluster node shown in Figure 5.7(b), for instance, would contain all cluster member nodes representing vehicles that have a position within the length $L \in [500, 1000]$ and width $W \in [0, 10]$. Such nodes are particularly useful when the application requires a maximum resolution of data quality. In addition to the metrics and values, a cluster node contains a value that identifies the number of single data records it represents. An application may require the velocities of all vehicles on 500 to 1000 meters cluster in lane 1, for instance. A tree supporting this requirement uses cluster nodes, like the node of 5.7(b), as direct children of the root node. Cluster member nodes are placed in their fitting cluster and may only be fused with other nodes from the same cluster. It is not valid to fuse two cluster member nodes from different clusters. The application can then analyze the velocity of the cluster members for each 500 meters in length and different lanes clusters individually.

5.3.1.2 **Conditions of the tree**

The tree layout have some conditions to full fill for simplifying the implementation. Let the set of all nodes of a tree be N, the set of super cluster nodes be denoted as

$SC \subset N$, the set of all cluster nodes be $C \subset N$ and set of all cluster member nodes be $CM \subset N$. The function gp(x) returns the grand parent of node $x \in N$ while the function p(x) and c(x) returns the parent and set of its children.

- $C \cap CM \cap SC = \phi$ The set of super cluster nodes, the set of cluster nodes, and the set of cluster member nodes are disjoint. No node can have status more than one type at same time.
- $|N| = 1 + |C \cup CM \cup SC|$ There exists exactly one node that is no super cluster, cluster nor a cluster member node.
- Let $x \in N$ then, $gp(x) = \phi$, $x \notin (C \cup CM \cup SC)$

  A node has no parent if and only if it is not an super cluster, cluster or a cluster member node. This node is the root.
- $\forall x \in SC$, $gp(x) \notin (C \cup CM)$ The parent of a super cluster node is never a cluster node nor cluster member node. This restricts the tree to have the super cluster nodes close to the root.
- $\forall x \in C$, $p(x) \notin CM$ The parent of a cluster node is never a cluster member node. This restricts the tree to have the cluster nodes close to the super cluster or root node.
- $\forall x \in C$, $c(x) \notin \phi$; $\forall y \in SC$, $p(x) \notin \phi$ super cluster and cluster nodes always have children. This restricts leaf nodes to be cluster nodes.

These restrictions allow to create a large variety of different aggregation trees. Most restrictions are imposed to simplify the implementation and could be relaxed in future work.

5.3.1.3 **Tree types**

The definition and restrictions stated above allows to create different types of aggregation tree. The examples of the tree are depicted in Figure 5.8 and 5.9.

One strategy to create the aggregation tree is to fuse data records into cluster member nodes, which are direct children of the cluster node. Each cluster member node has its own data node children as indicated in Figure 5.8. These children,

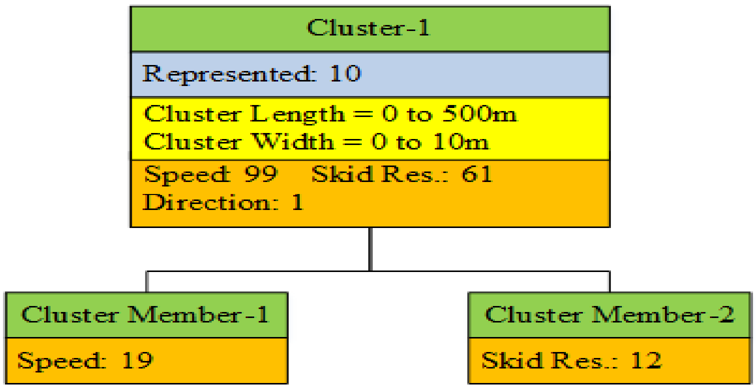


Figure 5.8: Extrema tree

however, only stand for one single vehicle and represent extrema value contained in the cluster member node. This allows to maintain extreme values like the velocity of a single fast car, for instance, while all other data of the vehicles is fused. In the Figure 5.8, the information of ten vehicles was fused in the cluster member at level 1 of the tree. For that reason, all values were averaged and the precise values were lost. However, due to the extrema nodes, it is still possible to make the statement that there was one vehicle with speed 19 and another vehicle with skid resistance 12. This information also increases the precision of the averaged value. For the speed, for instance, the average value of all four vehicles is 99. We know that the speed of one vehicle is 19. Thus, the actual average of the other three vehicles is $\frac{(99 \times 4 - 19)}{3} = 125.67$.

The second tree type uses super cluster node as direct child of the root node as depicted in Figure 5.9. All cluster nodes are assigned as children to the super cluster according to the metrics they fit in. This limits the maximal error introduced by the data fusion. Moreover, all cluster member nodes are children to the cluster node according to the metrics they fit in. It furthermore allows to gather results for a specified lane or direction, for instance, because only siblings can be fused. This means, when using super cluster nodes for the location as depicted in the Figure 5.9, the total cluster length is 500 meters from 500 to 1000 meters and cluster width of 0 to 10 meter. Moreover, no data from vehicles more than 500m in cluster length and 10 meters in cluster width will be fused. This tree type will be used in the aggregation scheme proposed in this thesis.

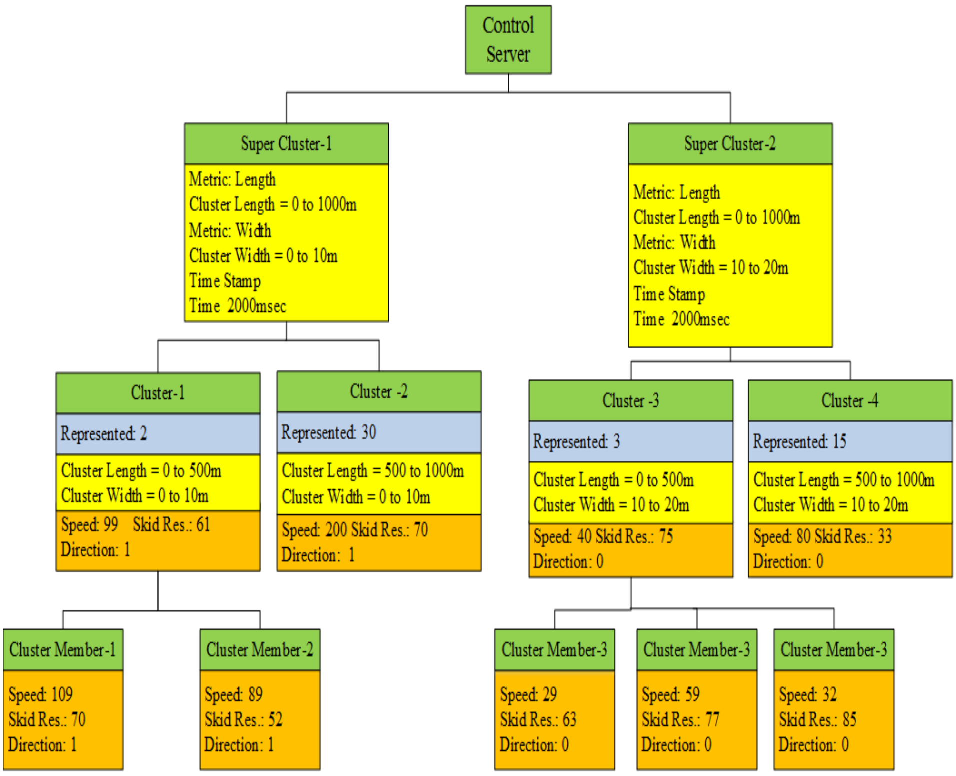


Figure 5.9: Super cluster node as direct child of the root node

5.3.2 Aggregation decision component

The decision component takes a set of nodes as input with nodes that are travelling in same direction and identifies two suitable nodes for fusion among the input set. This section introduces the adaptive cost aware, adaptive standard score, and weighted deviation scheme for decision component.

5.3.2.1 Adaptive cost aware decision

The adaptive cost aware decision uses a weight function to calculate the fusion costs of two vehicles a and b, considering all contained parameters. Let p denote the set of parameters and a_i be the value for the ith parameter of node a. Furthermore, let w_i be the weight for parameter i. Then, the cost can be calculated as indicated in equation (5.1).

$$cost = \sum_{i \in p} w_i * |\frac{a_i - b_i}{max_t}| \tag{5.1}$$

$$w_1 + w_2 + w_3 \ldots + w_i = 1 \tag{5.2}$$

$$w_1 + w_2 = 1 \Rightarrow w_2 = 1 - w_1, 0 \leq w_1 \leq 1 \tag{5.3}$$

Using this notation, assuming a system using only two parameters $P = \{vel, pos\}$, let the weights be 0.5 for both the velocity and position. The weights allow to determine the importance of velocity and position. max_t is the maximum possible values of a particular metric. Furthermore, by varying weights from 0 to 1 can determine the best performance of decision component.

5.3.2.2 Adaptive standard score decision

The adaptive standard score decision component aims at overcoming the problem to find proper weights in the weighted-difference cost function. Instead of multiplying the difference with some weight, the difference is described in units of standard deviation from the real time simulation of the metric at each period t. This allows comparing different metrics without using weights.

To identify the difference between two values in units of standard deviation, the adaptive standard score is used. It is a signed value that describes the difference from the mean in number of standard deviations. This unit of standard deviations allows the comparison of values of parameter values at a particular time t . Let z be the adaptive standard score, x the value of the metric, μ the mean and σ_t the standard deviation of the metric calculated for periodic intervals of time t. Then the adaptive standard score is defined as follows:

$$z = \frac{x - \mu}{\sigma_t} \tag{5.4}$$

Equation (5.4) only describes the adaptive standard score of one parameter at a time t. However, to calculate the fusion costs, two values must be considered. These two values are of the same metric, and thus at same time t. For two values x_1, x_2 this difference of the adaptive standard score can be expressed as follows:

$$|z_1 - z_2| \Rightarrow |\frac{x_1 - \mu}{\sigma_t} - \frac{x_2 - \mu}{\sigma_t}| \Rightarrow |\frac{x_1 - x_2}{\sigma_t}| \tag{5.5}$$

The difference between two values can be described in units of standard deviation by simply dividing the difference of the two values by the standard deviation calculated at a time t using equation (5.5). The resulting difference in standard deviations can be compared with other metrics calculated at same time t, which is the great advantage of the adaptive standard score. To calculate the adaptive standard deviation of a parameter that is used for the adaptive standard score calculation, the RSU could use the data it received within a certain period of time, for instance.

The adaptive standard score decision component aims at overcoming the problem to find standard deviation for each parameter. Instead of dividing the difference with some already assigned or defined standard deviation, the standard deviation is calculated in real time by using the data that is received by RSU or CH within a certain period of time, for instance.

5.3.2.3 **Weighted deviation scheme**

The goal of maintaining extreme data records while fusing many similar vehicles and there by maintain as much precision of the data as possible. Moreover, fusing two values of same metric of different vehicles is preferable. In this scheme, decision component uses a weight function to calculate the fusion costs of two vehicles a and b, considering all contained parameters. Lets consider two parameters velocity and position of the vehicles. p denote the set of parameters and a_i is the value for the ith parameter of node a. Furthermore, let w_i be the weight for parameter i. σ_t^i is the standard deviation of the ith metric calculated at periodic intervals of time t. Then, the cost can be calculated as indicated in equation (5.6).

$$cost = \sum_{i \in p} w_i * |\frac{a_i - b_i}{\sigma_t^i}| \tag{5.6}$$

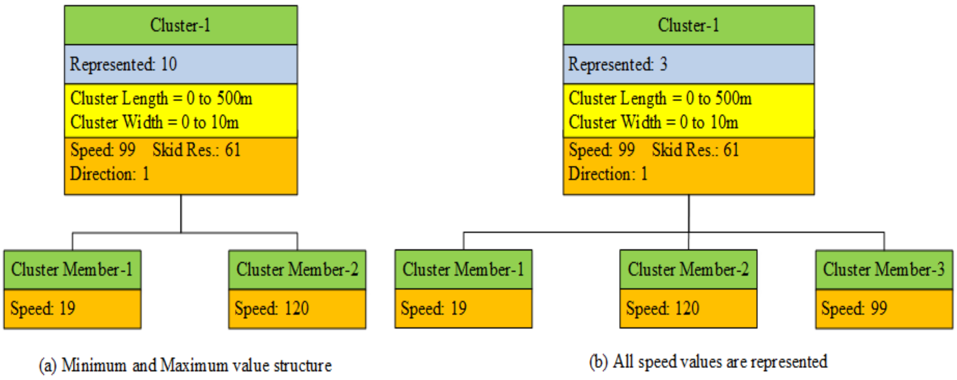
$$w_1 + w_2 + w_3 \ldots + w_i = 1 \tag{5.7}$$

$$w_1 + w_2 = 1 \Rightarrow w_2 = 1 - w_1, 0 \leq w_1 \leq 1 \tag{5.8}$$

Using this notation, assuming a system using only two parameters $P = \{vel, pos\}$, let the weights be 0.5 for both the velocity and position. The weights allow to determine the importance of velocity and position. Furthermore, by varying weights from 0 to 1 can determine the best performance of decision component. Moreover, the difference between the adaptive cost aware scheme and weighted deviation scheme is the use of maximum value $(max_t)$and standard deviation (σ_t^i). I compare the performance of both the weighted decision schemes in Section 6.

5.3.3 Fusion component

The fusion entity is the main component of the aggregation framework proposed in this thesis. When the fusion component identifies a node in aggregation tree with too many children, the decision components decide which two children to fuse. After two data nodes have been identified, the fusion component fuses the data. The fusion functions can be selecting the greatest, or smallest value of a certain metric using extrema nodes as indicated in Figure 5.10(a). Another, function can be keeping all data of only one metric while other metric values are averaged. An example of a resulting aggregation tree is illustrated in Figure 5.10(b), where all speed information is preserved while other data is averaged. The aggregation tree can furthermore be extended to store other information such as the variance. This



(a) Minimum and Maximum value structure

(b) All speed values are represented

Figure 5.10: Different tree‘s after data fusion of speed metric

can be done easily by introducing a new node type or by extending the cluster member node to store the additional information.

The other important function of the fusion entity is to keep the aggregation tree valid at all times. It provides a valid aggregation tree to the dissemination component, collects instructions of how much data to aggregate from the aggregation level control, and allows the decision component to select which all data to be fused. This way, whenever the dissemination component requests the aggregation tree for dissemination, a valid tree is ready to be sent. In this thesis, a tree is called valid when it satisfies two rules. First, it must obey the conditions for the construction of aggregation trees introduced in Section 5.3.1.2. Second, the tree must follow the construction scheme defined by the aggregation level control.

The tasks of the fusion component is to validate the tree and to fuse cluster member or cluster nodes if needed. In what follows in this section, both processes are introduced in more detail. The fusion component checks for tree validity whenever the tree has been changed. In particular, this happens when a CAM was received, transformed into a cluster member node, and added to the aggregation tree. Similarly, the validity check is performed when a second tree was integrated into the tree of the node. Furthermore, the process may be triggered after the dissemination process to account for possible partial dissemination executed by the dissemination component. In total, five properties of the tree are checked for validity:

- Check node type

The component checks if the type of each node fits the required type for its level as specified by the aggregation level control. If a cluster member node is found where cluster node was expected, the cluster member node is added as a child to the fitting cluster node. If such cluster node does not exist, one is created. In contrast, if cluster node is found where super cluster node was expected, the cluster node is fused and all its children are added as children to the super cluster.

- Check children count

  A second step in the validation process is to check if the number of children of each node meets the requirements specified by the aggregation level control. If the number of children exceeds the limit specified by the aggregation level control, multiple cluster and cluster member nodes are merged until the requirements are satisfied.

- Check extrema count

  During the extrema count check, the fusion component verifies that the number of extrema nodes does not exceed the limits specified by the aggregation level control. If the number of extrema does exceed the requirements, those extrema with least fusion costs are removed until the requirements are satisfied.

- Check cluster length and width

  During this phase, the fusion component checks if the length and width of a cluster node is too small. If such a cluster exists, length and width are increased to the current minimum cluster length and width size.

- Check for duplicate cluster length and width

  The component checks if there are two identical cluster nodes with same cluster length and width exists in the aggregation tree. This may happen when two aggregation trees are merged. If there are duplicate clusters of single direction, one of them is removed and all its children are added to the other cluster. After all properties have been validated, the tree fits the requirements of the aggregation level control and can be disseminated.

5.3.4 Dissemination component

The data dissemination component describes when and how data is disseminated by a RSU/CH to the next RSU/CH in the direction of the control center. Two basic principles to trigger data dissemination can be distinguished. This dissemination component is hybrid approach that combines both periodic and event driven approaches. The idea of hybrid approach is to combine the robustness of the periodic dissemination with the little delay of the event-based approach. Similarly to the event-based approach, the tail node disseminates its data periodically with a predefined frequency. Other nodes forward the aggregated data whenever they receive aggregated data from the previous node. To prevent large delays in case of connectivity loss between two CHs, each node keeps track of when it started its last dissemination process. If the last process is longer than $\frac{1}{f} + \sigma$, σ being a certain tolerance, it starts the dissemination process without waiting for the aggregated data from previous node.

5.3.5 Adaptive control component

This section describes the controllers and communication control. The goal of the component is to satisfy reliable delivery, predicable delay, high precision, and scalability. Reliable delivery means the number of data packets lost in the system should be low. Predictable delay means the maximum delay of data until it arrives at the application must be under a threshold of 2 seconds. High precision means the aggregated data that arrives at the control center should have little difference from the original data. Scalability means the aggregation scheme must be able to provide a sufficient quality of service when scaling up the length of the road. The packet loss in a VANET can be controlled by limiting the channel load using the density of vehicles. The packet loss increases with increasing density. The adaptive control, therefore, monitors the density and controls it with the adaptive aggregation scheme to maintain a target density with acceptable packet losses.

Table 5.2 provides an overview of the three controllers of the adaptive control. Each of these controllers can execute different actions when being triggered. Two controllers, the requirements and dissemination period controller, are operated by the control RSU or control center, while the aggregation level controller is executed on each RSU or CH individually. This is the case because the dissemination frequency and the requirements should be equal on all RSUs or CHs. Thus, the control unit defines these parameters. In contrast, the aggregation level should be individually controlled by each RSU or CH and does not need to be the same on all units. The requirements controller defines the required metrics and fusion parameters. It also defines an initial dissemination frequency. This controller is triggered during initialization of the infrastructure. Once other nodes receive the requirements, they start to collect the requested data. At run-time, the dissemination period controller observes the delay of incoming information at the control unit and adjusts the requested dissemination frequency when the delay exceeds the desired delay r_d with a certain tolerance σ_d. The aggregation level controller, on the other hand, is executed on each individual RSU or CH. This controller observes the density and adjusts the aggregation level if the density exceeds or lowers particular level. All of these three controllers will be introduced in more detail below.

Actor	Controller	Trigger	Effect
Control center	Requirements Controller	Initialization	Broadcast Requirements
	Dissemination period controller	$delay > r_d + \sigma_d$	Increase Dissemination Freq.
		$delay < r_d - \sigma_d$	Decrease Dissemination Freq.

Table 5.2: Adaptive control overview

5.3.5.1 **Requirements controller**

The requirements control is executed on the control center. Its task is to broadcast the requirements issued by the control RSU or control center to all other RSUs or CHs. This broadcast is executed for the first time during initialization of the infrastructure. With this initial broadcast, all RSUs or CH are informed what data to collect and when to forward it to the control unit. Specifically, the requirements contain the following information:

- **Aggregation tree design**

The aggregation tree design specifies how the aggregation tree should be structured. This design includes the following information for each level of the tree:

  - Node type of all nodes in the level.
  - Metric Id of those metrics that should be contained in the level.
  - Maximal number of children of nodes of this level.
  - Number of extrema values.
  - The maximum size of the cluster length and width.
  - Separation of nodes travelling in opposite direction.

- **Dissemination frequency**

  As argued before, the dissemination strategy used in this thesis is a hybrid approach of periodic and event-based dissemination. The tail RSU or CH periodically forwards its data and the other nodes aggregate their own data with the received ones and disseminate when they receive aggregated data. The dissemination frequency defines how often the tail RSU or CH disseminates its data. The control unit decides how the aggregation tree is built using the requirements. The aggregation tree design contains this information by defining the node type, metrics, and other information for each level of the tree. The requirements control can issue new requirements when necessary. This can be performed when the data delay is too high and the dissemination period control increases the dissemination frequency, for instance.

5.3.5.2 **Dissemination period controller**

Similar to the requirements control, the dissemination period control is executed only at the control unit. It monitors the delay of incoming packets and adjusts the dissemination frequency accordingly. Then, the requirement controller is notified to broadcast the new requirements to all other RSUs or CHs. Whenever the delay exceeds a certain threshold, the dissemination frequency is increased. When the delay is lower than a second threshold, the frequency is reduced.

5.3.5.3 Aggregation level controller

The aggregation level control probes the network and adjusts the level of aggregation whenever the density is high or low. For that reason, it defines different level of aggregation in which with increasing aggregation level more data is fused and the amount of data in the network reduced. These levels are defined so that the highest aggregation level still satisfies the minimal requirements specified by the requirements control. For instance, a node may have infinite children in aggregation level 0. At aggregation level n - 1, on the other hand, only x children are allowed while x is the minimal number of children as specified in the requirements. Different strategies to define aggregation levels will be introduced in the remainder of this thesis. The adaptive control and the other modules of the aggregation framework, introduced in this section, allow to design different aggregation schemes. They can be defined by the choice of the different components and by the specification of the aggregation tree they use.

$$DW = \frac{D_{max} - D_{min}}{n} \tag{5.9}$$

Aggregation Level 0	$0 < D < 20$
Aggregation Level 1	$20 \leq D < 40$
Aggregation Level 2	$40 \leq D < 60$
Aggregation Level 3	$60 \leq D < 80$
Aggregation Level 4	$80 \leq D < 100$
Aggregation Level 5	$100 \leq D < 120$
Aggregation Level 6	$120 \leq D < 140$
Aggregation Level 7	$140 \leq D < 160$
Aggregation Level 8	$160 \leq D < 180$
Aggregation Level 9	$180 \leq D \leq 200$

Table 5.3: Aggregation levels based on the density of vehicles in the cluster

In this thesis, I propose N-density Control, where two extreme values that are minimum and maximum number of possible vehicles in the cluster in a certain time and the number of aggregation levels are considered. The density window size (DW) for each aggregation level can be calculated from equation (5.9). Where

D_{max} is the maximum density of vehicles possible in a cluster, D_{min} is minimum number of vehicles in the cluster possible and n is number of aggregation levels. This method can be explained further using an example, from Table 5.3 n is considered as 10, $D_{max} = 200$, and $D_{min} = 0$. The DW size is calculated and each window size is assigned to each aggregation level. Firstly each CH checks the number of vehicles in the cluster using the CAM message it received recently and compares with the aggregation level size. If there is a change then it updates its aggregation level. In density control the CH checks its number of CMs every $1sec$.

5.4 Cluster based semantic data aggregation (CBSDA)

One of the key component of all aggregation schemes is their data structure. The data structure supports storing different data types, combining data of multiple sources into one structure and size reduction by data fusion. The level of aggregation is also easy to adjust using the tree as data structure. In general, the fewer nodes the tree contains, the less bandwidth it needs during transmission. The CBSDA scheme uses the aggregation framework discussed in Section 5.3. In this scheme, I use adaptive control with a more complex aggregation tree and aggregation levels. The aggregation tree used in CBSDA scheme shown in Figure 5.11 uses super cluster, cluster and CM nodes. The cluster node contains the number of vehicular data it represents, cluster length and cluster width, while CM nodes contains only vehicular data of a single vehicle. The super cluster node consists of three metrics the length and width of the cluster and time stamp. The aggregation tree layout used in CBSDA scheme allows multiple ways to reduce its size.

Aggregation level	0	1	…	4	…	9
Cluster length (CL)	$CL \geq 50$	$\frac{CL_{max}}{2^8}$	…	$\frac{CL_{max}}{2^5}$	…	$\frac{CL_{max}}{2}$
Cluster width (CW)	$CW \leq 10$	$CW \leq 10$	…	$CW > 10$	…	$CW > 10$
Max Child Count	∞	9	…	4	…	1

Table 5.4: Cluster based adaptive control

First, the number of super cluster and cluster node children could be limited. Super cluster nodes represent vehicles travelling in each direction. The cluster nodes under one super cluster contains vehicles travelling in same direction. Moreover, the number of super cluster is always limited to two. The number of cluster nodes itself cannot be limited because there must always be enough clusters to cover all cluster member nodes.

A second way to reduce the tree size is to re-size the cluster and super cluster nodes. When keeping the children limit constant and increasing the size of the cluster length and width, the tree shrinks and cluster nodes are fused. In contrast, when the size of the cluster width and length is reduced, more cluster nodes are created in order to cover the cluster member nodes. Thus, less cluster member nodes fall into the same cluster and fewer nodes must be fused, on average. If the cluster width and length are re-sized, it is important that the cluster length and width never exceeds the current requirements issued by the control unit.

For in-network aggregation, it is desirable that cluster length and width of different nodes can be easily merged. For that reason, the size of the cluster is restricted to values of 2^x. This ensures that many CHs use the same cluster metrics which can be merged easily. Table 5.4 shows an example configuration of different aggregation levels for the cluster based adaptive control, where CL_{max} is the maximum cluster length size defined by the requirements controller and N aggregation levels are used. The major advantage of this aggregation tree layout is the bounded imprecision due to the cluster and super cluster nodes. Using the example of Figure 5.11, the fused value for super cluster 1 with length metric between 0 to 1000 m. The siblings fused in super cluster 1 should be within this cluster length. Independent of the decision component, this can ensure high precision for some metrics. There are two variations in this adaptive control. Either, the cluster length metric is extracted from the cluster member nodes once they are inserted into a cluster, or the metric is kept same. When the length and width metric is kept same, no size reduction occurs. The tree will only prevent nodes from different clusters and vehicles in opposite direction to merge. If the length and width metric is removed from the cluster node, the information of the exact value of the cluster member node within the length and width is lost. This, however, may be acceptable since the maximum imprecision is already limited by the requirements. When this is the case, the amount of data in each cluster member node is reduced with each cluster length and width level. Throughout this thesis, a cluster based aggregation scheme is used where the length and width metric is not removed from the cluster nodes. Thus, the length and width are only used to restrict the maximum error, not to reduce the data load.

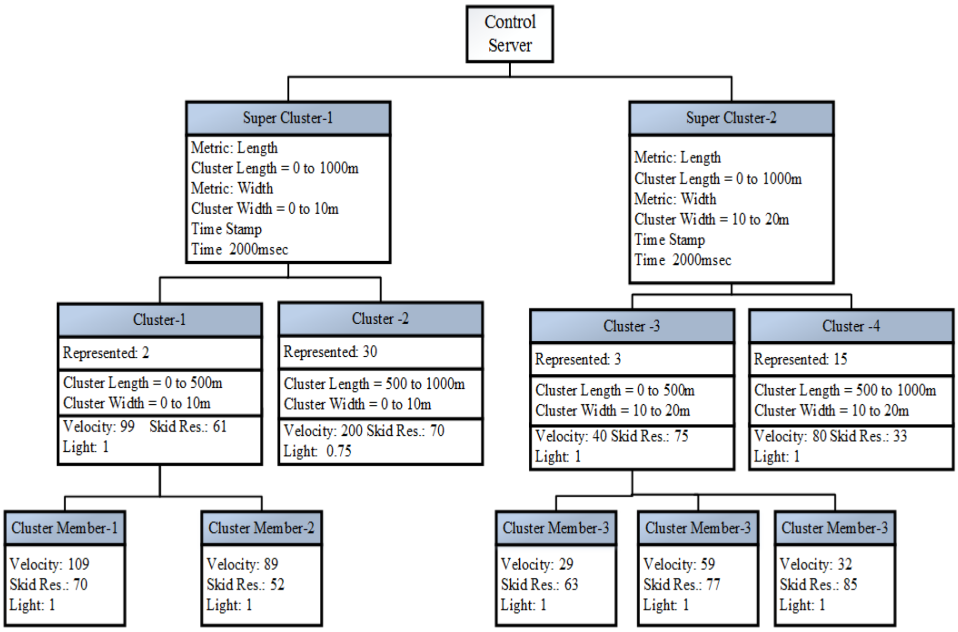


Figure 5.11: CBSDA data structure

The first super cluster metric is cluster length which is user defined. The three metrics of super cluster are useful when the application requires a defined resolution of data quality. The cluster length can be anywhere between minimum of 50 meter and maximum of 1000m shown in 5.4. Second metric defined is the cluster width, this defines how many lanes are considered in the super cluster. Third metric used here is the time-stamp metric, which records the time the data item was broadcasted by a CM to the CH. Thus, the imprecision of the length, width and time-stamp will never be higher than the specified maximal sizes. The number of super cluster nodes itself can be limited by assigning appropriate value to cluster length and width compared to the road length and width. Moreover, the number of aggregation level also depend on the cluster length and width. When keeping the children limit constant and increasing the cluster length of the super cluster, the tree shrinks and cluster, CMs are fused. In contrast, when the cluster length of the super cluster is reduced, the tree grows. The CBSDA scheme using adaptive aggregation level with different decision schemes are simulated to evaluate the performance. The precision is expected to be best when there are more super clusters then there will be less fusion of cluster and CMs.

5.5 Summary

In this chapter, I presented a cluster based semantic data aggregation scheme for VANETs (CBSDA), in which the data structure is a tree which consists of three different nodes super cluster, cluster and cluster member node. An aggregation framework is introduced, where it consists of fusion, dissemination, aggregation tree, adaptive control and decision components. Adaptive control consists of aggregation level control, which is based on the density of vehicles. Aggregation level increases and decreases based on the density of vehicles in the current segment. Three types of schemes for decision component is proposed to increase the precession by fusing data of similar nodes. The comparison of results of different decision schemes will be discussed in Chapter 6.

6

Results

This chapter describes the simulation results of the three protocols introduced in Chapters 3, 4, and 5. The simulation is done using different network simulators (ns-3, ns-2) under different traffic scenarios. This chapter, firstly introduces the network simulator, which is used for the evaluation. Subsequently, the simulation results of the D-CBM protocols is evaluated in Section 6.2. Next, the main parameters of the D-CBM protocol and their effect, on the cluster head and cluster topology are examined.

In Section 6.3, simulation results of DA-CMAC protocol is evaluated for different parameters. Moreover, Section 6.4 describes the simulation results of CBSDA protocol for different decision schemes. The evaluation shows, if the schemes satisfy the requirements listed in Chapter 5. In particular, the data load reduction and data precision aspects of the CBSDA schemes for different decision schemes are examined.

6.1 Simulator

This section describes the discrete-event network simulator ns-2 and ns-3 that was used in this thesis to implement and evaluate different protocols. The C++ and

TCL based ns-2 simulator was used in version 2.34 and is licensed under the GNU GPLv2 software license. It has a modular architecture that allows to extend the functionality of the simulator easily. The decision for this network simulator for D-CBM was made due to the availability of a LEACH patch that gives easy implementation of clustering and cluster head election in VANETs. The real simulation models using SUMO is generated and used as input to MOVE to generate mobility traces that were used as input to the ns-2.34 simulator.

Secondly, ns-3 was used to evaluate DA-CMAC and CBSDA protocols. ns-3 is an improved version of ns-2, where ns-3 uses only C++ programming language. The decision for this network simulator was made due to the availability of useful modules, some were already implemented in ns3 such as OLSR modules useful for the clustering and cluster head election, some of the MAC models were from KIT (STDMA)[Gaugel 13], and Hemanth Narra (TDMA) [Narra 13]and other models from Fraunhofer Institute for Embedded Systems and Communication Technologies ESK, that allow to simulate a vehicular network.

6.2 Simulation results of D-CBM protocol

This section describes the experimental setup and methodology used to evaluate the performance of the D-CBM protocol, including the network and traffic scenarios used in the evaluation. The evaluation metrics will be described in later sections. I assumed all vehicle on the road are equipped with GPS and DSRC transceivers.

I ran simulations using ns-2 network simulator [Laboratory 10]. For VANET, I used mobility traces for highway scenario generated by the micro-traffic simulator SUMO. The goal was to create a real vehicular network on highways with different number of vehicles. Vehicles travel in one lane with varying speeds. Some of the vehicles exit in intersections along the highway and some vehicles joined the highway. In simulations, I have tried to incorporate some of the characteristics of urban scenarios like low speed and exits. When the destination is reached the vehicle stops for a random time and then repeat the same process. The road segment patterns have features such as safety distances between two vehicles, the speed limit, and the maximum number of vehicles per lane per km is obtained from Figure 6.1 [Joaquim 13]. Finally, these generated movement patterns are used as

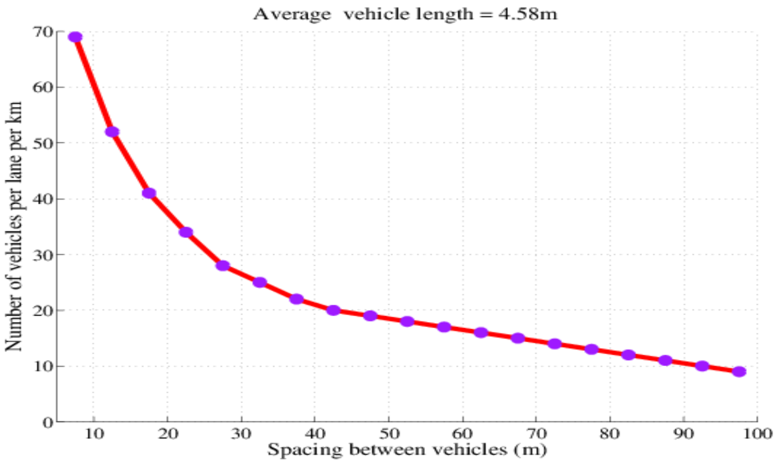


Figure 6.1: Number of vehicles per lane per km for an average vehicle length of 4,58m

input in ns-2. Different runs are used to obtain different simulation results for the same scenario. The parameters are summarized in Table (6.1). The other PHY layer parameters follow the default settings of IEEE 802.11p, where the Nakagami propagation model is used [Murray 08]. The length of the simulation scenario is $10km$. Vehicles speed ranges from 40 to $120km/h$, which is typical for highways.

I set the transmission range for vehicles to 300 m [Weimerskirch 09]. The transmission rate is set to be 6 Mbps, which was shown to be the optimal data rate for VANETs [Weimerskirch 09]. The update messages are of size 200 bytes, which is a reasonable size for safety/update messages [Weimerskirch 09]. These messages include the basic vehicle's information, such as speed, position, direction, and velocity on the road. Besides the basic information, the vehicle will include its cluster head election weight. The weighting value is calculated by every vehicle trying to join the cluster based on the calculations explained in Section 3.1.2. When the vehicles are in the process of creating the cluster, they will be sending their update messages using CSMA/CA mechanism to access the medium.

6.2.1 Convergence time

Convergence time is the time required by the CM to achieve a TDMA slot in a CH frame. For the purpose of simplifying analysis and highlighting the effects of time slot allocation on delay performance, stresses are laid on the delay from

Parameter	Value
Data rate	$6Mbit/s$
Frequency	$5.9GHz$
Transmission power	$15dBm$
Highway length	$10km$
Number of vehicles in each direction	900
Speed of vehicles	$20-40m/s$
Number of clusters in each direction	10
Cluster radius	$300m$
Propagation model	$Nakagami$
Safety packet size	$200bytes$

Table 6.1: Network parameters used in the simulation for the D-CBM protocol

the vehicle that joins the cluster to its slot schedule. The delay of slot allocation consists of maximum and minimum delay upon the allocation of a slot in the frame considering the arrival of the vehicle in the start of the frame. CH frame length for CM packets is T, time slot length is T_s,where $T_s = \frac{T}{N}$. The maximum slot allocation delay is

$$D_{max} = 3\left(T + T_{CH}\right) + 2\left(T_{RSU} + T_{CH-CM}\right) - T_s \tag{6.1}$$

The minimum packet delay is

$$D_{min} = T + T_{CH} - (N-1)T_s \tag{6.2}$$

Note that the allocation delay varies continuously from D_{max} to D_{min} with the relative between vehicle arrival time and time slot allocation. Thus, the average slot allocation delay could be expressed by

$$D_{avg} = \frac{D_{max} + D_{min}}{2} \tag{6.3}$$

The Figure 6.2 shows the slot allocation time versus vehicle arrivals to a cluster with different values of the packet size. For this experiment, I have considered one cluster where the vehicles arrives to the cluster based on the arrival process discussed in Section 4.1.3. Initially, the number of CMs in a cluster is considered

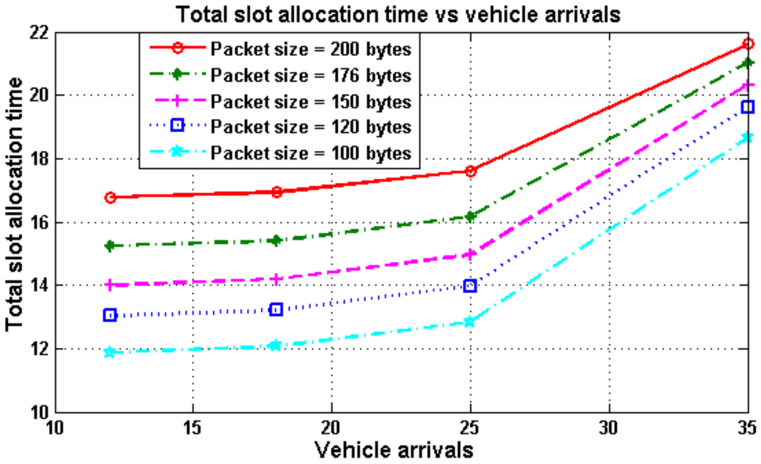


Figure 6.2: Time slot allocation time vs vehicle arrivals

0 and the number of CMs increases when the number of vehicle arrivals increases. From the Figure 6.2 its clear that when the number of vehicle arrivals increases the slot allocation time also increases. When the number of vehicles become more than the number of slots available, then convergence time increases further. In Figure 6.2, the packet size is varied from 100 bytes to 200 bytes. When packet size increases the number of slots in the frame decreases, thereby increases the slot allocation time.

6.2.2 Cluster stability

Stable clusters are important for an efficient and reliable information transfer. Stable clustering methods reduce communication load of re-clustering and lead to an efficient use of available bandwidth. Cluster stability depends on the selection of a suitable CH and cluster formation to ensure greater cluster residence times by reducing number of cluster change events. If vehicles are changing their mode very frequently and remain only for a short period of time in the CH state, stability of CH is low.

Cluster stability can be calculated as the average number of CH changes in the entire simulation. Association time is defined as the percentage of time in which vehicles were CMs of a particular cluster. Stability of CHs significantly affects the performance of MAC protocol. When CHs change their status to CM or GV or exit the road, then CMs has to re adjust to the new slot structure. In this thesis, I consider the average CH lifetime means the sum of the lifetime of each CH selected divided by the total number of elected CHs in the entire simulation.

Weights	**Vehicle density**	**Speed variation (m/sec)**	**CH life-time (sec)**	**Number of CH changes**
$W_1 = 0, W_2 = W_3 = 0.5$	900	25-35	23	34
$W_2 = 0, W_1 = W_3 = 0.5$	900	25-35	20	36
$W_3 = 0, W_2 = W_1 = 0.5$	900	25-35	131	11
$W_1 = W_2 = 0.33, W_3 = 0.34$	900	25-35	73	17
SBCA	900	25-35	50	22
$W_1 = 0, W_2 = W_3 = 0.5$	500	25-35	18	37
$W_2 = 0, W_1 = W_3 = 0.5$	500	25-35	19	42
$W_3 = 0, W_2 = W_1 = 0.5$	500	25-35	129	12
$W_1 = W_2 = 0.33, W_3 = 0.34$	500	25-35	70	18
SBCA	500	25-35	42	26
$W_1 = 0, W_2 = W_3 = 0.5$	900	5-12	52	24
$W_2 = 0, W_1 = W_3 = 0.5$	900	5-12	47	26
$W_3 = 0, W_2 = W_1 = 0.5$	900	5-12	220	6
$W_1 = W_2 = 0.33, W_3 = 0.34$	900	5-12	120	16
SBCA	900	5-12	71	17
$W_1 = 0, W_2 = W_3 = 0.5$	500	5-12	48	28
$W_2 = 0, W_1 = W_3 = 0.5$	500	5-12	43	29
$W_3 = 0, W_2 = W_1 = 0.5$	500	5-12	198	8
$W_1 = W_2 = 0.33, W_3 = 0.34$	500	5-12	89	14
SBCA	500	5-12	66	20
$W_1 = 0, W_2 = W_3 = 0.5$	900	5-35	34	32
$W_2 = 0, W_1 = W_3 = 0.5$	900	5-35	32	34
$W_3 = 0, W_2 = W_1 = 0.5$	900	5-35	179	9
$W_1 = W_2 = 0.33, W_3 = 0.34$	900	5-35	103	15
SBCA	900	5-35	86	14
$W_1 = 0, W_2 = W_3 = 0.5$	500	5-35	29	34
$W_2 = 0, W_1 = W_3 = 0.5$	500	5-35	28	36
$W_3 = 0, W_2 = W_1 = 0.5$	500	5-35	156	10
$W_1 = W_2 = 0.33, W_3 = 0.34$	500	5-35	97	15
SBCA	500	5-35	79	16

Table 6.2: Results of different simulation scenarios of D-CBM using single-hop clusters

The Figure 6.3 shows the average CH lifetime with respect to the number of vehicles for different weighing factors W_1, W_2, W_3. The factors are varied to different values which sums to 1. In this scenario, I consider the minimum speed of

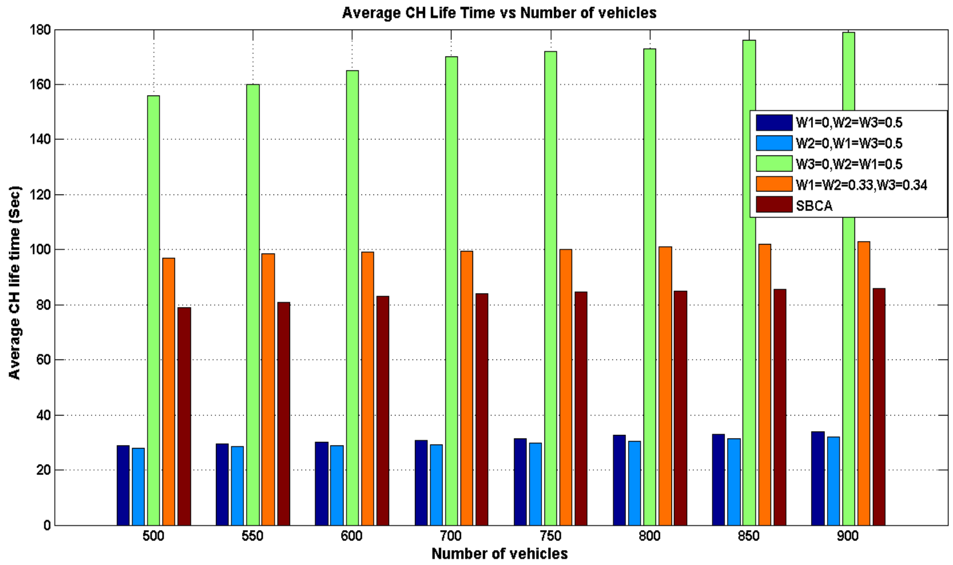


Figure 6.3: Average CH lifetime vs number of vehicles, when the speed variation of vehicles are between 5-35 m/sec

5m/sec to maximum speed of vehicles is 35m/sec, cluster width of 300 meters, and the number of CHs is 34.

From the Figure 6.3, when $W_1 = W_2 = 0.5, W_3 = 0$ are given to factors it performs better compared to the other values and have higher lifetime. So stability is higher, when the weight parameter of RSU distance is zero. Moreover, this is due to the fixed position of RSU in the road. Figure 6.3 shows the impact of the number of vehicles on the CH lifetime of the proposed D-CBM with different values of weighting parameters and SBCA protocol. D-CBM with $W_1 = W_2 = 0.5, W_3 = 0$ and $W_1 = W_2 = 0.33, W_3 = 0.34$ have higher CH lifetime compared to SBCA protocol. The CH lifetime for different variations of weighting parameter for different density of vehicles is given in Table 6.2. CH lifetime of D-CBM with equal weighting factors for 900 vehicles is $103sec$ compared to $86sec$ of SBCA protocol.

Figure 6.3 shows that the D-CBMs average CH lifetime increases slightly when the average number of vehicles increases. This is because the inter distance between CMs are decreasing; hence, the relative speed between them is decreasing, resulting in a high CH stability. Figures 6.4 and 6.5 shows the impact of the speed of

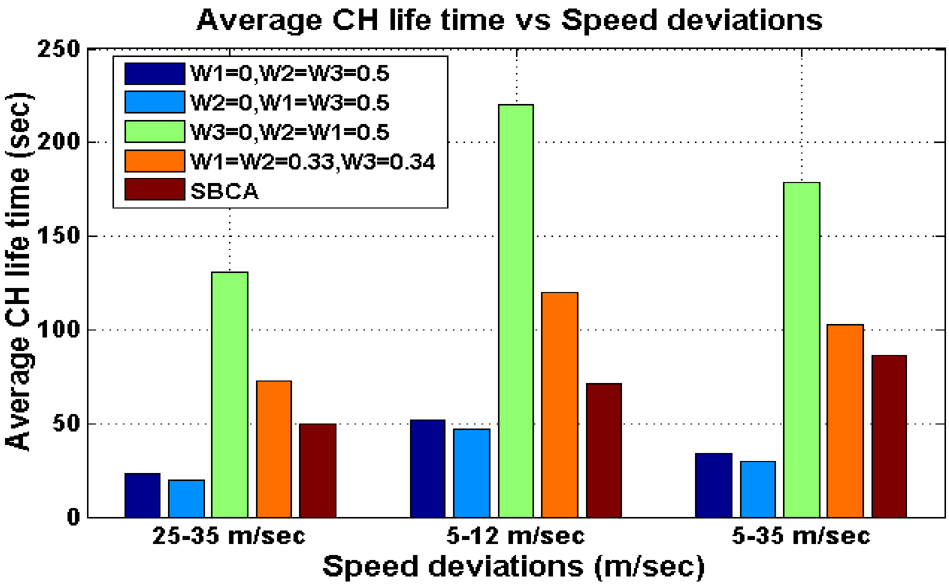


Figure 6.4: Average CH life time of D-CBM for different speed deviations with total number of vehicles = 900

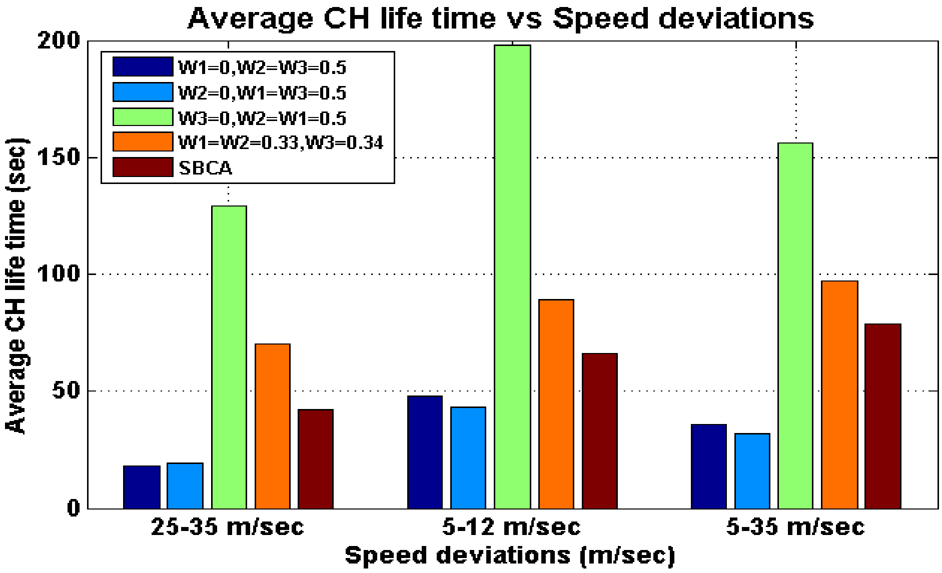


Figure 6.5: Average CH life time of D-CBM for different speed deviations with total number of vehicles = 500

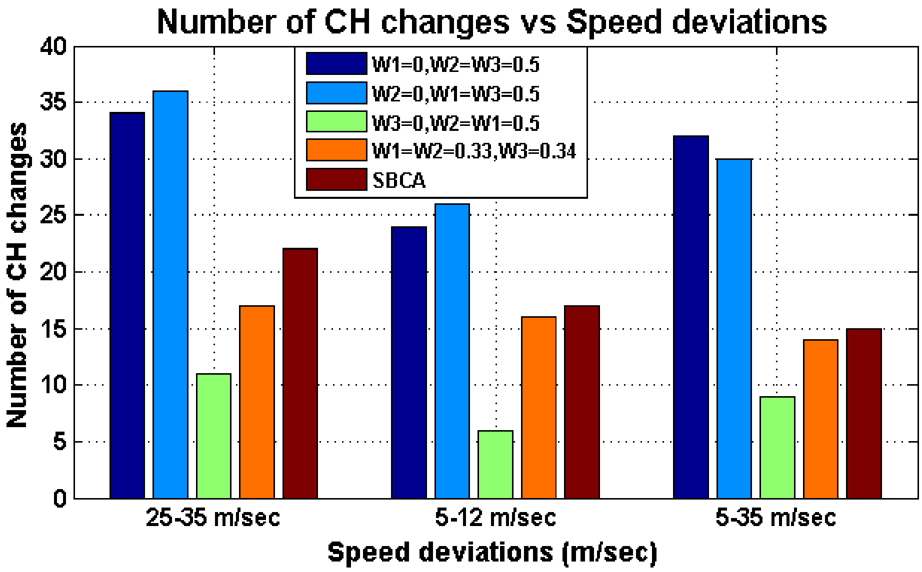


Figure 6.6: Number CH changes for different speed deviations with total number of vehicles = 900

vehicles on the average CH lifetime. The speed of the vehicles are categorized into three different variations 5-12 m/sec, 25-35m/sec and 5-35m/sec. In 5-12 m/sec, vehicles travels in low speed and the CH life time is higher. Moreover, the D-CBM with $W_1 = W_2 = 0.5, W_3 = 0$ have higher CH life time compared to SBCA protocol [Ahizoune 12]. Furthermore, D-CBM with $W_1 = W_2 = 0.5, W_3 = 0$ for speed deviations $5 - 12m/sec$ have higher CH lifetime compared to D-CBM with other values of weighing parameters for speed deviations $5 - 35m/sec$ and $25 - 35m/sec$. The average CH lifetime decreases when the difference between the maximum and minimum speed of the vehicles in the road increases.

Figures 6.6 and 6.7 shows the number of CH changes versus different speed deviations for D-CBM and SBCA protocols with different traffic densities. $W_1 = W_2 = 0.5, W_3 = 0$ with speed deviation $5 - 12m/sec$ have fewer CH changes compared to SBCA protocol. In this simulation, the number of CH are fixed irrespective of the density of vehicles. Comparing Figures 6.6 and 6.7, the $W_1 = W_2 = 0.5, W_3 = 0$ with 900 vehicles have few number of CH changes compared to the same with 500 vehicles.

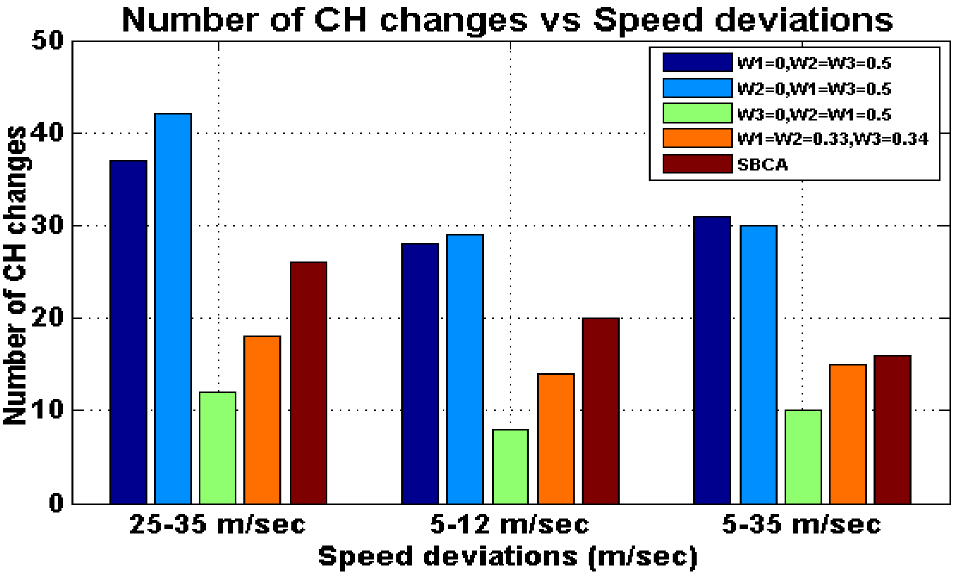


Figure 6.7: Number CH changes for different speed deviations with total number of vehicles = 500

6.2.3 Cluster reconfiguration

The frequent cluster and CH reconfiguration creates a enormous amount of overhead, which significantly reduces the efficiency of the available bandwidth. The cluster reconfiguration is needed in some cases when the distance between two CH is below the threshold. In some approaches, when the distance between two neighbouring CH vehicles goes below a particular threshold, the cluster who has fewer CMs is dismissed to reduce communication load while its CMs join other neighbouring clusters. If a large value of the threshold is considered then it can lead to a a higher probability of cluster reconfiguration and higher rate of CH changes. The threshold value is related to the transmission range and the probability of cluster changes is related to the transmission range. The larger transmission range provides longer distance for CHs to detect CMs, and therefore, more number of CMs means less number of reconfigurations. Figure 6.8, shows the impact of various communication ranges on the average lifetime of CHs. Figure 6.8 illustrates that CHs average lifetime increases by increasing the transmission range. As a result, the cluster stability is high in the higher transmission range. Moreover, the speed deviation for this simulation is between $5 - 35m/sec$ and number of vehicles = 900.

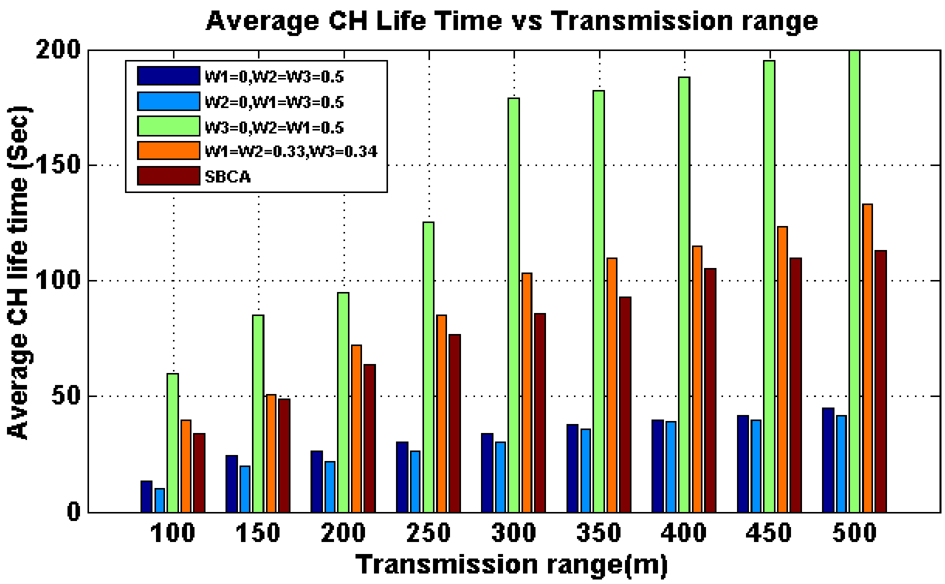


Figure 6.8: Average CH lifetime vs transmission range, when the speed variation of vehicles are between 5-35 m/sec

Furthermore, $W_1 = W_2 = 0.5, W_3 = 0$ and $W_1 = W_2 = 0.33, W_3 = 0.34$ have higher CH lifetime compared to SBCA protocol for different transmission ranges.

6.2.4 Cluster delay

Cluster delay means the time required for sending one message from the CM to the RSU. The delay parameter is very crucial for safety message delivery. The end to end delay should be minimised by selecting proper cluster size, moreover the selection of proper MAC protocol reduces the channel access time, as well as the selection of a stable CH nearer the RSU. In addition, the time delay required for sending one message from the CM in one cluster to RSU is considered here. $T_{delay,RSU}$ is the total time delay for delivering one message from CM to RSU. To assure the timely delivery of active safety messages, the maximum delays for delivery should be less than the required delivery delay of safety message i.e, $T_{delay,RSU} \leq T_{Safety}$.

In this simulation, I consider $W_1 = W_2 = 0.5, W_3 = 0$ for D-CBM with different types of slot allocation, number of vehicles is 900, and cluster width is 300m. In order to verify different slot allocation schemes, I create a traffic jam in the middle of the road for three minutes and the calculate the average packet

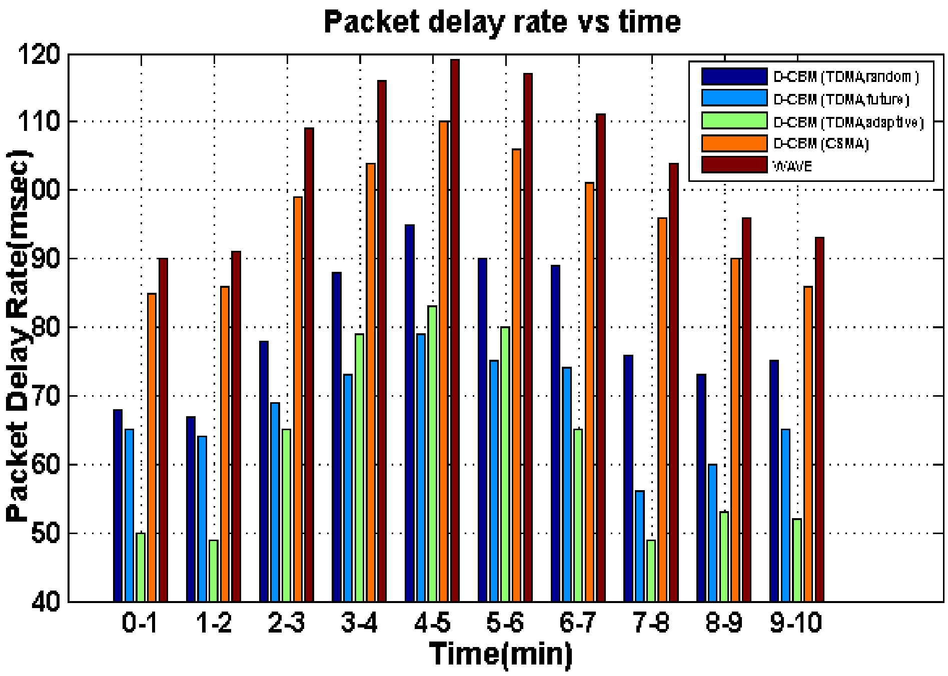


Figure 6.9: Average packet delay rate for clusters 10, 11and 12

delay rate for three neighbouring clusters. The clusters that are mainly affected by traffic jam are clusters 10, 11, and 12. During traffic jam, the speed of vehicles will be between $5 - 7m/sec$. In normal traffic, the speed of vehicles is between $5 - 35m/sec$. In this experiment, I compare different slot allocation schemes in TDMA with D-CBM based on hybrid (CSMA) and WAVE standard. Average packet delay rate is compared with time.

In general, messages of safety applications have latency constraints and should be received within the time limit. From Figure 6.9, the delay of both hybrid (CSMA/CA) and TDMA (random allocation, adaptive standard score, and future position) based on D-CBM approach with WAVE standard. In D-CBM, all vehicles in the cluster are using their own time slots to communicate with other vehicles. These slots are assigned to them by the CH, to communicate with other vehicles; this assignment mechanism is explained in Chapter 4. In Figure 6.9, the delay rate before traffic jam, during traffic jam and after traffic jam are presented. The traffic jam starts at 2 min to 5min in cluster 12. The traffic jam spreads to cluster 11 and 10 respectively after few minutes. When traffic jam occurs, the number of

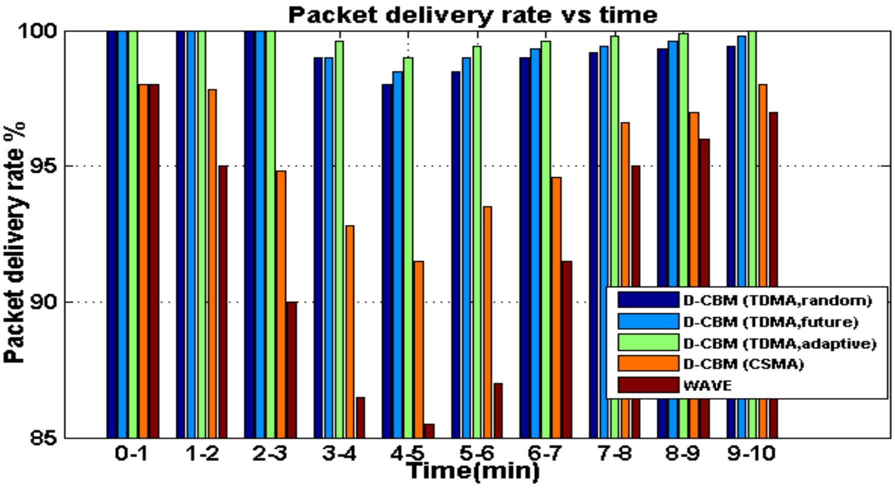


Figure 6.10: Average packet delivery rate for clusters 10, 11 and 12

CMs in each cluster increases there by increasing the number of channel access. From Figure 6.9, the packet delay rate during normal traffic is lower for D-CBM (TDMA. adaptive) compared to others. Moreover, the D-CBM (TDMA, adaptive) can allocate slots to the vehicles based on the speed deviations. During traffic jam, the speed deviation between vehicles is minimum and D-CBM (TDMA, future) performs better compared to others. Furthermore, after the traffic jam vehicles travels in normal speed and D-CBM (TDMA, adaptive) performs better compared to WAVE standard. Additionally, all D-CBM protocols perform better compared to WAVE standard. The TDMA based approach of D-CBM performs better compared to the hybrid (CSMA/CA) approach. This may be due to the busy channel or collisions during accessing the channel. Furthermore, both protocols achieve the deadline of $100ms$.

6.2.5 Packet delivery rate

The packet delivery rate (PDR) means the total number of packets successfully received in CH divided by the total number of packets generated in CMs. Average PDR is calculated for clusters 10, 11 and 12 with respect to time. Traffic jam is created during 2 to 5 minutes of the simulation. From Figure 6.10, the delivery rate decreases as the traffic jam is created in clusters 10, 11, and 12 during time 2 to 5 minute. When traffic jam occurs, then density of vehicles in clusters 10, 11, and 12 increases. Moreover, the PDR of D-CBM (CSMA) decreases sharply compared to D-CBM (TDMA). The PDR itself shows Figure 6.10 that D-CBM based on hybrid (CSMA) and WAVE standard is almost dropping equally during traffic jam. In case of D-CBM (TDMA, adaptive), the PDR is between 100-98% and this may be due to the fewer changes of CHs. From all preceding figures, it can be seen that the performance of D-CBM (TDMA, CSMA) exceeds the performance of WAVE standard. This is due to its feature of selecting a stable CH, and efficient slot allocation scheme. These features of D-CBM protocol helps to maintain a high reliability and predictability compared to WAVE standard, particularly in high density networks.

6.3 Simulation results - DA-CMAC

The network simulator ns3 version 3.18 was used for evaluation. The wireless channel assumes Nakagami highway propagation model with 6 Mbit/s data rate and 10 MHz bandwidth at 5.9 GHz for all communication. A realistic highway traffic scenario is used in the evaluation a 10 km highway with two lanes in each direction and 1800 vehicles randomly distributed on these four lanes. The mobility models assume that vehicles travel with a velocity between 20 - 40 m/s in the free traffic flow. After 2 minutes a sudden two directional traffic jam in the middle of the equipped road segment forces the velocity to drop to 8 - 10 m/s. For the next 4 minutes, vehicles queue in each direction are simulated. Afterwards, the traffic jam dissolves slowly for the next 4 minutes, restoring the original velocity distribution. Vehicles receive safety/update messages from other vehicles in both direction, but only relevant safety/update messages are taken into consideration for the further process. The frame length for DA-CMAC is 100 msec and 50 msec for both CCP and SCP. Ideally, the vehicle in DA-CMAC will be tuned to the CCH during the time interval; unless its own SCH slot time on the SCHs. Additional simulation parameters are depicted in Table 6.3.

Parameter	**Value**
Data rate	$6Mbit/s$
Frequency	$5.9GHz$
Transmission power	$15dBm$
Highway length	$10km$
Number of vehicles in each direction	900
Speed of vehicles	$20-40m/s$
Number of clusters in each direction	10
Cluster radius	$300m$
Propagation model	$Nakagami$
Safety packet size	$200bytes$

Table 6.3: Simulation setup of DA-CMAC protocol.

6.3.1 DA-CMAC frame

With the transmission cycle of 100 msec, the frame size of DA-CMAC is set to be 100 msec. Based on the above network configuration used in the simulation, we can find out the maximum number of vehicles that can be in one cluster using DA-CMAC. To calculate the maximum number of vehicles in the cluster, we need to find the time needed to transmit a 200 byte safety message on the CCH, or the time needed to transmit a 200 byte non-safety message on the SCHs.

For a 200 byte message, the time needed to be transmitted on the CCP and SCP is:

$$1600b \times \frac{1{,}000msec}{6{,}000{,}000b} = 0.267\text{msec}$$

And for a 100 byte mini slot message, the time needed to be transmitted on any of the SCP and CCP is:

$$800b \times \frac{1{,}000msec}{6{,}000{,}000b} = 0.134\text{msec}$$

Since the maximum slot size on the CCP and SCP is 0.267 msec and the frame size is divided into two sets SCP and CCP of 50 ms each, the CCP has 10 ms for mini slots in each cycle and 40 ms for normal slots on both CCP and SCP in one DA-CMAC frame. The 10 ms is divided into three for L, R, and U. The total mini slot length for L and R in each CCP and SCP is 4.5 ms. The total mini slot length for U in CCP is 1 ms. The number of mini slots in each L and R of CCP is

$$\left\lfloor \frac{Total-mini-slot-length(LorR)}{Mini-slot-size} \right\rfloor = \left\lfloor \frac{4.5}{0.134} \right\rfloor = 33 \text{ slots}$$

The number of mini slots in U of CCP is

$$\left\lfloor \frac{Total-mini-slot-length(U)}{Mini-slot-size} \right\rfloor = \left\lfloor \frac{1}{0.134} \right\rfloor = 7 \text{ slots}$$

The number of normal slots in CCP is

$$\left\lfloor \frac{Total-slot-length}{Slot-size} \right\rfloor = \left\lfloor \frac{18}{0.3} \right\rfloor = 60 \text{ slots}$$

The maximum number of cluster members possible in one cluster in DA-CMAC is 60 vehicles.

6.3.2 Evaluation Metrics

The metrics used to evaluate DA-CMAC and WAVE are to measure the reliability of safety/update messages and the throughput of the non-safety messages. The metrics are as follows:

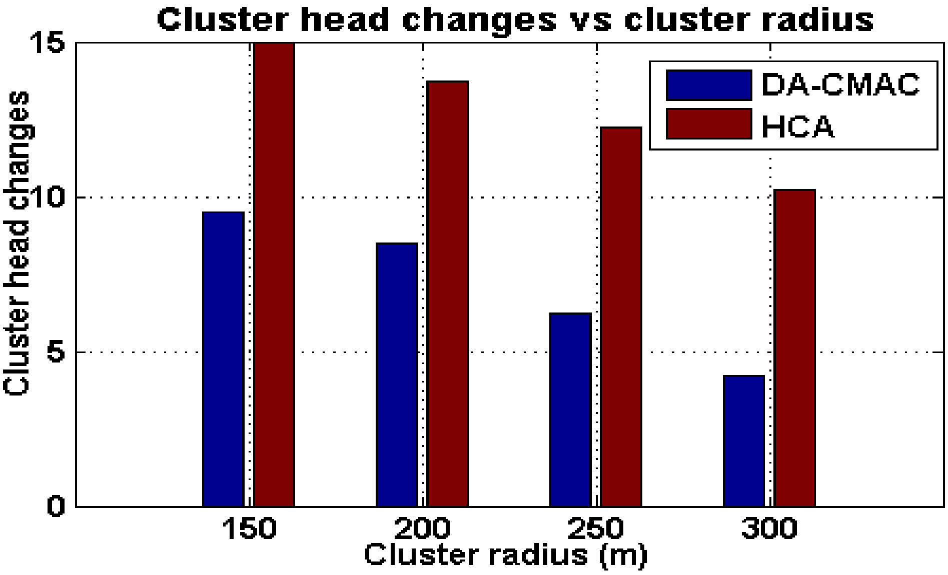


Figure 6.11: Cluster head changes vs cluster radius

Stable clusters are important for an efficient and reliable information transfer. Stable clustering methods reduce communication load of re-clustering and led to an efficient use of available bandwidth. Cluster stability depends on the selection of a suitable CH and cluster formation to ensure greater cluster residence times by reducing cluster change events. If vehicles are changing their mode very frequently and remain only for a short period of time in the CH state, stability of CH is low. As expected, the performance of DA-CMAC has varied depending upon the cluster radius. The number of nodes in the connected set decreases when the cluster radius decreases. Moreover, Figure 6.11 shows the number of CH changes decreases when the cluster radius is increased for the variable speed, where a traffic jam is created in the middle of the road for few minutes.

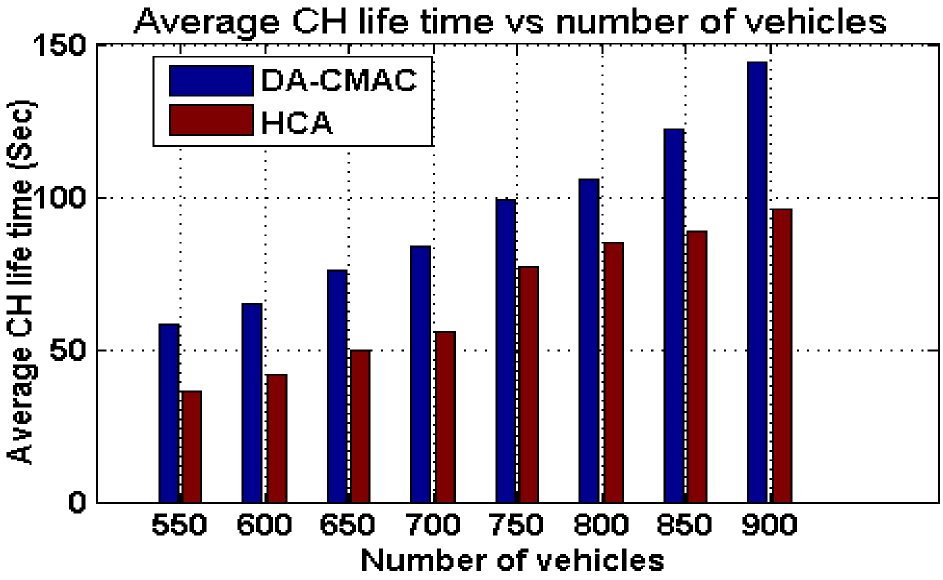


Figure 6.12: Average CH lifetime of DA-CMAC with different densities, when the speed variation of vehicles are between 5-35 m/sec

The Figure 6.12 shows the average CH lifetime with respect to the number of vehicles for DA-CMAC and HCA protocol. In this simulation, I consider the minimum speed of 5m/sec to maximum speed of vehicles is 35m/sec, and cluster width of 300 meters. From the Figure 6.12, DA-CMAC performs better compared to HCA protocol. So stability is higher for DA-CMAC protocol. Moreover, this is due to the number of connections, speed deviation and distance variation. Figure 6.12 shows that the DA-CMAC average CH lifetime increases slightly when the average number of vehicles increases. This is because the inter distance between CMs are decreasing; hence, the relative speed between them is decreasing, resulting in a high CH stability.

The lower CH changes increases the Packet delivery rate (PDR). PDR is the best parameter to measure the stability of the cluster. In this thesis, I define PDR as the total number of packets successfully received in CH divided by the total number of packets generated in CMs.

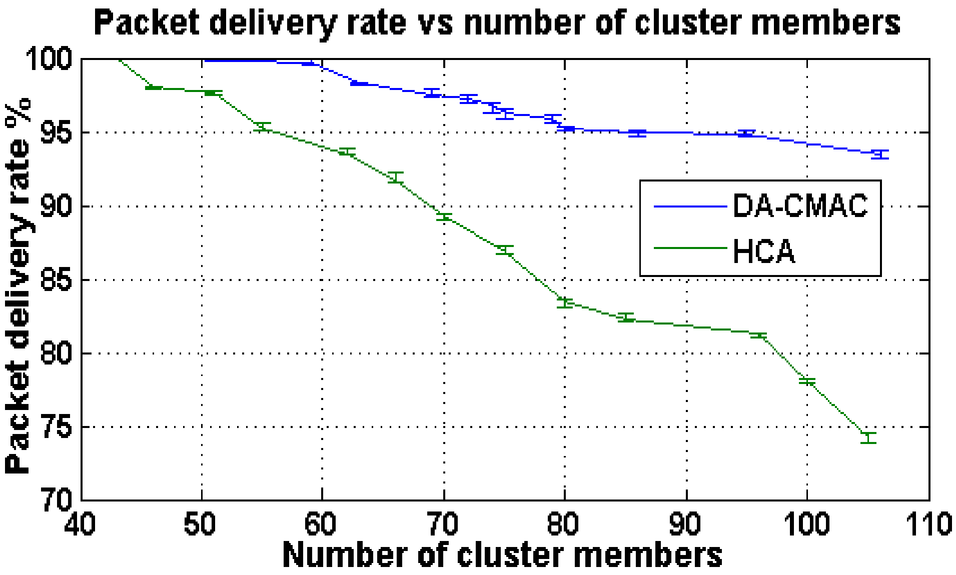


Figure 6.13: Packet delivery rate vs number of CMs

From Figure 6.13, the delivery rate decreases as the number of CMs increases. Moreover, the DA-CMAC and HCA [Dror 11] is almost ideal since it starts with a PDR which reaches about 100% reception probability and presents sharp falls in the PDR for HCA. In case of DA-CMAC, the PDR is between 100-94% and this may be due to the change of CHs. From Figure 6.13, it can be seen that the performance of DA-CMAC exceeds the performance of HCA protocol. This is due to its feature of selecting a stable CH and a backup CH to take over the main CHs responsibilities when CHs value is higher than the threshold. The MAC of DA-CMAC protocol helps to maintain a high reliability and predictability compared to HCA, particularly in high-density networks.

The access collision rate is defined as the average number of access collisions that happen within a slot in one hop neighbour. The overall access collision rates of all the DA-CMAC, HCA and WAVE under different traffic densities are shown in Figure 6.14. Access collisions in WAVE increases with the increasing traffic

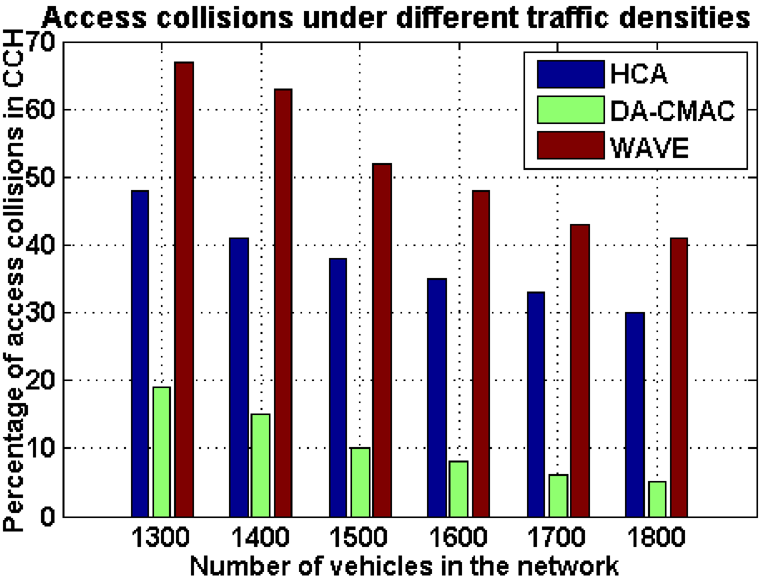


Figure 6.14: Access collisions under different traffic densities

density in one hop neighbourhood. However, access collisions in DA-CMAC is not higher as compared to WAVE due to the allocation of different sets of mini slots for vehicles travelling in opposite directions.

6.4 Simulation results of CBSDA scheme

For the evaluation of CBSDA scheme I have used network simulator ns3 version 3.18. It was enhanced by ITS modules enabling simulation of ETSI ITS-G5A and GeoNetworking protocols as well as positioning and simple mobility modules. The simulation is setup based on the parameters shown in Table 6.4. Two scenarios are simulated using ns-3. Firstly, the road is divided into different road segments. Each segment have an RSU and each RSU collects data from the vehicles in the corresponding road segment. The data is aggregated by each RSU and forwarded towards the direction of the control server. In second scenario, each segment elect a CH and aggregation take place in this CH, this aggregated data is forwarded to the control server. In the simulation setup, a highway scenario, where free flow is considered initially, and a traffic jam is created from 2 to 6 minutes in the middle of the road network is used to verify the aggregation scheme. Aggregation levels in CBSDA

Data Rate	$6MBit/s$
Frequency	$5.9GHz$
Transmission power	$15dBm$
Highway length	$10km$
Number Of Vehicles	700
Speed of vehicles	$20 - 40m/s$
Traffic Jam	$4min$
Number of RSUs	10
Number of Aggregation levels	10
Maximum CBR	0.43
Minimum CBR	0.27
Propagation model	$Nakagami$

Table 6.4: Simulation setup for both RSU and CH aggregation based CBSDA scheme

scheme are controlled by the density of vehicles in particular road segment. If the number of vehicles in the road segment increases the aggregation level increases to maintain the channel load below the maximum CBR.

In first scenario, aggregation takes place in each RSU. The aggregation in each RSU is controlled by the N density control based on aggregation level controller. The decision for fusion of data is taken by three different decision schemes adaptive cost-aware, weighted deviation, and adaptive standard score. In this section, I compare the effectiveness of the various decision schemes by comparing their precision of velocity and position in the server. Finally, the cluster length (CL) =1500 is fixed for the simulation setup. The main objective of the CBSDA scheme in the RSU is to reduce the load on the wireless channel. The CBR is defined as the ratio of the time a wireless device of a RSU is busy to the total time. The CBR for this simulation is set between 0.27 to 0.43. Figures 6.15, 6.16, and 6.17 show the channel busy ratio using CBSDA scheme with CL=1500 using adaptive cost-aware, weighted deviation, and adaptive standard score as decision component. The number of aggregation levels is set to 10 based on the density of vehicles.

From Figures 6.15, 6.16, and 6.17 its clear that the CBR of all RSUs are well below the maximum CBR even during congested traffic scenarios created from 2

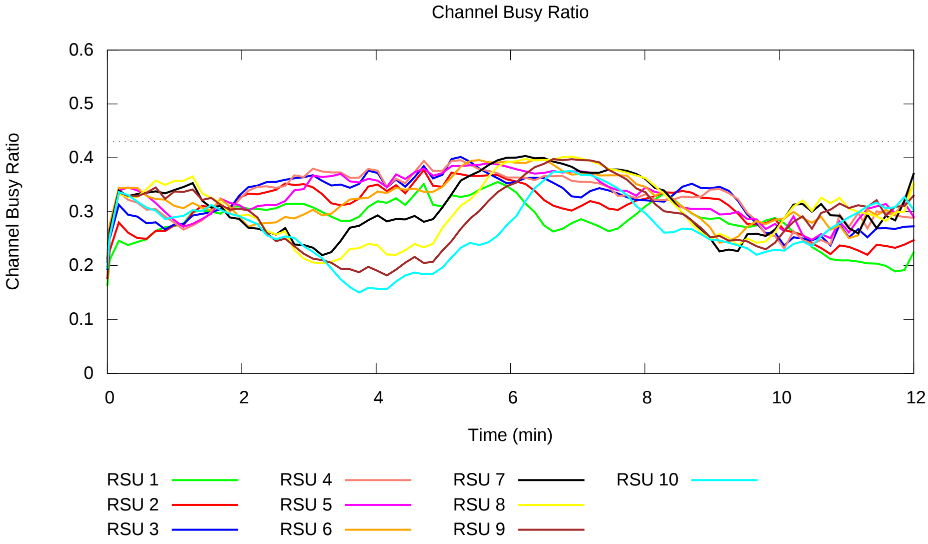


Figure 6.15: Channel busy ratio of adaptive cost-aware in RSU based scenario 1

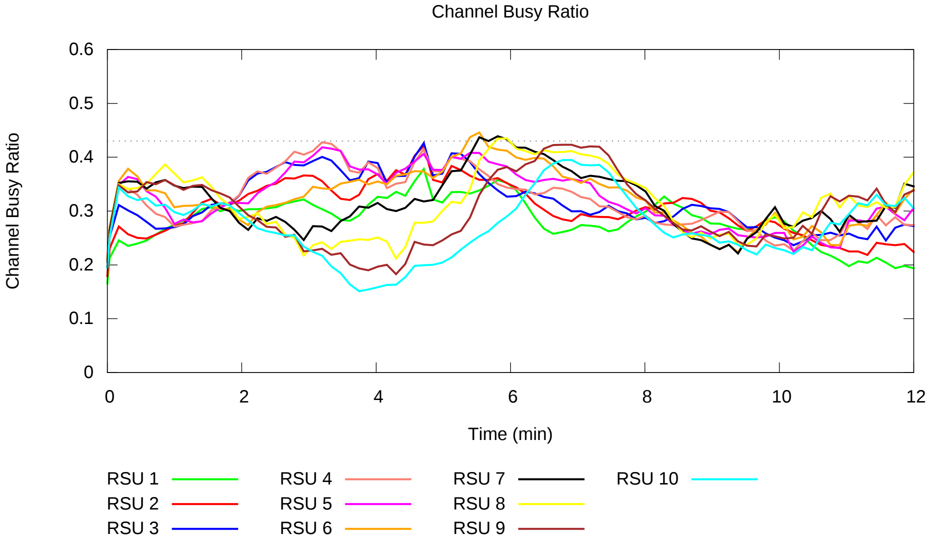


Figure 6.16: Channel busy ratio of weighted deviation in RSU based scenario 1

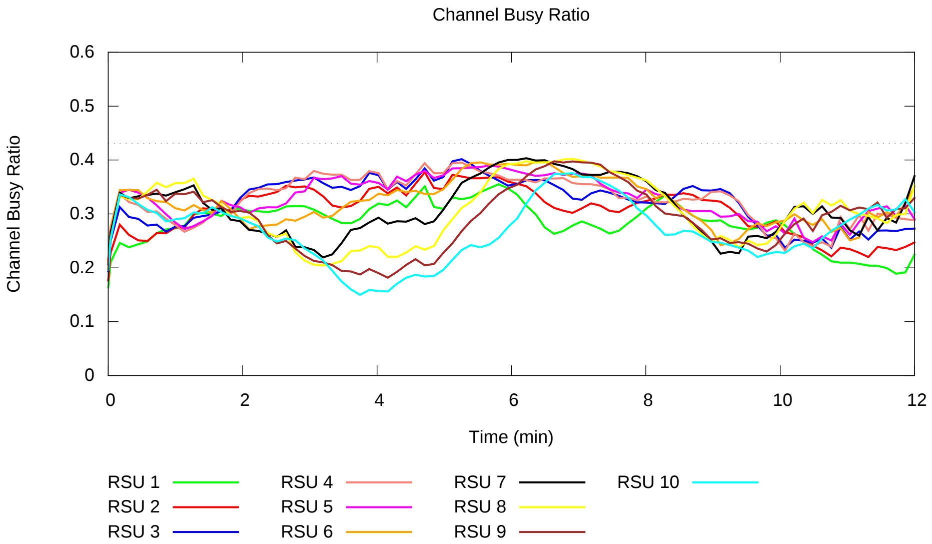


Figure 6.17: Channel busy ratio of adaptive standard score in RSU based scenario 1

to 6 minutes in the simulation. This is as a result of constant increase and decrease of aggregation levels of each RSU based on density of vehicles. During the period of the traffic free flow (0-2 min) the CBR is almost stable around 0.35 for all RSUs. The stability in CBR is a result of lower density during free flow. When the traffic jam starts initially, the number of vehicles in road segment 4 to 7 increases, and the CBR rises for clusters that are congested especially the middle and later. The CBR is higher for RSUs that are experiencing dense traffic and lower for RSUs having very low traffic. As the traffic jam moves in the direction of higher RSUs at min 4-7 every RSU is affected and their CBR rise having a peak at min 5-7. The aggregation schemes quickly adapt to the density of vehicles and change their aggregation level accordingly to meet the restrictions of each density window for each aggregation level. While the traffic jam dissolves, aggregation level and CBR low for all RSUs and back to the initial value. CBSDA scheme maintains their channel busy ratio well behind the maximum allowed CBR limit during traffic jam.

The aggregation levels for all RSUs change very rapidly due to the change in density of vehicles. The up- and downgrading of aggregation levels regulates the channel load of aggregated data. Figures 6.18, 6.19 and 6.20 show the aggregation levels for the three decision schemes. The traffic congestion starts at minute 2 and

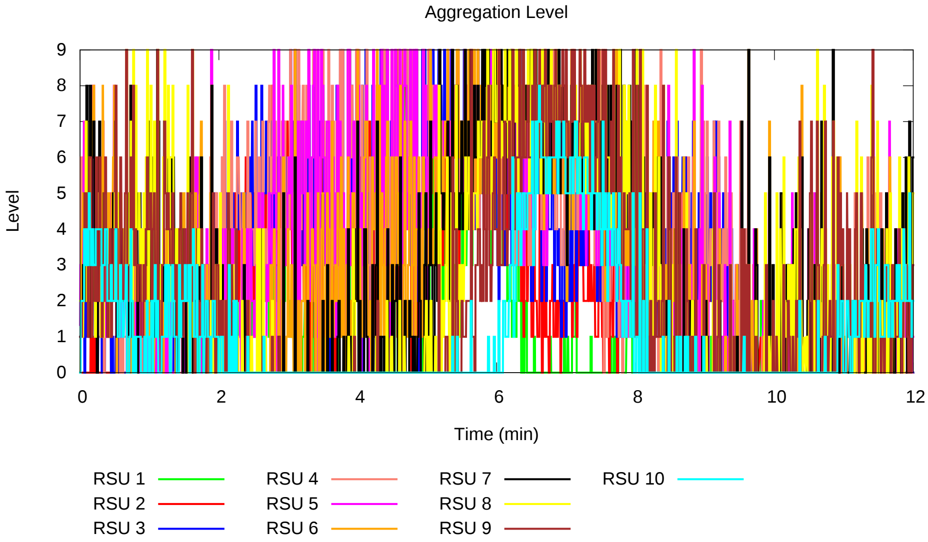


Figure 6.18: Aggregation level of adaptive cost-aware in RSU based scenario 1

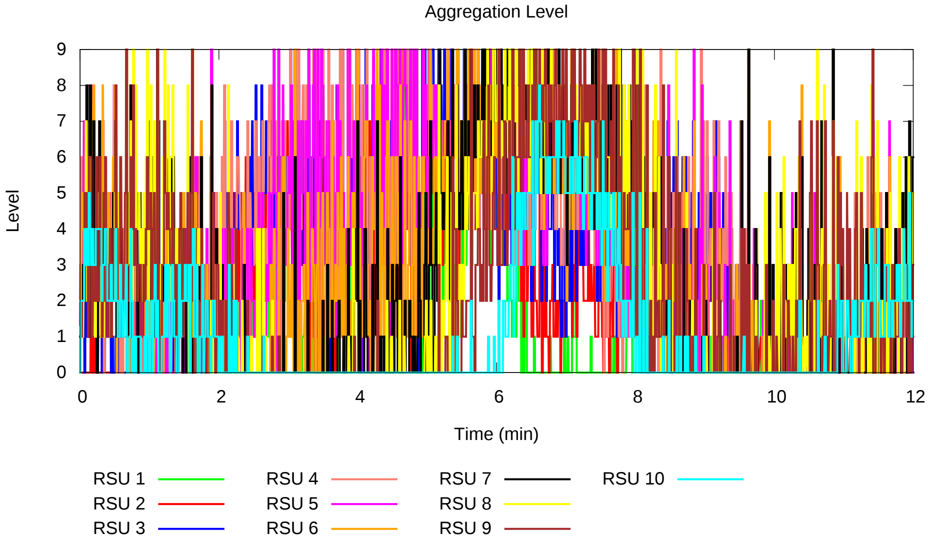


Figure 6.19: Aggregation level of weighted deviation in RSU based scenario 1

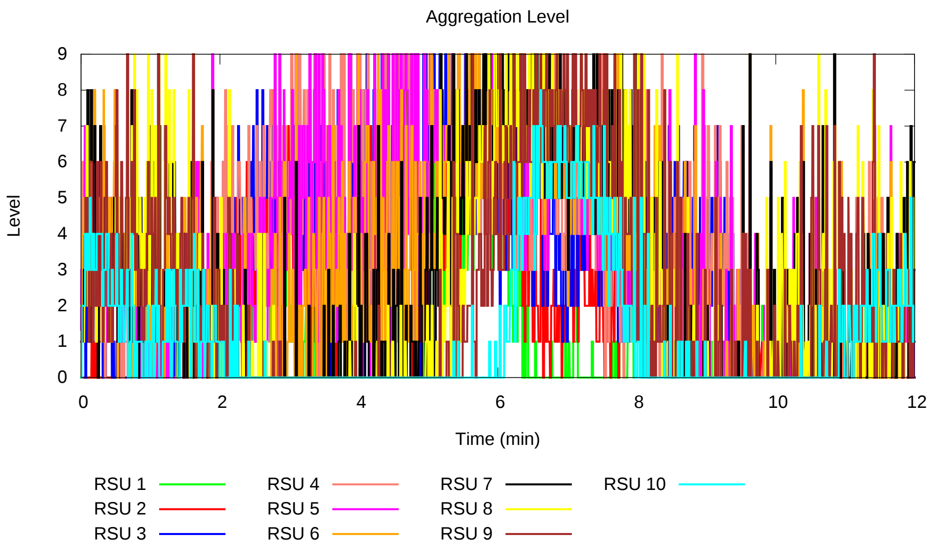


Figure 6.20: Aggregation level of adaptive standard score in RSU based scenario 1

as the traffic becomes more dense the aggregation levels increase starting about minute 3 in the area of RSU 4-5. While the traffic jam extends in both directions, the aggregation levels rise first in the direction to lower RSUs to minimise the additional data load for the rest of the road segment. Later, the traffic moves slowly in the direction of RSUs 7-9 and the aggregation levels rise there too. Beginning from minute 7 the traffic jam starts to dissolve and the aggregation levels slowly decrease back to zero. A heat map is used to represent the aggregated data, where the aggregated data contained in a matrix are represented as colors. The distribution of received aggregated data over time for all RSUs is shown in the heat map in Figures 6.21, 6.22, and 6.23 for all decision schemes. The data load is low in free flow traffic (0-2 min) with CBRs below thresholds and low aggregation levels. Data load increases significantly in the traffic jam situation $(2 - 5min)$ starting at RSUs $4 - 5$ and extending up to RSU 10 as the dense traffic moves forward slowly. Moreover, the aggregation data load is heavily reduced after the traffic jam dissolves and vehicles start to move with high velocity again causing high density thereby resulting in high channel load based on CAMs.

Data precision is an important performance indicator to evaluate decision schemes. The data fusion introduces an error that each decision scheme aims at keeping low.

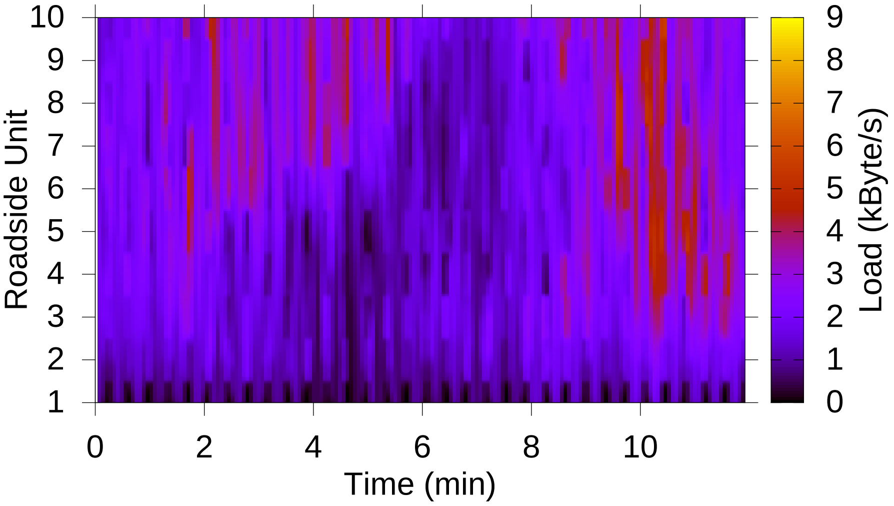


Figure 6.21: Aggregated data of adaptive cost-aware in RSU based scenario 1

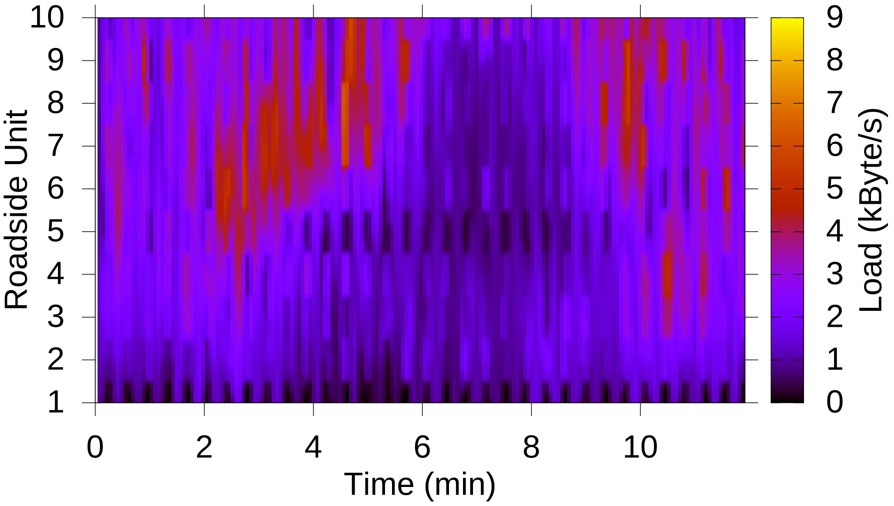


Figure 6.22: Aggregated data of weighted deviation in RSU based scenario 1

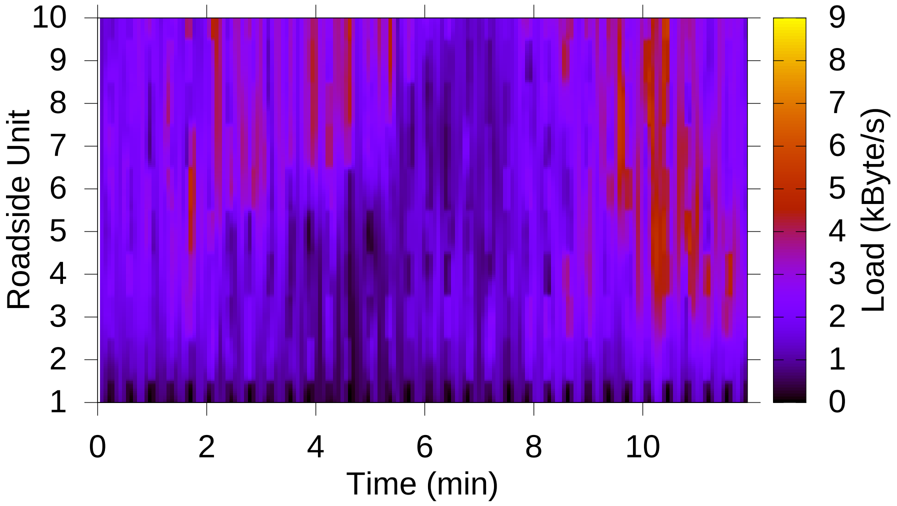


Figure 6.23: Aggregated data of adaptive standard score in RSU based scenario 1

The error introduced by each scheme is compared for two metrics: position and velocity. The imprecision is calculated as the Mean Absolute Error (MAE) of the data that reaches the control center compared to the actual data as it was broadcasted by the vehicles. These MAE values are calculated as shown in Equation 6.4, where $\hat{p}$ is the value of a metric as it was broadcasted by a vehicle and p is the value as it was received at the control control center.

$$MAE = \frac{1}{n} \times \sum_{i=0}^{n} |\hat{p} - p| \tag{6.4}$$

Each figure states the number of data records received with a certain error, the average difference from true value. During free flow traffic all schemes deliver data precision with less errors because the density wasn't high that lead to lower aggregation levels. During traffic jam, due to high density of vehicles and data fusion introduces errors in the packet received at control center.

The precision regarding the position metric from different decision schemes are illustrated in Figures 6.24, 6.25, and 6.26. The weighted deviation with N density control CBSDA scheme has the lowest MAE of 17.67 meters compared to adaptive cost-aware that has the MAE with 54.57 meters and adaptive standard score of

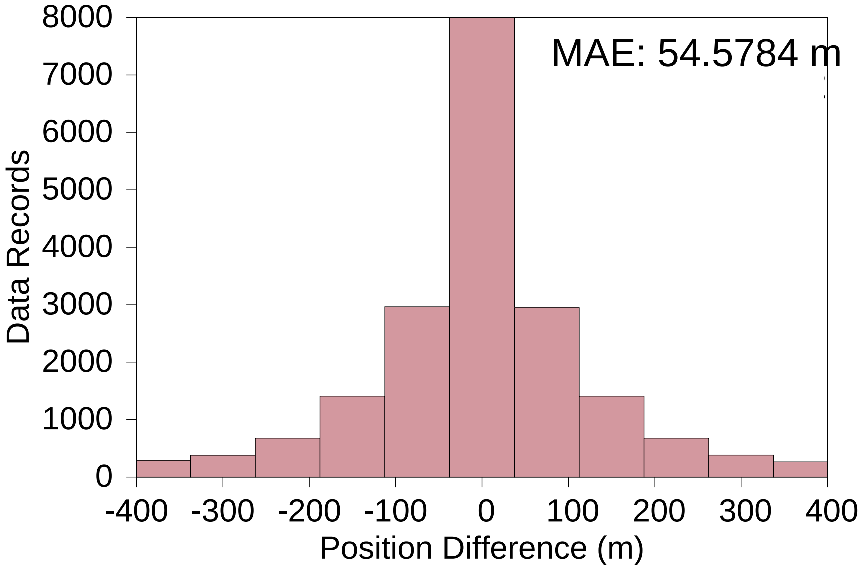


Figure 6.24: Position precision of adaptive cost-aware in RSU based scenario 1

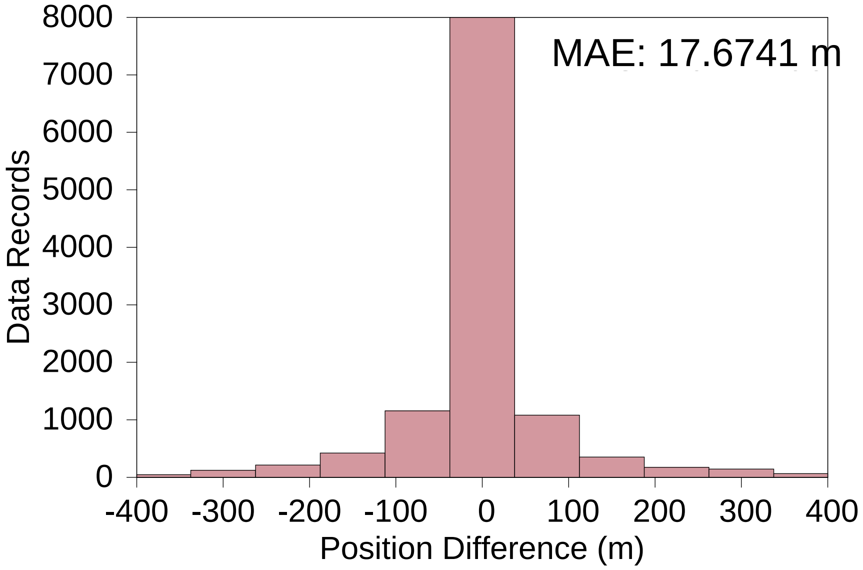


Figure 6.25: Position precision of weighted deviation in RSU based scenario 1

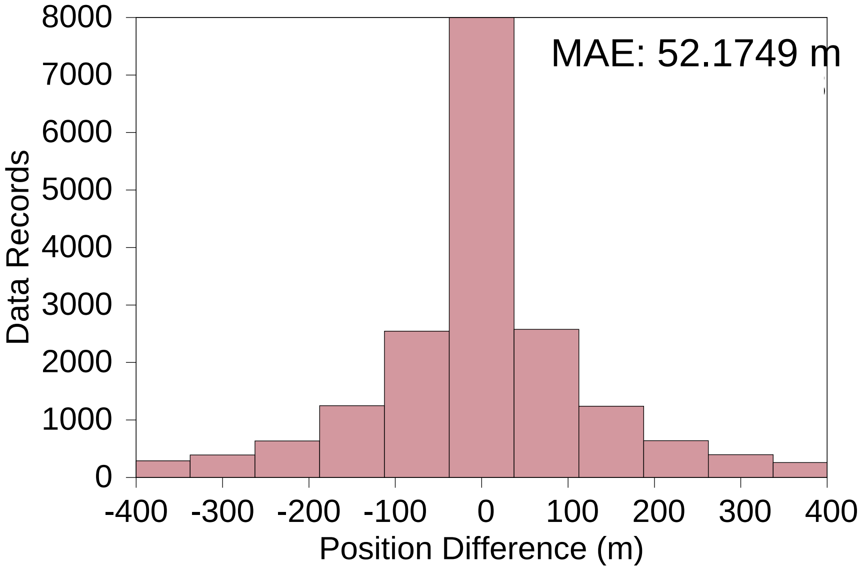


Figure 6.26: Position precision of adaptive standard score in RSU based scenario 1

MAE with 52.17 meters. Figures 6.27, 6.28, and 6.29 shows the precision analysis of the velocity metric. The precision of velocity metric for weighted deviation is higher compared to two other decision schemes. The MAE of weighted deviation scheme is 0.22m/sec compared to 0.51 m/sec of adaptive cost-aware and 0.49 m/sec of adaptive standard score. Smaller MAE means more precise the data recovered, so weighted deviation with density control using CBSDA performs better compared to other decision schemes.

In second scenario, I evaluate the CH stability and precision of decision schemes used for aggregation. Cluster stability can be calculated as the average number of CH changes in the entire simulation. Association time is defined as the percentage of time in which vehicles were CMs of a particular cluster. In this thesis, I consider the average CH lifetime means the sum of the lifetime of each CH selected divided by the total number of elected CHs in the entire simulation.

The Figure 6.30 shows the average CH lifetime with respect to the number of vehicles for different weighing values α. The factors are varied to different values from 0 to 1. In this scenario, I consider the minimum speed of 5m/sec to maximum speed of vehicles is 35m/sec, and cluster width of 300 meters.

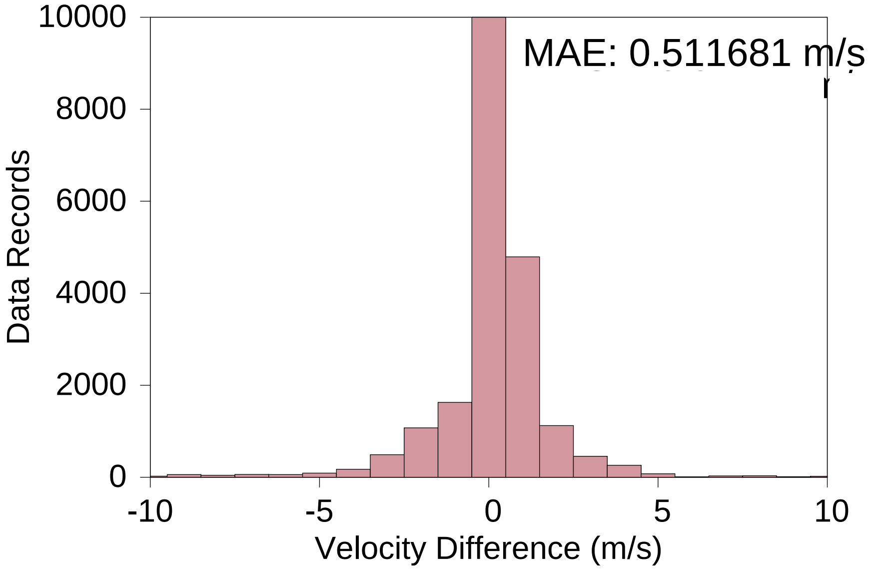


Figure 6.27: Velocity precision of adaptive cost-aware in RSU based scenario 1

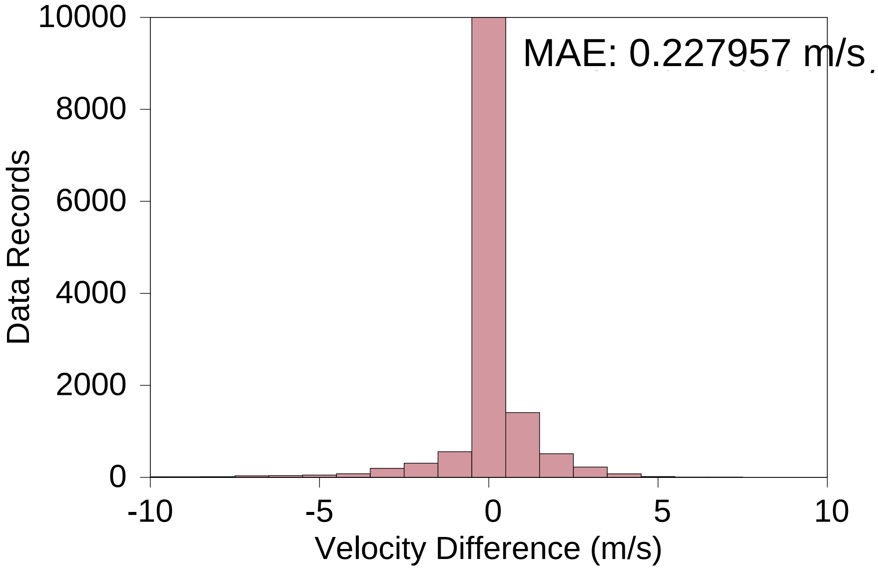


Figure 6.28: Velocity precision of weighted deviation in RSU based scenario 1

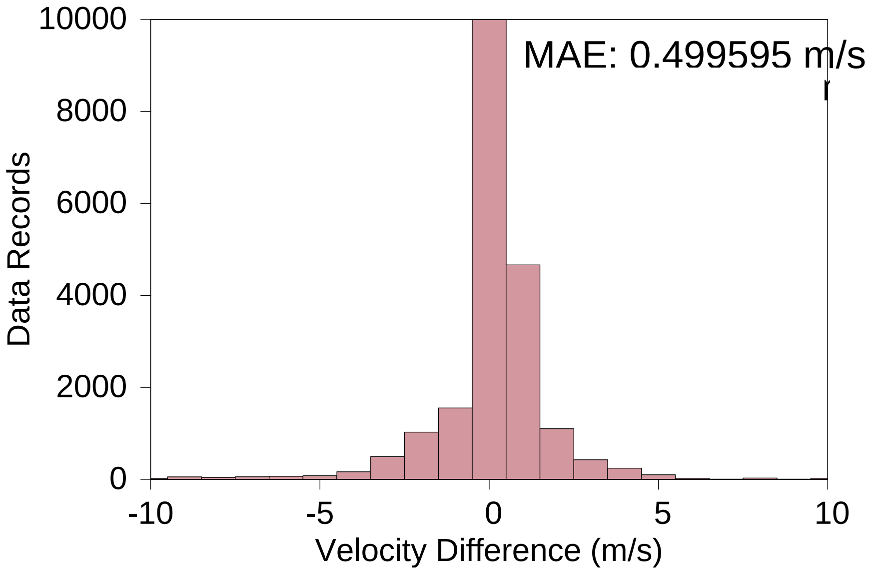


Figure 6.29: Velocity precision of adaptive standard score in RSU based scenario 1

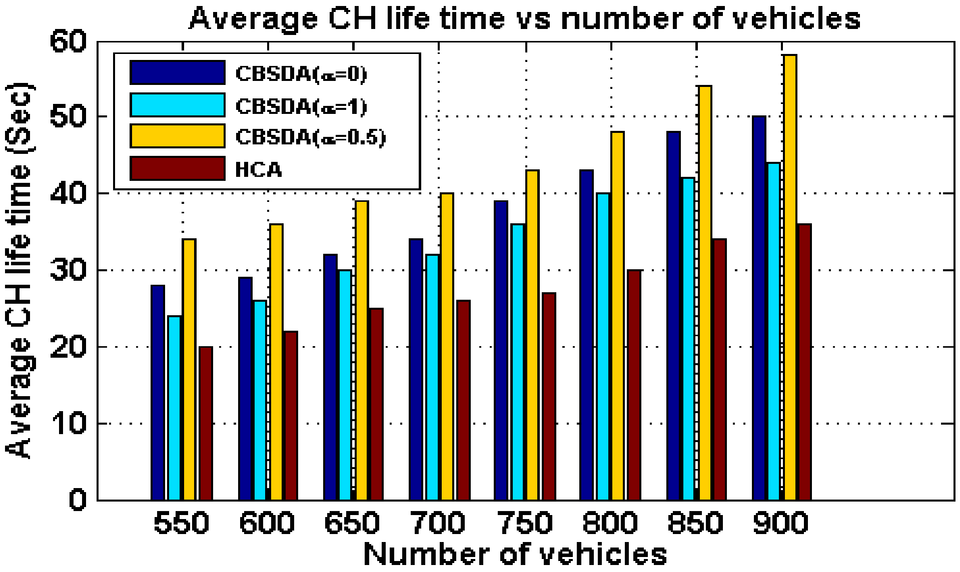


Figure 6.30: Average CH lifetime of CBSDA protocol vs number of vehicles, when the speed variation of vehicles are between 5-35 m/sec

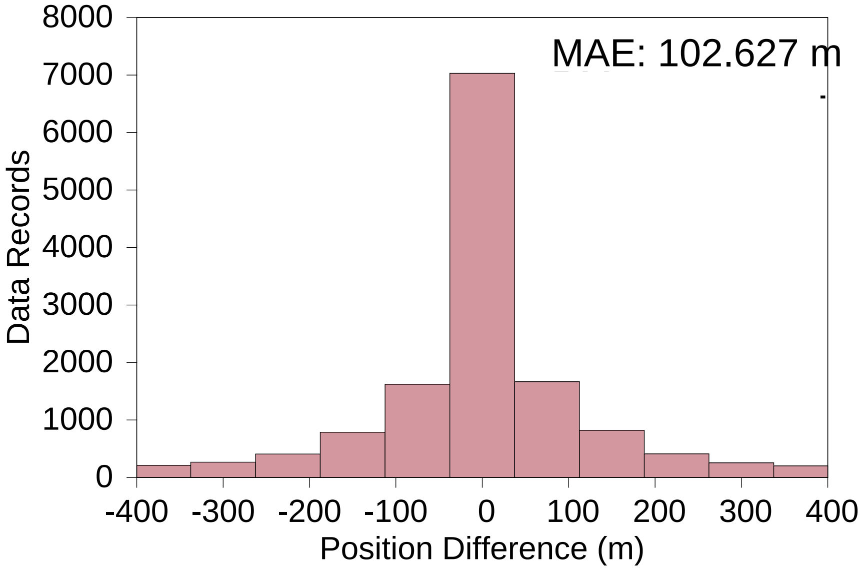


Figure 6.31: Position precision of adaptive cost-aware in CH based scenario 2

From the Figure 6.30, when $\alpha = 1$ are given to weigh factor means the CH election is considered only based on the distance between the vehicle and its neighbours. When $\alpha = 0$ means the CH election is solely based on the speed deviation between neighbours. Moreover, when $\alpha = 0.5$ it gives equal importance to both speed and distance deviations. Figure 6.30 shows the impact of the number of vehicles on the CH lifetime of the proposed CBSDA with different values of weighting parameters and HCA protocol. CBSDA with $\alpha = 0.5$ performs better compared to the other values and have higher lifetime. So stability is higher, when the weight is equal to all parameter.

In second scenario, the aggregation take place in CH rather than in RSU. The CH aggregates the data using aggregation level control based on the density with different decision schemes. In this scenario, the CH forwards the aggregated data towards nearest CH in the direction of control server.

Precision of position and velocity metric for different decision schemes using density control for maximum CL=1500 for CBSDA scheme is obtained from simulation results. The precision regarding the position metric is illustrated in Figures 6.31, 6.32, and 6.33. The weighted deviation has the lowest MAE of 15.6 meters compared to adaptive cost-aware that has the MAE with 102.62 meters and adapt-

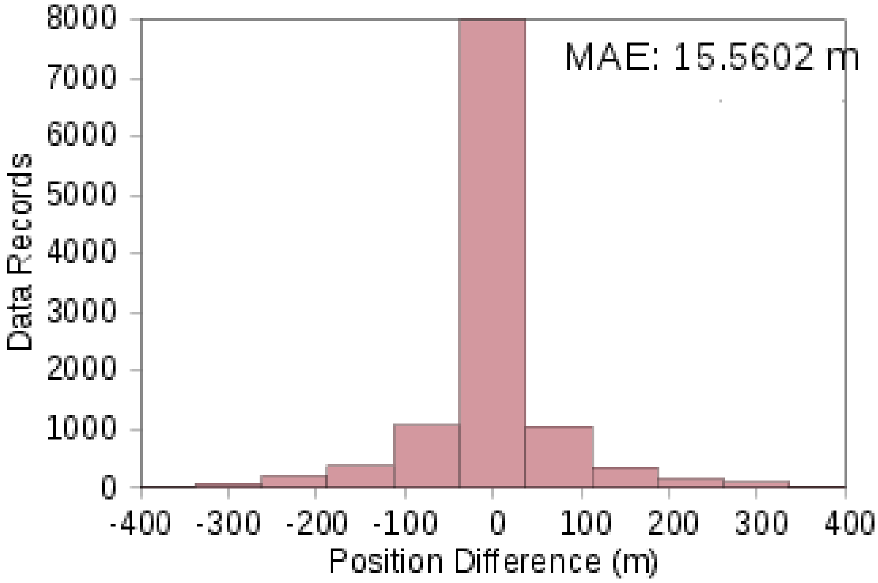


Figure 6.32: Position precision of weighted deviation in CH based scenario 2

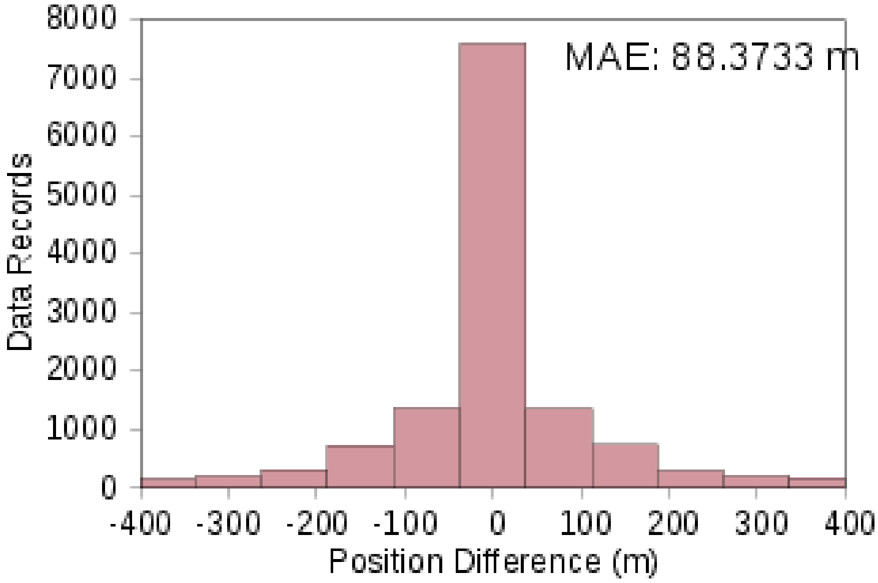


Figure 6.33: Position precision of adaptive standard score in CH based scenario 2

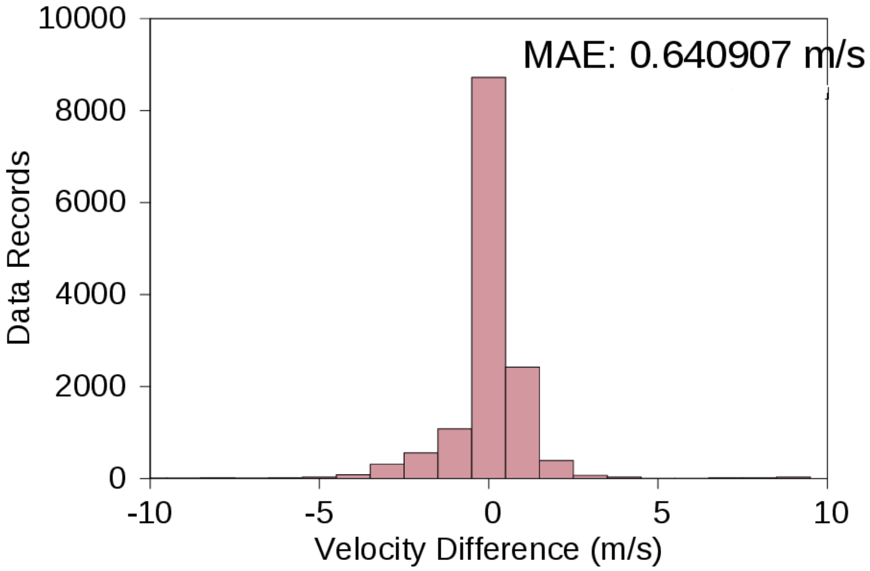


Figure 6.34: Velocity precision of adaptive cost-aware in CH based scenario 2

ive standard score has the MAE with 88.37 meters. Figures 6.34, 6.35, and 6.36 shows the precision analysis of the velocity metric. The velocity metric is more precise in weighted deviation compared to other decision schemes. The MAE of weighted deviation is 0.41 m/sec compared to adaptive cost-aware and adaptive standard score schemes. The distribution of received aggregated data over time for all CHs of all decision schemes are shown using the heat map in Figures 6.37, 6.38 and 6.39. The data load is low in free flow traffic (0-2 min) with CBRs below thresholds and low aggregation levels. Aggregated data increases significantly in the traffic jam situation $(4-7min)$ starting at clusters $4-5$ and extending up to cluster 9 as the dense traffic moves forward slowly.

6.5 Summary

In this chapter, the simulation results of D-CBM protocol based on clustering, hybrid (CSMA) and TDMA (random, adaptive standard, future) slot allocation schemes for VANETs are presented. Moreover, three different slot allocation schemes are presented for improving the current standard. I ran different simulations for D-CBM along with SBCA, in order to test the performance of D-CBM compared to

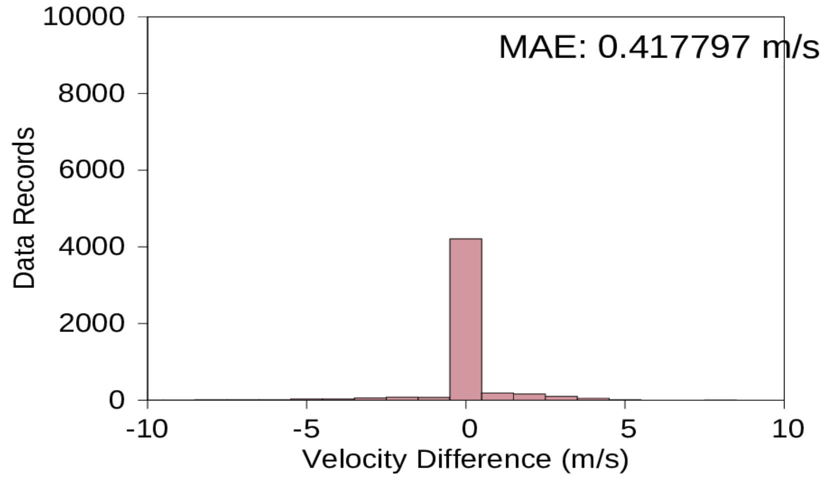


Figure 6.35: Velocity precision of weighted deviation in CH based scenario 2

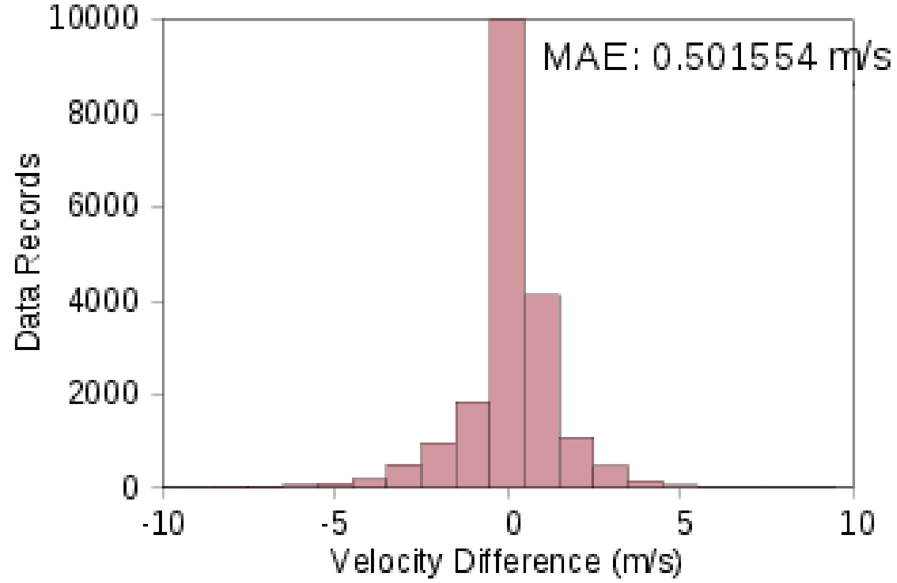


Figure 6.36: Velocity precision of adaptive standard score in CH based scenario 2

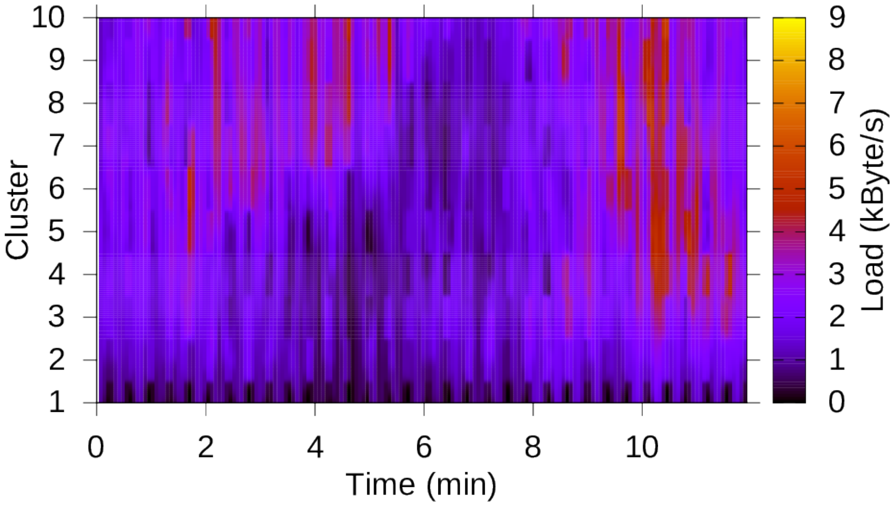


Figure 6.37: Aggregated data of adaptive cost-aware in CH based scenario 2

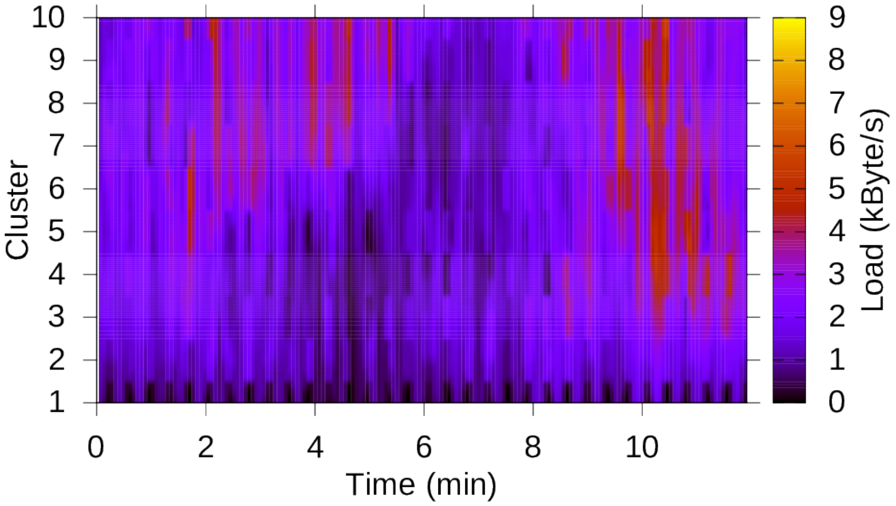


Figure 6.38: Aggregated data of weighted deviation in CH based scenario 2

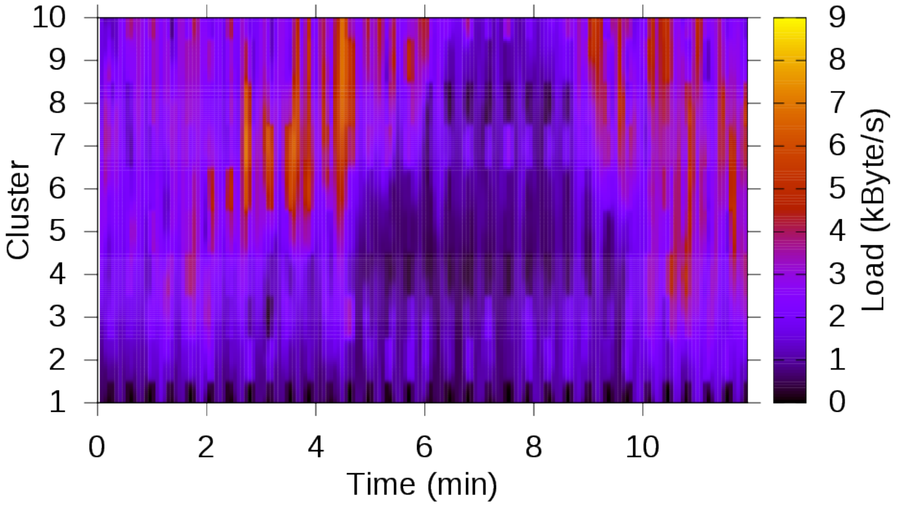


Figure 6.39: Aggregated data of adaptive standard score in CH based scenario 2

SBCA protocol. D-CBM showed that it can support higher cluster head life time than SBCA, even in high speed deviation scenarios. Also, D-CBM performed better in terms of packet delivery rate with WAVE standard.

Secondly, I presented DA-CMAC protocol based on TDMA. I ran different simulations for DA-CMAC along with WAVE to test the performance of DA-CMAC compared to WAVE standard. Every vehicle in DA-CMAC has its own chance to perform safety and non-safety communication in every 100 msec. On the other hand, WAVE suffered higher access collisions during high density. For WAVE, as the traffic density increases, the collision on the CCH increases.

Thirdly, I presented CBSDA aggregation scheme using three different decision schemes. Aggregation in the RSU or CH is controlled by aggregation level control which is based on the density of vehicles on the road segment. The performance of decision schemes are evaluated based on the precision of different metrics received in the control server. All simulations results show that CBSDA, D-CBM and DA-CMAC protocols perform better compared to existing standard and other protocols such as SBCA and HCA.

7

Conclusions and future work

In this chapter I summarize the motivation for this thesis, the problem I have addressed, and the solutions I have proposed. The work in this thesis also provides the direction for future research. The chapter is organized as follows: Section 7.1 addresses the summary and main motivation of this thesis, Section 7.2 lists the research contributions of this thesis, Section 7.3 explains the simulators and results of the proposed protocols. Future research guidelines of this thesis are shown in Section 7.4.

7.1 Summary

VANETs are an emerging paradigm which is currently receiving significant support from government, academia, and industrial organizations over the globe. By employing V2V and V2I communications, VANETs are expected to realize a variety of advanced applications for road safety, passenger infotainment, and vehicle traffic optimization.

Dedicated Short Range Communication (DSRC) is the 75 MHz wide spectrum band allocated by the U.S. Federal Communication Commission (FCC) for communications in VANETs. The spectrum band is divided into seven 10 MHz channels, one CCH, and six SCHs. The CCH is the default channel for the exchange of safety and update messages, while the SCHs are the default channels for non safety messages.

The IEEE has completed the 1609 family of standards for WAVE standard for vehicular communications. In WAVE, IEEE 1609.4 describes a concept of channel intervals in which time is divided into alternating CCHI and SCHI. Each of the intervals is 50 msec long. A pair of a CCHI and SCHI forms a SP with the length of 100 msec, which is motivated by a desire of having a safety messages rate of 10 Hz. This desire is based on the allowable latency requirements of real time active safety applications.

The main objectives of this research are to achieve the QoS requirements of VANET safety and traffic efficiency applications, by using clustering, multichannel MAC protocol and data aggregation in VANET. During an accident, the density of vehicles over a certain region may be high, which results in transmission collisions due to lack of scalability of the network. This can be alleviated using clustering and CH election. The CH can be a coordinator for the allocation time slots to other vehicles, thereby reducing channel access collisions. Furthermore, a cluster based multi channel MAC protocol is proposed to increase the scalability, reduce channel access collisions, and increase reliability of safety messages.

The data sent from one CH to the other CH grow depending upon the density of vehicles under each CH. This may eventually congest the network channel and packets will be lost. To reduce the load, adaptive aggregation scheme based on density of vehicles will reduce the load and prevent channel congestion. The data load is reduced by fusion of two similar data. However, the fusion process introduces an error in the vehicular data and makes it less precise. Thus, the process should be adaptive and fuse only enough information to maintain the channel load within acceptable limits. This way, it aims at providing the highest data precision possible without congesting the network channel. To achieve these objectives, this thesis presents density based aggregation, different precision schemes and a cluster based semantic data aggregation scheme.

7.2 Research contributions

In this thesis, I have made the following research contributions:

- Three different cluster formation algorithms are proposed based on direction of movement, position of vehicles and transmission range. Three different CH election algorithms are proposed based on speed deviation, the number of connections, distance between RSUs and vehicles, distance between start of the cluster segment and distance between the neighbours on the road. During the cluster formation process, the cluster members will be grouped based on the direction of travel and assigned local IDs by the CH. The design and implementation of CH election and cluster formation algorithms shows that fewer CH changes occur compared to existing algorithms. This lead to minimum overhead thereby resulting in fewer re-clustering and delivers an efficient hierarchical network topology. The cluster maintenance algorithm proposed handles the topology changes. The proposed algorithm takes advantage of the local IDs that are assigned in my cluster formation algorithm. In cluster maintenance algorithm, different events that can lead to merging, joining and leaving the cluster are considered. Moreover, the maintenance algorithm reduces the overall management overhead.

- Three cluster-based MAC protocols to coordinate intra-cluster communications. First proposed protocol is a hybrid protocol (D-CBM CSMA) that is a combination of both contention based and contention free channel access. Secondly, the MAC protocol (D-CBM TDMA) is based on TDMA slot allocation, where the CH allocates slots to CMs based on priority (future position, speed etc). Lastly, multi channel MAC protocol (DA-CMAC) where the period is divided into SCP and CCP. In other words, the time cycle is divided into two different intervals, CCH Interval and SCH Interval as with WAVE. In DA-CMAC, the SCP and CCP are divided into slots and mini slots. The slots are allocated to vehicles based on the local IDs.

- A cluster based semantic data aggregation scheme to reduce the channel congestion. The aggregation increases adaptively when the number of vehicles in the cluster increases. A tree data structure is used for storing the data in each CHs or RSUs. The data structure grows when the vehicle density increases and data structure reduces when the vehicle density decreases. Data fusion is performed to reduce the data load. Three different decision schemes are proposed to increase the precision of data.

7.3 Evaluation

The protocols are evaluated using computer simulations that are developed in network simulator ns-2, and ns-3. Moreover, the performance of the proposed protocols are compared with the WAVE standard, SBCA and the HCA protocol. Simulation results show that, the DA-CMAC and D-CBM protocol can deliver both types of safety messages to all the vehicles in the one-hop neighbourhood with an acceptable average delivery delay (less than 100 ms). Moreover, it is shown that the DA-CMAC has a low probability of a transmission collision, which results in a higher safety message throughput and better channel utilization, as compared to the WAVE standard. This research sheds light on TDMA as a promising technology for MAC in VANETs, and a suitable replacement of the WAVE standard, which has significant limitations in supporting VANET safety applications. Furthermore, the ability of the proposed protocols are demonstrated by detailed delivery delay analysis, including percentage of access collisions, cluster and CH lifetime, precision, and level of aggregation in different densities.

I have also evaluated CBSDA scheme for different decision schemes. The aggregation is based on the number of vehicles in a road segment. The aggregation level increases, when the density of vehicles increases. Decision schemes are evaluated using the MAE to calculate the maximum precision at the control server. After analyzing all schemes, I conclude that Weighted deviation schemes performs better compared to all other decision schemes.

7.4 Further research work

In the future, there is a need for further analysis of DA-CMAC, designing a numbering scheme for cluster members, enhancing the utilization of the SCHs, and developing speed-based clustering. The clustering formation algorithm I used in this thesis is based on the highways in Europe, where there are only few exits on the highway for a predefined length. However, in other countries, the highway may have many exits. So, using the same clustering formation for such highways will make the process of new vehicles joining and leaving the cluster more frequent, which may lead to an increase in the overhead. I would also like to investigate the impact of using a new clustering formation algorithm on the performance of DA-CMAC in those scenarios.

In the area of clustering of vehicles in VANETs we believe that crucial questions remain and should be considered as future work. Though the proposed methods of this thesis incorporate many different parameters of vehicles, such as direction, future position, velocity, and distance between neighbours, application driven methods should be developed in the near future in order to cope with different situations. Road safety and traffic congestion avoidance for example are two different circumstances that may arise in a VANET, with different requirements and limitations in terms of delay, dissemination coverage etc. Clustering parameters must be tuned for every different application that runs over the VANET and this must be done in a distributed and automated way. Messages that vehicles exchange in order to define their neighbourhood and elect CHs lead to awareness degradation due to interference in high load situations. Selective intervals of messages may cope with this problem. Short term prediction mechanisms finally, can be used along with the sociological patterns of vehicles in order to increase prediction accuracy and cluster stability.

A prototype should be created for the DA-CMAC protocol in order to investigate its implementation complexity and practically test its performance in a real vehicular scenario. The DA-CMAC protocol should be evaluated by using realistic mobility traces of vehicles in city scenarios. As in all evaluations, there is more analysis that can be performed. I would like to further investigate the effect of GIs on the number of cluster members and message size.

Another promising research topic is how to support periodic and event-driven safety messages via the Long Term Evolution Advanced (LTE-A) mobile communications standard and heterogeneous communications. To date, very few studies have considered this topic, and there are still many issues which need investigation to determine whether or not the LTE-A standard can be employed for road safety applications.

The data aggregation scheme, discussed in Chapter 5, shows some enhancements on reducing data load in the channel and reducing transmission collisions. Other aggregation schemes could be developed using the proposed framework in future work. Particularly interesting schemes could be designed by extending the definition of the aggregation tree by other node types, for instance. These could implement functions like the variance or mean of certain metrics. Furthermore, the restrictions of the tree design could be relaxed in future work. This would allow an even more flexible tree structure and thereby more flexibility in designing the aggregation scheme. A second aspect for future work could be a deeper investigation of the behavior of the aggregation schemes for values of different probability distributions. The findings of this thesis suggest a great impact of this distribution on the error introduced by the fusion algorithm. Last, other components for dissemination, fusion, decision, and level control component could be implemented. For instance, the level control component could use a different algorithm to increase the aggregation level. Instead of increasing the aggregation level by one, a strategy that increases the aggregation level exponentially might be useful. Adaptive data aggregation has a great potential to reduce the data load, when needed, to prevent the network channel from congestion. In the meanwhile, using adaptive aggregation, data is not fused when sufficient resources are available.

Bibliography

[Ahizoune 12] A. Ahizoune and A. Hafid. *A new stability based clustering algorithm (SBCA) for VANETs.* In Local Computer Networks Workshops (LCN Workshops), 2012 IEEE 37th Conference on, pp. 843–847, 2012. 119

[Almalag 12] M.S. Almalag, S. Olariu and M.C. Weigle. *TDMA cluster-based MAC for VANETs (TC-MAC).* In World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a, pp. 1–6, 2012. 17, 19

[Alon 86] Noga Alon, László Babai and Alon Itai. *A fast and simple randomized parallel algorithm for the maximal independent set problem.* Journal of Algorithms, Vol. 7, No. 4, pp. 567–583, dec 1986. 36, 47

[Bana 01] S.V. Bana and P. Varaiya. *Space division multiple access (SDMA) for robust ad hoc vehicle communication networks.* In Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE, pp. 962–967, 2001. 25, 27

[Basu 01] P. Basu, N. Khan and T. D C Little. *A mobility based metric for clustering in mobile ad hoc networks.* In Distributed Computing Systems Workshop, 2001 International Conference on, pp. 413–418, 2001. 20

[Bharati 13] S. Bharati and Weihua Zhuang. *CAH-MAC: Cooperative ADHOC MAC for Vehicular Networks.* Selected Areas in Communications, IEEE Journal on, Vol. 31, No. 9, pp. 470–479, September 2013. 25

[Bianchi 96] G. Bianchi, L. Fratta and M. Oliveri. *Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs.* In Personal, Indoor and Mobile Radio Communications, 1996. PIMRC'96., Seventh IEEE International Symposium on, volume 2, pp. 392–396 vol.2, Oct 1996. 25

[Blum 04] J.J. Blum, A. Eskandarian and L.J. Hoffman. *Challenges of intervehicle ad hoc networks.* Intelligent Transportation Systems, IEEE Transactions on, Vol. 5, No. 4, pp. 347 – 351, dec. 2004. 14

[Bononi 07] L. Bononi and M. Di Felice. *A Cross Layered MAC and Clustering Scheme for Efficient Broadcast in VANETs*. In Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on, pp. 1 –8, oct. 2007. 18

[Borgonovo 04] Flaminio Borgonovo, Antonio Capone, Matteo Cesana and Luigi Fratta. *AD-HOC MAC: new MAC architecture for ad hoc networks providing efficient and reliable point-to-point and broadcast services*. Wirel. Netw., Vol. 10, No. 4, pp. 359–366, July 2004. 25, 26

[Brenner 97] P Brenner. *A Technical Tutorial on the IEEE 802.11 Protocol*. 1997. 27

[Casavant 94] Thomas L. Casavant and Mukesh Singhal. Readings in distributed computing systems. IEEE Computer Society Press, Los Alamitos, CA, USA, 1994. 4

[Chen 07] Huifang Chen, Hiroshi Mineno, Yoshitsugu Obashi, Tomohiro Kokogawa and Tadanori Mizuno. *Adaptive Data Aggregation for Clustered Wireless Sensor Networks*. In Jadwiga Indulska, Jianhua Ma, LaurenceT. Yang, Theo Ungerer and Jiannong Cao, editeurs, Ubiquitous Intelligence and Computing, volume 4611 of *Lecture Notes in Computer Science*, pp. 475–484. Springer Berlin Heidelberg, 2007. 31

[Daeinabi 11] Ameneh Daeinabi, Akbar Ghaffar Pour Rahbar and Ahmad Khademzadeh. *VWCA: An Efficient Clustering Algorithm in Vehicular Ad Hoc Networks*. J. Netw. Comput. Appl., Vol. 34, No. 1, pp. 207–222, January 2011. 21

[Dietzel 09] Stefan Dietzel, Boto Bako, Elmar Schoch and Frank Kargl. *A fuzzy logic based approach for structure-free aggregation in vehicular ad-hoc networks*. In Proceedings of the sixth ACM international workshop on VehiculAr Inter-NETworking - VANET. ACM Press, 2009. 87

[Dietzel 10] S. Dietzel, F. Kargl, G. Heijenk and F. Schaub. *On the potential of generic modeling for VANET data aggregation protocols*. In Vehicular Networking Conference (VNC), 2010 IEEE, pp. 78–85, Dec 2010. 31, 87

[Dietzel 11] Stefan Dietzel, Frank Kargl, Geert Heijenk and Florian Schaub. *Modeling in-network aggregation in VANETs*. IEEE Communications Magazine, Vol. 49, No. 11, pp. 142–148, nov 2011. 87

[Dietzel 14] S. Dietzel, J. Petit, F. Kargl and B. Scheuermann. *In-Network Aggregation for Vehicular Ad Hoc Networks*. Communications Surveys Tutorials, IEEE, Vol. 16, No. 4, pp. 1909–1932, Fourthquarter 2014. 3

[Dror 11] E. Dror, C. Avin and Z. Lotker. *Fast randomized algorithm for hierarchical clustering in Vehicular Ad-Hoc Networks*. In Ad Hoc Networking Workshop

(Med-Hoc-Net), 2011 The 10th IFIP Annual Mediterranean, pp. 1–8, 2011. 17, 18, 128

[Flajolet 85] Philippe Flajolet and G. Nigel Martin. *Probabilistic Counting Algorithms for Data Base Applications*. J. Comput. Syst. Sci., Vol. 31, No. 2, pp. 182–209, September 1985. 32

[Gaugel 13] T. Gaugel, J. Mittag, H. Hartenstein, S. Papanastasiou and E.G. Strom. *In-depth analysis and evaluation of Self-organizing TDMA*. In Vehicular Networking Conference (VNC), 2013 IEEE, pp. 79–86, Dec 2013. 112

[Goonewardene 09] R. T. Goonewardene, F.H. Ali and E. Stipidis. *Robust mobility adaptive clustering scheme with support for geographic routing for vehicular ad hoc networks*. Intelligent Transport Systems, IET, Vol. 3, No. 2, pp. 148–158, 2009. 22

[Graffi 08] K. Graffi, A. Kovacevic, Song Xiao and R. Steinmetz. *SkyEye.KOM: An Information Management Over-Overlay for Getting the Oracle View on Structured P2P Systems*. In Parallel and Distributed Systems, 2008. ICPADS '08. 14th IEEE International Conference on, pp. 279–286, Dec 2008. 30

[Graffi 09] K. Graffi, D. Stingl, J. Rueckert, A. Kovacevic and R. Steinmetz. *Monitoring and management of structured peer-to-peer systems*. In Peer-to-Peer Computing, 2009. P2P '09. IEEE Ninth International Conference on, pp. 311–320, Sept 2009. 30

[Gunter 07] Y. Gunter, B. Wiegel and H.P. Grossmann. *Cluster-based Medium Access Scheme for VANETs*. In Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE, pp. 343–348, 2007. 17, 19

[Ibrahim 08] K. Ibrahim and M.C. Weigle. *CASCADE: Cluster-Based Accurate Syntactic Compression of Aggregated Data in VANETs*. In GLOBECOM Workshops, 2008 IEEE, pp. 1–10, Nov 2008. 30, 33

[IE 06] IE. *IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages*. IEEE Std 1609.2-2006, Vol. 1, pp. 01–105, 2006. 10

[IEE 06] IEE. *Trial-Use Standard for Wireless Access in Vehicular Environments WAVE - Resource Manager*. IEEE Std 1609.1-2006, Vol. 1, pp. 1–71, Oct 2006. 10

[IEEE 06] IEEE. *IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-Channel Operation*. IEEE Std 1609.4-2006 -Test, Vol. 1, pp. 1–82, Nov 2006. 11

[IEEE 07] IEEE. *IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Networking Services*. IEEE Std 1609.3-2007, Vol. 1, pp. 1–99, April 2007. 10

[IEEE 10] IEEE. *Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems – 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE ASSP Magazine, Vol. 1, p. 19, 2010. 10, 24

[IEEE 13] IEEE. *IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Architecture*. IEEE Std 1609.0-2013, Vol. 1, p. 30, 2013. 2, 10, 76

[Intanagonwiwat 03] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann and Fabio Silva. *Directed Diffusion for Wireless Sensor Networking*. IEEE/ACM Trans. Netw., Vol. 11, No. 1, pp. 2–16, February 2003. 30

[Jerbi 07] M. Jerbi, S. M Senouci, T. Rasheed and Y. Ghamri-Doudane. *An Infrastructure-Free Traffic Information System for Vehicular Networks*. In Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th, pp. 2086–2090, 2007. 20

[Jiang 08] D. Jiang and L. Delgrossi. *IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments*. In Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE, pp. 2036–2040, May 2008. 2

[Jiang 10] Hongbo Jiang, Shudong Jin and Chonggang Wang. *Parameter-Based Data Aggregation for Statistical Information Extraction in Wireless Sensor Networks*. Vehicular Technology, IEEE Transactions on, Vol. 59, No. 8, pp. 3992–4001, Oct 2010. 31, 33, 91

[Jiru 14] Josef Jiru, Lars Bremer and Kalman Graffi. *Data aggregation in VANETs a generalized framework for channel load adaptive schemes*. In 39th Annual IEEE Conference on Local Computer Networks. IEEE, sep 2014. 84, 87

[Jiru 15] J. Jiru, A.S. Koyamparambil Mammu and K. Roscher. *Adaptive decision algorithms for data aggregation in VANETs with defined channel load limits*. In Intelligent Vehicles Symposium (IV), 2015 IEEE, pp. 1367–1372, June 2015. 4

[Joaquim 13] F. Joaquim, S. Manuel, M. Tiago, KM. Aboobeker Sidhik and J. Unai Hernandez. *Fundamental Design Decisions Of The Deterministic MAC Protocol*. In Deliverable 3.1 for FP7- ICSI project, 2013. 112

[Kayis 07] O. Kayis and T. Acarman. *Clustering Formation for Inter-Vehicle Communication.* In Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE, pp. 636–641, 2007. 17

[Krishnamachari 02] B. Krishnamachari, D. Estrin and S. Wicker. *The impact of data aggregation in wireless sensor networks.* In Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on, pp. 575–578, 2002. 84

[Kutzner 03] Kendy Kutzner, Jean jacques Tchouto, Marc Bechler, Lars Wolf, Bernd Bochow and Thomas Luckenbach. *Connecting Vehicle Scatternets by Internet-Connected Gateways.* In in MMC'2003, 2003. 60

[Kwon 02] Younggoo Kwon, Yuguang Fang and H. Latchman. *Fast collision resolution (FCR) MAC algorithm for wireless local area networks.* In Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE, volume 3, pp. 2250–2254 vol.3, 02. 24, 25

[Laboratory 10] Lawrence Berkeley National Laboratory. http://www.isi.edu/nsnam/ns/. 2010. 112

[Lai 11] Yen-Cheng Lai, Phone Lin, Wanjiun Liao and Chung-Min Chen. *A Region-Based Clustering Mechanism for Channel Access in Vehicular Ad Hoc Networks.* Selected Areas in Communications, IEEE Journal on, Vol. 29, No. 1, pp. 83–93, 2011. 17, 19

[Li 06] Yijun Li, H. Wu, Nian-Feng Tzeng, D. Perkins and M. Bayoumi. *MAC-SCC: a medium access control protocol with separate control channel for reconfigurable multi-hop wireless networks.* Wireless Communications, IEEE Transactions on, Vol. 5, No. 7, pp. 1805–1817, 2006. 25

[Lindsey 02] S. Lindsey, C. Raghavendra and K.M. Sivalingam. *Data gathering algorithms in sensor networks using energy metrics.* Parallel and Distributed Systems, IEEE Transactions on, Vol. 13, No. 9, pp. 924–935, Sep 2002. 30

[Little 05] T.D.C. Little and A. Agarwal. *An information propagation scheme for VANETs.* In Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE, pp. 155 – 160, sept. 2005. 18, 23

[Lochert 07] Christian Lochert, Björn Scheuermann and Martin Mauve. *Probabilistic Aggregation for Data Dissemination in VANETs.* In Proceedings of the Fourth ACM International Workshop on Vehicular Ad Hoc Networks, VANET '07, pp. 1–8, New York, NY, USA, 2007. ACM. 32, 33, 91

[Lochert 08] Christian Lochert, Björn Scheuermann, Christian Wewetzer, Andreas Luebke and Martin Mauve. *Data Aggregation and Roadside Unit Placement for a Vanet Traffic Information System*. In Proceedings of the Fifth ACM International Workshop on VehiculAr Inter-NETworking, VANET '08, pp. 58–65, New York, NY, USA, 2008. ACM. 31

[Lu 10] Ning Lu, Yusheng Ji, Fuqiang Liu and Xinhong Wang. *A Dedicated Multi-Channel MAC Protocol Design for VANET with Adaptive Broadcasting*. In Wireless Communications and Networking Conference (WCNC), 2010 IEEE, pp. 1–6, April 2010. 25, 27

[Madden 02] Samuel Madden, Michael J. Franklin, Joseph M. Hellerstein and Wei Hong. *TAG: A Tiny AGgregation Service for Ad-hoc Sensor Networks*. SIGOPS Oper. Syst. Rev., Vol. 36, No. SI, pp. 131–146, December 2002. 30, 33, 90

[Mammu 13] A.S.K. Mammu, A. Sharma, U. Hernandez-Jayo and N. Sainz. *A Novel Cluster-Based Energy Efficient Routing in Wireless Sensor Networks*. In Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on, pp. 41–47, March 2013. 3, 4

[Mammu 15] Aboobeker Sidhik Koyamparambil Mammu, Unai Hernandez-Jayo, Nekane Sainz and Idoia de la Iglesia. *Cross-Layer Cluster-Based Energy-Efficient Protocol for Wireless Sensor Networks*. Sensors, Vol. 15, No. 4, p. 8314, 2015. 4

[Maslekar 09] N. Maslekar, M. Boussedjra, J. Mouzna and L. Houda. *Direction based clustering algorithm for data dissemination in vehicular networks*. In Vehicular Networking Conference (VNC), 2009 IEEE, pp. 1–6, 2009. 23

[Maslekar 11] N. Maslekar, M. Boussedjra, J. Mouzna and H. Labiod. *A stable clustering algorithm for efficiency applications in VANETs*. In Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International, pp. 1188–1193, 2011. 24

[Morales 11] M.M.C. Morales, Choong seon Hong and Young Cheol Bang. *An Adaptable Mobility-Aware Clustering Algorithm in vehicular networks*. In Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific, pp. 1–6, 2011. 22

[Murray 08] T. Murray, T. Murray, M. Cojocari and Huirong Fu. *Measuring the performance of IEEE 802.11p using ns-2 simulator for vehicular networks*. In Electro/Information Technology, 2008. EIT 2008. IEEE International Conference on, pp. 498–503, May 2008. 113

[Nadeem 04] Tamer Nadeem, Sasan Dashtinezhad, Chunyuan Liao and Liviu Iftode. *TrafficView: Traffic Data Dissemination Using Car-to-car Communication.* SIGMOBILE Mob. Comput. Commun. Rev., Vol. 8, No. 3, pp. 6–19, July 2004. 32, 34

[Narra 13] Hemanth Narra. https://codereview.appspot.com/4747046. 2013. 112

[Omar 11] H.A. Omar, Weihua Zhuang and Li Li. *VeMAC: A novel multichannel MAC protocol for vehicular ad hoc networks.* In Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on, pp. 413–418, 2011. 25, 26

[Panayotis 12] Juan Nicolas Ibanez Rivas Panayotis. *Measuring road congestion.* JRC scientific and policy reports, Vol. 1, p. 2, 2012. 83

[Picconi 06] Fabio Picconi, Nishkam Ravi, Marco Gruteser and Liviu Iftode. *Probabilistic Validation of Aggregated Data in Vehicular Ad-hoc Networks.* In Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks, VANET '06, pp. 76–85, New York, NY, USA, 2006. ACM. 28

[Rajagopalan 06] R. Rajagopalan and P.K. Varshney. *Data-aggregation techniques in sensor networks: A survey.* Communications Surveys Tutorials, IEEE, Vol. 8, No. 4, pp. 48–63, Fourth 2006. 84

[Rawashdeh 08] Z.Y. Rawashdeh and S.M. Mahmud. *Media Access Technique for Cluster-Based Vehicular Ad Hoc Networks.* In Vehicular Technology Conference, 2008. VTC 2008-Fall. IEEE 68th, pp. 1–5, 2008. 17, 18

[Salhi 09] I. Salhi, M.O. Cherif and S.M. Senouci. *A New Architecture for Data Collection in Vehicular Networks.* In Communications, 2009. ICC '09. IEEE International Conference on, pp. 1–6, 2009. 21

[Shea 09] C. Shea, Behnam Hassanabadi and S. Valaee. *Mobility-Based Clustering in VANETs Using Affinity Propagation.* In Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE, pp. 1–6, 2009. 23

[Shrivastava 04] Nisheeth Shrivastava, Chiranjeeb Buragohain, Divyakant Agrawal and Subhash Suri. *Medians and Beyond: New Aggregation Techniques for Sensor Networks.* In Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems, SenSys '04, pp. 239–249, New York, NY, USA, 2004. ACM. 30, 33, 91

[Souza 10] E. Souza, I. Nikolaidis and P. Gburzynski. *A New Aggregate Local Mobility Clustering Algorithm for VANETs.* In Communications (ICC), 2010 IEEE International Conference on, pp. 1–5, 2010. 21

[Su 07] Hang Su and Xi Zhang. *Clustering-Based Multichannel MAC Protocols for QoS Provisionings Over Vehicular Ad Hoc Networks*. Vehicular Technology, IEEE Transactions on, Vol. 56, No. 6, pp. 3309 –3323, nov. 2007. 18

[Tomar 13] RanjeetSingh Tomar, Shekhar Verma and GeetamSingh Tomar. *Cluster Based RSU Centric Channel Access for VANETs*. In MarinaL. Gavrilova and C.J.Kenneth Tan, editeurs, Transactions on Computational Science XVII, volume 7420 of *Lecture Notes in Computer Science*, pp. 150–171. Springer Berlin Heidelberg, 2013. 16

[Tung 13] Lung-Chih Tung, Jorge Mena, Mario Gerla and Christoph Sommer. *A cluster based architecture for intersection collision avoidance using heterogeneous networks*. In Ad Hoc Networking Workshop (MED-HOC-NET), 2013 12th Annual Mediterranean, pp. 82–88, June 2013. 23

[Union 13] European Union. *Road fatalities in the EU since 2001*. 2013. 83

[Uzcategui 09] R. Uzcategui and G. Acosta-Marum. *Wave: A tutorial*. Communications Magazine, IEEE, Vol. 47, No. 5, pp. 126–133, May 2009. 2

[V0.0.4 12] ETSI EN 302 636-5-1 V0.0.4. *Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 5: Transport Protocols; Sup-part 1: Basic Transport Protocol (Draft)*. 2012. 85

[Venkata 11] M.D. Venkata, M.M.M. Pai, R.M. Pai and J. Mouzna. *Traffic monitoring and routing in VANETs - A cluster based approach*. In ITS Telecommunications (ITST), 2011 11th International Conference on, pp. 27–32, 2011. 20

[Wang 08] Zhigang Wang, Lichuan Liu, MengChu Zhou and N. Ansari. *A Position-Based Clustering Technique for Ad Hoc Intervehicle Communication*. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, Vol. 38, No. 2, pp. 201–208, 2008. 20

[Watanabe 05] F. Watanabe, M. Fujii, M. Itami and K. Itoh. *An analysis of incident information transmission performance using MCS/CDMA scheme*. In Intelligent Vehicles Symposium, 2005. Proceedings. IEEE, pp. 249–254, June 2005. 27

[Weimerskirch 09] André Weimerskirch, Jason J Haas, Yih-Chun Hu and Kenneth P Laberteaux. Data security in vehicular communication networks, pp. 299–363. John Wiley & Sons, Ltd, 2009. 113

[Wischhof 05] L. Wischhof, A. Ebner and H. Rohling. *Information dissemination in self-organizing intervehicle networks*. Intelligent Transportation Systems, IEEE Transactions on, Vol. 6, No. 1, pp. 90–101, March 2005. 32, 33

[Yao 02] Yong Yao and Johannes Gehrke. *The Cougar Approach to In-network Query Processing in Sensor Networks*. SIGMOD Rec., Vol. 31, No. 3, pp. 9–18, September 2002. 30, 31

[Yu 07] Fan Yu and S. Biswas. *Self-Configuring TDMA Protocols for Enhancing Vehicle Safety With DSRC Based Vehicle-to-Vehicle Communications*. Selected Areas in Communications, IEEE Journal on, Vol. 25, No. 8, pp. 1526–1537, Oct 2007. 25, 26

[Yu 13] Hang Yu, Zhiqiang He and Kai Niu. *STDMA for vehicle-to-vehicle communication in a highway scenario*. In Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications (MAPE), 2013 IEEE 5th International Symposium on, pp. 133–138, Oct 2013. 25, 26

[Zhang 06] Xi Zhang, Hang Su and Hsiao-Hwa Chen. *Cluster-based multi-channel communications protocols in vehicle ad hoc networks*. Wireless Communications, IEEE, Vol. 13, No. 5, pp. 44–51, 2006. 17, 18

[Zhang 08] Yan Zhang and Jim Mee Ng. *A Distributed Group Mobility Adaptive Clustering Algorithm for Mobile Ad Hoc Networks*. In Communications, 2008. ICC '08. IEEE International Conference on, pp. 3161–3165, 2008. 21