On QoS Provisioning For Vehicular Safety Communication

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Sarah Sharafkandi

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

David H.C. Du

September, 2013

Acknowledgements

My deepest gratitude and appreciation go to my Ph.D advisor, Professor David H.C. DU, for his continuous support and guidance during my time in graduate school. Also, I give my sincere thanks to my Ph.D committee for their good suggestions and sincere help for my research: Professor David Lilja, Professor Zhi-Li Zhang, Professor Volkan Isler.

I express my most sincere appreciation to my internship manager John B. Kenny in Toyota Info Technology Center for his guidance and support and positive outlook. I am also most grateful to my dear friend and colleague Gaurav Bansal for his help and encouragement during the time I spent as an intern in Toyota Info Technology Center.

My warmest thanks goes to my friends Sara Nasiri Amini, Shahrouz Takyar, Baktash Boghrati, Iman Aganj, Rashad Moarref, Amir Kashi and Meisam Razavyan who made my graduate school years in Minnesota special and memorable.

I am eternally beholden to my family. My warmest thanks to my husband Alireza Razavi who has always been my best friend and supporter. I am also deeply grateful to my Mother Nosrat and my brothers Shahin and Ramin. Words cannot express how much they helped me pursue my studies with their endless encouragement and unconditional love. I would also like to express my sincer gratitude to my uncle Rassoul for his endless love and support during the years of my graduate school.

Dedication

To my husband Alireza Razavi.

Abstract

This dissertation studies the problem of safety communication in vehicular networks. Despite advances in automative safety during the past decades, still thousand of injuries and fatalities happen yearly in vehicle-related accidents. Dedicated Short Range Communication (DSRC) technology enables vehicles to communicate to each other through wireless medium, so they can inform each other of a potential danger on the road. Two important applications of vehicular network technology are collision avoidance and collecting traffic information. Collision avoidance relies on periodic sharing of safety messages to avoid accidents. Each vehicle that receives these safety messages from its neighbors, uses them to determine if any of the neighbors poses a collision threat. If a vehicle determines that this is the case, the onboard unit will warn the driver. For collision avoidance it is important that the status data of the vehicles be delivered on time and so the Medium Access Control (MAC) protocol design is very important. This dissertation studies two different approaches to the MAC design protocol: First, we use the QoS mechanism of IEEE 802.11 to reduce the collision rate among safety packets when IEEE 802.11p is used which is the likely scenario in United States. Then, we introduce a new contention-free TDMA-based MAC protocol tuned for vehicular communication which can guarantee an upper bound on the delivery delay of safety messages. Finally, we propose a strategy for collecting safety information of vehicles in a Road Side Units (RSU). This information can be used for analysis on the road traffic condition which can be then shared through a disseminating strategy with the vehicles on the road.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Each year in United States, vehicle related crashes result in loss of thousands of lives and billions of dollars. Although the automotive safety has been steadily improving during the past decades, the total numbers of injuries and fatalities have still remained relatively flat due to the increasing number of vehicles and total miles driven [2]. To further increase safety, a more active approach is needed to prevent collisions on the road, as opposed to the traditional passive approaches such as airbags and seatbelts which are designed to reduce the chance of fatalities and injuries after the collision.

Communication-based safety technologies is a promising way to move toward active safety technologies. Communication among vehicles provides extra information to the driver to warn him/her of a potential danger in an earlier phase, to prevent collisions. One of these technologies is Dedicated Short Range Communication (DSRC) [3] which supports Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication through a wireless medium. This technology is under active development in the United States and in other countries.

Fig. 1.1 shows a few examples of collision avoidance using DSRC technology. Each vehicle with a DSRC device broadcasts Basic Safety Messages (BSM) [4] containing status information, including location, speed, acceleration and heading, every hundred(s) millisecond over a range of a few hundred meters. Each vehicle that receives these safety messages from its neighbors, uses them to compute the trajectory of each neighbor. Then it compares the computed trajectory with its own predicted path and then determines if any of the neighbors poses a collision threat. If a vehicle determines that this is the case, the onboard unit will warn the driver. In some cases, it may even assist the driver in controlling the vehicle. The warning message to the driver can be conveyed audibly, visually (e.g., heads-up-display, dashboard screen, mirror signal), and haptically (e.g., shaking seat or steering wheel), and can range in intensity from inform to caution to warning [1]. While the communication between DSRC devices must follow carefully designed interoperability standards, the internal threat computation and warning system employed by a vehicle is determined by the automobile manufacturer.

The U.S. Department of Transportation (DOT) and several automakers in the United States have studied DSRC-based collision avoidance. The Vehicle Safety Communications Applications project, completed in 2009, demonstrated the feasibility of several V2V safety applications [3], including:

- Emergency electronic brake lights (hard-braking vehicle ahead)
- Forward collision warning (stopped vehicle ahead)
- Blind Spot Warning+Lane Change Warning
- Intersection movement assist
- Do not pass warning
- Control loss warning

A few of these are illustrated in Fig. 1.1.

DOT has estimated that by using DSRC technology, V2V communication can address up to 82% of all traffic related crashes in the United States [1] which involve unimpaired drivers. This

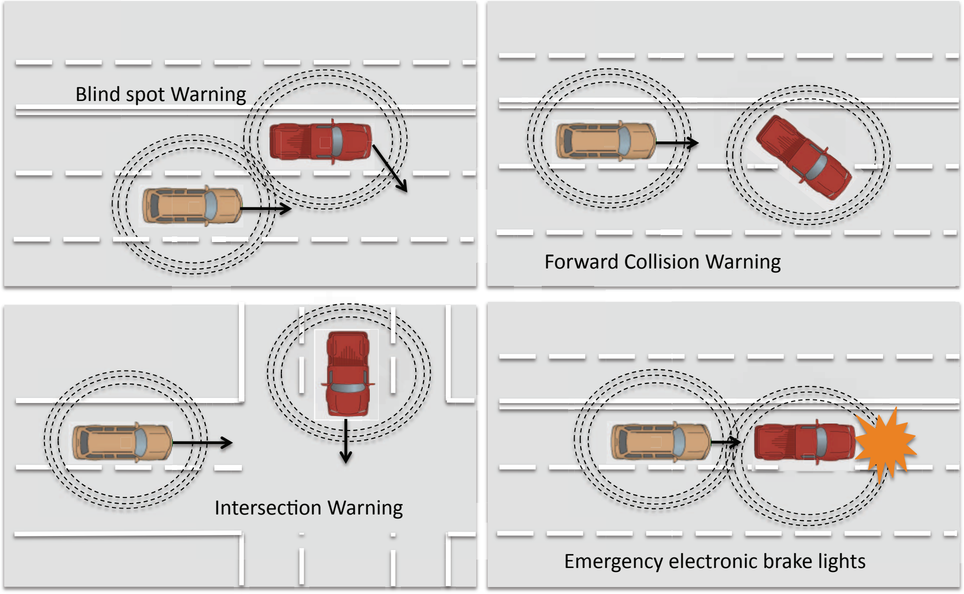

Figure 1.1: Collision avoidance using safety message communication

will potentially save thousands of lives and billions of dollars. In 2013, the National Highway Traffic Safety Administration (NHTSA) within the U.S. DOT plans to decide whether to use regulations to require or encourage deployment of DSRC equipment in new vehicles in the U.S. [5].

DSRC can be used for many other applications beyond collision avoidance. For example, DSRC can be used to assist navigation, make electronic payments (e.g., tolls, parking, fuel), improve fuel efficiency, gather traffic probes, and disseminate traffic updates. It can also be used for more general entertainment and commercial purposes.

Vehicles may also communicate to and from DSRC Roadside Units (RSUs) through safety messages and other types of message. A vehicle can learn from an RSU about the existence of a hazard such as ice, fog or emergency vehicle or the state of a signal at an intersection.

To make DSRC a feasible technology, the U.S. Federal Communications Commission (FCC) has allocated 75 MHz of licensed spectrum in the 5.9 GHz band for DSRC communication [3].

This spectrum is divided into seven 10MHz channels as shown in Fig. 1.2 . Each channel is either a Service Channel (SCH) or a Control Channel (CCH). The CCH (Ch. 178 in the US) is designated as a special rendezvous channel that the devices will tune to on a regular basis. All other channels in the band plan are designated service channels (SCH). V2V safety messages are expected to be exchanged on Channel 172 [6].

With a broadcast rate of 10 Hz, and the potential for tens or hundreds of vehicle in a given area, BSMs are expected to constitute the dominant source of traffic on Ch. 172. In Europe Co-operative Awareness Messages (CAM) [CAM], analogous to the BSM, are similarly expected to represent most of the data on the Control Channel in the ITS-G5A spectrum.

In addition to collision avoidance, the information carried by BSMs can also be useful for researchers if collected at a Road Side Unit (RSU) especially for statistical analysis on travel times and speeds on the roadways. This information can be disseminated to vehicles to inform them of accidents, dangerous road conditions or high traffic roadways.

In this dissertation, we focus on two applications of DSRC: collision avoidance and gathering traffic information. Both of these applications may use BSMs as their primary source of information. But they have different requirements. Collision avoidance requires the safety packets to be delivered successfully within a certain deadline. In case of traffic probing, the bandwidth consumption is the primary concern since the amount of data to be gathered is large.

For collision avoidance, the design of MAC layer protocol is significantly important because it affects the delay and reliability of safety message communication. Currently, DSRC uses the IEEE 802.11p medium access control (MAC) protocol [7], which is based on carrier sense multiple access/collision avoidance (CSMA/CA). IEEE 802.11p is prone to packet collision where a collision happens when two or more packet transmissions overlap in time at a receiver. When there is a collision, one or more of the packets involved cannot be received. The packet collision probability increases with the offered load, which in our study is primarily a function of node density. Different approaches has been suggested for this problem. Some of them are solutions that are within the IEEE 802.11p standard. Most of them control the collision rate using rate or power control [8, 9]. Some solutions propose a completely new MAC protocol that is primarily designed for V2V environment. In this dissertation we focus on both of these approaches.

For gathering traffic information at a RSU, it is important to design a data gathering protocol with minimum bandwidth consumption. In this dissertation, we propose an approach to exploit the similarities among status data among vehicles to reduce the total amount of data that is transmitted. This approach can also be used in environments such as sensor networks where energy is constrained.

Considering the above issues, we first consider the problem of using standard QoS mechanism of IEEE 802.11 called Enhanced Distributed Channel Access (EDCA) to minimize frame collisions among periodic vehicle safety messages. We make two main contributions: 1) We demonstrate an Access Category (AC) isolation technique that dramatically reduces the collision probability of high priority packets in dense environments. 2) We introduce a novel concept called virtual division by which significant reductions in collisions among lower priority packets can be achieved. AC isolation and virtual division are conformant with IEEE 802.11p and can be supported by existing integrated circuit implementations. These techniques are motivated by the periodic safety message environment under development in the US, Europe and parts of Asia, but they may have applicability for other vehicular applications or more general WiFi networks. The results are verified via extensive NS-2 simulations of both simple and hidden node topologies.

Second, we introduce a new contention free MAC protocol for periodic safety message communication which can bound the delay in a very dense vehicular networks and derive the theoretical lower bound on the maximum delay, in delivery of BSMs. This bound is a linear function of the maximum number of vehicles that interfere with one vehicle, in the network. The theoretical lower bound, can be achieved, if there is an entity that is aware of the position,

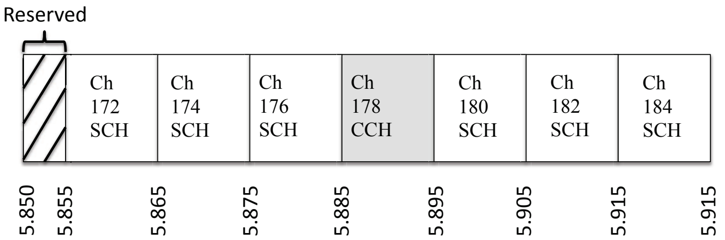

Figure 1.2: United States DSRC Band Plan [1]

and interference range of every vehicle. Since no such entity exists in vehicular networks, we propose a distributed cluster-based MAC protocol, which maximizes collision-free parallel transmissions of periodic safety messages. The proposed MAC protocol guarantees an upper bound on the delivery delay of routine safety messages which is only within 4 times from the optimal delay. Our protocol is distributed and dynamic, and easily adjusts to the changes in the network.

Third, we study the problem of collecting the traffic information from all the vehicles in access points for traffic probing. This raw data can be used later for further analysis on the traffic information on certain roads at different times of the day. Although, monitoring traveling information is possible through deployment of inductive loops, cameras and other sensors on the roads, these solutions are very costly. A more cost effective solution, is to use vehicles as traffic probe sensors. An efficient protocol is required to collect all these raw data across the network at a certain access point or RSU with minimum bandwidth consumption since bandwidth is a sparse resource especially when the highway is dense. For this purpose, we exploit the correlation on the data of vehicles to reduce the bandwidth consumption during data collection by suppressing the redundant data from multiple vehicles. In our methodology, a certain subset of vehicles send out their data without any suppression (full transmission). The vehicles that perform full transmissions are called reference nodes. The rest of vehicles only send out the difference between their data and that of reference nodes. Our strategy is based on two key principals. The first is to minimize the total number of full transmissions. This strategy minimizes the overall bandwidth consumption since in general, full transmissions consume significantly more bandwidth than transmission of the compressed data. The second principal is to remove the redundant data sent by each forwarding node on the path to the access point. In this dissertation, we first obtain a theoretical lower bound on the total number of full transmissions required for lossless collection of data at the access point. This lower bound is equal to the size of a specific subset of nodes. We then introduce an efficient algorithm that approximates the elements of this set. We also find a lower bound on the minimum number of bits that a node which is responsible for forwarding the data of its neighborhood should transmit for lossless data collection. Based on our theoretical findings, we propose a practical data collection algorithm which can reduce the overall bandwidth consumption significantly. Finally, we present the numerical simulations that verify our theoretical results. This result is also applicable for sensor network environment where the objective is to minimize energy when data is transmitted.

The rest of this dissertation is organized as follows:

- Chapter 2 describes a novel use of EDCA mechanism of IEEE 802.11p to improve vehicle safety communication.

- In chapter 3, we introduce a new contention free TDAMA-based MAC protocol for safety communication in vehicular network.

- Chapter 4 describes a data collection mechanism for gathering traffic probes in an access points.

- Chapter 5 concludes this dissertation and present future works of MAC protocol design for vehicular networks.

Chapter 2

A Novel use of EDCA to Improve Vehicle Safety Communication

The effectiveness of DSRC for collision avoidance depends on the communication performance of safety messages. EDCA, the standard IEEE 802.11 QoS capability, was designed for networks with a mix of voice, video and best effort traffic. This chapter examines how to use EDCA to reduce frame collisions for a channel dominated by periodic safety messages. We make two main contributions: Access Category(AC) Isolation and Virtual Division. AC isolation eliminates inter-AC countdown collisions, dramatically improving the success rate for high priority packets. Virtual Division uses isolation in a novel way that also reduces collisions for lower priority packets. Both techniques are consistent with the 802.11 standard. This chapter includes detailed analysis and insightful NS-2 simulations.

2.1 Introduction

Collision avoidance applications in the US are enabled through the exchange of periodic safety messages on Ch. 172. The Basic Safety Message (BSM) [3] is broadcast by each vehicle to convey its core state, e.g. location, speed, heading, to other vehicles within a few hundred meters so they can assess the extent to which the sender poses a collision threat. Other safety-related message types might also be sent on Ch. 172, for example by a roadside unit to advertise the geometry and signal state of an intersection.

The safety channel traffic can be homogeneous or heterogeneous with respect to its priority classes. If we do not classify messages based on relative priority, we call the traffic homogeneous. In the more general heterogeneous case messages will be classified into two or more categories. The classification is based on type (e.g. BSM vs. intersection message) and/or content. For example, a sender could classify one BSM as routine and another (perhaps indicating sudden deceleration) as critical, and assign them to different categories. Note that both routine and critical BSMs have the potential to prevent collisions, so we want both to be successful. But, we attach a higher value to the sender-identified critical class and thus seek to improve its relative reception probability. We are not concerned with the classification criteria.

DSRC uses a variation of IEEE 802.11 at the medium access control (MAC) layer [7], a carrier sense multiple access with collision avoidance (CSMA/CA) protocol. A packet collision happens when transmission of multiple packets overlap in time at a receiver. While CSMA/CA attempts to avoid collisions, they do occur, especially in high channel load environments. Collisions are primarily of two types: simultaneous countdown, in which the CSMA/CA mechanism leads two or more nodes to initiate transmission at the same time, and hidden node, in which the transmitting nodes do not sense each other. To evaluate the performance of the IEEE 802.11p MAC in a vehicular network, we use the Packet Error Ratio (PER) metric, which is the fraction of the transmitted packets that are not successfully received. This can be computed in aggregate or as a function of specific senders or distances. Latency is another potential metric of interest.

Given that safety applications are tolerant of a small number of lost packets, each effectively adding 100 msec to the model update interval, we consider queuing latencies on the order of tens of milliseconds to be relatively insignificant. In this dissertation we choose techniques that limit queuing latencies to this order, and we do not further consider latency as a metric.

Our approach is to use the priority mechanism of IEEE 802.11p, called Enhanced Distributed Channel Access (EDCA). An EDCA-equipped device accesses the medium with up to four queues, each associated with an Access Category (AC) and each characterized by three parameters: CW_{min} (minimum Contention Window), AIFSN (Arbitration Inter-frame Spacing Number), and TXOP (Transmission Opportunity), to give priority to different classes of traffic. TXOP is required to be 0 for IEEE 802.11p, and so we exclude that from our studies. Assigning a frame to an AC determines the CW_{min} and AIFSN parameters the frame will use to compete for channel access when it reaches the head-of-line. In a heterogeneous system, each class of frames is assigned to one AC; in a homogeneous system there is only one AC.

EDCA was designed primarily to control channel access delays for real-time voice and video streams, while also avoiding excessive collision loss for bursty best effort flows. A high priority packet was assumed to be primarily latency sensitive. Our challenge is to apply the EDCA tool in our somewhat unique periodic safety message regime, where collision loss is the most important criterion. A high priority packet in our environment is assumed to be primarily loss sensitive. The effect of IEEE 802.11p EDCA parameters on the PER has not been well studied especially for periodic message transmissions.

This dissertation offers two principal contributions. The first is a comprehensive study of the effect of AIFSN and CW_{min}, on the PER of periodic safety packets. In the heterogeneous case this includes the effect on the interaction between ACs, and we present an AC isolation technique such that a packet from a given AC can only collide with another packet if it is in the same AC. Under high load, AC isolation significantly reduces the collision rate of higher priority packets compared to more conventional parameter choices. The second contribution is a novel methodology called virtual division, which can be applied in either a homogeneous or heterogeneous setting. In this method, packets that are nominally of the same priority are nevertheless segregated into different ACs (i.e. virtually divided), and using AC isolation the net collision rate is reduced compared to placing all the packets in a single AC. In order to achieve fairness, each node with this class of packet will dynamically alternate the assignment to the various ACs according to a desired proportion. Recall that the queuing delay is controlled to be less than the message interval, so there is no concern with out-of-order delivery of packets within the same category. Virtual division can significantly improve the PER in highly loaded scenarios. For example, in simulations of a highway with 300 vehicles per kilometer, we can improve the PER from 60% to 36% for vehicles that are within 100-meters of each other. Virtual division does not need adaptive tuning based on the channel load. In addition, it is a practical mechanism in the sense that it conforms to IEEE 802.11 and can be implemented on existing DSRC devices.

We discuss related work in Section 3.2 and describe the standard EDCA mechanism in Section 2.3. We present results based on a simple simulated topology in Section 2.4. In 2.4.1 and 2.4.2 we study the effect of CWmin and AIFSN on the PER for the homogeneous cases, and in 4.3 for the heterogeneous case. This leads to the definition of AC isolation in Section . Building on the isolation technique, we introduce the virtual division mechanism in Section 2.4.4. In section 2.5 , we present simulation results for a highway scenario with hidden nodes and realistic fading model, using both AC isolation and virtual division. In sections 2.6, we briefly discuss interaction of virtual division with message rate congestion control, respectively. We offer conclusions in Section 4.7.

2.2 Related Work

The EDCA mechanism defined in IEEE 802.11e has been well studied both theoretically [10] and experimentally [11] for the WLAN. In the context of vehicular networks, EDCA has been studied in [12–16] [2,3,6,9,13,14]. The authors in [12, 14] have done a simulation study on the effect of IEEE 802.11e EDCA parameters on the performance of safety message transmission. In [12], the study was conducted for a scenario with only one high priority node. For this particular case, it was concluded that both a smaller CW_{min} and AIFSN would be beneficial for the high priority node because it decreases the chance of a collision between the high priority packet and low priority ones. The study is limited in that there will generally be more than one high priority node. Also, the study did not suggest a mechanism that can completely avoid collisions between high and low priority nodes, as we do in this dissertation. In [14], beside the smaller CW_{min} and AIFSN, a repetition mechanism was suggested for high priority packets, in order to increase the probability of at least one being received. However, the repetition mechanism adds to channel congestion and increases the chance of collision with other categories of traffic. In [15], simulations of 802.11p EDCA parameters showed that high priority nodes suffer more collisions at high loads, but the authors did not offer a solution to reduce the collision rate of high priority packets In [16], the authors proposed a scheme where they dynamically adjust the transmission power based on the estimated local vehicle density, while the CW_{min} size is adapted according to the collision rate. In Section 2.6 we briefly discuss the interaction between EDCA and active congestion control techniques. In [13], an analytical study is presented for EDCA 802.11p where they introduce a Markov chain to model each access category, and use that to estimate the error rate. However, their model cannot be used for periodic safety message transmission since they are assuming that the traffic is generated in bursts according to a Poisson process. Furthermore, they are assuming that the number of frames generated in each burst has a geometric distribution whereas the size of BSMs is constant.

2.3 EDCA Mechanism of IEEE 802.11P

In this section, we present an overview of IEEE 802.11p MAC protocol and its EDCA mechanism.

2.3.1 IEEE 802.11 MAC Protocol

The legacy 802.11 Distributed Coordination Function (DCF) uses a carrier sense multiple access/collision avoidance protocol (CSMA/CA). Before starting a transmission, each wireless node, senses the channel. If the channel has been idle for a period equal or longer than a specified duration, called Distributed Inter-Frame Space (DIFS), then the node begins transmitting. The DIFS period consists of a 32 μ sec SIFS (Short Inter-Frame Space) interval plus two 13 μ sec time slots (these times are standard for a 10 MHz channel). If the channel is sensed busy during the DIFS interval, the node starts a countdown procedure by choosing a random number between 0 and CW_{min} where CW_{min} is the upper limit of the contention window. Let CW represent the current countdown state. When the channel becomes idle the node decrements CW once per time slot until CW reaches zero or the channel becomes busy again. The countdown procedure is suspended when the channel is busy, and resumes when the channel goes idle again. When CW is decremented to zero the node begins transmission. Since vehicular safety messages are broadcast the upper limit of the contention window is always CW_{min}.

2.3.2 Priority Mechanism of IEEE 802.11p

EDCA is the prioritized CSMA/CA mechanism of IEEE 802.11. EDCA is different from DCF in that messages are classified into access categories based on their relative priorities. There are up to four access categories, AC_i, each with a queue whose head-of-line packet competes for channel access using a pair of parameters CW_{min} and AIFSN. CW_{min} has the same role in EDCA as in DCF. AIFSN is used to calculate AIFS, which plays an analogous role to DIFS for each AC, i.e. when a transmission ends AIFS is the interval before the AC considers the channel idle and begins or resumes a countdown. AIFS is calculated as follows for each ACi:

$$AIFS(AC_i) = SIFS + AIFSN_i.slot \tag{2.1}$$

Notice that if AIFSN is equal to 2, then AIFS = DIFS. There is a subtle difference between the DCF and EDCA countdown methods [16]. In DCF, CW is decremented at the end of each idle slot and a node can start transmitting as soon as CW decrements to 0. In EDCA however, CW is decremented at the beginning of the slot time even if some other node begins transmission during the slot. When CW is decremented to 0, transmission by that AC is allowed at the end of the slot time if the channel is still idle, and is otherwise deferred to the end of the next busy time.

Table 2.1 summarizes the default EDCA parameters of IEEE 802.11p [1] for access categories AC0, AC1, AC2, and AC3. Note that 802.11p allows use of non-default parameters, for example if specified in a controlling standard or regulation.

Table 2.1: Default EDCA parameters of IEEE 802.11p

Access category	CW_{min}	AIFSN
AC0	15	9
AC1	15	6
AC2	7	3
AC3	3	2

Fig. 2.1 illustrates how the head-of-line packets of 4 EDCA ACs access the channel using the default parameters. Each AC is shown with its AIFS and $CW_{min} + 1$ countdown slots. The vertical line at left represents a transition from busy to idle on the channel, but a given AC will not recognize the channel as idle until its AIFS elapses, and only then if no higher priority AC has seized the channel. AC3 recognizes the third slot after SIFS as available. The head-of-line packet will begin transmission, if CW=0, or decrement CW. If no transmission begins in the

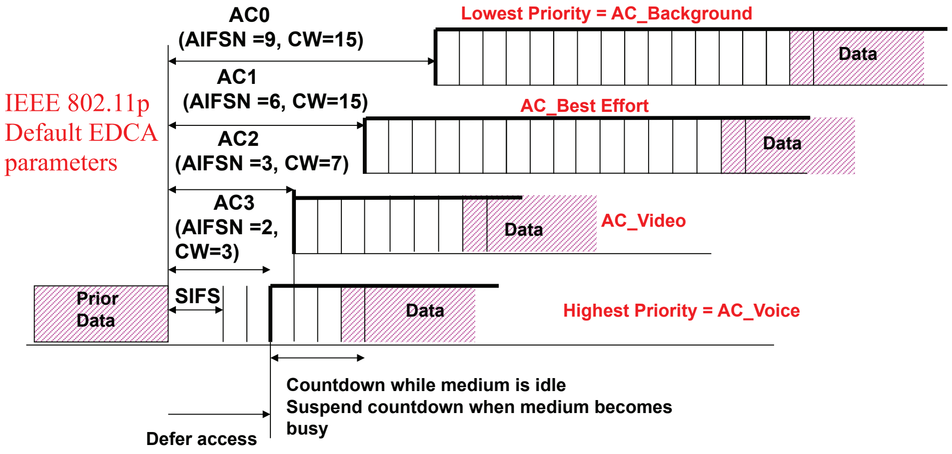

Figure 2.1: AIFS of the default EDCA access categories

third slot, then both AC3 and AC2 recognize the fourth slot as available. If either head-of-line packet has CW=0 it begins transmission, otherwise it decrements CW (if two head-of-line packets in the same node decrement to zero on the same slot, an internal arbitration determines which is transmitted). The head-of-line of AC1 does not recognize an opportunity to send/decrement until the seventh slot, and for AC0 it is the tenth slot. If the medium becomes busy at any time, all countdowns are suspended until the busy period ends. AC3 has the smallest CW_{min} and AIFSN, so it has the smallest average waiting time among the 4 categories. However, a frame from AC2 might gain access first if it has a small enough CW. AC0 has the longest average waiting time.

In section 4 we present a thorough study on the effect that each EDCA parameter has on the PER of periodic safety messages.

2.3.3 Packet Loss Reasons in IEEE 802.11

Using a high level analysis of IEEE 802.11, packet transmissions can be unsuccessful for any of the following reasons:

- Simultaneous count down collisions: If two (or more) transmitters sense the same transition from busy to idle channel, they may decrement their respective CWs to zero on the same slot time, and their transmissions will collide.
- Hidden node collisions: A hidden node collision happens when a receiver is within range of two transmitters that cannot sense each other. If transmissions from these transmitters overlap in time, their packets collide.
- Attenuation and Fading: Wireless signals will be attenuated with distance, and the packet may fail if the signal power is not strong enough at the receiver. Fading is the variation in signal attenuation, usually modeled as a random process.

Attenuation and fading are physical layer problems that are independent of the MAC layer protocol. Hidden node collisions are difficult for the 802.11 MAC to prevent, especially for broadcast packets that cannot use the RTS/CTS mechanism Thus, our implementation of EDCA targets the only remaining category of packet loss, the simultaneous count down collision. In the next section we use a simple topology where all the nodes are at the same location. This topology helps us to exclude packet losses that are due to attenuation, fading or hidden node collisions so we can focus on countdown collisions. In section 2.5, we will study a realistic highway scenario with hidden node collisions and a realistic fading model.

2.4 Effect of EDCA Parameters on PER For a Simple Topology

In this section, we study the effect of CWmin and AIFSN on PER for periodic traffic, through simulation. We use the NS-2.34 simulator [18] for our analysis, and for now, we consider a simple topology where all the nodes are located at the same place. We begin the study with this simple scenario in order to exclude packet losses that are caused by signal attenuation, fading and hidden terminals, and focus on the packet collisions that are due to simultaneous count down to zero. We study the more realistic highway scenario in section 5. In our simulations, each node transmits a 350-byte packet at 6 Mbps with an average interval of 100msec. While a real system would schedule transmissions at close to 100 msec intervals (perhaps with a small random jitter), in our simulations when a vehicle generates a message at time t, it schedules its next message at time t+T, where T is a uniform random variable over the interval [50, 150]. This 100 msec random interval allows us to approximate a Monte Carlo ensemble over successive transmissions, so that the statistics of packet loss are independent of the vehicles initial phases. The simulation parameters have been summarized in Table 2.2. These parameters are used for both simple topology and realistic highway scenario.

Table 2.2: NS2-34 simulation parameters

Parameter	Value
Noise floor	-99 dBm
Carrier sense threshold	-96 dBm
Packet reception SINR	7 dBm
Packet size	350 bytes
Data rate	6 Mbps
Transmission Power	10 dbm

2.4.1 Effect of CWmin on Homogeneous Traffic

In this subsection, we study the effect of CW_{min} on PER when packets are transmitted periodically. We assume that the traffic is homogenous, i.e. all packets are assigned to a single AC. For simplicity we use $AIFSN = 2$, which is the minimum allowed.

Fig. 2.2 shows the PER (y-axis) versus number of nodes (x-axis) for CW_{min}= 3, 7, 15, 31. We see that for low-to-moderate number of vehicles there is a clear advantage to choosing a larger CW_{min}, and we can explain this trend informally with the following logic. For these loads, the countdown interval that a node experiences is unlikely to be interrupted by another busy period. Thus, the most likely collision modality is between two nodes that each became active during the same busy period, and they collide if and only if they choose the same countdown

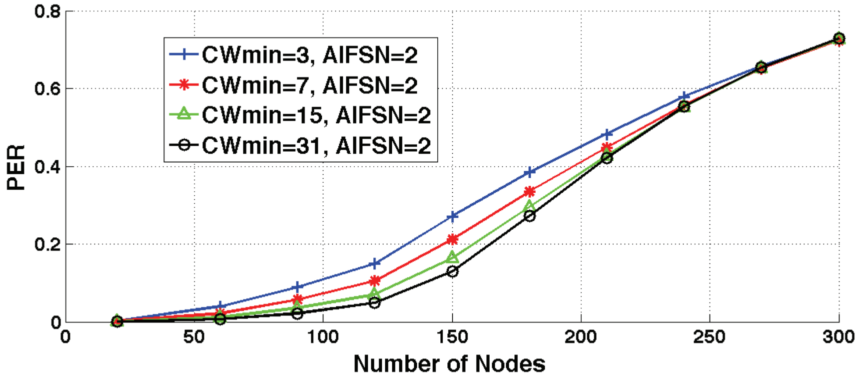

Figure 2.2: Effect of CW_{min} on PER

value CW. The probability of choosing the same value is 1/(CW_{min}+1), so larger CW_{min}results in lower collision probability. When more than two nodes become active in a given busy period, the collision probability is a more complex, but it is still inversely proportional to CW_{min}. As the number of nodes increases, the countdown of a given node is increasingly likely to span multiple busy periods, and the advantage of a large CW_{min} diminishes (e.g. for 200 nodes in Fig. 2). In the extreme, the node is only able to decrement CW once per busy period, i.e. the first slot after AIFS is always seized by one or more transmitters with $CW = 0$. In that case, the probability of collision is the same after every busy period, and is thus independent of the initial CW value a node chooses. We see this effect in Fig. 2 for 300 nodes, where PER is independent of CW_{min}. This effect has also been shown in [10]. Note that we cannot make CW_{min} arbitrarily large without incurring undesirable queuing delays. $CW_{min} = 31$, with queuing delays less than 20 msec, is the largest CW_{min} that we suggest for periodic safety transmission.

2.4.2 Effect of AIFSN on Homogeneous Traffic

In this subsection, we study the effect of AIFSN on PER for the homogeneous case. After each packet transmission, all active nodes wait for $SIFS + AIFSN.slot_{time}$ before they declare the channel idle, and either start transmitting or decrementing CW. During this AIFS period

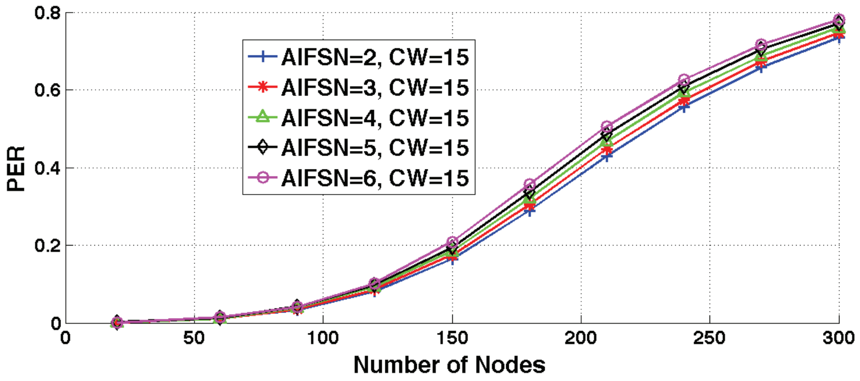

Figure 2.3: Effect of AIFSN on PER

nobody can transmit. When we increase AIFSN, we increase this waiting time for all the nodes, which effectively reduces the number of messages/second that the channel can support. Thus, the probability of collision increases with increasing AIFSN, since there is less time available for data transmission.

Fig. 2.3 shows the effect of AIFSN on PER for different number of nodes. We have tried $AIFSN =$ 2,3,4,5,6 and we see a clear trend of increasing PER with increases in AIFSN. This shows that, for a homogeneous scenario, there is no reason to use any AIFSN larger than 2.

2.4.3 AC Isolation for Heterogeneous Traffic

Safety messages can often be classified by the sender as routine or critical, and placed in different ACs to give the critical messages priority channel access. Here we investigate how to choose the EDCA parameters for this heterogeneous traffic, i.e. for these higher priority (HP) and lower priority (LP) ACs. For simplicity, we assume in this subsection that HP traffic is 10% of the whole. Note that the principles illustrated here do not depend significantly on this specific percentage, nor do they depend on whether the critical traffic is concentrated in 10% of the nodes or distributed.

We progress to our result in two steps, both illustrated in Fig. 2.4. In the first step we

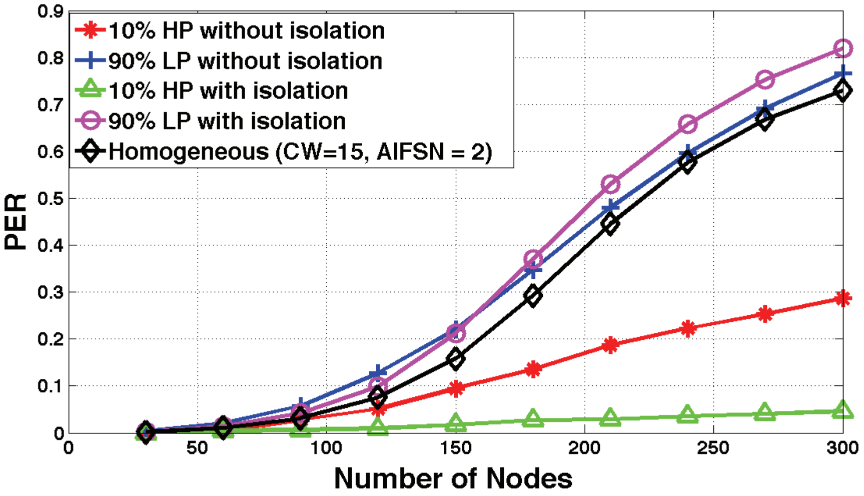

Figure 2.4: AC3 and AC2 vs. Homogeneous

use the default AC3 and AC2 parameters in Table 2.1 for HP and LP ACs, respectively. AC2 is a reasonable choice for the LP safety traffic because there might be traffic on the safety channel that is even lower priority than routine BSMs. In Fig. 2.4 we compare the PER for these two ACs (labeled without isolation) with that of homogeneous traffic. The differences are best illustrated for the highly loaded 300 node case, in which the PER for HP has significantly decreased (from 72% to 30%) compared to the homogeneous case, and this comes at a relatively small price of increasing the PER of LP to 75%. The explanation leads us to more general observations about how to choose parameters.

Fig. 2.5 illustrates how nodes with an LP packet and nodes with an HP packet compete for the channel after a busy period. HP nodes, with $AIFSN = 2$, recognize the channel as idle after $AIFS(2) = SIFS + 2.T_s$ where T_s is the duration of one time slot. LP nodes wait an extra time slot.

In the first slot after AIFS(2) LP nodes consider the channel busy and HP nodes only compete against each other, making the collision probability for this slot is very low. This is key to understanding the advantage of low AIFSN in a heterogeneous channel. If no HP node takes the

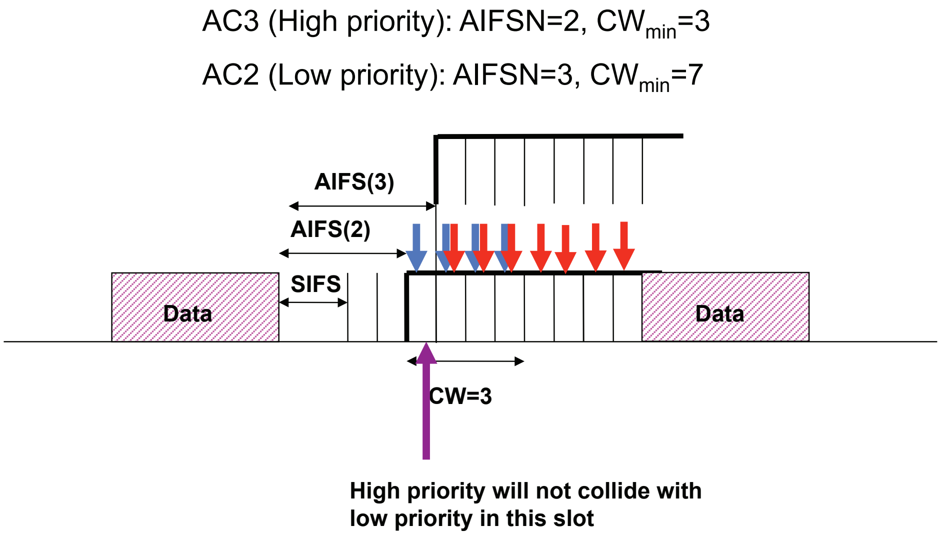

Figure 2.5: AC2 and AC3 competing

first slot, LP and HP nodes compete in the second slot. An HP node has a higher collision probability when transmitting in this slot, and in the others for which LP and HP have overlapping access. If the channel is still idle in the fifth slot LP nodes will only compete among themselves. Note that at heavy loads an HP node that originally picks $CW = 2$ will frequently count down twice to $CW = 0$ before the channel is seized by an LP node, and will thus only compete with other HP nodes after the next busy period, with low collision probability. Thus, for 300 nodes roughly half the HP packets ($CW = 0$ and 2) have low collision probability in the first slot after AIFS(2) and the other half have higher collision probability, which is why the PER is roughly half that of the homogeneous case. This approximate analysis explains qualitatively what we observe. It depends on the HP load being relatively small (e.g. 10% rather than 50%). LP has both higher AIFSN and lower CW_{min} than homogeneous, which accounts for the small PER increase.

Extending the above result leads to the first principle contributions presented in this chapter: AC Isolation. Two ACs are isolated if there is no overlap in the slots in which they attempt to

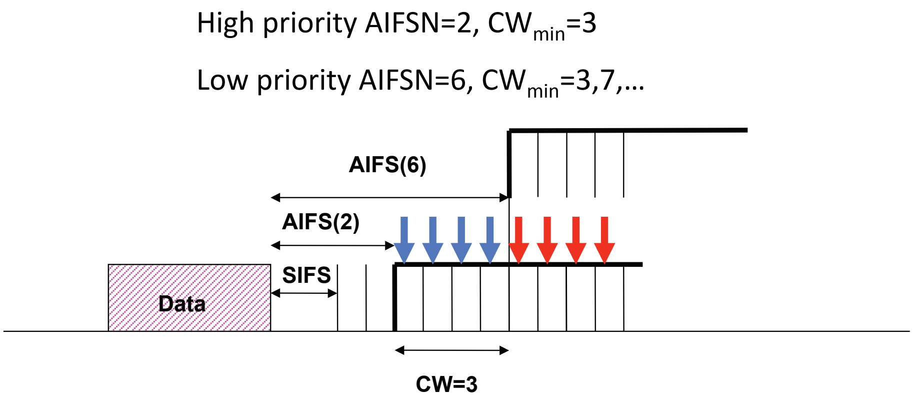

Figure 2.6: AC Isolation mechanism

access the channel. There can be no countdown collisions between packets in isolated ACs.

Fig. 2.6 illustrates how AC Isolation works. Using the same HP AC, we increase the LP AIFSN to eliminate overlapping slots. After a busy period, an active HP queue can completely exhaust its CW (blue arrows), even if $CW = CW_{min}$, before an LP queue considers the channel idle. The two ACs access the channel during non-overlapping slots, completely eliminating countdown collisions between HP and LP packets. This can be thought of as a distributed strict priority queue, i.e. an LP packet can only be sent if no node has an HP packet. Note an HP packet that is generated during the slot competition might transmit at the same time as an LP packet (red arrow slots). This low probability event is not a countdown collision, and EDCA cannot prevent it. AC Isolation is achieved by satisfying the following equation:

$$AIFSN(LP) = AIFSN(HP) + CW_{min}(HP) + 1 \tag{2.2}$$

In this dissertation, $AIFSN(HP) = 2$ and $CW_{min}(HP) = 3$, so equation 2.2 is satisfied with $AIFSN(LP) = 6$. $CW_{min}(LP)$ can be chosen large, e.g. 15, to reduce collisions between LP nodes. We return to Fig. 2.4 to see the effect of AC Isolation on PER, comparing with the non-isolated heterogeneous case. For HP (green vs red) there is a significant reduction in PER for all N, e.g. at 300 nodes the comparison is about 4% vs 30%. For LP (pink vs blue) the isolated and non-isolated ACs perform similarly. For N<150 the isolated AC has slightly less PER due to larger CW_{min}. For N>150 the isolated AC has slightly more PER due to larger AIFSN. In both ranges the difference is small. The dramatic improvement for HP thus comes at essentially no cost for LP, showing that AC Isolation is strongly preferable to using default AC3 and AC2.

Thus, if we only have two classes of traffic, we suggest ($CW_{min} = 3$, $AIFSN = 2$) for the HP class and ($CW_{min} = 15$, AIFSN=6) for LP. Generally it is not advisable to choose CWmin<3 in any class unless the proportion of traffic in the class is very small, and with our 10% assumption we will consistently set $CW_{min} = 3$ for the HP class. Similarly, we consistently set $AIFSN = 2$ for the HP class since it cannot be set smaller and there is no advantage to setting it larger.

If we have a third class of traffic (i.e. HP, LP, and medium priority (MP)) we can utilize a third isolated AC. We apply equation 2.2 twice. Starting with ($CW_{min} = 3$, $AIFSN = 2$) for HP, we must set $AIFSN = 6$ for the MP class. In order to keep AIFSN from becoming large for the LP class we set the MP class CW_{min} to either 3 or 7, depending on the proportion of traffic expected in the MP class. The more likely case is to set $CW_{min} = 7$, in which case from equation 2.2 we have $AIFSN = 14$ for the LP class. Table 2.3 summarizes our suggested parameters for three access categories based on the AC Isolation strategy.

Table 2.3: Suggested EDCA parameters for DSRC safety

Access Category	CW_{min}	AIFSN
AC1 (LP)	15	14
AC2 (MP)	7	6
AC3 (HP)	3	2

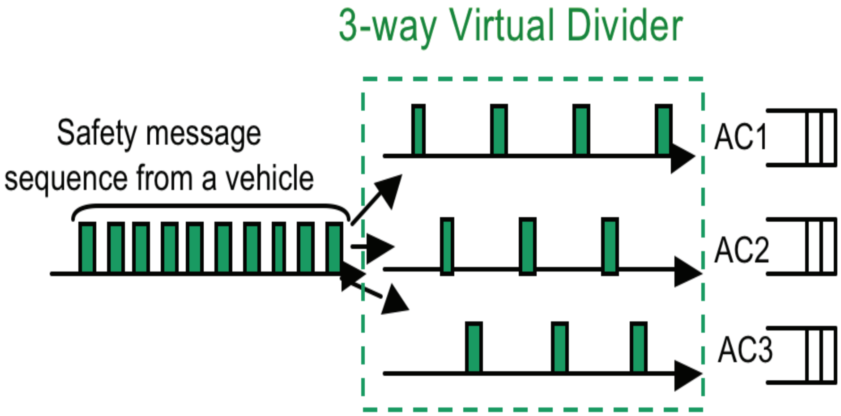

Figure 2.7: Virtual Division with one third distribution to each AC

2.4.4 Virtual Division

Observe that in all of the PER plots the curves are initially convex. Evidently, for some range of N, adding one vehicle creates enough potential new collision combinations that $PER(N+1) > PER(N) * N + 1/N$. This observation enables the our novel contribution: Virtual Division. It motivates us to try experiments of the following type: Classify the safety messages generated within each node into three classes using a ratio $N3 : N2 : N1$, assign those in the first class to AC3 (Table 2.3), those in the second class to AC2, those in the third class to AC1, and assess the aggregate PER across all classes.

We discovered choices for the ratio such that the aggregate PER resulting from this technique is indeed lower than if all messages use a homogeneous AC, e.g. ($CW_{min} = 15$, $AIFSN = 2$). In other words, even if all the messages are of equal value, an average PER advantage can be obtained by segregating them into different ACs, and specifically in this case isolated ACs. The assignment of the individual messages can be by random division or using a deterministic rotation. We call this technique Virtual Division because the assignment to ACs is not based on actual priority differences among the packets. It is illustrated in Fig. 2.7 for the equal division case, i.e. $N3 = N2 = N1 = 1/3$. We next examine the impact of Virtual Division on naturally homogeneous traffic, and then on naturally heterogeneous case.

Virtual Division For Naturally Homogeneous Traffic

Fig. 2.8 shows simulation results for the case illustrated in Fig. 2.7. The traffic is naturally homogeneous. We form three ACs using the parameters of Table 2.3. Each vehicle assigns one third of its packets to each AC, using a deterministic rotation after an initial random assignment. The figure compares the PER of Virtual Division (red) with that of a true homogeneous assignment (blue). The red curve is the aggregate PER experienced by a given vehicle, across all three ACs. We see that for low-to-moderate N, Virtual Division makes a modest improvement in the PER, but for high N the PER is improved significantly, from 72% to 58% (i.e. a 50% improvement in reception probability). Notice that virtual division will result in higher queuing delay for lower priority packets. If that delay reaches the packet generation period, 100 msec, we use a push-out queuing technique such that the old packet is discarded and the new packet is enqueued. Hence, the queuing delay for a successful transmission can not be higher than 100 msec, and this is not critical for safety applications. Discarding could be applied with a lower delay threshold if desired.

Virtual Division For Naturally Heterogeneous Traffic

If we have naturally heterogeneous traffic with, i.e. some with true high priority, we still can use the Virtual Division technique on the remainder. Fig. 2.9 shows simulation results for such an example. We again assume 10% of traffic in any message period is classified HP. A vehicle without an HP packet in a given period uses Virtual Division to assign its packet with equal probability to the MP or LP AC, e.g. it can alternate assignments from period to period. PER for HP traffic is equally low with and without division; the green and red curves are hard to

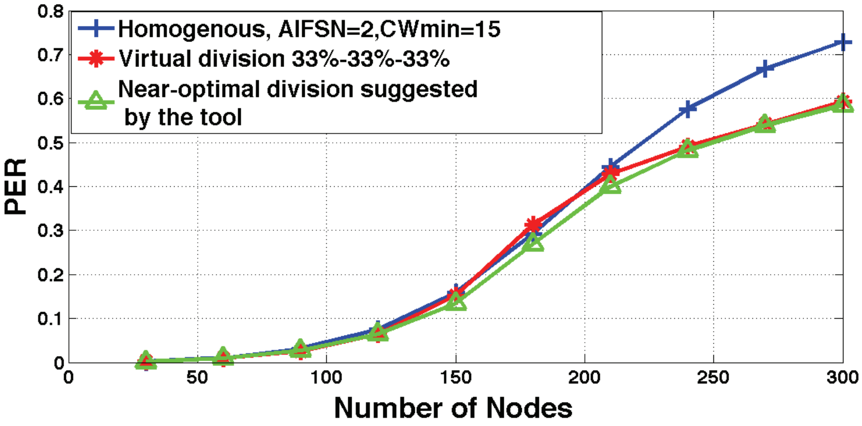

Figure 2.8: Virtual Division for naturally homogeneous traffic

distinguish. However, the PER for LP with division (pink) is significantly reduced compared to without division (blue) for higher values of N. For example, for N = 300 the PER has improved from 82% to 68% (the probability of reception has increased by more than 50%). Note that in this subsection we have used examples in which the Virtual Division is done on an equal basis among the subclasses, e.g. half to each of two, or one third to each of three. The question of optimal division ratio is discussed in the next subsection.

Optimal Division

Virtual Division can use any ratio N3:N2:N1. Equal division ratios were chosen in the prior subsection for simplicity. To investigate the optimal division we use a heuristic tool to efficiently search the (N3, N2, N1) space. This tool estimates PER for a given division quickly compared to a simulation. For example, an exhaustive search with 5% granularity requires checking 231 (N3, N2, N1) combinations, and takes a few minutes with the tool compared to many days via NS-2. The tool uses as input PER vs. N simulation data for each of the three homogeneous cases: AC3 alone (e.g. from Fig. 2.2), AC2 alone, and AC1 alone. For a given N3:N2:N1 it estimates the channel busy fraction (load) and PER contributed by each AC. Since the ACs are

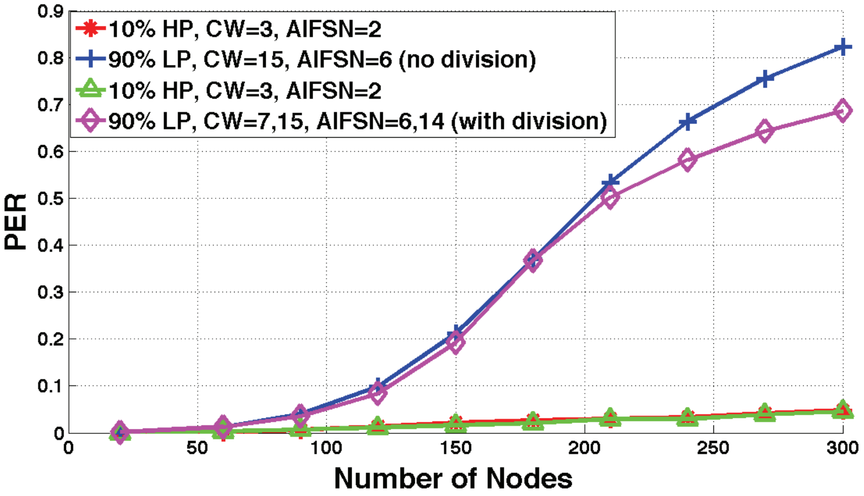

Figure 2.9: Virtual Division for naturally homogeneous traffic

isolated, aggregating the results is easy. A key assumption, validated by simulation, is that the load depends on the ratio of messages to message interval. Here are the specific steps for a given division and a given N:

1) From input data determine the channel load (L3) and PER (P3) when $N3 * N$ nodes use AC3 on a 100 msec channel.

2) Determine the load (L2) and PER (P2) if $N2 * N/(1 - L3)$ nodes use AC2 over 100 msec. This models $N2 * N$ nodes on a channel already reduced by higher priority AC3.

3) Finally, determine the PER (P1) if $N1 * N/(1 - L3 - L2)$ nodes use a 100 msec channel, modeling $N1 * N$ nodes on a channel reduced by higher priority AC3 and AC2. The aggregate PER for the division is then $P3 * N3 + P2 * N2 + P1 * N1$.

Fig. 2.10 shows the tool applied for $N = 300$, with ACs defined in Table 2.3, using AC3 input data from Fig. 2.2 and similar data for AC2 and AC1. The optimal division is identified as 0.4:0.2:0.4. The PER surface resembles a triangle, whose corners represent the three homogeneous cases, sagging in the middle. It appears to be uni-modal. Based on follow-up simulations, the division it identifies is near-optimal, with PER within 1% of the true minimum. The PER

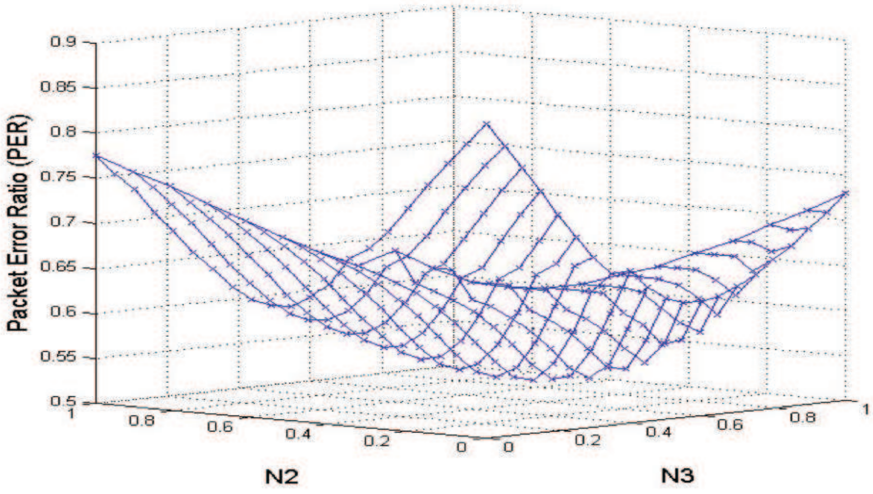

Figure 2.10: Virtual Division

corresponding to the near-optimal division found using this tool is also plotted in Fig. 2.8.

2.5 Realistic Highway Scenario

In this section, we will study the effect of CW_{min} and AIFSN on PER of safety traffic for a more realistic highway scenario. In this scenario, packet losses can be caused by signal attenuation, fading, hidden terminal problem as well as simultaneous countdown collisions.

For simulations in this section, we again use NS-2.34 simulator where we have the following scenario: We have a circular 3-kilometer highway where the circular highway is used to avoid the boundary effect [18]. We set the transmission power to 10 dBm which is equivalent to transmission range of 500-meters. Hence, we have many hidden nodes in this 3-kilometer road. Every vehicle is transmitting a 350-byte packet every 100msec with some random jitter. We use the Nakagami distribution, for radio propagation model, which has been shown more realistic for vehicular communication environment than pure Rayleigh distribution or two ray ground model [2,8]. The Nakagami distribution is already implemented in NS-2.34 and the mathematical description is given below [2].

$$f(x;\Omega,m) = \frac{2m^m x^{2m-1}}{\Gamma(m)\Omega^m} exp[\frac{-mx^2}{\Omega}],$$
$$x \geq 0, \Omega > 0 m \geq 1/2 \tag{2.3}$$

Where m is the shape parameter, and Ω controls the spread of the distribution. When $m = 1$, the Nakagami distribution becomes the Rayleigh distribution, which models a harsh scenario where no line of sight exists. When $m > 1$, it approximates the Rician distribution, which closely models line of sight scenario. For our simulations, we use the fading parameters as in [2] (these parameters were measured by collecting empirical data on a real highway), where we assume line of sight for closer ranges (< 50 meters) with $m = 3$, less severe fading for the distances up to 150 meters with $m = 1.5$, and Rayliegh fading for distance more than 150 meters with $m = 1$.

We calculate the PER for each transmitter-receiver pair over all the distances, and then calculate the average PER based on the distance. Notice that, the overall PER is a function of both the distance between the transmitter and the receiver and the node density. We define different distance bins where for each bin, we calculate the average PER between each pair of nodes falling within that bin.

2.5.1 Effect of CW_{min} on Homogeneous Traffic

In this subsection, we study the effect of CWmin on the PER in a realistic highway scenario when packets are transmitted periodically. We assume that the traffic is homogenous and all the packets are assigned to a single AC.

Fig. 2.11 shows the PER (y-axis) for $CW_{min} = 3, 15, 31$ for two scenarios: one where the highway is lightly loaded with 300 nodes over a 3km highway (100 node per kilometer) and one

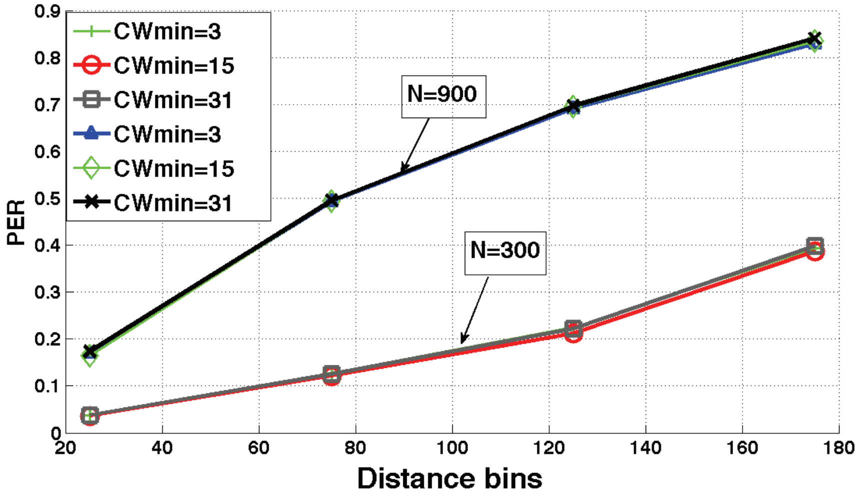

Figure 2.11: Effect of CWmin on PER in a highway scenario (300 and 900 nodes on a 3km highway)

where the highway is heavily loaded with 900 node over 3km highway (300 node per kilometer). The x-axis represents the distance bins [0-50, 50-100, 100-150, 150-200] meters. Each point represents the average PER for all the transmitter-receiver pairs that their distance falls within the specified bin. The AIFSN=2 for all the simulations. We can see that the variation in CW_{min} does not have any significant effect on the PER in this scenario whether the highway is lightly loaded or heavily loaded. The reason is that in highway scenario, we have both hidden node and countdown collisions, which are not independent from each other. In fact, reducing the simultaneous count down collisions may result in a higher hidden node collision rate. Reducing the count down collisions creates more successful transmissions. This decreases the channel idle time, which increases the chance of hidden node collision. Therefore, although larger CW_{min} at low and medium loads reduces the simultaneous count down collisions, but it increases the hidden node collisions because of decreasing the channel capacity and hence, the overall PER does not improve when we increase the size of CW_{min} in Fig. 10.

2.5.2 Effect of AIFSN on Homogenous Traffic

Similar to the simple topology scenario, in a realistic highway scenario, larger AIFSN results in longer waiting time before transmissions. This reduces the number of messages/second that channel can support and so the probability of collision increases. The performance plot is similar to Fig. 3 in subsection 4.2.

2.5.3 AC Isolation

In this subsection, we study the effect of AC isolation technique over the PER of HP and LP traffic in a realistic highway scenario. It should be noted that in this subsection like in [2] we study the highly loaded scenario (900 nodes which is equivalent to 1 car every 27 meters per lane in an 8 lane highway). In this scenario PER is high and our scheme improves PER by significant margin. We again assume that HP traffic is 10 percent of the whole traffic. Fig. 2.12 illustrates the PER of both HP and LP as well as the PER of homogenous case with only one AC. In this example, for heterogeneous case, AIFSN (HP)=2, CW_{min}(HP)=3 and AIFSN(LP)=6, CW_{min}(LP)=15. For homogeneous case, AIFSN=2 and CW_{min}=15. We can see that the PER of HP (blue) for all the distance bins has improved. As an example, the PER of [50-100] bin has improved from 50 percent to 20 percent compared to the homogeneous case (black), and this comes at the price of increasing the PER of LP (red) from 50 percent to 60 percent.

2.5.4 Virtual Division

As noted earlier, the virtual division examples studied in sections 4 and 5 employ an equal proportion strategy, e.g. 50%-50% or 33%-33%-33%. We demonstrated that this strategy is effective in improving communication performance in a variety of topologies, priorities, and densities. Many other strategies are possible. In general, one might expect the optimal proportions to be a function of vehicle density, message size and rate, fading environment, and AC parameters, among other factors.

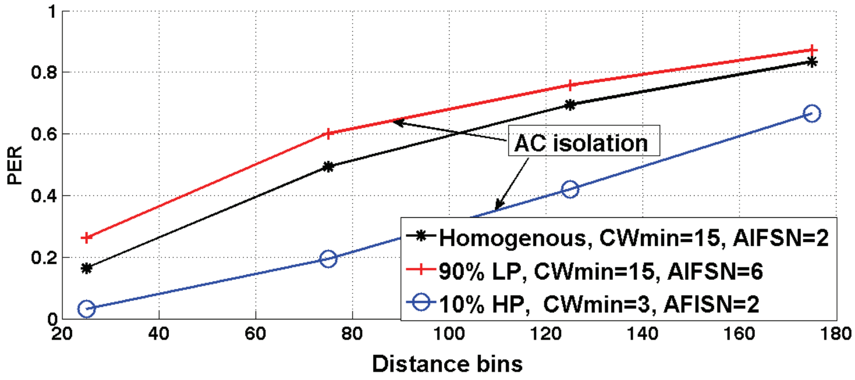

Figure 2.12: AC Isolation for highway (900 nodes on 3km road)

We have experimented with several unequal proportion scenarios, and have observed some that have better performance than equal proportion division. For example, when we simulated the case of homogeneous traffic in a realistic highway scenario with 900 nodes over 3 km, we found that a 20%-30%-50% division has lower aggregate PER than an equal 33%-33%-33% division. One can imagine constructing an adaptive proportion algorithm that reacts to changing load, topology, etc. The topic of optimizing the proportions of virtual division is identified as a subject for future research.

Virtual Division For Naturally Homogeneous Traffic on Highway

In this subsection, we study the effect that Virtual Division has on naturally homogeneous traffic in a realistic highway scenario. We form three ACs using parameters from Table 2.3. Each vehicle assigns one third of its packets to each AC, using a deterministic rotation after an initial random assignment. Fig. 2.13 compares the PER of Virtual Division with that of a true homogenous assignment for two different load scenarios: 900 and 300 nodes over 3 km highway. For the lightly loaded 300 node scenario virtual division (green) performs similarly to homogenous (black). For the highly loaded 900 node scenario the PER using virtual division (blue) has improved over all the distance bins. For example at distance bin [50- 100] the PER

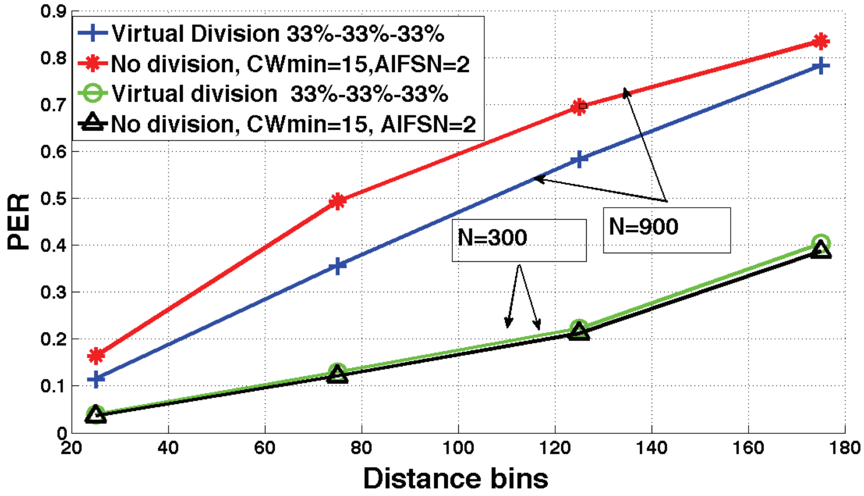

Figure 2.13: Virtual division for naturally homogeneous traffic (900 nodes on a 3km road)

has improved from 49% to 35% compared to the homogeneous approach (red).

Virtual Division For Naturally Heterogeneous Traffic on Highway

We study the effect of virtual division, on a naturally heterogeneous traffic in a realistic highway scenario in this subsection. The high priority traffic is 10 percent of the total traffic whose packets go into HP AC. The remaining 90 percent of the vehicles alternate assignment of their packets between the MP and LP ACs. Fig. 2.14 shows the simulation results, for this scenario. The green curve shows the PER over distance for LP when no virtual division is performed for LP nodes. The black curve represents the PER for LP when virtual division is performed for LP nodes. We can see that the PER of LP has improved over all the distance bins when virtual division is performed. For example, for the [50-100] bin, The PER has improved significantly from 60% to 36%. The blue curve shows the PER for HP when no virtual division is performed for LP nodes. The red curve represents the PER for HP when virtual division is performed for LP nodes. This time, we can see that virtual division for LP has slightly increased the PER of the HP. We can explain this phenomenon as follows: The significant reduction in simultaneous

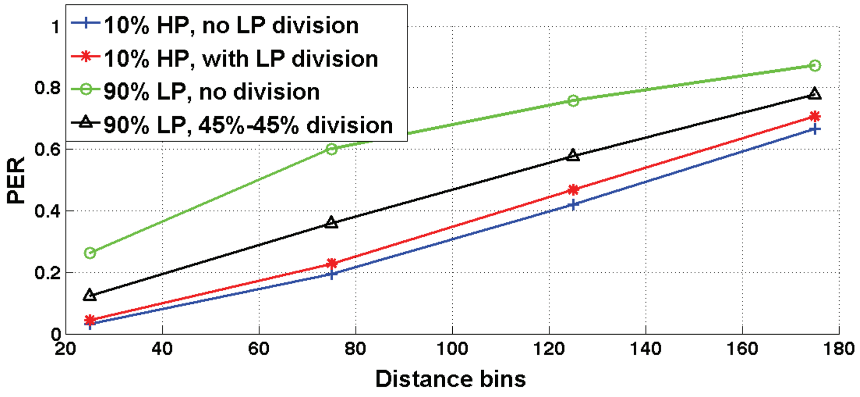

Figure 2.14: Virtual division for naturally heterogeneous traffic in highway scenario (900 nodes on 3km road)

count down collisions for LP using virtual division, decreases the channel idle time which will result in an increase in the hidden node collisions for HP, comparing to the case with no virtual division for LP. But this increase for the PER of the HP (e.g. from 20% to 23% percent in [50-100] bin) is fairly negligible when keeping in mind the significant decrease in the PER of the LP (60% to 36%). To bring that into perspective, if losses are independent, the probability that a given set of 4 packets will be lost in HP at the [100-150] bin is approximately 3% if virtual division is not performed for LP. That probability will be increased to approximately 4% if virtual division is performed for LP. Both these probabilities are fairly small. However, the probability of 4 losses for LP with no division is 33% for [100-150] bin. This probability is only 10% when virtual division is performed.

2.6 Advanced EDCA Techniques and Adaptive Message Rate Control

The examples used to demonstrate the EDCA concepts in this dissertation assume each node sends safety messages at a fixed rate. However, the AC isolation and virtual division techniques will also work in an adaptive message rate environment. Adaptive rate algorithms to control channel congestion are subject of current research [19,20]. Advanced EDCA techniques and message rate control are complementary technologies. There are potentially complex interactions between these technologies. For example, if the EDCA techniques produce lower PER (and thus higher channel load) for a given offered load, perhaps the target channel load of the adaptive message rate algorithm should higher than with default EDCA (e.g. 70% rather than 60%). Another interaction is the impact that adaptive rates have on virtual division proportions; for example if rates are reduced should the proportion going to the LP AC, which has the highest collision rate, be reduced? We leave these interesting questions for future research.

2.7 Conclusion

In this chapter we study the application of IEEE 802.11 EDCA to vehicular safety communication. We present two techniques that significantly improve communication performance: AC isolation and virtual division. Through comprehensive simulations these techniques are shown to reduce PER for both critical and routine safety messages. Simulations are presented for a simple topology to illustrate the basic principles, and also for realistic highway scenarios. Application of AC isolation requires use of non-default EDCA parameters. We recommend a specific alternative parameter set, which could be standardized for use on a safety channel (e.g. US Ch. 172 or European CCH). The techniques themselves are not only conformant with the EDCA standard, but area also easy to implement on existing hardware that can support per-packet priorities. Further research is suggested to learn more about selecting virtual division proportions, and to explore joint operation with congestion control algorithms.

Chapter 3

On Minimization of the Maximum Delay for Safety Communication in Vehicular Networks

This chapter introduces a new contention free MAC protocol for safety communication in vehicular networks. As discussed in previous chapter, IEEE 802.11p is a contention-based protocol, which can cause extensive and unbounded delays, especially in dense networks. Therefore, to bound the delay, in this chapter, we focus on a contention-free MAC protocol, and derive the theoretical lower bound on the maximum delay, in delivery of BSMs. This bound is a linear function of the maximum number of vehicles that interfere with one vehicle, in the network. The theoretical lower bound, can be achieved, if there is an entity that is aware of the position, and interference range of every vehicle. Since no such entity exists in vehicular networks, we propose a distributed cluster-based MAC protocol, which maximizes collision-free parallel transmissions of safety data. The proposed MAC protocol guarantees an upper bound on the delivery delay of BSMs which is only within 4 times from the optimal delay. Our protocol is distributed and dynamic, and easily adjusts to the changes in the network.We verify our results, and evaluate our protocol through numerical simulations.

3.1 Introduction

As described in Chapter 2 in detail, DSRC uses the IEEE 802.11p medium access control (MAC) protocol [7], which is based on carrier sense multiple access/collision avoidance (CSMA/CA) and is prone to packet collision. The packet collision probability increases with the offered load, which in our study is primarily a function of node density. Sakurai et al. [17] showed, that for the case of unlimited transmissions, the access delay in IEEE 802.11 DCF, has a heavy-tailed delay distribution. This means that, the delay can not be bounded, if the packets are not dropped after certain number of un-successfull tries. Thus, IEEE 802.11p MAC protocol is prone to long delays especially in dense networks, and can not guarantee an upper bound on the delivery delay of BSMs.

In this chapter, the goal is to bound the delivery delay of BSMs in vehicular networks, using a contention-free protocol. The key idea is to maximize parallel transmissions of safety messages without causing any collisions. We have three major contributions:

- We derive the theoretical lower bound on the maximum delay, using a TDMA-based protocol in vehicular networks.
- We propose a distributed cluster-based MAC protocol that is collision free and guarantees an upper bound on the delivery delay of routine safety messages, which is within 4 times from the optimal delay.
- Since the network is dynamic, and the structure of clusters changes, we propose a dynamic clustering and scheduling protocol geared for vehicular network environment that easily adjusts to the changes in the network with low overhead for clustering.

To obtain the theoretical lower bound, we introduce an interference graph which is a graph that represents the interference among all the vehicles in the network. We prove that, the lower bound on the maximum delay, is a linear function of the degree of the interference graph. This lower bound can be achieved, if the graph is fully known by an entity in the network. Since no such entity exists in vehicular networks, we propose a distributed cluster-based MAC protocol, which maximizes parallel transmission of data within clusters, without causing interference among clusters. Our dynamic clustering and scheduling protocol, takes advantage of the predictable moving pattern of the vehicles, to reduce the schedule adjustment overhead.

The contention-free protocol proposed in this chapter is TDMA-based, where we assume that all the vehicles are time synchronized using Global Positioning System (GPS) receivers. This is a realistic assumption since in future, all vehicles will have a GNSS (Global Navigation Satellite System) receiver on board, as it will be enforced by by new initiatives, such as Galileo in Europe, GPS modernization in USA, and QZSS in Japan. The coarse acquisition code (C/A) of GPS can be used for time synchronization in vehicular networks, as studied in [18], and can provide very precise synchronization (in order of nano seconds).

The rest of this chapter is organized as follows: Section 3.2 is an overview of the related work. In section 3.3, we derive the theoretical lower bound on the maximum delay. In section 3.4, we introduce our MAC protocol, and then obtain the analytical upper bound on the delay of routine safety messages using our protocol. In section 3.5, we explain how the proposed protocol adjusts to the changes of topology in the network. We evaluate the proposed protocol, through simulation in section 3.6, and section 4.7 concludes this chapter.

3.2 Related Work

Contention-free MAC protocols, in the context of vehicular networks and mobile wireless networks, have been studied in [19–26].

One of the proposed contention-free MAC protocols, is Space Division Multiple Access (SDMA) and its variations [19, 20]. The key idea is to divide the road into spaces and dedicate a time slot to each space for the vehicle located within that space. This approach have practical limitations, since it requires offline mapping of all the geographical locations to the time slots, as described in [27]. In addition, the overall network utilization is low, since many time slots are unused, in a sparse network.

Distributed TDMA, is another contention-free MAC protocol, and has been studied in variety of forms in [21–23, 26]. In STDMA [22], ADHOC MAC [21], and VeSOMAC [23], the key idea is that, each vehicle, reserves one unused time slot for itself, which is periodically repeated. During this time slot, the vehicle sends information. The slot time information are exchanged across 1-hop and 2-hop to avoid hidden terminal problem. The disadvantage of these protocols is that new vehicles, may compete for one time slot repeatedly. The proposed algorithm in [26] is similar to our approach in that both suggest a clustering approach where TDMA is chosen as the access protocol for the nodes within a cluster. However, Gerla et al. did not obtain a bound on delivery delay of messages using their protocol. Furthermore, they suggest lowest ID algorithm for clustering [28] where vehicles periodically broadcast their neighboring information. Our cluster adjustment protocol for vehicular networks has much less overhead by taking advantage of the fact that vehicles movements are predictable and restricted to the road. In addition, Gerla et al. suggest using CDMA to avoid interference among adjacent clusters, where separate codes are used for different clusters. Although CDMA is a feasible solution, but it requires a more complex hardware and also has the disadvantage of a lower data rate in comparison to TDMA.

The approach proposed by Su et al. [24], also uses a distributed TDMA in combination with a contention-based scheme. They use a clustering algorithm to place vehicles into different clusters, where each cluster has a vehicle as the cluster head. Within each cluster, TDMA is used, for collection and delivery of safety messages in the cluster, which are consolidated in the cluster head. Two clusters communicate through their cluster heads, using a contention-based protocol. Thus, two vehicles can not communicate directly with each other, if they are from different clusters, even though they are within each others' communication range. This will add

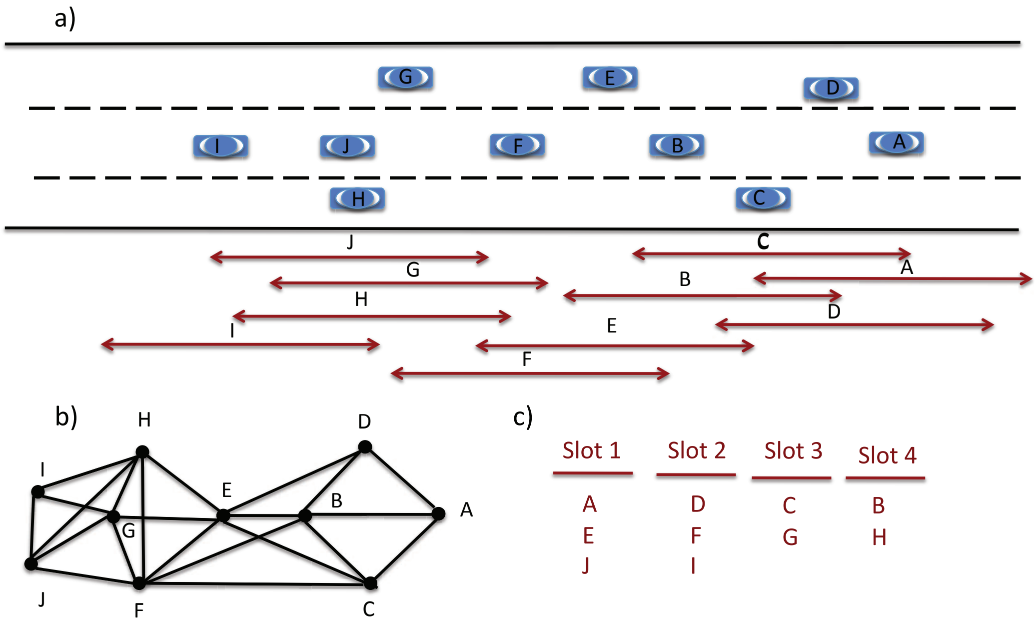

Figure 3.1: Optimal channel assignment using greedy coloring: a) Vehicles with their corresponding transmission ranges b) Interference graph representation of a) c) Optimal channel assignment

extra delay, for delivery of routine safety data.

3.3 Theoretical Analysis on Delivery Delay of Basic Safety Messages

Our objective is to design a MAC protocol, that can guarantee a maximum delay for delivery of safety messages. In this section, we derive a theoretical lower bound, on the maximum delay, in delivery of routine safety messages, using a TDMA-based MAC protocol. We assume that the vehicles are all time-synchronized, using GPS receivers. For simplicity, in this section, we consider a static network. We describe our algorithm, for the dynamic network, in section 3.5.

In addition, to obtain the theoretical lower bound on the maximum delay, we assume that there is an entity that is aware of the position, and interference range of all the vehicles. Since such an entity does not exist in vehicular networks, in the next section, we will explain how our proposed MAC protocol, uses local information to minimize the maximum delay in transmission of safety data.

To derive the lower bound, we maximize parallel transmissions that do not cause collision. For this purpose, we model the interference among nodes, with a graph that we call *interference graph*, and reformulate the problem as a graph coloring problem. Then, we prove that the interference graph is an interval graph in vehicular networks. Finding the optimal coloring, in interval graphs, is not NP-hard, and the optimal solution can be found efficiently [29]. Using this fact, we derive the lower bound on the maximum delay, in delivery of routine safety messages. This bound is a linear function of the degree of the interference graph.

Definition 1 *An* **interference graph** *is a graph in which, each vertex represents a vehicle, and each edge between two vertices, represents that their corresponding vehicles have overlapping interference areas.*

Using the above definition, we prove that:

Theorem 1 *For a TDMA-based collision-free scheduling protocol in a vehicular network, the maximum delay for transmission of safety messages, is lower bounded by* $(\Delta + 1)T$*, where* Δ *is the degree of the interference graph, corresponding to the network, and* T *is the duration of one time slot.*

Proof: We derive minimum number of time slots that a vehicle, with the maximum wait time, has to wait, before it can transmit its data. The transmissions should be collision-free. To avoid collisions, vehicles with overlapping interference areas, should not transmit data, in the same time slot. But vehicles with overlapping interference areas are adjacent in the interference graph. Thus, finding the minimum number of time slots, is equivalent to optimal coloring problem in the interference graph. The optimal coloring is finding the minimum number of colors, for coloring the vertices of the graph such that no two adjacent vertices share the same color. Now, the communication range of each vehicle is usually more than 150 meters, which is much larger than the width of the road. Therefore, we can represent the interference area of each vehicle with an interval. Thus, the interference graph is an interval graph. An interval graph is a graph that can be represented with a set of intervals assigned to the vertices so that vertices are adjacent, if and only if the corresponding intervals intersect [29]. In interval graphs, the optimal coloring problem is not NP-hard, and requires at most $\Delta + 1$ colors [29], where Δ is the degree of the interval graph. Therefore, the lower bound on the maximum delay, in transmitting routine safety data is $(\Delta + 1)T$. This completes the proof. ■

The following example, shows the interference graph and the optimal channel assignment for a given network.

Example 1 *Let's consider a snapshot of a 3-lane highway in Fig.3.1 a) where vehicles* $A, .., F$ *want to transmit their safety messages. Our objective, is to find the lower bound on the maximum delay, in delivery of safety messages. The interference areas of vehicles is shown with intervals. Vehicles with overlapping interference intervals, should not transmit their safety messages concurrently, because it will cause collision. For example, concurrent transmission of* A *and* C*, will collide in* B *and* D*. Fig.3.1 b) shows the interference graph corresponding to Fig.3.1 a).*

The optimal scheduling for the this problem can be found using the greedy coloring algorithm, described in [29]. The solution is given in Fig.3.1 c). All the vehicles $A, .., F$ *have to wait for only 3 time slots, before transmitting their data to avoid collision. Therefore, total delay, between two transmissions, is equal to 3T. This is smaller than the degree of the interference graph plus 1 which is 7 (the degree of node* E *is 6). Note that, the lower bound on the maximum delay, is equal to* $(\Delta + 1)T$*, only if the interference graph is fully connected, which is when all the vehicles are in the interference range of each other. It is clear that, in a fully connected* *network, in each time slot, only one vehicle should transmit data.*

Notice that, this optimal solution can be found, only if the interference graph is completely known, by an entity in the network. This means that, the entity should be aware of the position and interference range of all the vehicles on the road. No such entity exists in vehicular network, and the interference graph is only known locally. In the following section, we will propose a MAC protocol that uses local information to minimize the delay of safety data transmission.

3.4 MAC Protocol for Routine Safety Messages

In this section, we introduce our contention-free TDMA-based MAC layer protocol for routine safety messages. We assume that every vehicle is aware of its position and velocity, and that all the vehicles are time-synchronized using GPS devices. We also assume that interference range of vehicles is twice their transmission range. Our setting is a highway with two moving directions. The objective is to maximize the number of vehicles with non-overlapping interference ranges that can broadcast their data during the same time slot. We first explain our clustering algorithm which partitions the network into clusters. Then, we describe our scheduling protocol which maximizes parallel transmissions within clusters with no overlapping interference area.

We take advantage of the current existing physical layer of DSRC standard, for which Federal Communications Commission (FCC) has allocated 75 MHZ of 5.9 GHZ bandwidth. The 75 MHZ bandwidth of DSRC is divided into seven 10 MHZ channels. Among them, we use two of the 10 channels (CH172 and CH174) for each moving direction on the highway. Since our protocol is similar in both directions, we only discuss one direction.

We partition the network into clusters, within which the safety message transmissions are scheduled. Then, we schedule the transmissions within adjacent clusters such that parallel transmission that cause interference, are avoided. Each cluster is identified by a Cluster Head (CH). Any node in the transmission range of a CH belongs to the cluster and is called a Cluster

Member (CM). A vehicle that lies within the transmission range of two CHs, is called a gateway. Gateway nodes coordinate data transmissions among adjacent clusters, such that concurrent data transmissions within two different clusters, do not cause interference in any node. Since transmission range of vehicles is more than the width of a road, each cluster only has maximum of two adjacent cluster neighbors with the same moving direction: One in the front and one in the back. This makes the coordination easy since gateways only have to coordinate the schedule of two adjacent clusters.

Here, we explain how scheduling works within each cluster and its adjacent clusters, assuming no changes occur in the topology of the clusters. In the next section, we discuss how the structure of clusters changes, when topology of network changes due to vehicle movements.

In each cluster, the channel is assigned by the CH, and is announced to the CMs and gateways through the beacon messages. On each beacon message, CH piggybacks the ID of the next node within the cluster that should broadcast its status data. CH dedicates the designated time slot to the CMs and gateway nodes in a round robin fashion. Thus, the time that a vehicle has to wait in a cluster, for its turn to transmit its data, depends on the number of nodes in the cluster. Gateway nodes can either follow the transmission schedule of their front cluster or their back cluster. Gateway nodes make this decision based on the number of nodes in the two clusters. Every CH dedicates an empty time slot to the vehicles that are willing to join the corresponding cluster. These vehicles will send a request to join the cluster during this empty time slot.

We enforce a wait/transmit schedule within each cluster, such that transmission in one cluster, does not interfere with transmission in any other clusters. The key idea is that, only one cluster, among four consecutive clusters, should allow data transmission, during each time slot. This rule, first ensures that, two directly adjacent clusters, do not transmit data concurrently. Second, it ensures that non-adjacent clusters, with one or two clusters in between, do not transmit at the same time, either. The second rule avoids data collision at the middle clusters, when side clusters are transmitting. Note that, this is necessary, since interference range is usually

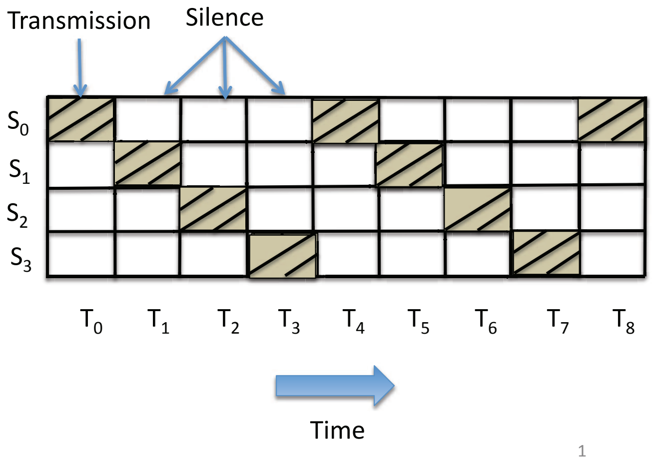

Figure 3.2: Transmit/Wait schedules

considered to be twice the transmission range and parallel transmissions can cause interference even when transmitting clusters are two clusters apart. For this purpose, we introduce three different schedules: $\{S_0, S_1, S_2, S_3\}$. Fig. 3.2 shows these four schedules. The schedules will be repeated periodically. We assign one of the four schedules to each cluster. Each schedule determines the time slots during which, a vehicle should transmit data in a cluster, and the time slots during which, the cluster should be silent. Each period, consists of one time slot for data transmission within the cluster, followed by three silent time slots, during which, vehicles in the adjacent clusters transmit data. Schedules differ at their time of transmission and silence. At each time slot T_i, if the result of $(i \, mod \, 4)$ is equal to schedule number, then a transmission should happen, within the cluster. For example, a cluster with schedule S_0 can transmit during time slots T_0, T_4, T_8,..., and a cluster with schedule S_1 can transmit during time slots T_1, T_5, T_9,.... Each cluster determines its schedule by obtaining the schedule of its front cluster, adding it by one, and then modulo 4 the result. This is the responsibility of gateway nodes to ensure that adjacent clusters, have different schedules, as we will explain later.

Example 2 *Fig. 3.3, shows vehicles $A, ..., N$ that are all moving in one direction. The road is*

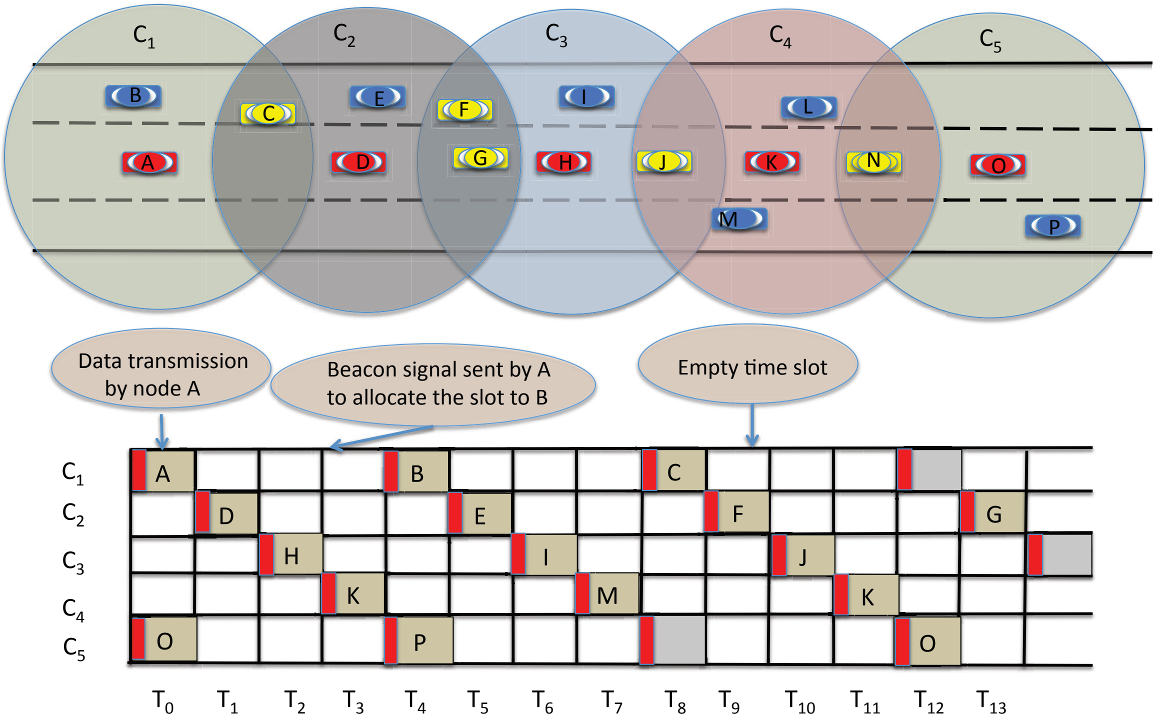

Figure 3.3: Scheduling protocol example: Moving direction is from left to right.

partitioned into five clusters $C_1, ..., C_5$*. In this example, red, blue and yellow vehicles represent CH, CM and gateways, respectively. Clusters with identical schedules, have identical colors (Clusters* C_1 *and* C_5 *are using* S_0*). Clusters* C_2*,* C_3 *and* C_4 *are using* S_1*,* S_2 *and* S_3 *schedules, respectively. When a node in cluster* C_2*, such as* D*, sends data, based on* S_2 *schedule, adjacent clusters,* C_1*,* C_3 *and* C_4 *remain silent. Similarly, clusters* C_1*,* C_2*,* C_4 *and* C_5 *remain silent, when a vehicle in* C_3 *is transmitting data. An empty time slot, shown in gray, is used for vehicles that are willing to join the cluster. During this empty time slot, CH transmits a beacon, with a null destination.*

When a vehicle is asked to send its safety data, it broadcasts all the information regarding its status. This includes time stamp, location, speed, acceleration, heading, and CH ID. During the beacon transmission, CH sends its own ID, and the ID of the next transmitter within the cluster. CH also sends a number that determines the ”back-up rank” of the transmitter, which is used, for the dynamic clustering, as we will explain in section 3.5. Note that the overhead of clustering using this algorithm is low. CMs do not broadcast ay clustering information. The only clustering information that CH broadcasts is the ID of the next transmitter within the cluster, and the back-up rank.

3.4.1 Delay Analysis

In this subsection, we first obtain the lower bound on the maximum delay in delivery of safety messages, using our MAC protocol. Then, we show our MAC protocol, satisfies the delay requirements of safety message transmission in a vehicular network.

The maximum waiting time, for a vehicle in the cluster, to transmit its status data, is equal to the total time for transmission of all other vehicles in the cluster, plus the total silent time, required after each transmission. Let N_C be the number of nodes in the cluster. After each transmission in the cluster, the cluster remains silent for three time slots, to avoid interference with adjacent clusters. Thus, the maximum wait time for a vehicle, is $(N_C - 1 + 3.N_C - 3)T$, plus the time assigned to the empty time slot, and the silent slots after that which is $3T$. Thus, the total wait time is:

$$min(max(delay)) = (N_C + 3.N_C)T <= 4(\Delta + 1)T, \tag{3.1}$$

where Δ is the degree of the interference graph, and T is the duration of one time slot.

As an example, if the update interval of safety messages is 100 ms, the size of safety packets is 350 bytes, and the channel rate is 6 Mb/s, the duration of a time slot is about 0.467 ms. Thus, the maximum number of vehicles, in a cluster, to satisfy delay requirements of less than 100 ms is 54.

3.5 Dynamic Clustering and Scheduling

So far, we explained how scheduling works within each cluster, and its adjacent clusters, assuming structure of the clusters does not change. In this section, we explain how the schedules are adjusted, when topology of the network changes.

Any change in the network topology, falls under one of the following three categories.

- A vehicle joins the road.
- A vehicle leaves the road.
- Vehicles' relative positions with respect to each other change on the road.

In the following subsections, we explain how clusters and schedules will be adjusted dynamically, for the above changes.

3.5.1 Joining the Road

When a vehicle joins a Road, it will join an existing cluster, if there is already one, or it will form a new cluster, if there is none. To discover if a cluster already exists, newly joined vehicle, which we call V, will listen to the channel for a predefined period. During this time, one of the following scenarios may happen:

- V receives the beacon transmission of one CH.
- V receives the beacon transmission of two CH(s).
- V hears no beacon transmission.

If V only receives a beacon of one CH, it requests to join that cluster, during the empty time slot of the cluster. The empty time slot, is identified by a beacon message with a ”null” destination. V waits for a random time, which is shorter than the duration of the time slot, and broadcasts a Request To Join (RTJ) message. This message carries ID of V. Then, CH dedicates a time slot to V, to send its safety data. The only clustering overhead in this case is the RTJ message.

Note that using this algorithm, a CH stays a CH unless it exits the road, or fall into transmission range of another CH, as we will explain in the following subsections.

If V receives beacon transmission of two CHs, it will become a gateway. A gateway node should make sure that the schedule of the adjacent clusters are in coordination. Thus, V first checks if the schedule of the back cluster, is equal to schedule of the front cluster plus 1 modulo 4. If it is, V just simply sends a RTJ message during the first empty time slot to come, which in average leads to joining the cluster with the least number of members. If not, it keeps sending a Change of Schedule (CS) message for the back CH, during all the upcoming unused time slots, until it changes its schedule. CS message, is a short message, that includes the suggested schedule for the back cluster. As soon as CH of the back cluster, receives this message, it will change its schedule. Then, V can join either cluster. Note that the CH of the back cluster, should also ensure that the schedule of its own back cluster is equal to the its new schedule plus 1 modulo 4. Thus, it selects one of its own back gateways to inform its own back cluster,

Algorithm 1 Vehicle v_i joins the road. T_w is wait time which is equal to the duration of 4 time slots. RTJ is Request to Join message.

```
$v_i$ waits for duration of $T_w$
if during $T_w$, $v_i$ hears a beacon transmission by one CH then
    Sends a RTJ message during empty time slot
    Upon receipt of schedule by CH, $v_i$ transmits status data
end if
if during $T_w$, $v_i$ hears a beacon transmission by two CHs then
    Becomes a gateway                                          ▷ algorithm 2
end if
if during $T_w$, $v_i$ hears no beacon transmission after $T\_w$ then
    Becomes a CH                                               ▷ algorithm 3
end if
```

Algorithm 2 Vehicle v_i becomes a gateway. CS is the Change of Schedule message. BC is Back Cluster and FC is Front Cluster

```
if Schedule(BC) $\neq$ Schedule (FC)+1 module 4 then
    Send CS message to the back CH repeatedly
end if
if Schedule(BC) is changed then
    Send a RTJ message during the first upcoming empty time slot
    Upon receipt of schedule by CH, $v_i$ transmits status data
end if
```

about the change of schedule. The clustering overhead in this scenarios includes RTJ and CS messages.

Algorithm 1 summarizes our protocol.

If V hears nothing, after a predetermined time period, it simply becomes a CH, picks one of the four schedules randomly, and starts sending beacon messages.

3.5.2 Leaving the Road

When a vehicle that is a CM or a gateway, leaves the Road, only the time slot that was assigned to it, will be released. Departure of a CH, is not as trivial. Each CH, ranks all the nodes of its cluster, except for the back gateway nodes, as its back-ups. The ranking is based on their closeness to the CH, such that the closer a node is to the CH, the higher its rank is. CH informs

Algorithm 3 Vehicle v_i becomes a CH

```
Picks a random schedule and send beacons based on the schedule
Assigns one empty time slot for vehicles willing to join
if receives RTJ messages during empty time slot then
    Assign a time slot for each joining vehicle
    Assign a backup rank for each vehicle based on their closeness to CH
    Announce schedule and back up rank through beacon transmission
end if
```

Algorithm 4 Vehicle v_i looses a CH

```
if $v_i$ hears no beacon transmission from a CH after 4 time slots then
    while $myrank \geq 1$ do
        $myrank \leftarrow myrank - 1$
        if $v_i$ hears beacon transmission of a CH then
            joins the cluster
        end if
    end while                                          ▷ $myrank = 0$
    $v_i$ Becomes a CH                                 ▷ algorithm 3
end if
```

the CMs of their back-up ranking, through beacon transmissions. When CH leaves, beacon transmissions will be stopped. At this point, the first back-up node in the ranking, becomes the new CH, and resumes beacon transmission according to the previous schedule. Other back-up nodes, become CH, if they do not hear beacon transmission of the their higher ranked nodes.

After a CH leaves, nothing changes for the gateway nodes that were following the status of the other CH. However, if a gateway node was following the schedule of the CH that left, it will apply for membership of the other cluster, during the empty time slot, if it does not hear beacon transmission of a back-up node. But, if all the gateway nodes, transmit their membership

Algorithm 5 v_i is a CH in transmission range of another CH

```
if $v_i$ hears beacon transmission from another CH  then
    if $v_i$ less members than the other CH then
        $v_i$ stops sending beacon message
        $v_i$ sends a RTJ message to the other CH
    end if
end if
```

request during the same empty time slot, the requests may collide. To avoid collision, gateway nodes, wait for a time depending on their back-up ranking, and then, send their membership request in the first upcoming empty slot.

3.5.3 Change of relative positions on the road

The membership status of vehicles may change as follows:

- Two CHs fall into transmission range of each other.
- A gateway stops hearing beacon transmissions of one of the CHs.
- CM stops hearing beacon transmissions of its CH.
- CM starts hearing beacon transmissions of a new CH in addition to its own CH.

When two CHs fall into the transmission range of each other, both start hearing beacons. The CH with less number of CMs stops transmitting beacon messages, and becomes a member of the back cluster. CHs know the number of the members of each other based on the back up ranks that they hear from the other CH. In this case, some of the CMs of the other cluster, may loose their CH. At this point, one of the back-up nodes, in the smaller cluster, should switch to the CH status, as explained in the previous section.

When a CM stops hearing beacon transmission of its CH, it waits for a time based on its back-up status in the cluster. As described in the previous subsection, if by that time, it does not hear any beacons, it switches to the CH status.

If a CM starts hearing beacon transmission of a new CH, it switches to the gateway status, as described in subsection 3.5.1.

3.6 Simulation

In this section, we quantitatively evaluate the performance of our MAC layer protocol, through numerical simulations. For our simulations, we use MATLAB environment. We consider a one-directional highway of 2000 meters. Each vehicle, joins the highway from a random position, and stays on the highway, until it reaches to the end of the road, at which point it exits, and rejoins the highway, at the beginning of the road. Average speed of the vehicles is 30 m/s (108 km/h). We simulate vehicles movement, using the IDF motion model [30, 31], where the acceleration of a particular vehicle at a given time is calculated by evaluating the desired gap to the vehicle in front.

The data transmission rate, is assumed to be 6 Mb/s, and transmission range is 150 meters, for every vehicle. We consider the total size of a safety packets plus the physical layer header to be 125 bytes. Thus, the duration of each time slot is 0.167 ms. Every plot shows the results averaged over 10 runs.

In Fig.3.4, we show the impact of maximum cluster size on the maximum delay. The green curve, is the optimal scheduling delay, computed using greedy coloring [29], assuming there is an entity that knows the position of all the vehicles. The blue curve is the maximum transmission delay, using our MAC protocol. The red curve is the theoretical upper bound on the delay, computed based on the maximum cluster size of our protocol (equation (3.1)). For this simulation, the network is static. The simulation results show that the delay, using our MAC protocol, is always less than $4(\Delta + 1)$, where $\Delta + 1$ is the maximum cluster size.

In Fig.3.5, we compare our proposed MAC protocol with IEEE 802.11p in terms of IPG for a dense scenario of 900 vehicles in a 3km highway. The x-axis represents the distance bins [0-50, 50-100, 100-150, 150-200] meters. Each point represents the average IPG in seconds for all the transmitter-receiver pairs that their distance falls within the specified bin. The blue curve shows the average IPG when the MAC protocol is IEEE 802.11p. The simulation parameters are similar to chapter 2. The red curve shows the average IPG if the proposed cluster based

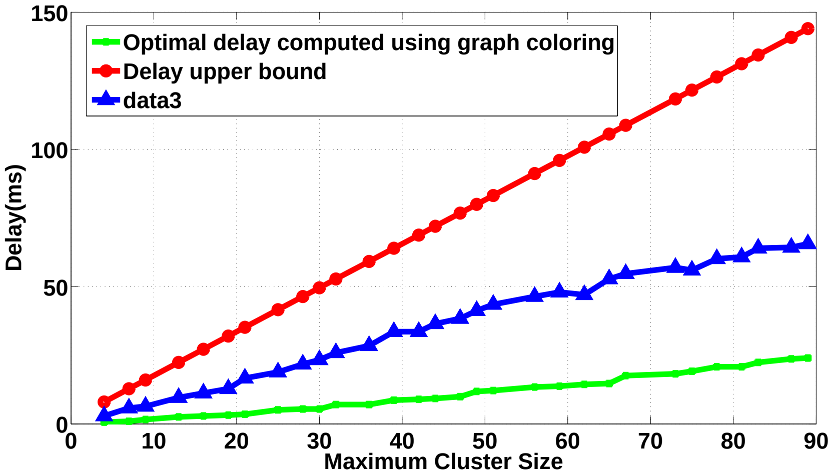

Figure 3.4: Maximum Delay

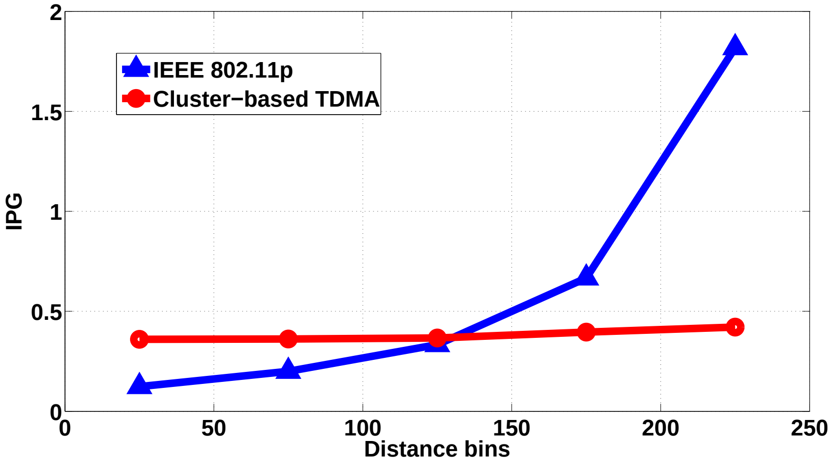

Figure 3.5: IPG Delay of IEEE 802.11p vs Cluster-based TDMA

MAC protocol is used where there is 300 nodes per cluster. The simulation results shows for the distances less than 125 meters, the average IPG is smaller if IEEE 802.11p MAC protocol is used. However, for larger distances, our proposed MAC protocol performs significantly better than IEEE 802.11p in terms of average IPG. As expected, our proposed MAC protocol bounds the delay for dense scenarios.

3.7 Conclusion

We focused on a contention-free MAC protocol design for vehicular networks. We found the theoretical lower bound on the maximum delay in delivery of routine safety messages. This bound is a linear function of the degree of the interference graph, and can be achieved if the interference graph is fully known by an entity in the network. Since no such entity exists in a vehicular network, we proposed a MAC protocol which partitions the network into clusters, and applies a TDMA-based scheduling within each cluster. The suggested protocol guarantees an upper bound on delivery delay of routine safety messages, and is within 3 times from the optimal delay. We verified our results and evaluated our protocol through numerical simulation. For future works, we want to extend our algorithm for urban environments.

Chapter 4

Efficient Data Collection Algorithm for Traffic Probing in Vehicular Networks

Vehicle-based sensors can be used for trafc monitoring. These sensors capture information such as location and speed of the vehicles which can be collected at the road side units. This data can be used for statistical analysis on the traffic conditions of the roadways. In this chapter, we study the problem of collecting the traffic information from all the vehicles for traffic probing. DSRC technology and the safety data of vehicles can be used for traffic monitoring. But collecting all the raw safety data consumes considerable amount of bandwidth while it is a sparse resource especially in dense vehicular networks. In this chapter, we propose a Distributed and Energy efficient algorithm for Collection of Raw data in vehicular networks called DECOR. This algorithm can also be used for collection of data in sensor network where energy is the main concern. DECOR exploits spatial correlation among the status data of the close by sensors to reduce the bandwidth consumption and communication energy. In our approach, at each neighborhood, one node shares its raw data as a reference with the rest of the nodes without any suppression or compression. Other nodes use this reference data to compress their data by representing them in the forms of mutual differences. In a highly correlated network, transmission of reference data consumes significantly more energy and bandwidth than transmission of compressed data. Thus, we first attempt to minimize the number of reference transmissions. Then, we try to minimize the size of mutual differences. We derive analytical lower bounds for both these phases and based on our theoretical results, we propose a two step distributed data collection algorithm which reduces the number of transmitted bits significantly compared to existing methods. In addition, we modify our algorithm for lossy communication channels and we evaluate its performance through simulation.

4.1 Introduction

Travel times and speeds on the roadways have always been of interest to researchers and travelers. Monitoring traveling information is possible through deployment of inductive loops, cameras and other sensors on the roads. But these solutions are very costly. A more cost effective solution, is to use vehicles as traffic probe sensors and develop a framework to use their information for further data analysis [32–34]. In many US cities, taxi companies often deploy GPS-based sensors on their taxies for the effective dispatch of vehicles. These mobile sensors can constitute a vehicle-based mobile sensor network, where the data sensed can be collected through either DSRC technology or cellular networks.

In this chapter, we focus on the problem of collection of the status data of the vehicles in the network in an access point. The solution to this problem is also applicable to any scenario where data of wireless nodes need to be collected at one access point and a special resource is limited which requires to conserve the number of bits that are transmitted in the air. This specifically applies to the data collection problem in sensor networks where the limited resource is energy since sensors are battery operated and will be left for long duration to operate on their own. In this chapter, we use the term *nodes* and *sensors* interchangeably. Similarly *base station* and *access point* are both the units where data will be collected and they can be used interchangeably as well.

If a user/application can express its interest to the data as a query, the query will be sent into the network. Based on the query, nodes perform aggregation on their readings to reduce the amount of data that is transmitted to the access points [35–38]. For example, if in a network, a user is only interested in the minimum or maximum speed of the vehicles, all the redundant data can be easily dropped on their way to the access point and only one value is reported. However, in many situations, raw data from all the nodes is required.

Lossless collection of data in wireless networks comes at the price of power and bandwidth consumption. To reduce this cost, different techniques have been suggested to exploit the spatial correlation among the data of nodes for data collection [36, 39–42, 42, 43]. One of these techniques is distributed source coding [40] which avoids transmitting redundant information if nodes know the correlation of their data. However, usually this knowledge does not exist a priori. Therefore, most of the data collection schemes require an expensive initialization phase during which raw data from all the nodes is collected [39–42]. During this phase, nodes transmit their raw data toward the base station where the correlation structure is determined. Different approaches have been proposed in [36, 42, 43] to remove the redundant data at each relaying node on the forwarding path to the base station. However, in these techniques, each node should still perform one full transmission where a full transmission is a transmission of one node's raw data without any suppression. Full transmissions are the inevitable task during lossless data collection. In a highly correlated network, full transmissions consume considerably more energy and bandwidth than transmissions of compressed data since uncompressed data is significantly larger than the compressed data.

For the above reason, in order to minimize the total number of transmitted bits and to reduce the overall energy and bandwidth consumption, we suggest to combine following two strategies: 1) Minimize the number of full transmissions 2) Remove redundant data on the forwarding path to the access point or base station. For the first strategy, we do a theoretical study on the minimum number of full transmissions required for lossless collection of data at the base station. For the second strategy, We find a lower bound on the minimum number of bits that a node, which is responsible for forwarding the data of its neighborhood, should transmit for lossless data collection.

To minimize the total number of full transmissions, we represent the network with a graph and introduce what we call *Minimum Semi-Connected Dominating Set* (MSCDS). The elements of this set are a dominating set of the graph and they are the only nodes that require to do full transmissions for lossless data collection at the base station. These nodes perform only one full transmission during a data collection. We identify these elements as reference nodes. Other nodes are only required to transmit the difference between their data and that of the adjacent reference nodes. Since finding a minimum dominating set is NP-hard, it is very likely that finding MSCDS is also NP-hard. Thus, we introduce a practical distributed algorithm which approximates this set. We show that the cardinality of the approximated set is within a constant factor of that of MSCDS.

We then find the minimum number of bits required for representing a set of data of nodes in the forms of mutual differences. This lower bound is equal to the sum of edge-wights of the minimum spanning tree of a logical weighted graph whose edge-weights correspond to the size of mutual differences between each pair of nodes' data. This is the minimum amount of data that each node, which forwards the data of a neighborhood toward the base station, should transmit.

We propose a practical algorithm for collecting the data at the base station or access point. This algorithm is designed to minimize the total number of bits to be transmitted by the nodes. Then, we discuss how it can deal with lossy communication channels. We evaluated the algorithm through numerical simulations with other existing approaches that exploit spatial correlation and show significant improvement over them. Our numerical simulations also suggest that our algorithm can capture the redundancy when the number of sensors increases and thus avoids transmitting the redundant data in highly dense networks.

Our algorithms are distributed and run locally at each node and have $O(n)$ time and message complexity where n is the number of nodes.

The chapter is organized as following: Section 4.2 describes the system model and problem formulation. In section 4.3, we describe how we can minimize the number of full transmissions and number of bits transmitted by each forwarding node. In section 4.4, we explain our data collection algorithm. In sections 4.5 and 4.6, we evaluated our algorithm through analysis and numerical simulations, respectively. Conclusion and future work are in section 4.7.

4.2 System Model and Problem Statement

Consider a data collection problem in a wireless network where n nodes $s_i, i \in \{1, \ldots, n\}$ report their observations to a base station. The objective is to collect the data of all the nodes at the base station with the minimum number of transmitted bits to preserve the energy and make efficient use of bandwidth. Assume that the nodes are densely deployed and so their data is highly correlated and redundant. We avoid transmitting the redundant data by computing the difference between data of adjacent nodes and then suppress the redundant data. We assume that every node is capable of running an algorithm to compute the difference of its data with respect to a reference provided by another node. We call this difference *delta*.

Base station reconstructs the data of all the nodes if all the deltas and required references are received. We model the communication network among the nodes and the base station with a unit-disk graph C where two nodes in the network are adjacent in the graph C if the Euclidean distance between them is less than the communication range r. We assume that C is connected otherwise some nodes can not share their data with the base station. Nodes use broadcast communication. As a result, when a node sends out a message, it is received by every node or base station within its communication range. We also assume that each node is capable of computing the difference between its data and that of another node. We use this defined model to address the problem of minimizing the number of transmitted bits using two strategies.

First strategy is to to minimize the number of full transmissions through which reference data are being transmitted. Since the network is not fully connected, transmission of each reference data is only received by the nodes that lie in the transmitter's neighborhood. This implies that at least a *dominating set* of nodes in graph C should send out their reference data so that the rest can compute their deltas. Set D is a dominating set of graph C, if every vertex in the graph that is not in D, has a neighbor in D. But this reference data should also be transmitted over a multi-hop path to the base station such that further full transmissions are avoided. Hence, for minimizing full transmissions, it is not sufficient to just identify a minimum dominating set of graph C as reference nodes.

Second strategy is to minimize number of transmitted bits at each neighborhood where one node is responsible to forward the data of all its neighborhood toward the base station. This node has to represent a set of deltas optimally with the minimum number of bits.

4.3 Theoretical background

In this section, we first present a theoretical study on the minimum number of full transmissions required for lossless collection of data at the base station. Then, we find a lower bound on the minimum number of bits that a node, which is responsible for forwarding the data of its neighborhood, should transmit during lossless data collection.

4.3.1 Minimum Number of Full Transmissions

In this section, we obtain the minimum number of transmissions of reference data that is required for data collection at the base station. Each node should receive at least one reference data to compute its delta. To ensure that, at least a dominating set of nodes in the graph C should transmit their reference data. However, as we demonstrate in Example 3, this dominating set should also have an extra property to ensure that all references and deltas can be collected at the base station without any reference data being retransmitted. We call this specific dominating set a Minimum Semi-Connected Dominating Set (MSCDS). We first introduce a MSCDS and then prove that the cardinality of this set is equal to the minimum number of full transmissions required for full data gathering at the base station.

We first introduce a *bridge graph* which will be used for defining a MSCDS.

Definition 2 *For a graph* $C = (V, E)$*, its* **bridge graph** *on the set of vertices* $W \subseteq V$*, denoted by* C_W*, is a new graph such that two vertices in the set* W *are adjacent in* C_W *if and only if*

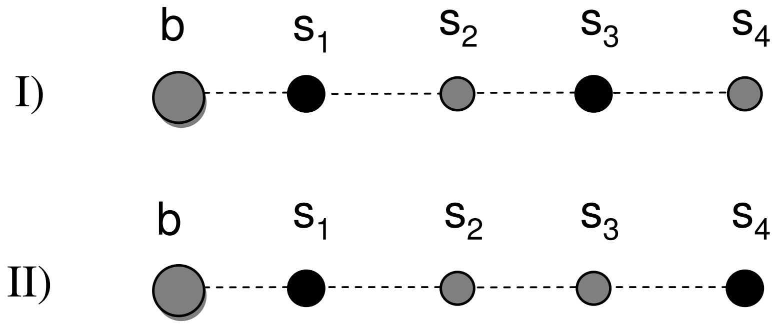

Figure 4.1: Wireless Network in Example 3

they are within the 2-hop distance of each other in graph C.

Definition 3 *A dominating set D of graph C, is a* **Semi-Connected Dominating Set** *(SCDS) of graph C, if its bridge graph C_D, is connected. Among all the SCDSs, we refer to a SCDS of minimum size as a Minimum SCDS (MSCDS).*

Here, through a simple example, we explain why nodes should be SCDS rather than just a dominating set.

Example 3 *Consider a network with the communication graph C presented in Fig. 4.1 where nodes $s_i, i \in \{1, \ldots, 4\}$ should transmit their data $f_i, i \in \{1, \ldots, 4\}$ to the base station b. In this example, set $\{s_1, s_4\}$ is a dominating set but not a SCDS because s_1 and s_4 are not within the 2-hop neighborhood of each other and so their corresponding bridge graph is not connected. However, set $\{s_1, s_3\}$ is a SCDS of the communication graph since s_1 and s_3 make a dominating set and are within the two-hop distance of each other and so their corresponding bridge graph (a graph consisting of s_1 and s_3 and an edge between them) is connected.*

Assume that nodes s_1 *and* s_4 *(black nodes in Fig. 4.1.I) send out their reference data delivering* f_1 *and* f_4 *respectively. In this case, at least three reference data transmissions are required for gathering all the data at* b*. That's because* s_3 *should transmit either* f_4 *or* f_3 *to* s_2*.*

Now, assume that nodes s_1 *and* s_3 *(black nodes in Fig. 4.1.II) send out their reference data delivering* f_1 *and* f_3*. Node* s_4 *computes* $\Delta_{4,3}$ *and transmits it to* s_3 *where* $\Delta_{i,j}$ *is the difference of* f_j *with respect to* f_i*. Nodes* s_3 *transmits* $\Delta_{4,3}$ *to* s_2*. Now* s_2 *has* $\{f_1, \ldots, f_4\}$ *as it already has received* f_1 *and* f_3 *and can reconstruct* f_4 *using* $\Delta_{4,3}$*. Therefore,* s_2 *can compute* $\Delta_{2,1}$*,* $\Delta_{3,1}$*, and* $\Delta_{4,1}$ *and transmit them to* s_1*. To gather all the data in* b*, it is enough that* s_1 *retransmits these deltas to* b*. Since base station* b *already has received* f_1*, it can reconstruct* $\{f_2, \ldots, f_4\}$ *using the received deltas. Therefore, in this case, the two original full transmissions by* s_1 *and* s_3 *are enough for gathering all the data at* b*.*

Inspired by Example **3**, we generalize this result in Theorem **2**.

Theorem 2 *The minimum number of transmissions of reference data necessary for data gathering at node* $b \in C$*, is equal to the size of MSCDS of* C*.*

Proof: We first show that if D is a SCDS, the transmissions of reference data by the nodes in D, is sufficient for data gathering in b. Then, we prove that the number of the transmissions of reference data which are necessary for data gathering, is not less than $|MSCDS|$ where $|MSCDS|$ is the size of MSCDS.

Consider a SCDS set D in graph C. The set of nodes in D, form a connected bridge graph C_D. Since D is a dominating set, b or at least a neighbor of b is in C_D. That means for any node $s_i \in D$, there is another node $s_j \in D$ in the 2-hop neighborhood of s_i such that s_j is on a path from s_i to b. Note that s_j is closer than s_i to b on this path. Thus, when s_i and s_j send out their reference data, there is at least one node s_k that receives both. s_k transfers s_i's data to s_j by only transmitting $\Delta_{i,j}$. With the similar strategy, $s_j \in D$ transmits its data (and $\Delta_{i,j}$) all the way to b. Now, for any node $s_r \notin D$ there is at least one node $s_l \in D$ in its neighborhood.

Node s_r can transmit $\Delta_{r,l}$ to any node on its neighborhood on a path to b along which $\Delta_{r,l}$ can be simply forwarded. Thus, no more reference data transmission for s_r'data is required. Thus, one reference transmission by the nodes in D is enough for full data reconstruction in b.

Now, we prove that the number of transmissions of reference data necessary for data gathering, is not less than $|MSCDS|$. Consider a data gathering which requires t transmissions of reference messages to reconstruct all the data at b. We show that $|MSCDS| \leq t$. We refer to the set of nodes that transmitted the t references as set R. Note that $|R| \leq t$ (a node may have done more than one full transmission). Set R should be a dominating set, otherwise some of the nodes do not receive any references. Now, we have to show that set R is a SCDS. If R is not a SCDS, the set of nodes in R form a non-connected bridge graph C_R. That means there should be at least one node $s_i \in R$ from which there is no path to b or any of b's neighbors in C_R. Thus, considering any path through which node s_i transmits its data to b, there should be at least two adjacent nodes $s_r \notin R$ and $s_k \notin R$ that have no common neighbor in R. Let's consider one path from s_i to b. Without loss of generality let's assume that s_r is closer to s_i. s_k and s_r have no common neighbor in R whose reference data transmission can be heard by both. Thus, to transmit s_i's data, node s_r has no other option other than one full transmission of one reference data; otherwise data of s_i can not be reconstructed at b. But s_r does not belong to R. This is in contradiction with our original assumption that t reference data transmissions by nodes in R is enough for data gathering. Thus, set R is a SCDS and so $|MSCDS| \leq R$ which implies $|MSCDS| \leq t$. ■

In order to minimize the overall energy and bandwidth consumption, during data collection, we need to reduce the total number of transmitted bits. To this end, we suggest to minimize the total number of full transmissions as the first step since in a densely deployed network, references are usually significantly larger than differences. To this end, we need to identify the MSCDS of the communication graph C. As finding the minimum dominating set problem is well known to be NP-hard [44], in subsection 4.4.2, we suggest a heuristic algorithm that approximates the MSCDS as part of our data collection algorithm.

4.3.2 Optimal Data Representation

In this section, we determine the theoretical lower bound on the total number of bits required for representing a set of data in the forms of mutual differences. This is the minimum data that each node, responsible for forwarding the data set of its neighborhood toward base station, should transmit.

We first define a delta graph H as a complete graph in which each edge-weight between two nodes s_i and s_j, is equal to $d_{i,j}$ where $d_{i,j}$ is size of $\Delta_{i,j}$ in bits. Note that we assume $d_{i,j} = d_{j,i}$ for any i and j. We also assume that the references have the same size.

Any set of data can be represented as a reference plus a proper set of mutual differences. For example, $F = \{f_1, \ldots, f_3\}$ can be represented as f_1 (a reference) plus set of $S = \{\Delta_{1,2}, \Delta_{2,3}\}$. That means set F can be reconstructed from f_1 and set S.

A delta representation graph G for a representation set S is a graph in which two vertices s_i and s_j are adjacent in G if and only if $\Delta_{i,j}$ or $\Delta_{j,i}$ is in the set S.

Optimal delta representation graph is a representation graph corresponding to a representation set with minimum total number of bits.

Theorem 3 *An optimal delta representation graph G_{opt}, is a MST of the delta graph H.*

Proof: The representation graph G_{opt} spans all the nodes so that every data can be reconstructed. Graph G_{opt} is connected, otherwise every data can not be reconstructed. Graph G is a tree, otherwise it would have a loop. Assume we have a loop. Now, if we remove an edge within the loop, still every data can be represented. Therefore, we have a new representation graph with a less total edge-weight which is a contradiction.

It remains to prove that G_{opt} is the **minimum** spanning tree. This is straight forward since the total number of bits for representing a set of data (without considering the reference data) is the sum of edge-weights of G_{opt}. Thus, if this sum is minimum, then G_{opt} is a MST. ■

4.4 Data Collection Algorithm (DCA)

In this section, we introduce our proposed Data Collection Algorithm (DCA) which is our methodology for collecting the data of all the nodes in an energy-efficient way at the base station. This is a practical algorithm to minimize the total number of bits to be transmitted. DCA has three phases: First, the Level Calculation Phase (LCP) that finds the shortest distance of each node in number of hops from the base station. We need this information in the next stages of the algorithm. Second, the MSCDS Approximation Phase (MAP) which finds the set of reference nodes that transmit their reference messages. Third, the Data Gathering Phase (DGP) which gathers every node's data at the base station. All these algorithms are distributed and run locally at each node. In this section, we first assume that our shared communication link is lossless and error-free. In subsection 4.4.4, we explain how we deal with a lossy communication channel.

4.4.1 Level Calculation Phase

LCP identifies the distance of each node in number of hops from the base station which we refer to as *level*. Also for each node, LCP determines its *parents* which are defined as the adjacent nodes that are one level closer to the base station. We need this information about the level and parents of each node in the next two stages of the DCA.

LCP is straightforward and can be done similar to the level calculation stage in [45]. In this phase, starting from base station, each node announces its level which is zero for the base station. Upon receiving the level announcement message, each node that has not determined its level yet, identifies it by increasing the minimum received level by one. In addition, each node recognizes the nodes from which it receives the minimum level message, as its parents. Notice that each node may have multiple parents.

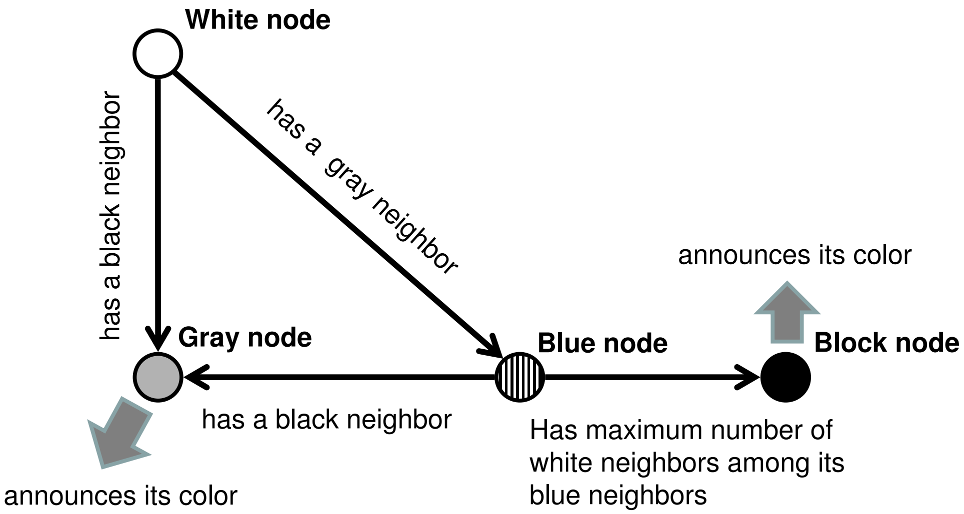

Figure 4.2: State transition diagram for MSCDS Approximation Phase

4.4.2 MSCDS Approximation Phase (MAP)

MAP approximates MSCDS through distinguishing two sets of nodes: BLACK nodes and GRAY nodes. BLACK nodes represent the nodes which perform full transmissions, and GRAY nodes represent the rest of nodes. Initially, all the nodes are WHITE which means that it has not yet been determined whether they are BLACK or GRAY. During the algorithm run, some nodes become BLUE meaning that they may be eligible to become BLACK, later.

Fig. 4.3 shows the state transition diagram of MAP. BLUE state is shown with the dashed lines. After LCP, where every node's level and parents are determined, the base station b colors itself BLACK and sends out a message to its neighbors. Note that b either owns a reference data or acquires one from one of its neighbors so it colors itself BLACK. Each node, upon receiving a message, makes a decision about its color and then announces it. The decisions are made as following:

- Each WHITE node colors itself GRAY if it receives a message from a BLACK parent.
- Each WHITE node colors itself BLUE if it receives a message from a GRAY parent.
- Each BLUE node colors itself GRAY if it receives a message from a BLACK neighbor.

If a WHITE node receives two messages at the same time from a BLACK and GRAY parent, it always gives the priority to the message of the BLACK parent and becomes GRAY.

These simple steps make sure that if a node has a reference node in its direct neighborhood, it doesn't need to transmit a reference data.

The algorithm for BLUE nodes is more complex. BLUE nodes negotiate with their BLUE neighbors to make their decision about becoming BLACK. To approximate the MSCDS, we suggest a greedy choice for BLUE nodes in the sense that a BLUE node that covers the maximum number of neighbors among its other BLUE neighbors, becomes BLACK. Each BLUE node only competes with its BLUE neighbors in its own level. This way competition is limited within each level and no BLUE node competes with its BLUE children.If two neighboring BLUE nodes, at the same level, have equal number of neighbors, one of them randomly colors itself BLACK and informs the other. Note that based on this algorithm, except for the base station, every BLACK node has been BLUE before.

To explain how MAP works on a network, consider the following example:

Example 4 *Consider the communication graph in Fig. 4.4. Each node's level is marked in the parenthesis. The levels are identified through the first stage as explained. MAP on this networks runs as follows:*

I) We start with LCP. The levels are identified in Fig. 4.4.I

II) Base station b colors itself BLACK and announces its color (Fig. 4.4.II).

III) Nodes s_1 and s_2, at level 1, upon receiving the message from BLACK node b, color themselves GRAY and announce their colors (Fig. 4.4.III).

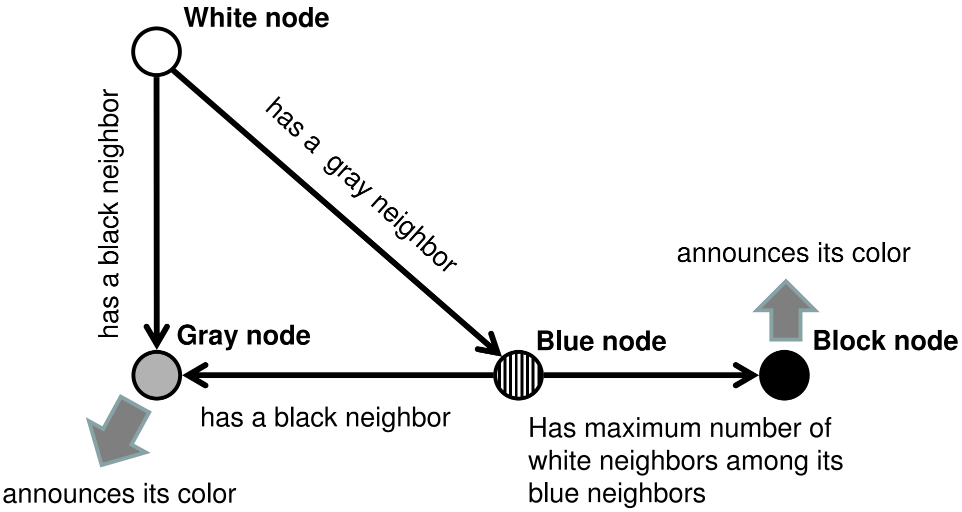

Figure 4.3: State transition diagram for MSCDS Approximation Phase

IV) Nodes s_3 and s_4, at level 2, upon receiving the message from GRAY nodes, color themselves BLUE (Fig. 4.4.IV, color BLUE is shown by dashed lines).

V) Node s_3 has 4 neighbors whereas s_4 has only 3. Thus s_3 colors itself BLACK and sends a message to its neighbors. (Fig. 4.4.V).

VI) Nodes s_4 and s_5 upon receiving this message, color themselves GRAY (Fig. 4.4.VI).

Note that $\{b, s_3\}$ is a MSCDS of the communication graph of Fig. 4.4.

Note that $\{b, s_3\}$ is a MSCDS of the communication graph of Fig. 4.4. Thus, in this example, MAP was able to determine MSCDS.

In section 4.5, we will first prove that set of BLACK nodes identified through MAP construct a SCDS of the communication graph C. Then, we find the upper bound on the number of BLACK nodes determined through MAP. We show that the cardinality of the set of BLACK nodes found through MAP is within eight times the cardinality of the MSCDS.

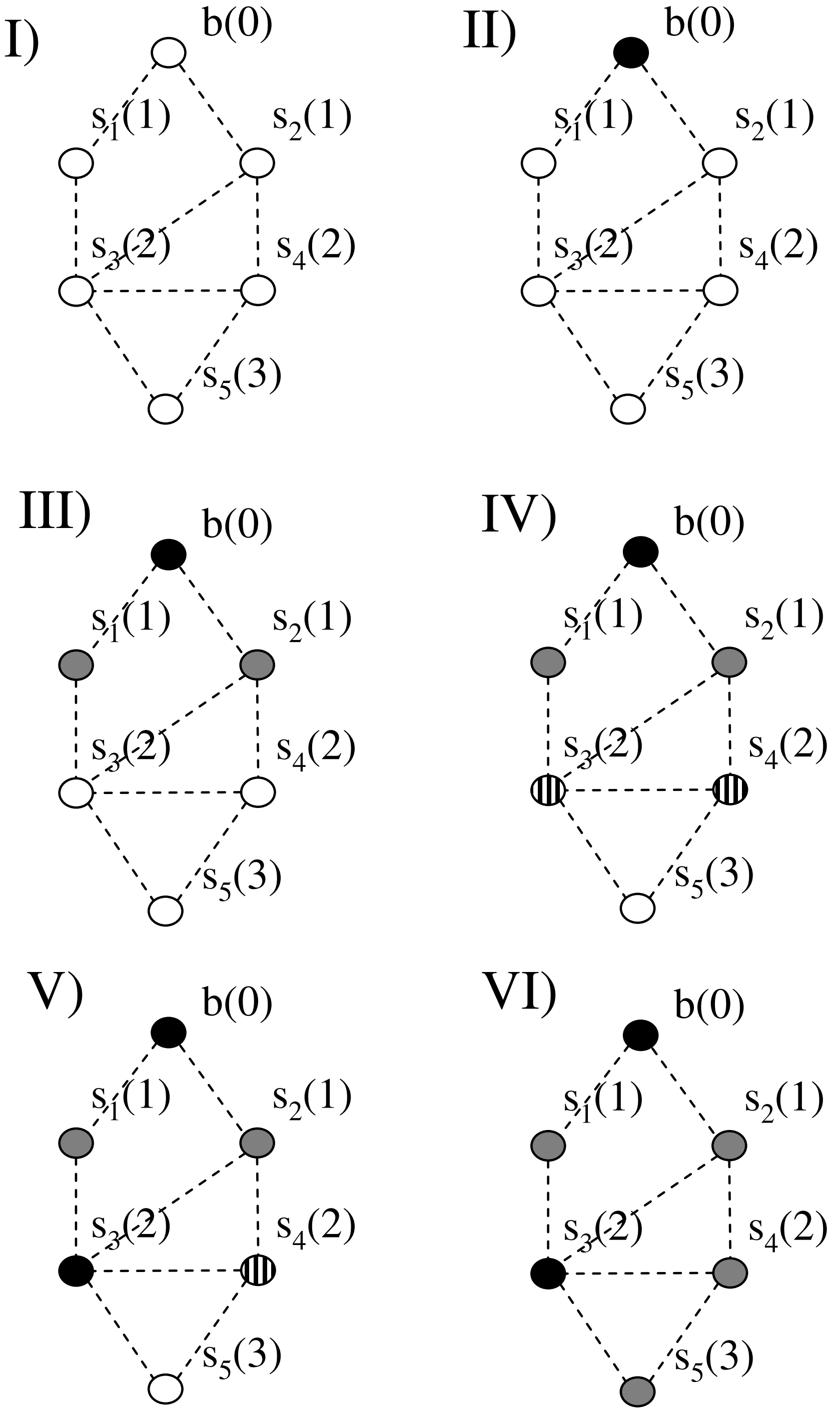

Figure 4.4: Example of MSCDS approximation

4.4.3 Data Collection Phase (DCP)

So far, we explained how to minimize the number of full transmissions by approximating MSCDS through MAP. In DCP, delta and references will be collected at a base station. Our goal is to minimize the number of transmitted bits by minimizing the size of data transmitted by each set of nodes.

In DCP, each node tries to contribute in compressing the data set of its neighborhood optimally by finding the Minimum Spanning Tree (MST) of the corresponding delta graph in a distributed way.

Nodes use four types of messages to communicate during DCP.

- REF: Data message containing the reference data and the ID of the node that transmits a reference.
- DIFF: Data message containing the delta and the ID of the node to which the delta belongs and the reference ID with respect to which the delta is computed.
- SIZE: Message sent by a parent to its BLACK child containing the ID of the child and the size of the child's data represented by the parent.
- SELECT: Message sent by a child containing the ID of the *selected* parent responsible for transmitting data of the child.

Pseudocode of DCP is in [25]. In summary, if a node is BLACK, it shares its reference data with its neighbors. If it is GRAY, it attempts to minimize the size of GRAY node's delta. If a node (BLACK or GRAY) has children, it transmits its children's data toward the base station.

DCP starts when each BLACK node performs a full transmission by sending out a REF message that contains its reference data. The REF message is received by the set of BLACK nodes' neighbors. Every GRAY node computes its delta with respect to a transmitted reference data. Some GRAY nodes may receive multiple references and so they will be able to compute multiple deltas for their data. If a node has more than one delta, it can choose the smallest one. s

After full transmissions, data of BLACK and GRAY nodes will be collected. The strategies for collecting the data of BLACK and GRAY nodes are different as following:

For BLACK nodes, we should make sure that every reference data is compressed with regard to another reference data of a higher level node. This avoids the loop referencing that may occur otherwise. Data of each BLACK node will be sent through one of its GRAY parents as explained in Example 3. Notice that a BLACK child has no BLACK parent since during MAP each BLACK node ensure that all its neighbors are GRAY. Similarly, a BLACK parent has no BLACK child. Every GRAY parent of a BLACK node receives at least another reference data from a BLACK parent or neighbor. The parent, computes the difference of these two references. Then, it sends out a SIZE message representing the size of the delta between them (If the parent has more than one delta for its BLACK child, it sends out the size of the smallest one). The BLACK child that may have received the SIZE message from multiple parents, selects a parent with the smallest size delta. Then, it notifies the parent through a SELECT message to forward its delta to its own parent. The forwarding continues until delta reaches to the base station.

Strategy for transmitting the data of GRAY nodes is is as follows: A GRAY node, waits for a time directly proportional to the size of its delta to send out its delta through a DIFF message. This transmission will be heard by all the parents and neighbors of this node. Then, it chooses a parent randomly using a SELECT message to forward its transmitted delta. The advantage of having a waiting time proportional to the size of delta, is that nodes with the smaller deltas will transmit their deltas first. Nodes with the larger deltas have a chance to wait and overhear other transmissions to see if they can re-represent their data with smaller deltas. Every time a node receives a delta, it will reset its waiting time. Notice that a GRAY node only updates its delta if it can represent its data with a smaller delta. We will discuss in subsection 4.5.2 that in a fully-connected neighborhood of nodes, this strategy results into the optimal delta compression since it is in fact a distributed version of the Prim's algorithm [46] for computing MST.

Each parent forwards the DIFF that it has received from its GRAY children immediately by choosing a parent randomly through SELECT.

We explain our algorithm through Example 5.

Example 5 *Consider the wireless network in Fig. 4.5. In Example 4, we found that MSCDS for this graph is* $\{b, s_3\}$. *Now, nodes* $s_i, i \in \{1, \ldots, 5\}$ *should transmit their data to* b.

Initially nodes b *and* s_3 *transmit their reference data* f_0 *and* f_3 *through REF messages respectively. These two are the only full transmissions that are required for our data collection. Note that we assume base station* b *is a node too and so provides* f_0. f_0 *is received by* s_1 *and* s_2 *and* f_3 *is received by* s_1, s_2, s_4 *and* s_5.

Node s_1 *computes* $\Delta_{1,0}$ *and* $\Delta_{1,3}$ *and keeps the smaller one (say* $\Delta_{1,0}$*). Similarly* s_2, s_4 *and* s_5 *keep* $\Delta_{2,0}$, $\Delta_{4,3}$ *and* $\Delta_{5,3}$ *respectively.*

We first explain how data of BLACK nodes are transmitted to the base station. Node s_3 *is BLACK, so Nodes* s_2 *and* s_1 *as the parents of* s_3 *compute the difference between two reference data* f_0 *and* f_3 *(*$\Delta_{3,0}$*). Node* s_2 *and* s_1 *both transmit size of* $\Delta_{3,0}$ *through the SIZE message. Since they both transmit the same size,* s_3 *randomly chooses one, (say* s_1*) and sends out a SELECT message containing* s_1*'s ID. Thus,* s_1 *now sends* $\Delta_{3,0}$ *to* b.

Now, we explain how data of GRAY nodes is transmitted: Assume s_4 *is the first one which transmits its delta* $\Delta_{4,3}$. *Sensor* s_4 *chooses* s_2 *as its parent through a SELECT message since it is the only parent of* s_4. *Nodes* s_2 *forwards* $\Delta_{4,3}$ *to* b *as* b *is also the only parent of* s_2. *After* $\Delta_{4,3}$ *is transmitted by* s_4, *it is received by* s_2 *and* s_5. *Nodes* s_2 *and* s_5 *use* $\Delta_{4,3}$ *to construct* f_4 *and then* $\Delta_{2,4}$ *and* $\Delta_{5,4}$ *respectively. Let's* $Size(\Delta_{5,4}) < Size(\Delta_{5,3})$. *Node* s_5 *updates its delta to* $\Delta_{5,4}$ *and resets its timer to start over its waiting time. But let's* $Size(\Delta_{2,0}) < Size(\Delta_{2,4})$. *Node* s_2 *does not update its delta but it still updates its timer. Node* s_5 *chooses a parent randomly (say* s_4*) and after timer of* s_5 *expires, it will transmit* $\Delta_{5,4}$ *which will be forwarded by* s_4 *to* s_2 *and then by* s_2 *to* b. *Nodes* s_2 *and* s_1 *similarly transmit* $\Delta_{2,0}$ *and* $\Delta_{1,0}$ *respectively to* b.

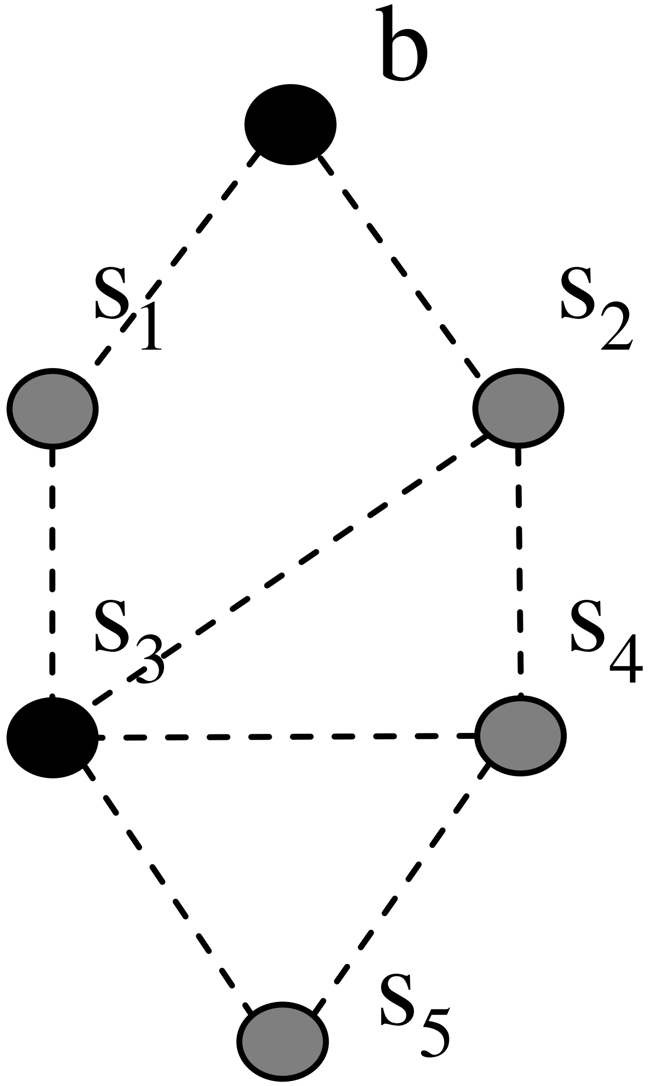

Figure 4.5: Example of Data Collection Algorithm

Having f_0, $\Delta_{2,0}$, $\Delta_{3,0}$, $\Delta_{1,0}$, $\Delta_{4,3}$ *and* $\Delta_{5,4}$, *base station* b *can reconstruct* $\{f1, \ldots, f_5\}$.

There exists a tradeoff between the size of deltas and the cost of extra storage and memory footprint of each node. During DCP, each GRAY node has to store data of all of its neighbors that have already transmitted their data. That's because it should be able to re-represent its delta using the data of any of its neighbors as reference. When a node receives a delta from a neighbor, it checks to see whether it has its reference or not and if it does, it reconstructs the neighbor's data. Thus, it has to keep an index list of the existing data in its memory. However, if each GRAY node only computes its delta with respect to the reference data provided by the BLACK neighbors, it will only need to store the smallest delta and does not need to keep any index list in memory. We call this version of the algorithm Simple DECOR (S-DECOR). Note that in S-DECOR, no attempt for MST computation is performed. We evaluate the performance of DECOR and S-DECOR in section 4.6 and compare them with some existing state of the art solutions.

4.4.4 Data Collection Algorithm for Lossy Communication Channels

In wireless networks, communication links are noisy and collision may occur especially when we use broadcast communication. Thus, designing an algorithm that is resilient to communication failures, is essential for a lossless data collection algorithm. In this section, we modify DECOR to deal with lossy networks. Our strategy is to avoid retransmissions especially for larger messages. When retransmission is inevitable, we reduce the chance of collision by introducing a random waiting time for each node after collision before retransmitting the data. We will discuss our strategy for each type of message that might be lost.

During DECOR, some control messages are sent, including the messages sent during MAP and also SIZE and SELECT during DCP. These are short messages and nodes can afford to employ simple error control techniques like ARQ (Automatic Repeat Request) methods [47] to correct the errors during transmission. Thus, we ignore the effect of noise and interference for this kind of messages.

For DIFF messages, if one parent receives the data correctly it does not need to be retransmitted. Thus, a DIFF message is lost when it is not received by any of its parents. If none of the parents have received the DIFF message correctly, it has to be retransmitted using an ARQ type technique.

A REF message might be lost too. In this case, our strategy is to avoid retransmitting the references by the BLACK nodes as much as possible because it results in the energy exhaustion at the BLACK nodes.

In three cases, if a REF message is lost or received incorrectly, another full transmission should be performed. We explain these three cases in the following. Notice that, strategies described in 1) and 2) ensure that data of BLACK nodes will receive to the base station whereas the strategy described in 3) ensures that data of a GRAY node that has not received any reference will be received by the base station.

1) If the full transmission by a BLACK node is not received correctly by any parent, there is no other option other than BLACK node repeating its transmission. The reason is that the data of the BLACK node should reach to the base station through one of its parents and so at least one parent should receive its data. Notice that this is the only case where BLACK node has to repeat its full transmission.

2) If a GRAY parent has received an error-free full transmission from a BLACK child but none from a BLACK parent or neighbor, then the GRAY node has to do one full transmission in order to transmit its BLACK child's data. The reason is that as explained in DCP, the GRAY parent should receive at least another reference data from its parent or neighbor to compute the delta for its BLACK child. If it does not receive that, it has to to repeat the full transmission.

3) A GRAY node that has not received any error-free references need to do a full transmission itself, so its data will be received by its parents.

4.5 Theoretical Performance Analysis

In this section, we evaluate the performance of DECOR. To this end, we first evaluate performance of MAP and then evaluate DCP. Finally, we determine the time complexity, message overhead, storage and memory requirement of DECOR.

4.5.1 Performance Analysis of MAP

In this subsection, we evaluate the performance of MAP. We first prove that set of BLACK nodes identified through MAP is a SCDS. Then, we derive an upper bound for the number of BLACK nodes found through MAP based on the size of MSCDS.

Theorem 4 *Set of BLACK nodes identified through MAP construct a SCDS of the communication graph C.*

Proof: Set of BLACK nodes D are a dominating set since each GRAY node is adjacent to at least one BLACK node.

Now, to show that set of BLACK nodes construct a SCDS, we need to show that bridge graph C_D is connected. For that it is enough to show that there is a path to b in graph C_D for any BLACK node and so all the BLACK nodes are connected in C_D through b. To this end, we show that for every BLACK node at level l_i, there is another BLACK node at level $l_i - 1$ or $l_i - 2$ within the two-hop neighborhood. Assume to the contrary that node s_i is a BLACK node at level $l_i \geq 2$ which does not have any BLACK node within its two-hop neighborhood at levels $l_i - 1$ or $l_i - 2$. Node s_i has been BLUE before. Thus, s_i has received a message from its GRAY parent s_j at level $l_i - 1$. But s_j has also received a message either from a BLACK parent s_k at level $l_i - 2$ or a BLACK neighbor at the same level $l_i - 1$. This contradicts the assumption that s_i has no BLACK node within its two-hop neighborhood at level $l_i - 2$ or $l_i - 1$. ■

Now, we show that the cardinality of the set of BLACK nodes found through MAP is within eight times the cardinality of the MSCDS. In other words, MAP has an approximation factor of at most 8.

The sketch of the proof is similar to [45]. We denote the size of MSCDS by opt. We first show that the set of BLACK nodes in MAP is an *independent* set where a set of nodes are *independent* if they are pairwise non-adjacent. Then, we find an upper bound for any independent set of a connected graph, namely set of BLACK nodes found through MAP, based on opt.

Lemma 1 *Set of BLACK nodes determined through MAP is an independent set.*

Proof: Assume to the contrary that MAP has determined two adjacent BLACK nodes s_i and s_j. Neither one of s_i and s_j can be the base station since all the neighbors of base station become GRAY, after base station announces its color. Thus, s_i and s_j have both been BLUE before. But a BLUE node turns GRAY if it has a BLACK neighbor. Therefore, set of BLACK nodes determined through MAP are pairwise non-adjacent and so are independent. ■

Now, we bound the size of any independent set in terms of opt. For our proof, we use this fact that in a unit-disk graph, each node is adjacent to at most five independent nodes [48].

Lemma 2 *The size of any independent set in a unit-disk graph $C = (V, E)$ is at most $8opt + 1$.*

Proof: Consider a spanning tree T_{MSCDS} of the bridge graph C_{MSCDS} (Note that C_{MSCDS} is connected, so T_{MSCDS} exists). We construct a subgraph $T \subset C$ as follows: 1) T includes every vertex in T_{MSCDS}. 2) For any v_i and v_j that are adjacent in T_{MSCDS}, we add exactly one vertex $v_k \in C$ to T such that v_k is adjacent to both v_i and v_j in C (by definition of bridge graph such v_k certainly exists). We also add the two edges that connect v_k to v_i and v_k to v_j to graph T. Graph T is a tree ;otherwise it has a loop. Now, we delete every vertex that is not in MSCDS from the loop in T, along with their corresponding edges. Since all deleted vertices were adjacent to exactly two vertices of MSCDS, the remaining graph which is T_{MSCDS} has also a loop. This is a contradiction since T_{MSCDS} is a tree. The size of T is at most $2opt$ since for each vertex $v_i \in MSCDS$ at most one vertex $v_k \notin MSCDS$ is added to T.

From here the proof is identical to the proof in [45].

Let U be any independent set of V. Consider any arbitrary preorder traversal of T given by $v_1, v_2, ..., v_{2opt}$. Let U_1 be the set of nodes in U that are adjacent to v_1. For any $2 \leq i \leq 2opt$, let U_i be the set of nodes in U that are adjacent to v_i but none of $v_1, v_2, ...v_{i-1}$. Then U_1, U_2,.., U_{2opt} form a partition of U. In other words, each element of U appears in exactly one subset of the list. As v_1 can be adjacent to at most five independent nodes, $|U_1| \leq 5$. But for any $2 \leq i \leq 2opt$, at least one node in $v_1, v_2, ...v_{i-1}$ is adjacent to v_i. Thus, $|U_i| \leq 4$. Thus the size of any independent set is equal to:

$$|U| = \sum_{i=0}^{2opt} |U_i| \leq 5 + 4(2opt - 1) = 8opt + 1$$

■

Theorem 5 *The approximation factor of MAP is at most 8.*

Proof: Lemma 1 shows that the set of BLACK nodes is an independent set and based on Lemma 2, the size of any independent set is bounded by $8opt + 1$. Thus, the the total number of BLACK nodes found through MAP is at most $8opt + 1$. Therefore, the approximation factor of MAP is equal to 8. ■

4.5.2 Performance of DCP

In this section, we show that DCP can minimize the number of transmitted bits by each node in a fully-connected neighborhood.

In a fully connected network, each data transmission is received by every node. Since every node resets its waiting time after receiving a delta, at each step always the node with the smallest delta transmits its data. Thus, our algorithm is in fact a distributed version of the Prim's algorithm [46] for finding the MST where at each step a new edge with the minimum size will be added to the constructed tree.

In general, a communication graph C of the nodes is not fully connected. However, a set of nodes in the same communication range r are certainly fully connected. During DCP we take advantage of this property to minimize the total number of bits in each neighborhood. Therefore, although DECOR is not guaranteed to be optimal, it can minimize the total number of transmitted bits in each neighborhood of nodes within the same communication range r.

4.6 Simulation

In this section, we evaluate the performance of DECOR and S-DECOR (section 4.4.3) through simulations. We randomly place 100 nodes in a two-dimensional 1×1 space based on the uniform distribution in MATLAB environment. Each node has a 10 kbit data which should be transmitted to the base station located at the center of the area (Figure 4.6).

In the following subsections, first, we evaluate the performance of MAP, as the first phase of DECOR. Then, we compare performance of DECOR and S-DECOR with RDC and CRDC. As explained in the introduction section, RDC removes the redundant data at each relaying node on the shortest path to the base station and CRDC suppresses the redundant data at each cluster head where the cluster data is collected.

4.6.1 Performance Evaluation of MAP

We compare the number of BLACK nodes found through MAP with the actual size of MSCDS and the analytical upper bound we derived in section 4.5 for MAP. We run the simulations for different communication graphs of the network. To change the communication graph, we change the communication range of the nodes. As shown in Fig. 4.7, the cardinality of MSCDS (minimum number of full transmissions), decreases when the communication range increases. This result is expected since the connectivity of the communication graph grows with the nodes' communication range. Thus, the size of MSCDS decreases. Fig. 4.7 also shows that MAP performs much closer to the optimal answer than the analytical upper bound we derived in section 4.5 for MAP.

4.6.2 Performance Evaluation of DECOR

We compare performance of DECOR and S-DECOR in data collection with RDC and CRDC by investigating the effect of the following three parameters on the number of transmitted bits:

- Correlation coefficient
- Number of nodes
- Probability of channel failure

Effect of Correlation Coefficient

Correlation Coefficient is used as measure of correlation among data of sensors. We model the relation between correlation coefficient, delta and the distance between nodes s_i and s_j as following:

$$\text{Size}(\Delta_{i,j}) = \left(1 - \exp\left(-\beta \left\|s_i - s_j\right\|_2^2\right)\right) F, \tag{4.1}$$

where $\|s_i - s_j\|_2^2$ is the distance between nodes s_i and s_j, and F is the size of data (10 kbit) and β is correlation coefficient which is a constant factor that controls the correlation among data of sensors. This model is similar to the model proposed in [49, 50]. Notice that $\Delta_{i,j}$ has limiting values of F when β is infinite and zero when β is zero.

Fig. 4.8 shows the number of transmitted bits normalized with respect to the size of data versus correlation coefficient β for DECOR, S-DECOR, RDC and CRDC. Both RDC and CRDC algorithms require n number of full transmissions since every node should do one full transmission. Therefore, for high correlation (small β), the number of transmitted bits in DECOR and S-DECOR, which only require BLACK nodes to perform full transmissions, is much less than those of RDC and CRDC. As the correlation among the nodes data decreases (large β), the size of delta converges to the size of reference data. Therefore, normalized number of transmitted bits in both DECOR and S-DECOR increases and converges to $\sum_{i=1}^{n} l_i$ where l_i is level

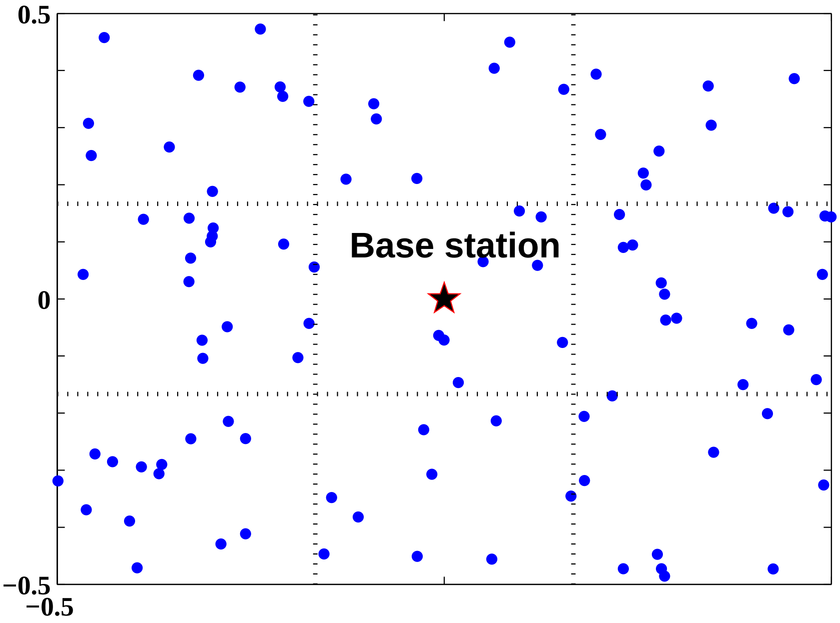

Figure 4.6: Distribution of the nodes and the base station in the area

of sensor i. As shown in Fig. 4.8, the normalized number of bits sent in both DECOR and S-DECOR are equal to that of RDC in low or no correlation. The reason is that when there is no correlation, DECOR, S-DECOR and RDC all similarly relay the data along the shortest path to the base station with no suppression. CRDC performs worst since it first collects data of each cluster at the cluster head and so data is not necessarily transmitted through the shortest path.

Effect of Number of nodes

In this section, we study the number of transmitted bits as a function of the number of nodes for fixed communication range and correlation coefficient β. We choose β=1 which enforces a large correlation among data of nodes. The results in Fig. 4.9 show that as the number of nodes increases, the number of transmitted bits for RDC and CRDC increases linearly. This is

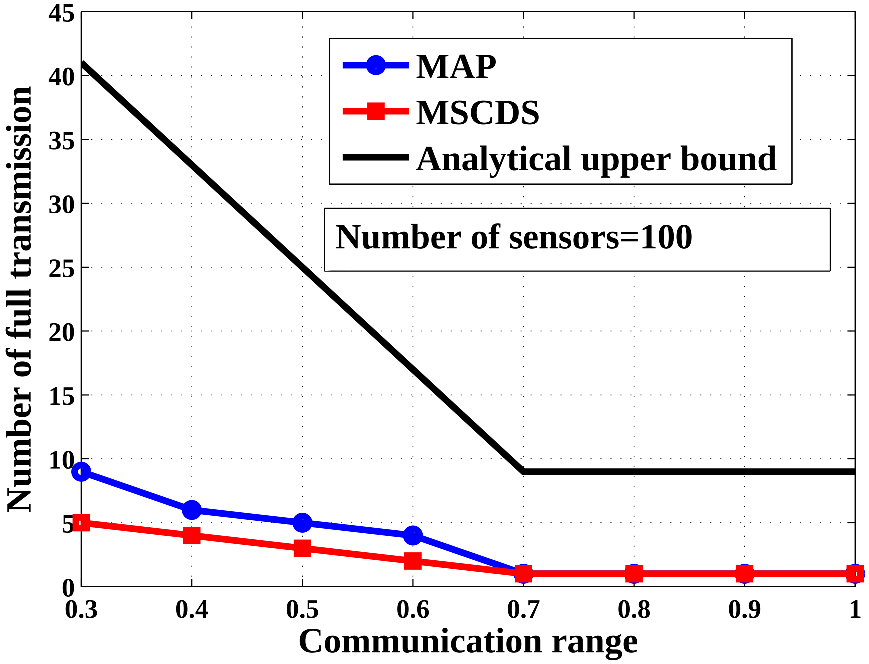

Figure 4.7: Number of full transmission versus communication range

because in these algorithms, number of full transmissions increases with the number of nodes. Notice that RDC and CRDC show a similar performance for the selected β. However, number of transmitted bits in DECOR remains almost constant when number of nodes increases. The reason is that when number of nodes grows, their data becomes highly correlated and most of it becomes redundant. DECOR successfully captures this and does not transmit the redundant data.

Effect of Channel Failure

Finally, we evaluate the performance of the algorithm presented in subsection 4.4.4 for lossy communication channels. We consider a network of 100 nodes with fixed communication range and correlation coefficient β. Moreover, we assume that the probability of channel failure is

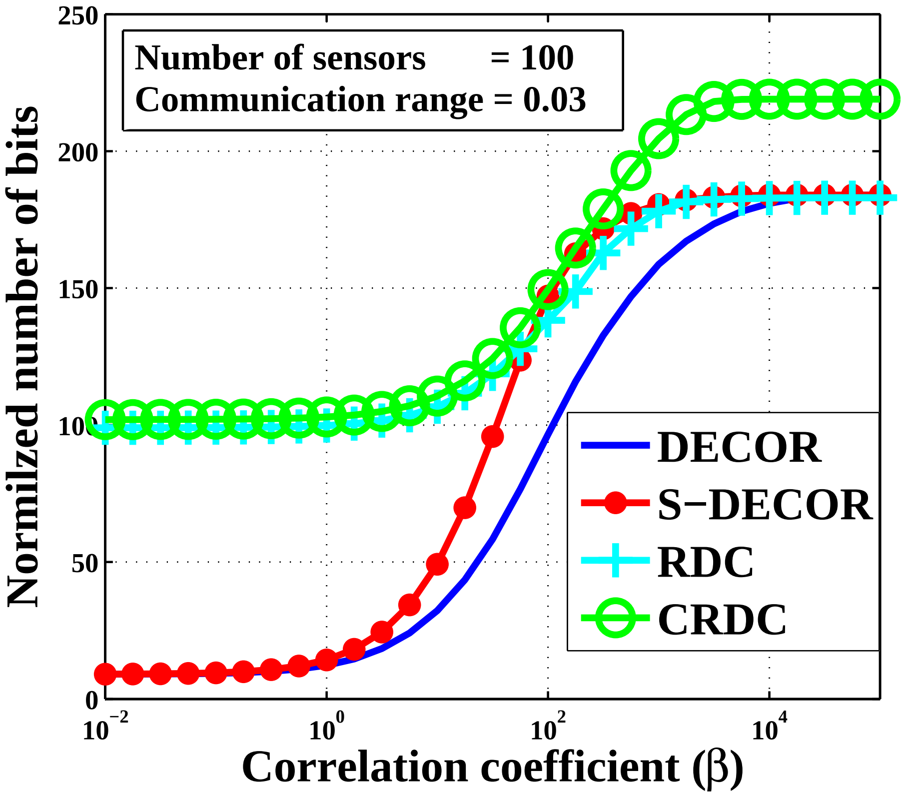

Figure 4.8: The number of transmitted bits (normalized with data size) versus correlation coefficient β

constant over the channels. Figure 4.10 shows the number of transmitted bits for both DECOR and S-DECOR normalized with respect to the number of transmitted bit for error-free communication scenario averaged over 20 runs. As it is expected, the number of transmitted bits increases with probability of channel failure. Notice that DECOR is more robust to communication failure than S-DECOR. For example, if the probability of channel failure is equal to 0.5, DECOR requires only 3 times more transmitted bits comparing to the error-free case where S-DECOR requires 4.5 time more transmitted bits.

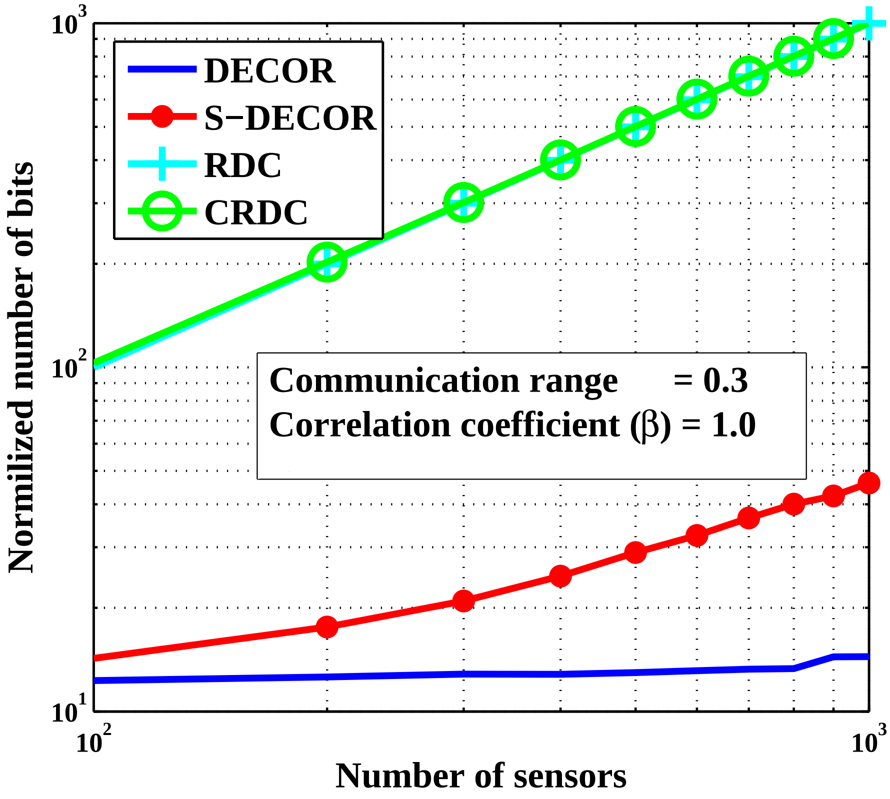

Figure 4.9: The number of transmitted bits (normalized with data size) versus the number of nodes

4.7 Conclusion

In this chapter, we exploited the spatial correlation on the data of wireless nodes to reduce the communication energy and bandwidth consumption during data collection. We suppress the redundant data sent through data collection using a two-step strategy. First minimizing the number of full transmission and second minimizing the size of data produced by set of nodes at each neighborhood. We found a lower bound on the minimum number of full transmissions required for lossless data collection. We also found the minimum number of bits required for

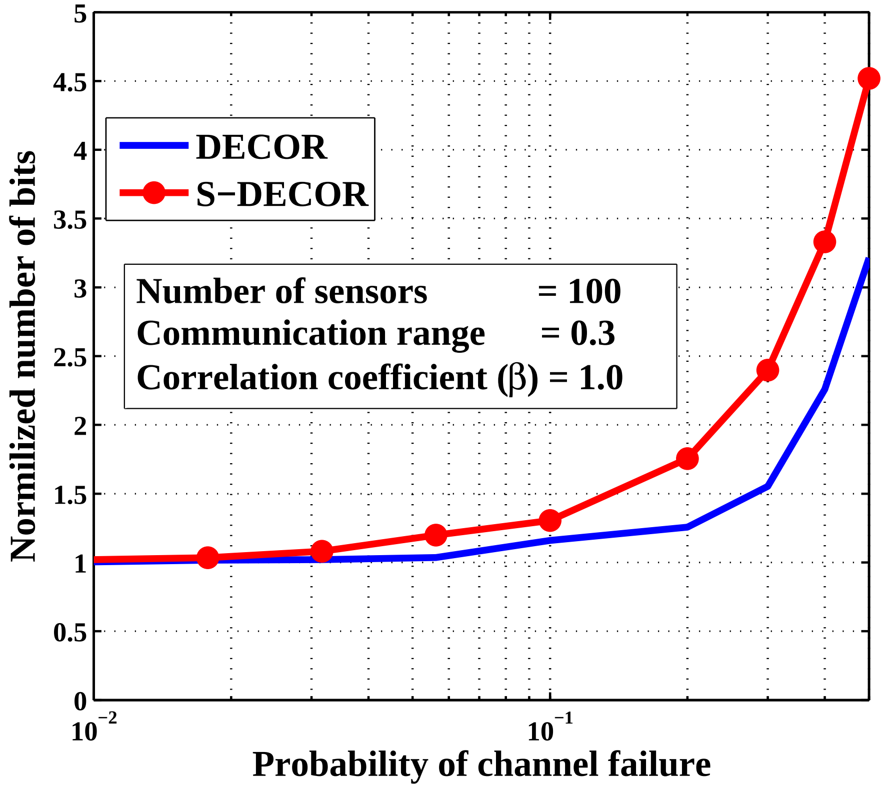

Figure 4.10: The number of transmitted bits (normalized with data size) versus the number of nofrds

compressing a data set using the mutual differences. Based on our theoretical results, we introduced a practical algorithm for data collection which reduces the overall energy consumption. Finally, we presented the numerical simulations to verify our theoretical results.

Chapter 5

Conclusion

This dissertation studies two important problems in vehicular networks: collision avoidance and gathering roadway traffic information. Both of these applications can use DSRC technology for communication and use basic safety messages as their primary source of information. However, the primary challenges for these two applications are different: Collision avoidance requires the safety packets to be delivered successfully within a certain deadline. For traffic information gathering, the bandwidth consumption is the primary concern since the amount of data to be gathered is large.

We first studied the problem of using standard QoS mechanism of IEEE 802.11 called Enhanced Distributed Channel Access (EDCA) to minimize frame collisions among periodic vehicle safety messages. By minimizing the frame collision among periodic safety messages, we reduce the delay in delivery of the safety messages. We used to basic parameters of EDCA called $AIFSN$ and CW_{min} and we made two main contributions: 1) We demonstrated an Access Category (AC) isolation technique that dramatically reduces the collision probability of high priority packets in dense environments. 2) We introduced a novel concept called virtual division by which significant reductions in collisions among lower priority packets can be achieved. We verified via extensive NS-2 simulations of both simple and hidden node topologies.

Second, we introduced a new contention free MAC protocol for periodic safety message communication which can bound the delay in very dense vehicular networks. We have the following main contributions: 1) We derived the theoretical lower bound on the maximum delay, in delivery of basic safety messages. This bound is a linear function of the maximum number of vehicles that interfere with one vehicle, in the network. 2) We proposed a distributed cluster-based MAC protocol, which maximizes collision-free parallel transmissions of periodic safety messages. The proposed MAC protocol guarantees an upper bound on the delivery delay of routine safety messages which is only within 4 times from the optimal delay. Our protocol is distributed and dynamic, and easily adjusts to the changes in the network.

Third, we studied the problem of collecting traffic information in access points. Using vehicles as traffic probe sensors, is a cost effective solution to this problem. Since bandwidth is a sparse resource especially when the highway is dense, an efficient protocol is required to collect all the data across the network at a certain access point or RSU with minimum bandwidth consumption . For this purpose, we exploit the correlation on the data of vehicles to reduce the bandwidth consumption during data collection by suppressing the redundant data from multiple vehicles. In our methodology, a certain subset of vehicles send out their data without any suppression (full transmission). The vehicles that perform full transmissions are called reference nodes. The rest of vehicles only send out the difference between their data and that of reference nodes. Our strategy is based on two key principals. The first is to minimize the total number of full transmissions. This strategy minimizes the overall bandwidth consumption since in general, full transmissions consume significantly more bandwidth than transmission of the compressed data. The second principal is to remove the redundant data sent by each forwarding node on the path to the access point. We have three main contributions: 1) we obtained a theoretical lower bound on the total number of full transmissions required for lossless collection of data at the access point. This lower bound is equal to the size of a specific subset of nodes. 2) We introduced an efficient algorithm that approximates the elements of this set. 3) We determined a lower bound on the minimum number of bits that a node which is responsible for forwarding the data of its neighborhood should transmit for lossless data collection. Based on our theoretical findings, we proposed a practical data collection algorithm which can reduce the overall bandwidth consumption significantly.

The three proposed strategies in this dissertation are new steps that make vehicular network technology closer to reality. MAC protocol solutions suggested in this dissertation will reduce the delay in delivery of safety messages which is crucial for collision avoidance applications. It can be envisioned that in US, adaptive rate algorithms to control channel congestion will interact with EDCA techniques to lower PER. For example, if the EDCA techniques produce lower PER (and thus higher channel load) for a given offered load, perhaps the target channel load of the adaptive message rate algorithm should be higher than with default EDCA. Another interaction is the impact that adaptive rates have on virtual division proportions. TDMA-based MAC protocols for vehicular network are the subject of research in Europe and we can predict that techniques similar to the proposed protocol will be used. More research needs to be done to extend this algorithm for the intersection and urban scenarios. Another important application of vehicular network is gathering traffic information, analyze the data at some central location and then disseminate this information to the vehicles. More research has to be done to do analysis on this subject especially how to disseminate this information to the vehicles so that vehicles can avoid traffic and select routes accordingly.

References

[1] J. Kenney, “Dedicated short-range communications (DSRC) standards in the united states,” *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, July 2011.

[2] U. D. of Transportation, “Early estimate of motor vehicle traffic fatalities for the first nine months (january-september) of 2012,” December 2012. [Online]. Available: http://www-nrd.nhtsa.dot.gov/Pubs/811706.pdf

[3] A. Zaid, “Vehicle safety communications-applications VSC-A, first annual report,” *US Department of Transportation, National Highway TrafficSafety Administration, Rep. DOT HS 811 073*, 2009.

[4] D. Committee, “Dedicated short range communication (DSRC) message set dictionary,” *SAE Std. J2735, SAE Int.*, November 2009.

[5] R. Resendes, “Vehicle-to-vehicle and safety pilot,” in *Proceedings of IntelliDrive Safety Workshop*, July 2010. [Online]. Available: http://www.its.dot.gov/presentations/safety_workshop2010/Vehicle-toVehicleandSafetyPilot--RResendes.pdf

[6] “Amendment of the commissions rules regarding dedicated short-range communication services in the 5.8505.925 GHz band (5.9 GHz band),” Tech. Rep., July 2006.

[7] “IEEE standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (MAC) and physical layer

(PHY) specifications amendment 6: Wireless access in vehicular environments," *IEEE Std 802.11p*, pp. 1–51, 2010.

[8] C. Huang, Y. Fallah, R. Sengupta, and H. Krishnan, "Adaptive intervehicle communication control for cooperative safety systems," *Network, IEEE*, vol. 24, no. 1, pp. 6–13, 2010.

[9] J. Kenney, G. Bansal, and C. Rohrs, "LIMERIC: a linear message rate control algorithm for vehicular DSRC systems," in *Proceedings of the Eighth ACM international workshop on Vehicular inter-networking*. ACM, 2011, pp. 21–30.

[10] J. Hui and M. Devetsikiotis, "A unified model for the performance analysis of IEEE 802.11e EDCA," *Communications, IEEE Transactions on*, vol. 53, no. 9, pp. 1498 – 1510, September 2005.

[11] A. Banchs, A. Azcorra, C. Garcia, and R. Cuevas, "Applications and challenges of the 802.11e EDCA mechanism: an experimental study," *Network, IEEE*, vol. 19, no. 4, pp. 52–58, July-August 2005.

[12] M. Torrent-Moreno, D. Jiang, and H. Hartenstein, "Broadcast reception rates and effects of priority access in 802.11-based vehicular ad-hoc networks," in *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*. ACM, 2004, pp. 10–18.

[13] J. Gallardo, D. Makrakis, and H. Mouftah, "Performance analysis of the EDCA medium access mechanism over the control channel of an IEEE 802.11p WAVE vehicular network," in *Communications, 2009. ICC '09. IEEE International Conference on*, june 2009, pp. 1–6.

[14] C. Suthaputchakun and A. Ganz, "Priority based inter-vehicle communication in vehicular ad-hoc networks using IEEE 802.11e," in *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, April 2007, pp. 2595–2599.

[15] S. Eichler, “Performance evaluation of the IEEE 802.11p WAVE communication standard,” in *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, October 2007, pp. 2199–2203.

[16] D. Rawat, D. Popescu, G. Yan, and S. Olariu, “Enhancing VANET performance by joint adaptation of transmission power and contention window size,” *Parallel and Distributed Systems, IEEE Transactions on*, vol. 22, no. 9, pp. 1528–1535, September 2011.

[17] T. Sakurai and H. Vu, “MAC access delay of IEEE 802.11 DCF,” *Wireless Communications, IEEE Transactions on*, vol. 6, no. 5, pp. 1702–1710, May 2007.

[18] R. Scopigno and H. Cozzetti, “Gnss synchronization in vanets,” in *New Technologies, Mobility and Security (NTMS), 2009 3rd International Conference on*, dec. 2009, pp. 1 –5.

[19] S. Bana and P. Varaiya, “Space Division Multiple Access (SDMA) for robust ad hoc vehicle communication networks,” in *Intelligent Transportation Systems Proceedings IEEE*, 2001, pp. 962–967.

[20] J. J. Blum and A. Eskandarian, “A reliable link-layer protocol for robust and scalable intervehicle communications,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, no. 1, pp. 4–13, Mar. 2007.

[21] F. Borgonovo, A. Capone, M. Cesana, and L. Fratta, “ADHOC MAC: new MAC architecture for ad hoc networks providing efficient and reliable point-to-point and broadcast services,” *Wireless Networks*, vol. 10, no. 4, pp. 359–366, 2004.

[22] K. Bilstrup, E. Uhlemann, E. Strom, and U. Bilstrup, “Evaluation of the IEEE 802.11p MAC method for vehicle-to-vehicle communication,” in *Vehicular Technology Conference, 2008. VTC 2008 IEEE 68th*, Sep. 2008, pp. 1–5.

[23] F. Yu and S. Biswas, "A self-organizing MAC protocol for DSRC based vehicular ad hoc networks," in *27th International Conference on Distributed Computing Systems Workshops, 2007. ICDCSW'07*, 2007.

[24] H. Su and X. 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.

[25] S. Sharafkandi and D. Du, "A New MAC Layer Protocol for Safety Communication in Dense Vehicular Networks," in *4th Workshop On User MObility and VEhicular Networks, IEEE LCN ON-MOVE*, 2010.

[26] M. Gerla and J. Tzu-Chieh Tsai, "Multicluster, mobile, multimedia radio network," *Wireless networks*, vol. 1, no. 3, pp. 255–265, 1995.

[27] X. Chen and H. Refai, "SDMA: On the suitability for vanet," in *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on*, Apr. 2008, pp. 1–5.

[28] D. Baker and A. Ephremides, "A distributed algorithm for organizing mobile radio telecommunication networks," in *Proceedings of the 2nd International Conference on Distributed Computer Systems*, 1981, pp. 476–483.

[29] D. West, *Introduction to Graph Theory*. Prentice Hall Upper Saddle River, NJ, 2001.

[30] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review*, vol. 62, no. 2, pp. 1805–1824, 2000.

[31] C. Sommer, I. Dietrich, and F. Dressler, “Realistic simulation of network protocols in vanet scenarios,” in *2007 Mobile Networking for Vehicular Environments*, May 2007, pp. 139–143.

[32] H. Arbabi and M. Weigle, “Monitoring free flow traffic using vehicular networks,” in *IEEE Consumer Communications and Networking Conference (CCNC)*. IEEE, 2011, pp. 272–2760.

[33] H. Zuylen, F. Zheng, and Y. Chen, “Using probe vehicle data for traffic state estimation in signalized urban networks,” *Traffic Data Collection and its Standardization*, pp. 109–127, 2010.

[34] X. Li, W. Shu, M. Li, H. Huang, P. Luo, and M. Wu, “Performance evaluation of vehicle-based mobile sensor networks for traffic monitoring,” *Vehicular Technology, IEEE Transactions on*, vol. 58, no. 4, pp. 1647–1653, 2009.

[35] B. Krishnamachari, D. Estrin, and S. Wicker, “Modelling Data-Centric Routing in Wireless Sensor Networks,” *IEEE INFOCOM*, pp. 1–11, 2002.

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

[37] K. Kalpakis, K. Dasgupta, and P. Namjoshi, “Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks,” *Computer Networks*, vol. 42, no. 6, pp. 697–716, 2003.

[38] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, “Directed diffusion for wireless sensor networking,” *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, pp. 2–16, 2003.

[39] A. Silberstein, R. Braynard, and J. Yang, “Constraint chaining: on energy-efficient continuous monitoring in sensor networks,” in *Proceedings of the 2006 ACM SIGMOD international conference on management of data*. ACM New York, NY, USA, 2006, pp. 157–168.

[40] J. Chou, D. Petrovic, and K. Ramchandran, “A distribued and adaptive signal processing approach to reducing energy consumption in sensor networks,” in *Proceedings of IEEE INFOCOM*, San Fransisco, CA, USA, March, April 2003, pp. 1054–1062.

[41] H. Gupta, V. Navda, S. R. Das, and V. Chowdhary, “Efficient gathering of correlated data in sensor networks,” in *Proceedings of the 6th ACM international Symposium on Mobile Ad Hoc Networking and Computing*. New York, NY, USA: ACM Press, 2005, pp. 402–413.

[42] S. Pattem, B. Krishnamachari, and R. Govindan, “The impact of spatial correlation on routing with compression in wireless sensor networks,” in *Proceedings of the third international symposium on Information processing in sensor networks*. ACM Press New York, NY, USA, 2004, pp. 28–35.

[43] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, “Impact of Network Density on Data Aggregation in Wireless Sensor Networks,” in *International conference on distributed computing systems*, vol. 22. IEEE Computer Society; 1999, 2002, pp. 457–458.

[44] S. Guha and S. Khuller, “Approximation algorithms for connected dominating sets,” *Algorithmica*, vol. 20, no. 3, pp. 374–387, 1998.

[45] P.-J. Wan, K. M.Alzoubi, and O. Frieder, “Distributed construction of connected dominating set in wireless ad hoc networks,” in *Proceedings of IEEE INFOCOM 21st Annual*

Joint Conference of the IEEE Computer and Communications Societies, vol. 3, June 2002, pp. 1597–1604.

[46] R. Prim, “Shortest connection networks and some generalizations,” *Bell Systems Technical journal*, vol. 36, pp. 1389–1401, 1957.

[47] L. L. Peterson and B. S. Davie, *Computer Networks: A Systems Approach*, 3rd ed. Morgan Kaufmann, 2003.

[48] M. Marathe, H. Breu, H. Hunt III, S. Ravi, and D. Rosenkrantz, “Simple heuristics for unit disk graphs,” *Networks*, vol. 25, no. 2, pp. 59–68, 1995.

[49] K. Yuen, B. Liang, and L. Baochun, “A distributed framework for correlated data gathering in sensor networks,” *Vehicular Technology, IEEE Transactions on*, vol. 57, no. 1, pp. 578–593, 2008.

[50] M. Vuran and I. Akyildiz, “Spatial correlation-based collaborative medium access control in wireless sensor networks,” *Networking, IEEE/ACM Transactions on*, vol. 14, no. 2, pp. 316–329, 2006.