Relays Recovery and MAC-Layer Cooperation Models in VANETs

Doaa Al-Terri

MSc. Thesis

June, 2016

A Msc. thesis submitted to Khalifa University of Science, Technology and Research in accordance with the requirements of the degree of Msc. in Engineering in the Electrical and Computer Engineering Department.

Relays Recovery and MAC-Layer Cooperation Models in VANETs

by

Doaa Al-Terri

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science by Research in Engineering (Electrical & Computer Engineering)

at

Khalifa University

Thesis Committee

Prof. Hassan Barada (Supervisor),
Khalifa University
Dr. Hadi Otrok (Co-Supervisor),
Khalifa University
Prof. Mahmoud Al-Qutayri (Co-Supervisor),
Khalifa University
Dr. Yousof Al Hammadi (Co-Supervisor),
Khalifa University
Prof. Raed Shubair (Co-Supervisor),
Khalifa University

Prof. Barry O'Sullivan (External Examiner),
University College Cork, Ireland
Dr. Khaled Salah (Internal Examiner),
Khalifa University

June 2016

Abstract

Doaa Al-Terri, "Relays Recovery and MAC-Layer Cooperation Models in VANETs", M.Sc. M.Sc. by Research in Engineering, Department of Electrical and Computer Engineering, Khalifa University of Science, Technology and Research, United Arab Emirates, June 2016.

In this thesis, we address the problem of selfishness in Vehicular ad-hoc Networks (VANETs) at both network and MAC layers. At the network layer, a vehicle is considered selfish when it is over speeding or under speeding the maximum/minimum road speed limits. This leads to a disconnected network due to the non-cooperative behavior in forwarding packets. At the MAC layer, a selfish vehicle can misbehave to acquire more bandwidth by not complying with the standard MAC protocol design. Thus, the presence of selfish nodes would hinder normal nodes' services.

In this research, we explore the Qos-OLSR protocol to study the impact of the nodes' misbehavior on a routing protocol. QoS-OLSR is a cluster based proactive routing protocol that takes into account the quality of service while selecting the MultiPoint Relay (MPR) nodes. This protocol does not consider the case of MPR disconnection due to mobility caused by the misbehaving nodes, which leads to network disconnection. Therefore, the performance of QoS-OLSR has been improved by proposing a new routing protocol based on Intelligent Water Drop algorithm identified as IWD-QoS-OLSR protocol. This protocol introduces an MPR recovery algorithm that is capable of selecting alternatives in the case of link failure disconnection, caused by the selfish nodes behavior.

The existence of selfish behavior at the MAC layer disrupts the services of the network. The distributed nature of the CSMA/CA MAC protocol allows rational nodes to deliberately manipulate their backoff parameters to gain an unfair share of the network throughput. Therefore, we develop a game theoretical mechanism that encourages the rational nodes to behave normally under the threat of retaliation. Thus, the selfish nodes are motivated to cooperate if they aim to maximize their obtained payoff. Otherwise, all the nodes will end up with less payoffs. Finally, simulations are conducted to evaluate the performance of the proposed models.

Indexing Terms: VANETs, QoS-OLSR, Intelligent Water Drop algorithm(IWD), CSMA/CA, Game Theory.

Acknowledgements

I owe my gratitude to all those people who were supporting and encouraging me through my masters study. First, I would like to express my deepest gratitude to my advisors, Dr. Hadi Otrok for his time, support and patience, our meetings and discussions made this thesis possible. Dr. Hassan Barada, Dr. Mahmoud Al-Qutayri, Dr. Raed M. Shubair, and Dr. Yousof Al-Hammadi for their continuous guidance and efforts in this research work.

I would like to thank my examination committee, Dr. Khaled Salah and Dr. Barry O'Sullivan for their constructive feedback on the thesis.

My sincere thanks go to my family as they were always supporting and encouraging me with their best wishes. I am also grateful to my friends: Dina Shehada and Hanin AbuBaker, who were always willing to help and give their best suggestions and for all the fun we had in th last two years.

Lastly, many thanks for my friends: Tasneem Salah and Alaa Khalid for supporting me spiritually throughout my study years. Thanks for the great time we spent together, I would have felt so lonely at the hostel without them.

To my Family...

Declaration and Copyright

Declaration

I declare that the work in this thesis was carried out in accordance with the regulations of Khalifa University of Science, Technology, and Research. The work is entirely my own except where indicated by special reference in the text. Any views expressed in the thesis are those of the author and in no way represent those of Khalifa University of Science, Technology, and Research. No part of the thesis has been presented to any other university for any degree.

SIGNED: Doaa Al-Terri DATE: 14/7/2016

Copyright ©

Contents

List of Figures

List of Tables

List of Abbreviations

MANETs	**Mobile ad-hoc networks**
VANETs	**Vehicular ad-hoc networks**
DSR	**Dynamic Source Routing**
OLSR	**Optimized Link State Routing**
QoS	**Quality of Service**
MAC	**medium access control**
CSMA/CA	**Carrier Sense Multiple Access with Collision Avoidance**
MPR	**MultiPoint Relay**
DOS	**Denial of Service**
DSRC	**Dedicated Short Range Communications**
RSUs	**Road-side Units**
OBUs	**On Board Units**
VCI	**Vehicular Communication Infrastructure**
IVC	**Inter-vehicle communications**
TC	**Topology Control**
WDs	**Water Drops**
NAV	**network allocation vector**
DIFS	**Distributed Inter Frame pace**
BO	**Backoff**
CW	**Contention Window**
EDCA	**Enhanced Distributed Channel Access**
DCF	**Distributed coordination function**
AIFS	**Arbitration Inter-Frame Space period**

GPS	**Global Position System**
ACK	**Acknowledgment**
CTS	**Clear to send**

List of Symbols

n	Set of nodes in the network
N_1	Set of 1-hop neighbors
N_2	Set of 2-hop neighbors
$QoS(i)$	Quality of Service Metric of a node i
$BW(i)$	Available bandwidth of node i
$N(i)$	Number of neighbors for node i
$MPR(\mathrm{i})$	MPR set for node i
S	Source cluster-head
D	Destination cluster-head
P	Set of all paths leading to D
N	The players set
a_i	is the strategy chosen by node i
P_i	is the payoff of node i
R_i	The reputation of node i (level of selfishness)
Th	Reputation threshold value
$R_{final}(i)$	is the aggregated reputation for node i
W_i	is the set of observers monitoring i
R_{w_j}	is the reputation calculated from observer w_j.
AK_i	is the number of ACK frames received by node i
T	is the selfishness threshold value

Chapter 1

Introduction

The rapid evolution of wireless data communication technologies paved the way for the development of wireless ad-hoc networks. Wireless Ad-hoc networks are self-configuring networks which are independent of infrastructure where each node participates in forwarding data to other nodes. Ad-hoc networks were initiated by the Defense Advanced Research Project Agency (DARPA), which tried to use packet switched radio communication to provide reliable communication between computers [1].

Vehicular Ad-hoc Networks (VANETs), are new emerging technology in the field of ad-hoc networks where they are considered a sub-class of Mobile Ad-hoc Networks (MANETs). MANETs are mobile wireless networks in which the nodes are free to move independently [2]. Compared to MANETs, VANETs are characterized with the high mobility of the nodes where the movement of the nodes is restricted by the road topology. The main purpose of VANETs is to provide the mobile users with continuous connectivity while they are traveling on the roads. For the past few years, a significant amount of research has been conducted in the field of VANETs due to the numerous distributed applications they can support. Their applications can be categorized as: Safety applications and Comfort applications [3] [4]. Safety applications aim to enhance the road safety by exchanging emergence messages among the vehicles

to save people's lives as shown in Figure 1.1 [4]. These messages include collisions warning alerts, lane change, and curve warnings alarms. On the other hand, Comfort applications aim to improve the comfort level for passengers. For example, such applications can supply the users with information about the weather, nearest gas station, hotels, or restaurants. They also can provide the passengers with comfort services such as: infotainment and online games [5] [6]. The effectiveness of such applications is highly dependent on the proper delivery of the exchanged messages among the vehicles. Therefore, the subject of message delivery has attracted the research community lately. Routing in ad-hoc networks can be classified into two categories: reactive and proactive routing protocols. Reactive routing protocols seek to set up routes on-demand (i.e., Dynamic Source Routing(DSR))[5], whereas proactive routing protocols seek to maintain a constantly updated topology understanding (i.e., Optimized Link State Routing (OLSR)) [7]. Several approaches based on clustering were developed for the purpose of enhancing the route discovery process and prolonging the network life time.

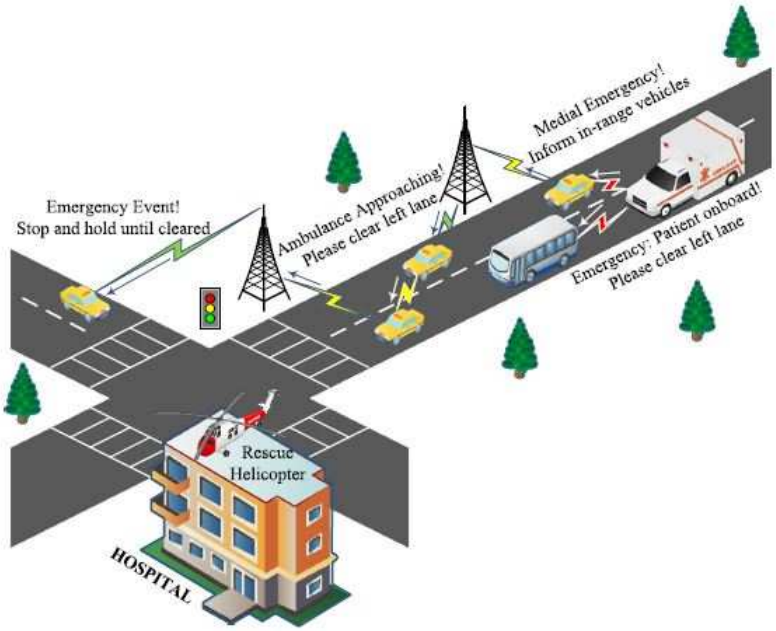


FIGURE 1.1: Emergence messages exchanged in [8]

1.1 Motivation

The concept of integrating wireless communication in vehicles has fascinated researchers since the 1980s [3]. Intelligent transportation systems (ITS) refer to transportation systems that utilize the information and communication technologies to collect and distribute real-time traffic information intelligently in order to improve the quality, effectiveness, and safety of future transportation systems. VANETs are exploited in ITS to provide an efficient wireless connection without the access to any fixed infrastructure.

Designing a stable and reliable communication among the vehicles is very crucial in supporting ITS. Having a stable network is essential in order to have an effective delivery of safety messages exchanged in the network. In the context of multimedia applications, Quality of Service (QoS) should be considered while selecting the routes in order to achieve a stable streaming rate to fulfill the requirements of such applications. Clustering in VANETs is widely implemented to improve and organize the communications among the vehicles [9] [10]. Maintaining stable clusters is very crucial for VANETs to assure a well-connected network. The existing clustering models do not take into consideration the case of links failure due to mobility. For example, some drivers who are self interested may misbehave by over-speeding or under-speeding the road speed limit. Such behavior results in disconnected clusters and thus a disconnected network.

Moreover, VANETs are self-organized and distributed networks. Therefore, establishing reliable communication among the vehicles is highly dependent on the level of cooperation among the nodes. The presence of selfish nodes that are unwilling to cooperate may significantly impact the network performance. These nodes may misbehave and deviate from following the standard medium access protocol (MAC) CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) in order to increase their utilization of the network resources.

CSMA/CA protocol is the primary medium access mechanism of IEEE 802.11 [11] [12]. The channel access is determined by this protocol using a binary exponential backoff rule. This protocol also defines the fundamentals of handling the retransmissions of collided packets. Selfish or greedy nodes manipulate some of the MAC pre-defined parameters to achieve this goal which will severely degrade the overall performance of the network. Numerous works have been proposed to cope with the existence of MAC-layer selfish nodes. These proposals deal with the problem of selfishness from two perspectives: detection-based and incentive-based mechanisms [13] [14] [15]. The proposals suffer from several drawbacks that affect their efficiency and have some undesirable behaviors including ambiguous monitoring, false alarms, and packet collisions.

1.2 Problem Statement

Problem 1 *Selfishness at network layer*

QoS-OLSR is a proactive routing protocol designed that takes into account the quality of service while selecting the MultiPoint Relay (MPR) nodes. This protocol is based on a clustering algorithm that elects a set of cluster heads. Hence, dividing the network into clusters where each cluster head selects a set of optimal nodes called MPR nodes that connect clusters with each other. In fact, some vehicles decide to act selfishly and do not participate in the routing process by being disconnected from the network after being selected as MPRs. This reduces the network connectivity and increases the number of control messages needed for selecting the new set of MPRs. It is clear that these problems disrupt the formation of stable clusters, and hence a well-connected network.

Problem 2 *Selfishness at MAC layer*

Some vehicles may adopt a greedy or selfish behavior to increase their channel access probability. They deliberately manipulate some of the MAC predefined parameters. They can employ some techniques in order to modify some of the parameters defined in CSMA/CA. In CSMA/CA nodes competing for the channel wait for backoff interval before transmission to avoid collisions. The selfish nodes mainly manipulate the waiting time interval to wait for less time. This allows them to acquire the channel for a longer time and increase their throughput accordingly. The existence of such behavior has a severe effect on the performance of the normal nodes and the overall performance of the network as well. In addition, adopting such behavior at the MAC layer can propagate to the upper network layer and affect the routing mechanism negatively as it can prevent the relay nodes from forwarding the packets to the designated destination.

1.3 Objectives of the work

The main purpose of this research is to mitigate the effect of the selfishness behavior encountered at the Network and the MAC layers. This can maintain a well-connected network and ensure a fair resource utilization among the vehicles. The objectives of this research can be summarized as follows:

- Maintain the stability of the clusters in case of link failures due to mobility.
- Regulate the MAC-layer cooperation in the presence of selfish nodes.

1.4 Research Approaches and Contributions

In this thesis, the performance of a cluster-based OLSR routing protocol called QoS-OLSR is improved under the presence of selfish nodes. This protocol is

based on a clustering algorithm that elects a set of cluster heads. Hence, dividing the network into clusters where each cluster head selects a set of optimal nodes called MPR nodes that connects clusters with each other. We propose a new cluster-based protocol based on the Intelligent Water Drop algorithm (IWD) which is referred to as IWD-QoS-OLSR. This protocol is an enhanced version of the QoS-OLSR protocol that can improve the network connectivity and overcome the problem of MPRs disconnection due to mobility. The protocol consists of three main parts: cluster formation, MPR selection, and MPR failure management. The cluster formation is done using the same algorithm proposed in the QoS-OLSR. The IWD algorithm is used to launch a proactive discovery to select the best path connecting the clusters with each other. The selection of MPR set is implemented in a way to satisfy the Quality of Service (QoS) requirements and the mobility constraints of the network in order to improve the connectivity, reliability, and stability of the network. Thus, the IWD is able to select long-living paths to assure the stability of the clusters. Moreover, by exploiting the IWD algorithm, a failure management algorithm is introduced to deal with link failure cases caused by the misbehaving nodes. This algorithm allows the cluster heads to select alternative MPR set once an MPR is disconnected from the network.

To guarantee a proper implementation of the routing protocol, we consider the problem of selfish vehicle at the MAC layer. In fact, some nodes aim to transgress the MAC standard protocol IEEE 802.11 to increase their channel access probability and their throughput by modifying some of the parameter defined by the protocol. Such greedy behavior negatively affects the services availability of the genuine nodes and causes a Denial of Service (DOS) for them. Therefore, in this research, we develop a game theoretical mechanism that encourages the rational nodes to behave normally under the threat of retaliation. The game model mainly motivates the vehicles to be cooperative since mutual

cooperation provides the welfare for the entire network while refraining to cooperate would result in a loss for all of them. As a result, the selfish nodes are motivated to cooperate if they aim to maximize their obtained payoff.

Our proposed approach is an adaptive CSMA/CA protocol based on adapting the contention window of the cooperative nodes to assure fairness in the network. This protocol is based on tit-for-tat strategy [16] to deal with the problem of selfish nodes. In this research, we present three existing tit-for-tat strategies in the literature which are: (1) Classical tit-for-tat [16] (2) Generous tit-for-tat [16] (3) Reputation based tit-for-tat [15]. However, these strategies are not resilient to packet collisions. Therefore, this rises the need for developing two new collaborative based tit-for-tat strategies identified as: (1) Group Reputation based tit-for-tat and (2) Cooperative detection based tit-for-tat.

The contributions of the thesis can be summarized as follows:

- Overcoming the high mobility of the vehicular environment by improving the connections among the clusters in the network using IWD algorithm.
- Having a reliable MPR failure management algorithm to mitigate the effect of selfish nodes.
- Proposing two new collaborative tit-for-tat strategies that are immune to the ambiguous monitoring caused by collisions.
- Regulating the MAC-layer cooperation and motivating the selfish nodes to behave normally.

1.5 Research Publications

- D. Al-Terri, *et al*, Cooperative Based Tit-for-Tat Strategies to Retaliate Against the Greedy Behavior of Vehicles in VANETs, submitted to the Journal of network and computer applications.

- D. Al-Terri, H. Otrok, H. Barada, M. Al-Qutayri, R. M. Shubair, Y. Al-Hammadi, Review on IEEE 802.11 MAC Misbehavior in ad-hoc networks, in: Graduate Students Research Conference (GSRC), 2016.

- D. Al-Terri, H. Otrok, H. Barada, M. Al-Qutayri, R. M. Shubair, Y. Al-Hammadi, Qos-olsr protocol based on intelligent water drop for vehicular ad-hoc networks, in: International Wireless Communications and Mobile Computing Conference (IWCMC), IEEE, 2015, pp. 1352–1357.

- D. Al-Terri, H. Otrok, H. Barada, M. Al-Qutayri, R. M. Shubair, Y. Al-Hammadi, Q-DSR protocol in vehicular ad-hoc networks, in: Innovations in Information Technology (IIT), IEEE, 2015, pp. 162–165.

- D. Al-Terri, Enhanced Approach for DSR Based Routing Protocol, in: Graduate Students Research Conference (GSRC), 2015.

- D. Al-Terri, Distributed Intelligent Transportation Management System, in: Information and Communication Technology Research Forum (ICTRF), 2014.

1.6 Thesis Organization

The remainder of the thesis is organized as follows:

In Chapter 2, the main concepts that form the thesis are highlighted and reviewed. The chapter includes an overview of: VANETs, Routing, Intelligent water drop optimization algorithm, Medium Access Control (MAC) IEEE 802.11 protocol, MAC-layer greedy behavior and repeated game theory. In addition, related work in the fields of routing, MAC-layer misbehavior detection and reaction are presented.

In Chapter 3, we propose a clustered OLSR routing protocol based on Intelligent water drop algorithm. The main elements of the protocol are clarified

along with an illustrative example showing how the proposed protocol works. Finally, the performance of the proposed protocol is evaluated through the use of simulations where the simulation parameters and the achieved results are demonstrated.

In Chapter 4, a game theoretical motivational mechanism is proposed to regulate the MAC-layer cooperation in the presence of selfish nodes. The game model is based on the tit-for-tat strategy where multiple strategies are evaluated and analyzed via simulations. In Chapter 5, we conclude the work in this thesis and outline some future directions.

Chapter 2

Background and Related Work

2.1 Introduction

This chapter introduces the fundamental concepts that form this research. First, an overview about vehicular ad-hoc network technology and its applications is presented followed by a general idea about the routing protocols in ad-hoc networks. Moving to our proposed model, we overview the the Intelligent Water Drop algorithm that form our clustering model. This algorithm is responsible for a proactive discovery to select the best set of nodes that connect the clusters with each other.

Moreover, the thesis tackles the problem of the presence of MAC-layer selfish nodes in the network. Therefore, a background on how the medium access control (MAC) protocol works and how nodes can adopt a selfish behavior is illustrated. In order to cope with the existence of such nodes, we propose a repeated game theoretical model to regulate the MAC layer cooperation. Hence, a review of repeated game theory is highlighted to show how the interaction among the vehicles can be modeled. Finally, a summary of the related work in the fields of routing, selfishness in CSMA/CA, and game theory is presented along with the importance of our contributions.

2.2 Vehicular Ad hoc networks

2.2.1 Definition

Ad-hoc networks are self-configuring networks in which the nodes can communicate with each other without a base station [17]. Ad hoc networks have attracted attention due to their flexibility, low cost, and ease of deployment. They are widely used in the military and the commercial fields. Vehicular ad-hoc networks (VANETs) are wireless ad-hoc networks where the vehicles or the road side units (RSU) represent the nodes of the network. These nodes communicate with each other in order to exchange data among them for the purpose of information inquiry or distribution [18]. VANET allows vehicles that are few hundred meters apart from one another to communicate, which generates a wider range network [4]. Vehicular communication is supported by standard called dedicated short range communications (DSRC) which is a type of Wi-Fi. VANET specifically uses the IEEE 802.11p wireless access in vehicular environment (WAVE) standards in the licensed frequency band at 5.9 GHz [19] [20].

2.2.2 VANET structure

The structure of the WAVE system consists of two major components which are the Road-side units (RSUs) and the On Board units (OBUs) as shown in Figure 2.1

Road-side Units

RSUs, called central controllers are static components that provide Internet connectivity to the OBUs mounted on vehicles [21]. The distribution of the RSUs is highly dependent on the used communication protocol. For example, some protocols demand RSUs to be located along the road side while others demand the placement of the RSUs on dedicated locations such as intersections. RSUs

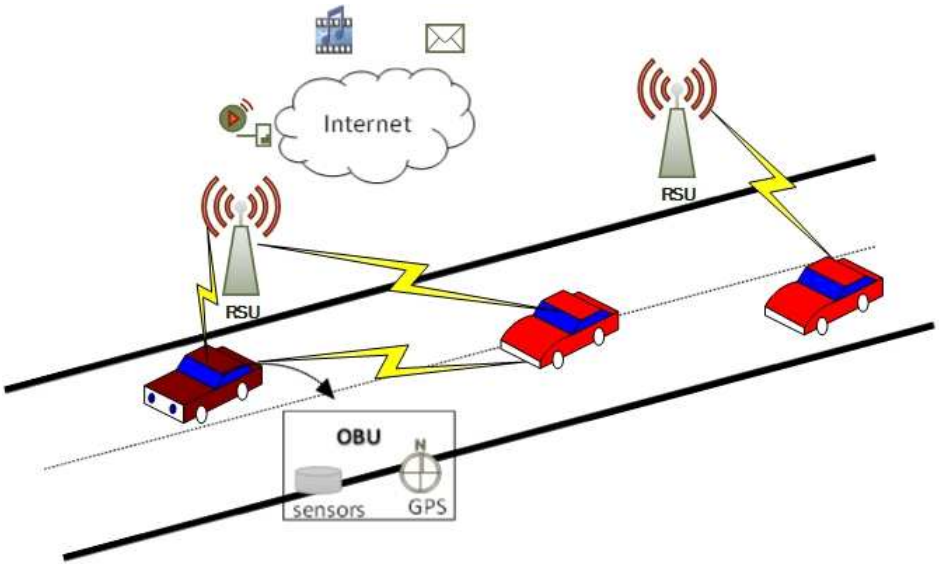


FIGURE 2.1: VANET structure

play a vital role in extending the communication rang in the network. They can re-distribute the information to other OBUs or to the other RSUs in order to forward to another OBUs. Those units exchange the data among each other via specific radio channel known as control channel (CCH) [4] [22].

On Board Units

The OBUs are mounted on the vehicles in order to exchange the information with other OBUs or with other RSUs. OBUs are equipped with central processing unit (CPU) to run the designed protocols, mobile radio module, warning devices and sensors to measure various parameters. In addition, the OBUs contain a Global Positioning System (GPS) since most of the applications supported by VANETs depend on the geographical positions of the transmitter and receiver [23] .

2.2.3 Vehicular Communication Infrastructure (VCI)

VANETs do not depend on fixed infrastructure for the messages dissemination since they are designed to provide ubiquitous connectivity to the mobile users.

The Vehicular Communication Infrastructure of VANETs can be classified as: Inter-vehicle communications and Vehicle-to-roadside communication.

Inter-vehicle communications (IVC)

This type of communication is also called Vehicle to Vehicle communication ($V2V$) shown in Figure 2.2. In IVC, the traffic is transmitted either in multicast or broadcast mode. There are two types of forwarding the messages either by naïve broadcasting or by intelligent broadcasting. In naïve broadcasting, periodic messages are transmitted by the vehicles in which a vehicle ignore the message if it received from a vehicle behind it. However, it broadcasts the message if it is received from a predecessor vehicle. In intelligent broadcasting, the broadcasted messages are limited to the messages related to an emergency event in order to reduce the total number of transmitted messages [22].

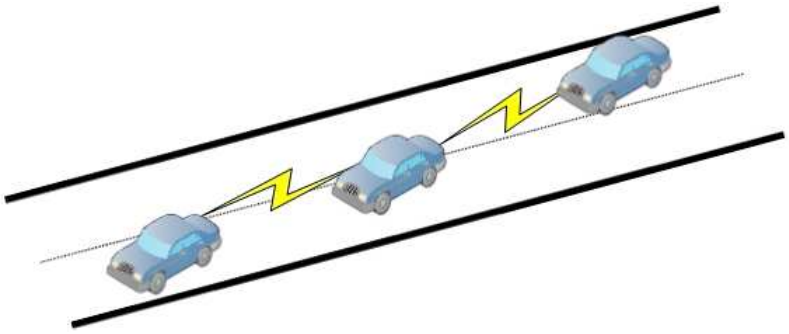


FIGURE 2.2: Inter-vehicle communications

Vehicle-to-Roadside communication (VRC)

This type of communication is also called as Vehicle to Infrastructure ($V2I$) . VRC shown in Figure 2.3 is established between the RSUs and the vehicles in which the RSUs can broadcast/receive messages to/from all the nearby vehicles [24]. The messages that are broadcasted by the RSU are related to the road instructions such as road speed limit. In this case, the RSU determines the appropriate speed limit according to its internal database information and traffic

conditions and will periodically broadcast this information to the vehicles coming within its communication range[22].

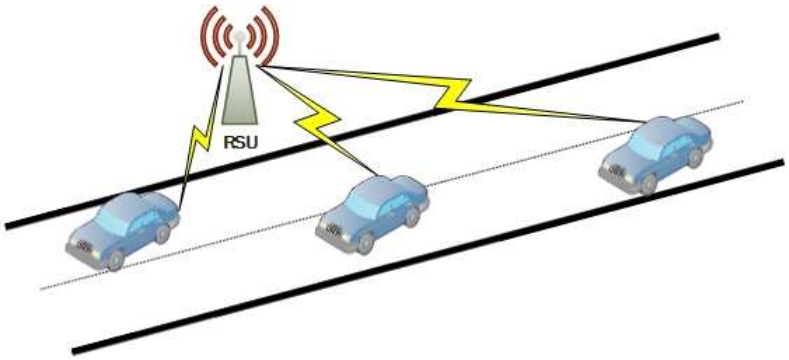


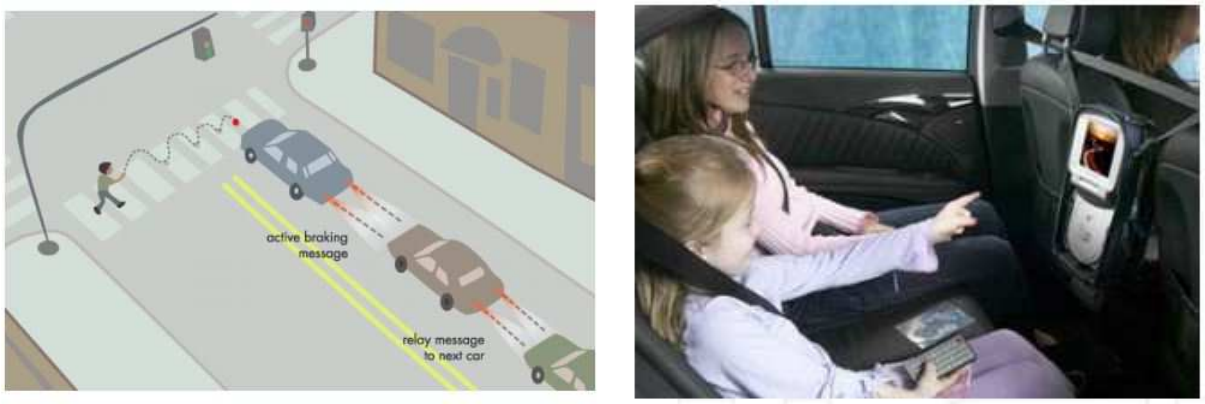
FIGURE 2.3: Vehicle-to-roadside communication

2.2.4 VANET applications

VANETs can support numerous distributed applications such as:

- Active road safety applications: These applications are considered to be the major applications that are used as components of Intelligent Transportation Systems (ITS). These applications allow the vehicles to exchange warning messages among them to avoid dangerous situations that may cause collisions. Examples of warning messages include curve warning, lane changing assistance, traffic signal violation warning, and road condition warning. Those applications require both the vehicle to vehicle and the vehicle to roadside communications [25] [6]. Figure 2.4(a) shows an example of such applications. In the figure, when a vehicle tries to avoid a frontal collision (i.e., by using the brakes), it should disseminate this information to the vehicles behind it.

- Public services applications: Vehicular networks can support the reachability of the public services such as police and emergency units when collisions occur. This can be done by identifying the collision location and



(A) (B)

FIGURE 2.4: (a) VANET safety applications [25] (b) VANET entertainment applications ([25]

selecting the route that has less traffic in order to reach the destination faster [3].

- Driving efficiency applications: These applications are used to update the drivers about the road conditions so they can select alternative routes in case one of the routes is facing an accident. They can also provide the driver with important information needed while driving such as the weather condition.

- Entertainment and business applications: These applications allow vehicles to share multimedia services among each other such as downloading music or movie (see Figure 2.4(b)). Passengers can also play interactive games with other passengers while they are traveling. As an example of business applications, VANETs can facilitate the payment of parking fees [3].

2.3 Routing in VANETs

Routing refers to the process of selecting the best path from the source node to the destination node. Routing in VANETs is very challenging due to the high

mobility of the nodes and the frequent changes in network topology. There are several routing protocols proposed for VANETs. The same routing protocols used for Mobile ad-hoc network (MANET) can be used for VANET with some modifications due to environment compatibility [3]. Routing protocols in ad-hoc networks are classified into two major categories: reactive and proactive routing protocols.

2.3.1 Reactive Routing Protocols

Reactive routing protocols are called on demand routing protocol as the route discovery takes place when it is needed. They can reduce the network traffic as the flooding takes place only when needed. They can also save the bandwidth as there are no periodic transmitted messages among the nodes. However, reactive routing protocols suffer from high latency in finding the route to the destination. AODV (Ad Hoc On Demand Distance Vector) and DSR (Dynamic Source Routing) routing protocols are examples of Reactive routing protocols [6] [26] [27].

Dynamic source Routing (DSR)

A Dynamic Source Routing (DSR) protocol is an on-demand routing protocol in which a Route Request packet is generated by the source in order to find the path to the destination [28] [29]. DSR is composed of two main components: route discovery and route maintenance.

- Route discovery: is only used once the source node (S) attempts to find the route to the destination node (D). This procedure is mainly based on flooding the network with Route Request (RREQ) as shown in Figure 2.5. When an intermediate node receives the RREQ, it adds its ID and flood the packet. Upon the reception of the RREQ to the destination node (D),

it uses the stored IDs to direct the reply packet (RREP) to the source (S). Therefore, the Source node uses the RREP packet to determine the route to the destination.

- Route maintenance: this procedure takes place once a route error is discovered. This means that when a route is not valid anymore, a route error packet (RERR) is generated and sent to the source node (S). Therefore, once the RERR is received, the source node (S) removes the invalid route from the buffer and starts a new route discovery process.

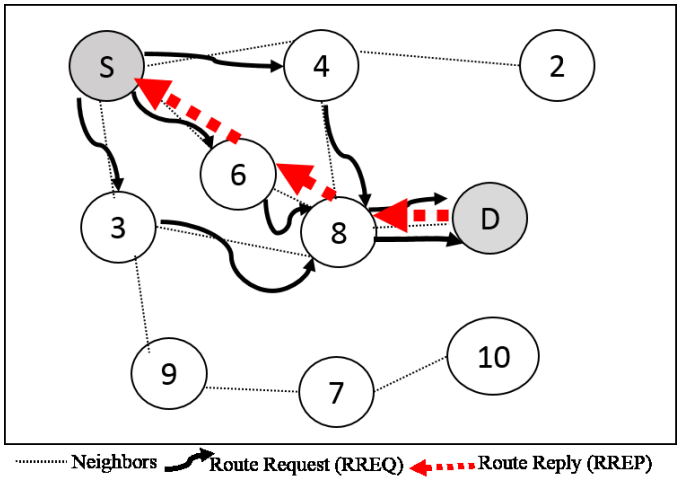


FIGURE 2.5: DSR protocol

2.3.2 Proactive Routing Protocols

In proactive routing protocols, the network topology is distributed among the nodes in the network by exchanging periodic control messages. The protocol is based on a routing table that connects each node to other nodes in the network. Nodes share their table with their neighbors, so any change in the network topology will be distributed to the nodes to update their tables. This will lead to low latency in reaching the destination but it will consume bandwidth due to the periodic flood of messages. OLSR (Optimized link state routing) protocol is an example of a proactive routing protocol [6] [30].

Optimized link state Routing (OLSR)

It is a type of proactive routing protocol which means that the route to any destination is known immediately when it is needed. OLSR is an optimization of a pure link state protocol. In pure link state protocol, each node maintains a link with all of it neighbors. However, in OLSR, each node has a link with specific neighboring nodes called MPR (multiPoint relay). Having a declared link with specific nodes (MPRs) not with all the neighbors will reduce the number of the control messages. It will also reduce the number of retransmitted packets of a node as its MPRs are the only nodes who will retransmit the broadcasted messages [7]. OLSR protocol works in a distributed manner. It does not require a reliable transmission for its control messages as each node keeps sending its control messages periodically. It also does not require the delivery of the control message to be in order as each message contains a sequence number that is used to order the information on the receiver side. Each node that is included in the route uses the recently updated information to route the packets. The following attribute of OLSR makes it suitable for routing in mobile ad-hoc networks [31]:

- Protocol functionality: Each node should know its one-hop neighbor and a list of its two-hop neighbors that are a one-hop away from its neighbors to be able to determine its set of MPRs [31].

- MultiPoint Relay : As mentioned previously, the OLSR uses a set of nodes called MPRs to be responsible for routing its packets. Each node selects a set of its neighbors to be MPRs that can connect it with the 2-hop away nodes in the network [31]. In other words, the MPRs will connect that node to other nodes in the network that are not in its transmission range as shown in Figure 2.6.

- Neighbor Declaration : Each node should know its neighbors and the status of the link with that neighbor; either unidirectional or bidirectional.

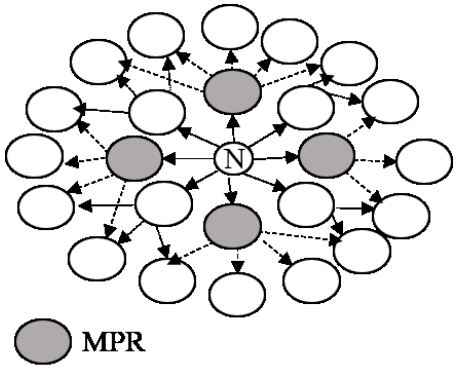


FIGURE 2.6: MPR selection

This can be achieved by exchanging HELLO messages periodically with its neighbors. The broadcasted HELLO messages contain the neighbors with their link status. Therefore, each node can know its 1-hop and 2-hop away neighbors and based on this knowledge it determines its set of MPRs. Each node constructs a routing table that contains the one hop neighbors, the 2-hop neighbor that can be reached by those 1-hop neighbors and the list of selected MPRs. It also has a sequence number that is incremented once the list of MPRs is updated. The list of MPRs is updated in the case of a change in the neighborhood or in the 2-hop neighborhood sets. The MPR nodes communicate with each other using Topology Control (TC) messages. In the TC message, the list of MPR selectors is shared among the MPR nodes in order to build the topology table. Each node builds the topology table that includes the MPRs of other nodes in order to construct the routing table [31].

2.4 Intelligent water drop optimization algorithm

The Intelligent Water Drop (IWD) algorithm is a warm-based optimization algorithm inspired from observing the water drops flowing in the rivers. It has been used to solve multiple optimization problems [32] [33] [34] [35]. IWD [36]

mimics the real behavior of the natural river in finding the optimal or near optimal paths. In fact, the Water Drops (WDs) are created with two main features, namely velocity and soil which are changing during the WD lifetime. The WD begins with an initial velocity and zero amount of soil. During its trip from the source to the destination it removes from the bed some soil and gains some speed. The speed of the WD is non-linearly proportional to the inverse of the soil between two locations. Therefore, a path with less soil lets the WDs flow faster than a path with more soil. The soil that the WD gathers during its trip is removed from the path joining the source and destination. The amount of carried soil by the WDs is inversely proportional to the time needed to pass between the two locations. According to this, the time needed is proportional to the WD velocity and inversely proportional to the distance connecting the two locations. Moreover, the paths that are used with more WDs will have less soil on their beds and will attract more WDs to flow on it. In general, the WD uses a mechanism to select a path to its next location. This mechanism is based on selecting the path that has less soil compared to the path with more soil. Thus, the probability of selecting the next path is inversely proportional to the amount of soil in that path. IWD algorithm has attracted the researchers due to it adaptive nature that makes it suitable to solve many optimization problems such as: Vehicle Routing Problem [35], Traveling Salesman Problem (TSP) [36] and ad-hoc networks routing [33].

In order to illustrate how the IWD algorithm works, we present how this algorithm is applied to solve the TSP. TSP is a routing problem in which a map of cities is provided to a salesman and he is required to visit the entire cities by visiting each city only once except the original city. The TSP can be represented by a graph (N, E) in which N is the set of n cities and E is the set of edges that denote the distance between a pair of cities. Applying the IWD algorithm to this problem, each WD starts its tour from a randomly selected city. Each link of the

edge set E has an amount of soil. Thus, a WD can move between the nodes (cities) and update the amount of soil on the links while completing its tour. The soil value for shorter routes would be less than other routes. Therefore, the WDs prefer the routes with less soil left on them. This process is repeated for a particular number of iterations in order to find the best solution. The flow chart of this process can be depicted in Figure 2.7

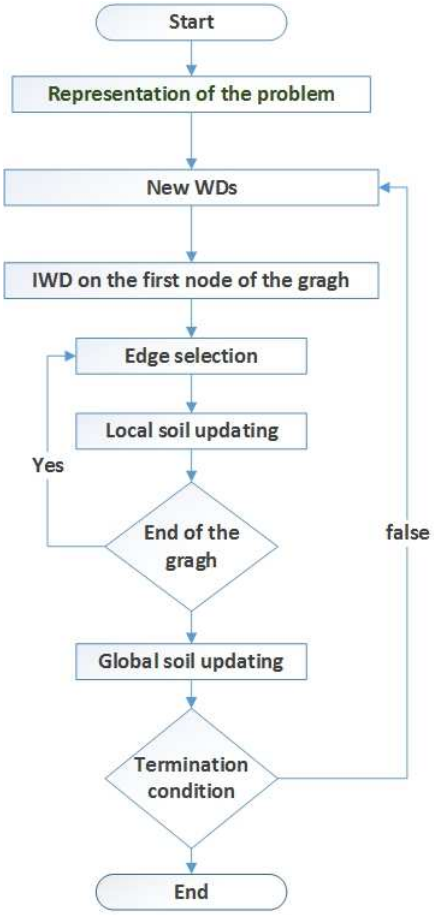


FIGURE 2.7: Intelligent water drop algorithm

In this thesis, we exploit this algorithm to connect the clusters with each other in a cluster-based QoS-OLSR protocol. Incorporating this algorithm with the routing protocol increases the connectivity level of the network.

2.5 Ant Colony Optimization

Ant Colony Optimization (ACO) is a probabilistic approach that is used to solve optimization problems. It is a swarm based algorithm that is inspired from the behavior of ant in searching for a source of food. In nature, ants start the search for food by moving randomly to a different food sources. During their search process, when a food source is explored, ants deposit chemical substance called pheromone in order to mark this place for the other ants. Thereafter, In order for the other ants to be able to find this place, they will follow the smell of this substance and hence the shortest path will be reinforced by more ants. The path with higher pheromone value is selected and will get marched by more ants repeatedly [37].

An important element of this algorithm is the pheromone evaporation as it can be used to select the future solutions. The efficiency of this element is as follows. As the time passes the pheromone starts to evaporate in which the path goodness that is represented by the pheromone value will be disappearing unless they they are reinforced by more ants. Therefore, the shortest path will have more pheromone than the other paths.

2.6 IEEE 802.11 Architecture

IEEE 802.11 is the de-facto standard for wireless local area networks (WLANs). It is a set of media access control (MAC) and physical layer (PHY) specifications for implementing WLANs [38]. IEEE 802.11 uses Distributed Coordination Function(DCF), which is a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) MAC protocol as the fundamental mechanism for medium access method [39].

2.6.1 DCF

The DCF protocol defines how the medium is shared among the stations. Figure 2.8 demonstrates how the DCF protocol works. When a station wants to initiate a transmission, it senses the channel to check whether it is idle. There are two ways to identify whether the medium is idle or not [39].

- By checking the presence of a carrier at the physical layer (Layer 1).
- By checking the network allocation vector (NAV). When NAV = 0, the station can attempt a transmission.

The network allocation vector (NAV) is a virtual carrier-sensing mechanism. In the IEEE 802.11 frames, the header includes a duration field that specifies the time needed for the station's frame exchange to take place. NAV gets updated from the frames transmitted through the medium where it is considered as a counter. A station decrements its NAV counter until it reaches zero. When it reaches zero, the medium is considered to be idle.

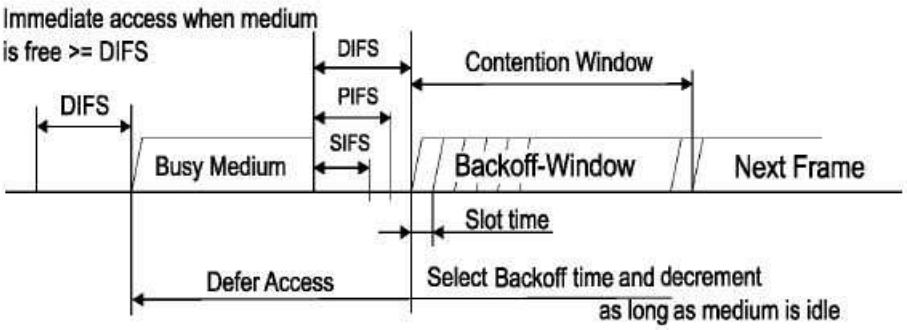


FIGURE 2.8: DCF in IEEE 802.11 [40]

If the channel is sensed idle for a period of Distributed Inter Frame pace (DIFS), the station can emit its transmission. However, if the channel is sensed busy, the station should defer its transmission until the channel is sensed idle for DIFS interval. After that, the station initializes a counter called

backoff counter by selecting a random interval (backoff interval) which is measured in time slots to schedule its transmission [41]. The backoff counter is decremented by one once the channel is sensed idle for a period of DIFS. If the channel is sensed busy, the backoff counter is suspended and it can be resumed when the channel is sensed idle for a DIFS period. Once the backoff timer reaches zero, the station can start its transmission. The backoff time is randomly chosen from the interval (0,CW), where CW stands for Contention Window. CW is an integer value that can be determined by the adopted physical layer characteristics, CW_{min} and CW_{max}. For the first transmission attempt, CW= CW_{min}, each time a collision occurs CW is doubled until it reaches CW_{max}. Upon a successful transmission, CW is reset to its minimum value [42].

2.7 Overview of IEEE 802.11p

IEEE 802.11p is an approved amendment to the IEEE 802.11 standard which is proposed for Wireless Access in Vehicular Environment (WAVE) [11]. In this section, we explore the specifications of the IEEE 802.11p related to the physical layer and MAC sub-layer.

2.7.1 Physical (PHY) Layer in 802.11*p*

The PHY layer of IEEE 802.11p is built on top of the IEEE 802.11a but with some modifications to make it compatible with the vehicular environment. The IEEE 802.11p supports transmission rate ranging from 3 to 27 Mbps. 802.11p operates at 5.9 GHz and it adopts the orthogonal frequency division multiplexing (OFDM) transmission technique [8]. The bandwidth of a single channel in 802.11p is 10 MHz which is half of the channel bandwidth used in IEEE 802.11a. This reduction was essential to the vehicular environment where the nodes are moving with relatively high speed. Therefore, if the signal bandwidth is

high, the delay spread of multiple paths would be higher and the inter-symbol-interference is exacerbated accordingly. Therefore, a bandwidth of 10 MHz is more practical for vehicular networks [11].

2.7.2 MAC Sublayer in 802.11*p*

The MAC layer of 802.11p uses Enhanced Distributed Channel Access (EDCA) which is an enhanced version of Distributed coordination function (DCF) that is used in the standard IEEE 802.11. EDCA is a prioritized contention based channel access mechanism [8]. It supports four access categories according to the type of the traffic. The access categories are (AC: Access category): Background traffic (AC_0), Best Effort traffic (AC_1), Video traffic (AC_2) and Voice traffic (AC_3). Each access category uses different Arbitration Inter-Frame Space period (AIFS), CW_{min} and CW_{max} as shown in Table 2.1 Each AC works as independent DCF station with Enhanced Distributed Channel Access Function (EDCAF) to contend for transmission using its own EDCA parameters. EDCA is implemented by using a new inter frame space called AIFS which is considered as an extension to the backoff procedure implemented in the DCF. EDCA uses the CSMA/CA protocol as channel access method [43]. Figure 2.9 illustrates how the EDCA protocol works.

TABLE 2.1: Default EDCA parameters in IEEE 802.11p

AC	CW_{min}[AC]	CW_{max}[AC]	AIFSN[AC]
AC_BK	CW_{min}	CW_{max}	9
AC_BE	(CW_{min} +1)/(2-1)	CW_{max}	6
AC_VI	(CW_{min} +1)/(4-1)	CW_{min}	3
AC_VO	(CW_{min} +1)/(4-1)	(CW_{min} +1)/(2-1)	2

CW_{min}= 15 and CW_{max}= 1023

In EDCA, if a station wants to transmit a packet, it senses the channel. If the channel is idle for a period of AIFS, it can send the packet. The waiting time

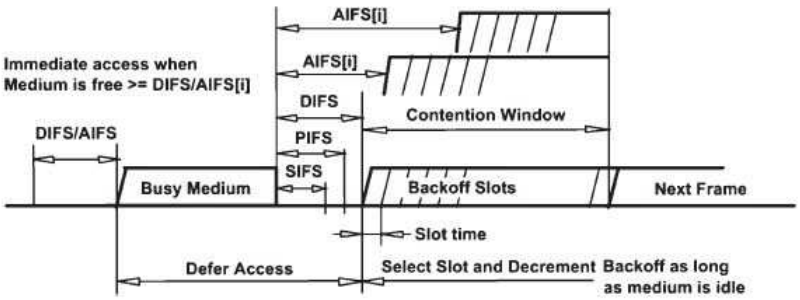


FIGURE 2.9: 802.11p [11]

AIFS[AC] can be derived from (2.1):

$$AIFS[AC] = AIFSN[AC] \times aSlotTime + aSIFSTime \tag{2.1}$$

where:

– AIFSN[AC]: is the value assigned in the EDCA parameter table (Table 2.1)

– aSlotTime: is the duration of a slot time.

– aSIFSTime: is length of SIFS (Short Inter Frame Space)

In the case of a busy medium, the station should implement the backoff procedure. The back off procedure in EDCA is similar to the one illustrated in DCF section. The default EDCA parameters are depicted in Table 2.2. According to the version of IEEE 802.11p, CW_{min}= 15 and CW_{max}= 1023 [11].

TABLE 2.2: Default Parameters of IEEE 802.11p

Parameter	Value
Slot time (σ)	13μs
SIFS	32μs
CW_{min}	15
CW_{max}	1023
Number of retry limit	7

2.8 Overview of IEEE 802.11 Misbehavior

Nodes are called greedy or selfish if they employ some techniques in order to reduce the backoff value which increases their access probability and allows them to gain more bandwidth share. Some common strategies adopted by selfish nodes are highlighted below [44] [45]:

- *CW Cheating Strategy*: Following this strategy, a selfish node manipulates the backoff value by selecting a value in the range of [0, α(CW)] instead of the range [0, CW], where $0 < \alpha < 1$. In this way, the selfish node obtains a lower CW and a lower BO value accordingly. Thus, the BO value interval of a selfish node is less than that of a normal node.

- *Fixing CW Strategy*: In such strategy, a selfish node selects a fixed CW size called CW_f to attain the backoff value from (i.e.,[0, CW_f]). Moreover, upon an unsuccessful transmission, this selfish node refuses to double the contention window as specified in the protocol. Therefore, it will have a higher access probability to the channel in future transmissions.

- *BO Cheating Strategy*: In this strategy, a selfish node does not modify the contention window size and it will follow the binary exponential backoff algorithm defined by CSMA/CA protocol. It will also double its contention window in the case of transmission of collided packets. However, it manipulates the attained backoff value to a lower value. Therefore, the new BO (BO_{new}) value is computed as follows: BO_{new}= γ(BO), where $0 < \gamma < 1$ and BO is the originally chosen value according to the 802.11 standard.

It is worth noting, that all the different strategies lead to a greedy behavior in the network in which the nodes who adopt such behavior gain more bandwidth share. In this thesis, *fixing CW Strategy* is selected to simulate the behavior of a selfish node.

2.9 Game Theory

Game theory is a branch of applied mathematics which is exploited to model the interactions and the conflicts for rational participants aiming to achieve their objectives in long term basis [16]. The motivation behind using game theory is to identify the optimal strategies for all of the players. Game theory has been widely used in several fields due to its great potential to solve different problems. For example, it has been used in social science [46], communication networks [47] cellular networks [48] and wireless ad-hoc networks [49]. In general, each game model consists of three main elements [16]:

- Players: are the decision makers.
- Strategies: are the set of all possible actions.
- Payoff: refers to the gain obtained by the players. It can also represent the motivations of the players.

The symbolic representation of a game model is shown below in 2.2 [16]:

$$G =< N, \{A_i\}, \{P_i\} > \tag{2.2}$$

where:

- N: is the total number of players.
- A: is the action space.
- P: is the payoff.

2.9.1 Nash equilibrium

The players in a game aim to adopt a strategy that maximizes their own payoff. The equilibrium strategies are selected by the players to optimize their individual payoffs [50]. Nash equilibrium is a solution concept of a game consisting

of two or more players [50]. At the Nash equilibrium, none of the players can gain or increase its payoff by a unilateral deviation. In other words, if no player can increase its payoff while other players do not change their strategies, then the chosen strategy set along with the corresponding payoffs constitute a Nash equilibrium.

2.10 Game models

A Non-cooperative game is a type of game in which the players make their decisions independently. Non-cooperative game models can be divided into two main types: static and dynamic games. In static games, the interactions among the players are studied when the actions are taken only in a single period. In contrast, dynamic games analyze the players' actions in multiple periods. In such games, the players can adjust their strategies according to their opponents' previous strategies [16].

2.10.1 Static Game

In this part, we demonstrate an example of a static game using the Prisoner's dilemma game [51]. Prisoner's dilemma game represents the situation in which two criminals suspected to commit a crime are arrested at the same time. The criminals have two strategies to select from: either to confess or not in which they are isolated and hence, should select their strategies independently. The investigator suggests the following:

1. If one of the criminals confesses against the other one, the confessed criminal will be freed ($payoff = 0$) while the offender will stay for 5 years in the prison ($payoff = 5$).

2. If both criminals confess, they will both be punished by staying in the prison for 2 years ($payoff = 2$).

3. If both criminals refuse to confess, they will have a reduced sentence due to the lack of evidence by being in the prison for 1 year ($payoff = 1$).

The outcome of this step is summarized in Table 2.3. Now, the question is: *how should both criminals behave in such situation?* Each one of them will think as follows:

- If the other criminal confesses, then the best strategy is to confess since being 2 years in prison is better than 5 years.
- If the other criminal refuses to confess, then the best strategy is to confess since getting freed is better than staying 1 year in prison.

TABLE 2.3: Payoff matrix of Prisoner's Delimma

	Confess	Don't Confess
Confess	$(2,2)$	$(0,5)$
Don't Confess	$(5,0)$	$(1,1)$

2.10.2 Repeated Game

In repeated game theory, the game is played repeatedly over multiple iterations. Therefore, the players can decide their future actions based on the other player's actions in order to achieve their goal. The player in a game model can either cooperate (C) or defect (D). In general, a repeated game model is restricted with the following conditions:

$$C_i > C_{i-1} \quad \textit{and} \quad D_i > D_{i-1} \quad \forall i \tag{2.3}$$

$$D_i > C_i \quad \forall i \tag{2.4}$$

$$C_{n-1} > D_0 \tag{2.5}$$

where: C is the cooperative payoff, and D is the defect payoff.

According to the first condition the payoff of a player *i* cannot be increased while more opponents are defecting regardless of its chosen action. For the second condition, any player *i* can have a higher payoff if *i* follows the defect strategy, given that the other players do not change their actions. Regarding the third condition, the obtained payoff when all players are cooperative would be higher than the payoff when all of them are defecting.

Referring to the Prisoner's Dilemma example presented in the previous subsection, if the game is played only once, the best strategy for both players is to confess regardless of the other player's strategy (i.e., staying 2 years in prison). However, if the game is played repeatedly, each player can observe the action of its opponent. Thus, the players may get a better result by following the non-confessing strategy (i.e., both stay 1 year in prison).

2.11 Related Work

In this section, a literature review is done on the following topics: Routing in VANETs, Selfishness in CSMA/CA and repeated games in MAC.

2.11.1 Routing in VANETs

Routing based on Multi-point relay nodes

The Optimized Link State Routing (OLSR) Protocol [31] has been modified in order to be compatible with mobile ad-hoc networks (MANETs). The authors in [52] proposed a routing protocol that is built on top of the original OLSR protocol, based on clustering in which the network is divided into clusters and each cluster has a cluster head responsible for selecting a set of MPRs. Those heads along with the MPRs will be charged for the routing process in the network. This model can reduce the overhead by reducing the duplicate transmissions introduced in the original OLSR.

In [53], the authors designed a QOLSR protocol on top of OLSR. QOLSR is Quality of service (QoS) routing protocol that basically satisfies the QoS constraints in selecting the optimal paths as it depends on the available bandwidth and the delay metrics to select the MPR nodes.

The QoS-OLSR was presented in [30] by improving the QOLSR protocol. This was done by implementing a clustering algorithm and adding the residual energy metric along with the bandwidth and the connectivity level metrics while selecting the cluster heads and the MPR nodes. This leads to prolong the network life time and assure a good QoS among the links.

VANET-QoS routing protocol was proposed for vehicular networks [54]. This protocol considers the QoS of the paths along with the mobility parameters in the path selection process. This improves the stability and the reliability of the network.

Routing based on IWD algorithm

Intelligent Water Drop (IWD) has been used to improve the routing protocols. The authors in [55] used the IWD algorithm for routing in MANET. The simulation results show that IWD has better performance than other routing protocols in reducing the overhead while finding the optimal path.

The QoS multipath routing algorithm IWDRA [33] was proposed for MANETs used IWD to find the path in terms of the QoS for MANET. The packets gain the properties of the IWD among nodes.

The above described routing protocols designed for MANET are not compatible with the vehicular environment, hence, they are inadequate to be used in VANET. This because these approaches do not consider the link failure case while selecting the paths. To the best of our knowledge, there is no work that exploits IWD for the MPR selection in cluster-based QoS-OLSR protocol.

2.11.2 Selfishness in CSMA/CA

There are numerous of papers published in the literature that addressed the issue of selfishness in the CSMA/CA protocol in wireless networks. We review some of the major ones below.

In [13], the authors proposed a detection scheme to detect the selfish nodes in WLAN (infrastructure based networks) called DOMINO. The detection scheme does not require any modifications to the standard IEEE 802.11 protocol to detect the selfish nodes. It is capable of detecting the manipulation of other MAC protocol parameters such as: (a) the modification of the backoff value and the Clear To Send (CTS) frame delay (b) and an increase of the NAV (Network Allocation Vector) time. Similarly, the authors in [15], proposed a detection technique to detect the selfish nodes which modify the contention window (CW) value to obtain higher throughput in public hotspots environments. The detection scheme is based on counting the number of Acknowledgment(ACK)

frames in order to make the detection decision. In addition, the authors proposed a motivational approach based on game theory in order to motivate the nodes to follow the standard protocol.

In [56], the authors proposed a detection scheme to detect the misbehavior nodes in ad-hoc networks. The detection scheme is based on modifying the original IEEE 802.11 protocol to detect the selfish nodes. The key idea of the proposed scheme is pre-defining the backoff value for the sender node and monitoring whether the node will follow the provided value or not.

In [14] [57], an analytical model for real time misbehavior detection in wireless network is proposed. The analytical model is based on Markov chain to systematically study the non-parametric cumulative sum test. The proposed mechanism is supposed to detect the selfish nodes without any prior knowledge of the statistics of the selfish misbehavior.

In [43], the authors proposed a detection algorithm to detect the greedy behavior in VANETs. The detection technique does not require any modification to the original IEEE 802.11p and is executable by any node in the network. It can detect the manipulation of the EDCA parameters. The detection algorithm is based on the linear regression approach and the watchdog supervision tool.

In [44], the authors proposed a real time detection scheme for multi-hop ad hoc networks. The detection scheme does not require a prior knowledge of the selfish nodes in order to detect them. However, it requires few number of samples to come up with the detection decision. In addition, the detection decision by an individual observer is sent to a local cluster head in order to make the final decision about a suspicious node.

However, all the above techniques do not include any penalty scheme for the selfish nodes. In this case, more nodes are still motivated to misbehave. In this paper, we propose a reaction model against the selfish nodes to encourage them to behave normally.

2.11.3 Repeated games in MAC

The problem of cooperation among non cooperative participants was comprehensively introduced by Axelrod in [58]. Axelrod highlighted multiple strategies proposed in the literature including the well-known tit-for-tat strategy. According to the proposed performance analysis, the tit-for-tat outperformed the other strategies. Game theory has been widely used to model the interactions and the conflicts among the participants (such as [49] [59] [60] [46] [47]). In medium access control mechanism which is a distributed approach, the nodes in the network are contending to access the wireless channel. Therefore, a game theory framework can be implemented in the MAC layer of wireless networks to ensure a fair share among the nodes.

In [61], the author proposed a non-cooperative repeated game model to investigate the greedy behavior in CSMA/CA. The paper defined a strategy called CRISP that provides a fair bandwidth share among the nodes and prevent the backoff attack in mobile ad-hoc networks.

In [62] [63] the authors proposed a game theoretic model for infrastructure networks that defines the MAC protocol. Each station maintains a desired ratio between uplink and downlink throughput in which it tunes its access probability in a way that its payoff is maximized. As the station's payoff is determined by the throughput of the access point (AP), the station will be enforced to behave cooperatively.

In [15], a Reputation tit-for-tat strategy in a repeated game model frame is proposed in order to enforce the selfish nodes to comply with the CSMA/CA protocol. The proposed strategy is an enhanced version of the classical tit-for-tat in which it has an immunity to the misinterpretation errors and it allows the cooperative nodes to maximize their payoff. The game is applied to a group of nodes trying to communicate with the same access point.

In [64], a game theoretic contention window adjustment approach is proposed. This approach determines an optimal value for the CW under the heavy load conditions. Thus, each node can select its CW value autonomously in order to improve the overall network performance in which there will be a compromise between the obtained throughput and the MAC frames delay.

However, there is still a need for a robust strategy that can overcome the problems of the ambiguous monitoring and false alarms. To the best of our knowledge, there is no work that exploits the tit-for-tat strategy to regulate the MAC-layer cooperation among the nodes in VANETs.

2.12 Conclusions

In this chapter, we explained VANETs, Routing, IWD algorithm, IEEE 802.11 architecture and repeated game theory that constitute the core of the thesis. Thereafter, we present the related work in the fields of Routing in VANETs, Selfishness in CSMA/CA and repeated games in MAC. According to the conducted literature review, the QoS-OlSR protocol lacks an efficient algorithm that can compensate for any relay nodes disconnection due to mobility. This disconnection can be caused due to the misbehaving nodes that do not cooperate with other nodes. Concerning the greedy behavior, the existing approaches have several limitations that make them inefficient to deal with such behavior such as: ambiguous collisions and false alarms. Chapter 3 and Chapter 4 addresses the aforementioned issues.

Chapter 3

Relay Recovery Technique to Mitigate the Network-Layer Selfish Behavior

3.1 Introduction

This chapter addresses the problem of MultiPoint Relay (MPR) node disconnection due to mobility in VANETs using the cluster-based Quality of Service Optimized Link State Routing (QoS-OLSR) protocol. The QoS-OLSR protocol uses MPR nodes to establish communication among the clusters. MPR disconnection represents a major challenge in VANETs, due to the frequent change in the network topology. Furthermore, the disconnection could occur due to the misbehavior of some nodes which decide to over-speed or under-speed the road speed limit to not cooperate in the packet forwarding process [65]. This results in a break between the direct links connecting two clusters. Hence, increases the percentage of disconnected clusters. This also increases the overhead due to the increase of the exchange of the messages needed to re-select the new MPR nodes. Consequently, the performance of the routing protocol will be weakened as it adversely affects the connectivity level of the network.

In order to highlight this problem, a simulation of the effect of disconnection

due to mobility in a network is conducted using MATLAB. The simulation is carried out using 50 vehicles moving in a highway road in which their speed ranging between 60 Km/h and 120 Km/h. Figure 3.1 plots the the percentage of directly disconnected clusters along with different percentage of disconnected nodes. According to Figure 3.1, it is clearly shown that as the percentage of nodes disconnected due to mobility increases, the percentage of directly disconnected clusters increases. This is due to the disconnection of the MPR nodes that connect the clusters with each other. Therefore, this arises the need for having an MPR failure management algorithm (Recovery algorithm) that can compensate for any MPR disconnection cases and keep the network well-connected.

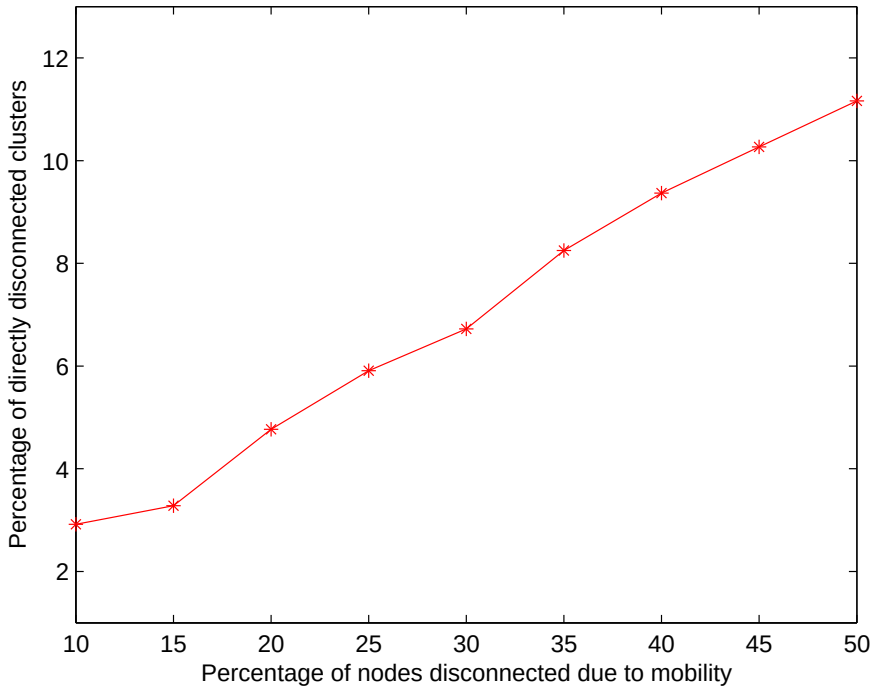


FIGURE 3.1: Effect of mobility on the network: The percentage of disconnected clusters

Thus, in this chapter we propose a new cluster-based protocol based on the intelligent water drop algorithm (IWD) which is referred to as IWD-QoS-OLSR [66] that is capable of (1) selecting the best set of MPR in terms of QoS (2) dealing with the MPR disconnection as it selects alternatives to assure a connected

network (3) maintaining a reliable MPR failure management process.

In summary, our contribution is a novel QoS-based routing protocol that is based on the intelligent water drop algorithm that is capable of:

- Overcoming the high mobility of the vehicular environment by improving the connections among the clusters in the network using IWD algorithm.
- Reducing the overhead by having a failure management algorithm that allows the selection of another set of MPRs in case of link failures due to mobility.
- Having a reliable MPR failure management algorithm that is time invariant.

The remainder of this chapter is organized as follows. Section 3.2 describes the proposed IWD-QoS-OLSR protocol including its three main components: cluster formation, MPR selection, and MPR failure management. In Section 3.3, we provide an illustrative example to show how the proposed protocol works. Moving to Section 3.4, the simulation parameters, and the achieved results are demonstrated. Finally, in Section3.7 the conclusions of this chapter is presented.

3.2 IWD-QoS-OLSR Protocol

This section describes the IWD-QoS-OLSR protocol that is used to improve the network connectivity. The proposed protocol consists of three main components: QoS-based clustering, MPR selection algorithm and MPR failure Management algorithm. The protocol can be summarized as follows. Firstly, a set of nodes is elected as cluster heads based on the local maximum QoS value. Then, using the intelligent water drop algorithm, the MPR nodes that connect the clusters with each other are selected. Finally, an MPR failure management

algorithm is proposed in order to compensate for any link failure situation. Table 3.1 illustrates the routing protocol notations.

TABLE 3.1: Notations

Symbol		Meaning
n	:	Set of nodes in the network
N_1	:	Set of 1-hop neighbors
N_2	:	Set of 2-hop neighbors
$QoS(i)$	:	Quality of Service Metric of a node i
$BW(i)$	:	Available bandwidth of node i
$N(i)$	:	Number of neighbors for node i
MPR(i)	:	MPR set for node i
S:	:	Source cluster-head
D:	:	Destination cluster-head
P	:	Set of all paths leading to D

3.2.1 Cluster formation

Quality of Service(QoS) metric

Building the clusters mainly depends on the Quality of Service (QoS) value for each node. The chosen QoS metric attains a tradeoff between the QoS of the selected route and the network mobility, since it combines connectivity level, bandwidth, distance ratio and velocity ratio of the node as illustrated in Table 3.2. Including the connectivity level of the node increases the coverage of cluster head/MPR and considering the available BW will enhance the reliability. Moreover, distance and velocity ratios are added to the QoS to ensure the stability of the network.

The Velocity Ratio of a node i is calculated using:

$$VelocityRatio(i) = \frac{Velocity(i)}{Expectedvelocity}. \tag{3.1}$$

where: Velocity (i) is the velocity of node i, and Expected velocity is the road expected velocity.

TABLE 3.2: Quality of Service function

Quality of Service metric function
Let i be a node in the network, define $QoS(i)$ = Quality of Service Metric of a node i $BW(i)$ = Available bandwidth of node i $N(i)$ = Number of neighbors for node i $VelocityRatio(i)$ = Ratio of velocity for node i $DistanceRatio(i)$ = Ratio of remaining distance for node i **define**: $QoS(i) = N(i) \times BW(i) \times \frac{DistanceRatio(i)}{VelocityRatio(i)}$

Incorporating the Velocity ratio in the calculation of QoS has the following two objectives:

- Group the vehicle with convergent velocity scale into clusters.
- Ensure that the elected/selected head/MPR has a reasonable velocity.

The calculation of the Distance Ratio is depicted in Algorithm 1. Global Position System (GPS) is used to provide the distance parameter needed in the deployed system. Adding the distance ratio to the calculation of the QoS value has two objectives:

- Group the vehicle with convergent residual distance into clusters.
- Ensure that the elected/selected head/MPR has a considerable distance to traverse.

Cluster-Head Election algorithm

In order to form the clusters, an election process takes place. The algorithm works as follows. Firstly, the nodes broadcast HELLO message containing their QoS values. Then, each node votes for its neighbor that holds the highest QoS or it may vote for itself if it holds the maximum QoS value among its neighbors. The node uses a special HELLO message called Election message to broadcast

Algorithm 1 Distance Ratio calculation

Input: Max distance, Current position (i)
Output: DistanceRatio (i)

Initialization:
Max distance: is the maximum distance between the source and destination.
procedure DISTANCE RATIO CALCULATIONS
    for each node $i \in n$ **do**
        Current position (i) = current position of node i
        Residual distance (i) = Max distance - Current position (i)
        DistanceRatio (i) = $\frac{Residual distance(i)}{Max distance}$
    end for
end procedure

their votes. Hence, the elected head and the nodes that vote for that head form a cluster. The cluster heads are considered as a relay node that forwards the packets for their electors to other nodes. They should update their Topology Control (TC) message to include their electors. The algorithm of head cluster election is illustrated in Algorithm 2. Here, it should be noted that some modifications need to be done on the classical HELLO message defined in the OLSR protocol [67]. The first modification is to add a H flag to denote that the node has been selected as a cluster head. The second one is to add a new neighbor type in the link code which is H_NEIGH. This flag indicates that a neighbor has been elected as a cluster head. The Election message is used to indicate for which node the neighbor has voted for.

Algorithm 2 Cluster-Head (CH) election algorithm

Input: N_1, QoS
Output: MPR

procedure CH-ELECTION
    for each node $i \in n$ **do**
        Broadcast HELLO message to its neighbors
        let $N_1(i)$ is the set of 1-hop away nodes from node i
        let $k \in N_1(i) \cup \{i\}$ s.t
        QoS (k) = max $\{QoS(j) | j \in N_1(i) \cup \{i\}\}$
        MPR (i) = $[k]$
    end for
end procedure

3.2.2 MPR selection using IWD

IWD optimization

In this part, the MPR selection based on IWD algorithm [36] is illustrated. IWD [34] is a nature-inspired optimization algorithm imitates the behavior of the river flowing in nature. In this research, we exploit this algorithm to establish a reliable communication among the clusters in a cluster-based QoS-OLSR protocol. There are dedicated packets called IWD-HELLO that are responsible for gathering the information needed for each path, then select the best path. The best path would be selected according to the amount of soil it has, in which the path with the least soil is the preferable one. Here, the IWD packets have the same properties of the natural water drops (WDs) which are the velocity and the soil.

MPR selection algorithm using IWD

After determining the cluster heads, each head is responsible for selecting a set of MPR nodes to connect it with other cluster heads. The proposed MPR selection based on IWD algorithm is illustrated in Algorithm 3. A Modified version of the HELLO message is used for MPR selection algorithm. Here, the IWD-HELLO message is used and it is called IWD-Req in the Drifting Phase while later it is called IWD-Rep in the Finding Route Phase. The IWD-HELLO packet format is described in the following subsection.

In order to illustrate how the algorithm works, suppose that a source cluster-head (S) wants to find the MPR nodes that connect it with destination cluster-head (D). Source (S) will send IWD-Req packet to its neighbors. Each IWD-Req packet has a list of visited node V_c that is initially empty V_c (IWD-Req) = {}. Each IWD-Req has initial velocity sets to initVel and has initial soil sets to zero (initSoil = 0). Once the IWD-req packet is created and broadcasted by the source

node (S), the soil of the link is updated. IWD-Req packet will be converted into IWD-Rep once it reaches the destination. Then the IWD-Rep packet will follow the same path of IWD-Req packet. Therefore, each node will compute the path preference probability P_r to determine the best path and select its corresponding MPR node. Finally, the MPR route that would connect the two cluster heads is defined as follows: MPR-Route = [S MPR(S) MPR(D) D].

Algorithm 3 MPR Selection algorithm

Input: QoS
Output: MPR

initialization
initSoil = 0
Vc (IWD) ={}

Phase 1 – Drifting phase

procedure DRIFTINGPHASE
  for each source S **do**
    Broadcast IWD-Req 2-hop away
    for each Path (P) **do**
      Find soil (P)
      Compute $P_r(P) = \frac{f(soil(i,j))}{\sum_{k \notin vc(IWD)} f(soil(i,k))}$ s.t
      $f(soil(P)) = \frac{1}{soil(P)}$
    end for
  end for
end procedure

Phase 2 – FindingRoute phase

procedure FINDINGROUTE PHASE
  for each destination D **do**
    $MPRset(D) = y|\ y \in x|\ P_r(x) = \max\{P_r(z)|z \in P\}$
    Send IWD-Rep 2-hop away
  end for
end procedure

Phase 3 – Final phase

procedure FINALPHASE
  for each source S **do**
    $MPRset(S) = \{y|\ y \in x|\ P_r(x) = \max\{P_r(z)|z \in P\}\}$
  end for
  MPRRoute= [S, MPR(S), MPR(D), D]
end procedure

3.2.3 MPR failure management algorithm/Recovery algorithm

An MPR failure management algorithm is proposed in order to deal with the case of MPR disconnection due to mobility. Applying such algorithm reduces the overhead as there will be no need for re-selecting new MPR nodes. Algorithm 4 illustrates the failure management algorithm or Recovery algorithm. The algorithm works as follows. Once the cluster head fails to connect with its MPR, it will use the sorted MPR route list that is stored in the cache to select alternative one. The MPR route list is sorted in descending order based on the path preference probability. This means that the cluster head will select the second best path to connect it with the other cluster head. This procedure occurs each time the selected MPR is disconnected until the cache is empty. In this case, the cluster head will run the MPR selection algorithm in order to be connected with the other cluster heads. Hence, the MPR failure management is capable of reducing the overhead generated while running the MPR selection algorithm.

Algorithm 4 MPR Failure Management algorithm

Input: MPR routes of cluster head
Output: MPRset

```
1:  procedure MPR FAILUREMANAGMENT
2:      for each cluster head s do
3:          Sort (MPR routes of s)
4:          if MPR route(s) is invalid then
5:              MPRset(s) = {j| j ∈ Cache(s(2))}
6:              Cache (s) ← Remove j
7:              if Cache is empty then
8:                  MPR selection algorithm( )
9:              end if
10:         end if
11:     end for
12: end procedure
```

3.2.4 IWD messages

There are two main messages that MPR selection algorithm uses: IWD-HELLO and TC messages.

IWD-HELLO message

IWD-HELLO is an extended version of the Hello message. Several modifications are done on the original HELLO message as clarified in Table 3.3 :

- *D* : is used to specify whether the IWD-HELLO in an IWD-Req or IWD-Rep packets. D will have a value of 0 when the packet is an IWD-Req or a value of 1 when the packet is an IWD-Rep.
- *Visited node list* : This field saves the nodes visited by the IWD-Req that directs it to the destination in which the IWD-Rep can use this field to return back to the source.
- *Vel(IWD)* : In this field the updated velocity of the IWD is defined.

TABLE 3.3: IWD-HELLO message used in the MPR selection

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1			
D	Reserved	Htime	QoS value
Link Code		Reserved	Link Message Size
Visited Node List			
Neighbour Interface Address			
IWD-Velocity			
Neighbour Cluster Head Address			
...			

Topology control (TC) message

TC: is a message exchanged among the MPR nodes whenever an update in the topology takes place. Therefore, during the MPR failure management, the cluster-head will exchange TC messages to update the newly selected route.

The TC message will keep track of the link quality and hence update the soil on the path. The TC packet format is shown in Table 3.4.

TABLE 3.4: Topology contro (TC) message

0 1 2 3 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1	
ANSN	Reserved
Updated Soil value	
Advertised Neighbour Main Address	
...	

3.3 Illustrative Example

In this sub-section, an example is illustrated to clarify how the proposed protocol works including the MPR selection and MPR failure management processes.

3.3.1 MPR Selection

Assume that a network consists of 14 nodes as shown in Figure 3.2. According to the QoS value corresponding to each node (see table 3.5), it can be deduced that node 7 and node 10 are elected as cluster heads as they have the maximum QoS value among their neighbors. Then, those cluster heads should find the MPR nodes that connect them to each other. Node 7 has 5 possible paths in order to reach the other cluster head as shown in Figure 3.2 which are either through 5-8, 5-9,6-8, 6-9 or 6-11. Therefore, according to Algorithm 3, the source head (node 7) sends two IWD-Req 2-hop away. During the drifting phase, the IWD-Req in each path gains a speed and carries an amount of soil according to the QoS of that path. Thus, as high as the QoS of the path, the soil left on the path will be less, hence the best path is then chosen. In this case, the path that goes through node 6 and node 9 has the less amount of soil on its bed and will be chosen to connect cluster head-1 and cluster head-2. In addition, the other

routes will be stored in the cache for both cluster heads in descending order according to the path preference probability.

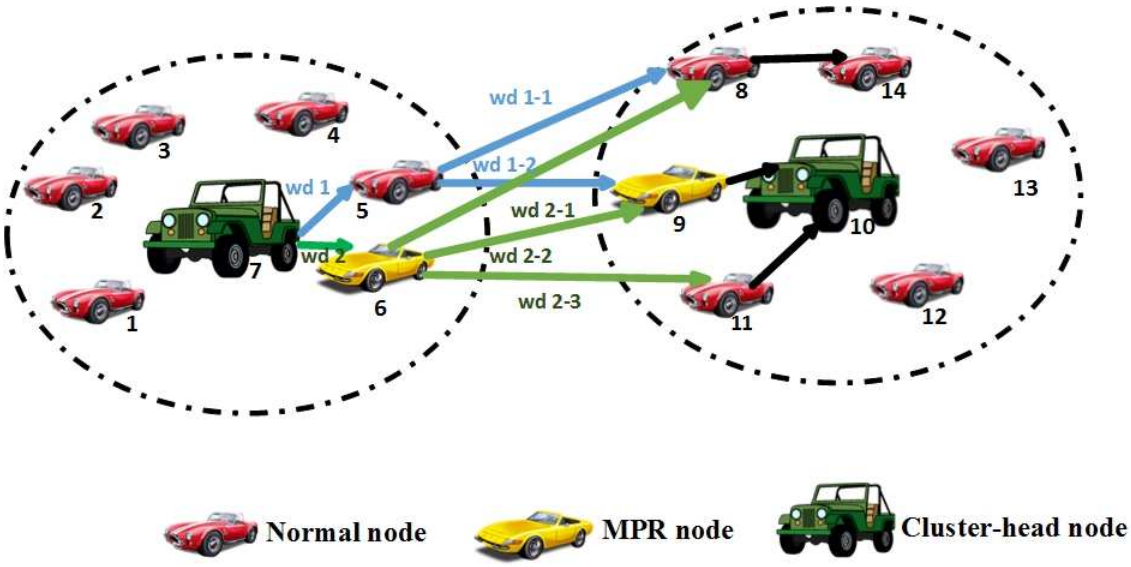


FIGURE 3.2: Illustrative example for MPR selection

TABLE 3.5: Quality of Service QoS value

Node	$n1$	$n2$	$n3$	$n4$	$n5$	$n6$	$n7$
QoS value	370	295.35	200.25	179.45	236.33	400.75	531.15
Node	$n8$	$n9$	$n10$	$n11$	$n12$	$n13$	$n14$
QoS value	300.01	290.55	400.01	199.8	194.8	354.8	254.8

3.3.2 MPR Failure management

Now, suppose that the MPR (node 6) is disconnected from the network due to the mobility factor in Figure 3.3. Using the MPR failure management algorithm described in Algorithm 4 Cluster head-1 can select an alternative MPR immediately to connect it with Cluster head-2 using the MPR routes stored in the cache. In this case Cluster head-1 will choose node 5 to be served as its MPR and then it updates the other cluster head to select node 8 to be its new MPR, hence a TC message is generated to update the new link quality.

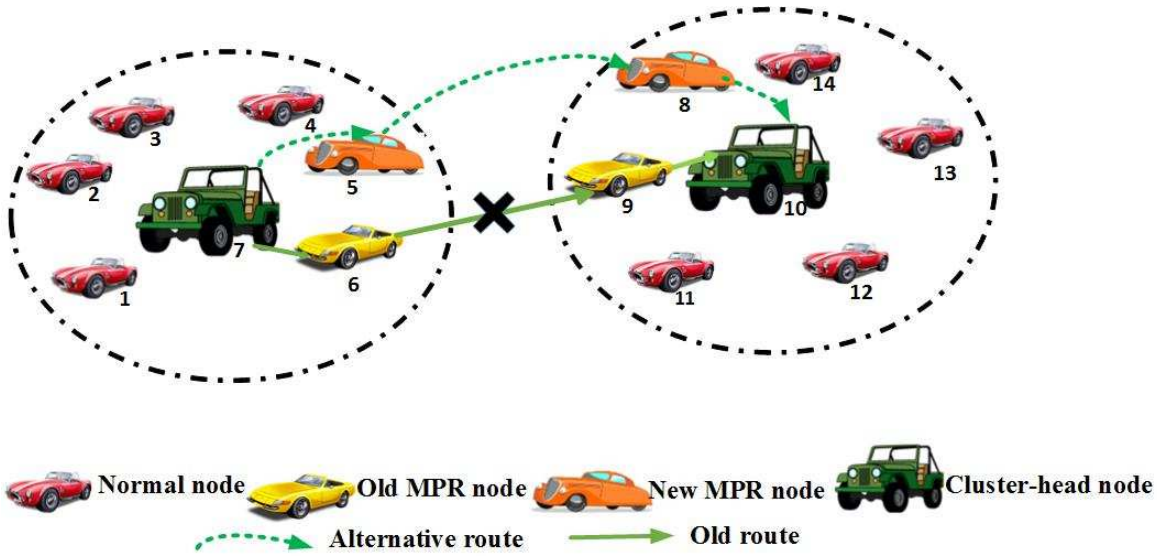


FIGURE 3.3: Illustrative example for MPR failure management

3.4 Simulation Parameters

In this section, the simulation parameters and scenarios are illustrated followed by a discussion of the obtained results.

3.4.1 Simulation Scenario and parameters

MATLAB and Mobisim have been used to simulate IWD-QoS-OLSR and QoS-OLSR protocols. Mobisim is a framework that has been used to simulate the mobility model of mobile ad hoc networks [68][69]. The output trace file generated from Mobisim contains details of the position and the speed for each node at specific time step. This data is then used to build the mobile network. A simulation area of 1000×100 m^2 is used to simulate a set of nodes. For the simulation results, a confidence interval of 95 % has been obtained in order to get precise results by running the simulation 100 times. For each run, the built in random generator function in MATLAB was called. The simulation parameters are summarized in Table 3.6. In order to simulate the behavior of a selfish node,

it has been assumed that the selfish node is disconnected and hence will not be included in any future routing discovery mechanism.

TABLE 3.6: Simulation Parameters

Parameter	Value
Simulation area	1000×100 m^2
Number of nodes	Between 30 and 100
Topology	Freeway (5 Lanes)
Transmission Range	100 m
Speed Range	60 km/h - 120 km/h
Link Bandwidth	2Mbps
Available Bandwidth	Random value in[0..1]$\times$*Link Bandwidth*
Number of simulation runs	100 (95% of confidence level)

3.5 Simulation Results

3.5.1 Comparison with the QoS-OLSR protocol

In this section, our proposed routing protocol is compared with the QoS-OLSR proposed in [30]. The following performance metrics are used to evaluate both protocols.

Percentage of MPRs

MPR is a node selected by the cluster head in order to connect it with other cluster heads. The cluster head is also considered as an MPR since it serves as a relay node for its electors. The number of nodes needed to be selected as MPR is decreased according to the level of connectivity of the selected MPR nodes. Reducing the percentage of MPR nodes will lead to a decrease in the number of exchanged TC messages. Hence, leads to reduce the overhead. Percentage of MPR is the ratio of the number of nodes selected as MPRs to the total number of nodes. Figure 3.4 shows that IWD-QoS-OLSR protocol leads to a reduction

in the number of nodes selected as MPR than QoS-OLSR. This is because the connectivity level for each node is multiplied with the other metrics in the QoS function which means that the node with higher number of connections will be selected as MPR, hence there is no need for a wide set of MPRs. On the other hand, QoS-OLSR divides the BW over the connectivity level which results in selecting a wider range of MPR nodes.

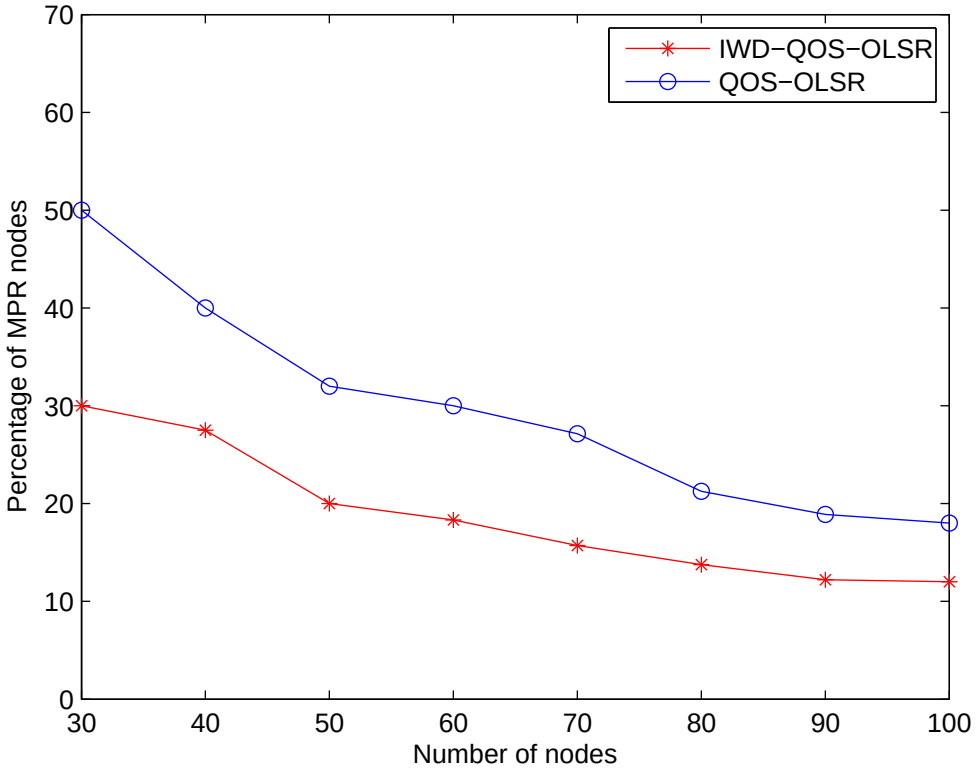


FIGURE 3.4: Percentage of MPR

End-to end delay

End-to-end delay represents the number of needed hops to reach a particular destination. As the average number of hops is less, the probability that a packet is lost decreases. This will increase the packet delivery ratio and the transmission reliability. Using the IWD algorithm will ensure that the shortest path is

chosen which means that in our protocol the end-to-end delay should be decreased. It can be clearly shown from Figure 3.5 that our protocol results in a path with less number of hops compared to the QoS-OLSR protocol.

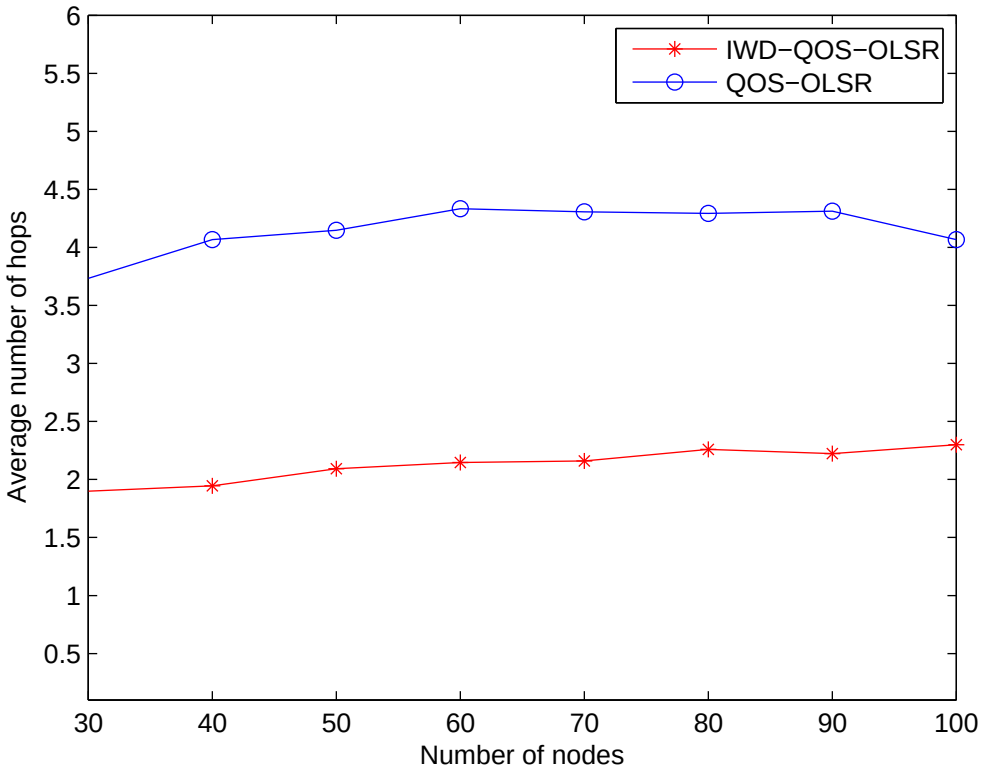


FIGURE 3.5: Average number of hops

Packet delivery ratio (PDR)

Packet delivery ratio is an essential criterion to measure the efficiency of any routing protocol. It is computed by dividing the number of received packets over the number of sent packets. To improve the PDR, the number of received packets should be high and this depends on the connectivity level of the selected path from the source to the destination. In our protocol, the end-to-end delay is decreased and the connectivity percentage is improved. As a result, the PDR is improved as illustrated in Figure 3.6 comparing with QoS-OLSR.

In Figure 3.6, when the number of nodes is 80 a decrease in the PDR occurred. This is due to an increase in the number of hops as depicted in Figure 3.5.

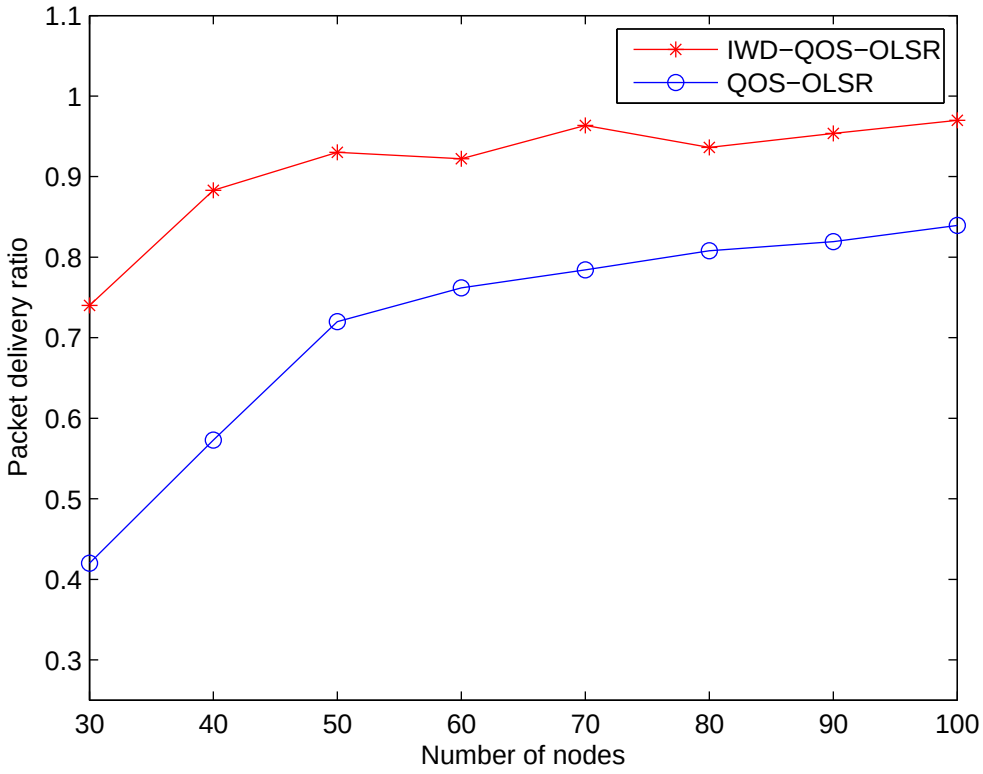


FIGURE 3.6: Packet delivery ratio

Probability of packet loss (PPS)

Probability of packet loss is the ratio between the number of lost packets and the number of transmitted packets. This ratio can be reduced by improving the packet delivery ratio. As stated previously, our protocol improves the PDR which means that PPS is reduced as shown in Figure 3.7.

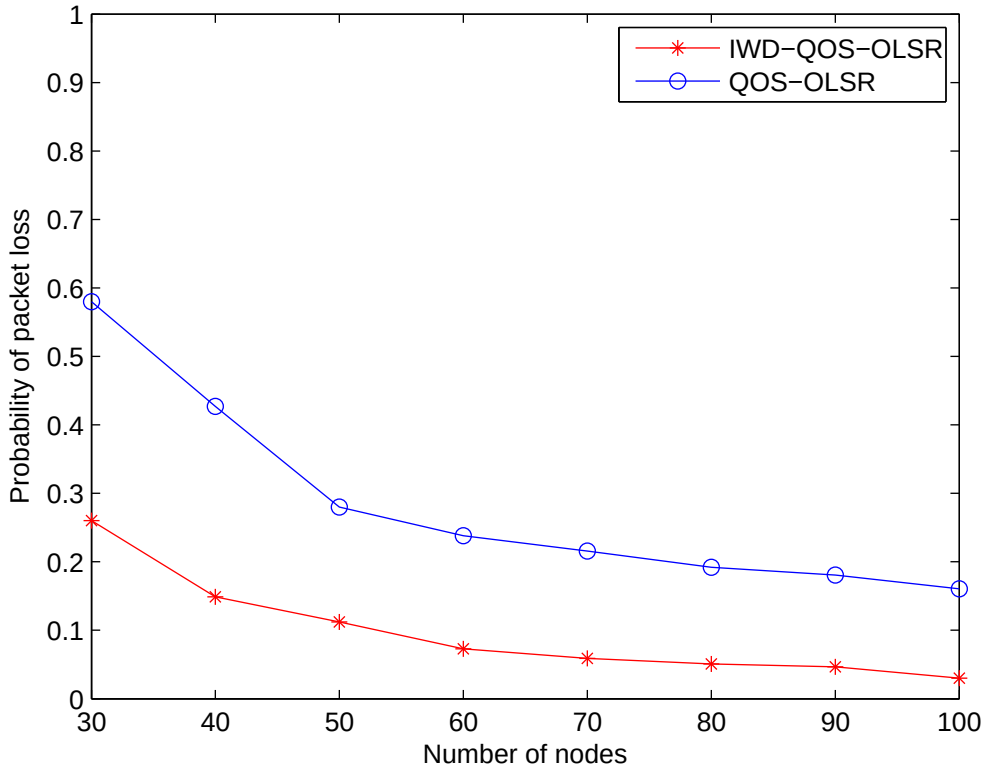


FIGURE 3.7: Probability of packet loss

Bandwidth average difference

Bandwidth average difference is the difference between the bandwidth of the path with the maximum bandwidth and the bandwidth of the selected path. Hence, as this factor decreases, the reliability of the network increases. Table 3.7 presents the percentage average difference for a network of 70 nodes for both protocols: the IWD-QoS-OLSR and the QoS-OLSR, evaluated for different transmission ranges of: 150 m, 200 m and 300 m. From Table 3.7, it can be seen that the average difference between the QoS-OLSR and our protocol is small. This difference is acceptable since in our model the BW is combined with other metrics in order to achieve other aspects for the network such as: connectivity and stability.

TABLE 3.7: Bandwidth average difference

Models	Transmission Ranges		
	150	200	300
IWD-QoS-OLSR	6.77 %	5.4 %	4.8 %
QoS-OLSR	7.1 %	5.12 %	4.3 %

3.5.2 Significance of MPR failure management

As the level of connectivity increases, the percentage of disconnected clusters decreases. In order to show the significance of the MPR failure management algorithm, the performance of the proposed protocol is studied with and without the MPR failure management for a network composed of 60 nodes in which the percentage of the nodes disconnected due to mobility varies from 10% to 50%. It can be clearly seen from Figure 3.8 that having such algorithm in a routing protocol reduces the percentage of disconnected clusters and maintains the network connected.

3.5.3 Comparison with the VANET QoS-OLSR

In this sub-section, the efficiency of the recovery algorithm for the VANET QoS-OLSR proposed in [54] is compared with the recovery algorithm proposed in our protocol. The authors in [54] introduced a recovery algorithm based on ant-colony optimization algorithm [70]. In their model, the ants deposit pheromone on the paths connecting two cluster heads together in which the optimal path will have the highest pheromone value. Thus, the optimal path will be re-enforced by more ants compared to other routes (alternative routes). However, the pheromone evaporates on the other paths as time passes which make them invalid for selection. Hence, when a recovery is required, the pheromone may be evaporated and there will be a need for running the MPR selection algorithm again.

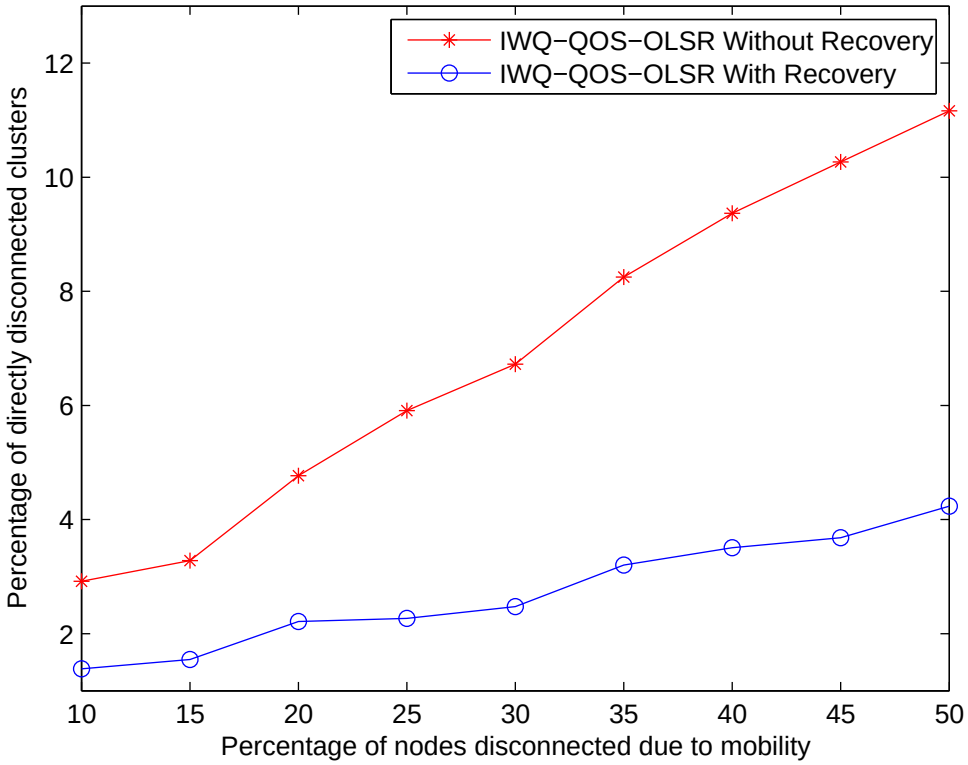


FIGURE 3.8: Percentage of disconnected clusters with and without the recovery algorithm for IWD-QoS-OLSR protocol

The metric that is used to compare both recovery algorithms is the aliveness of the routes. As mentioned previously, recovery algorithm based on ant colony is time variant. In other words, the pheromone on a specific route evaporates if that route is not refreshed by other ants as time passes. Therefore, this is considered as a major drawback in this recovery algorithm. Figure 3.9 shows the percentage of alive routes for both algorithms. It can be deduced that Recovery algorithm based on IWD is more reliable than recovery algorithm based on ant colony.

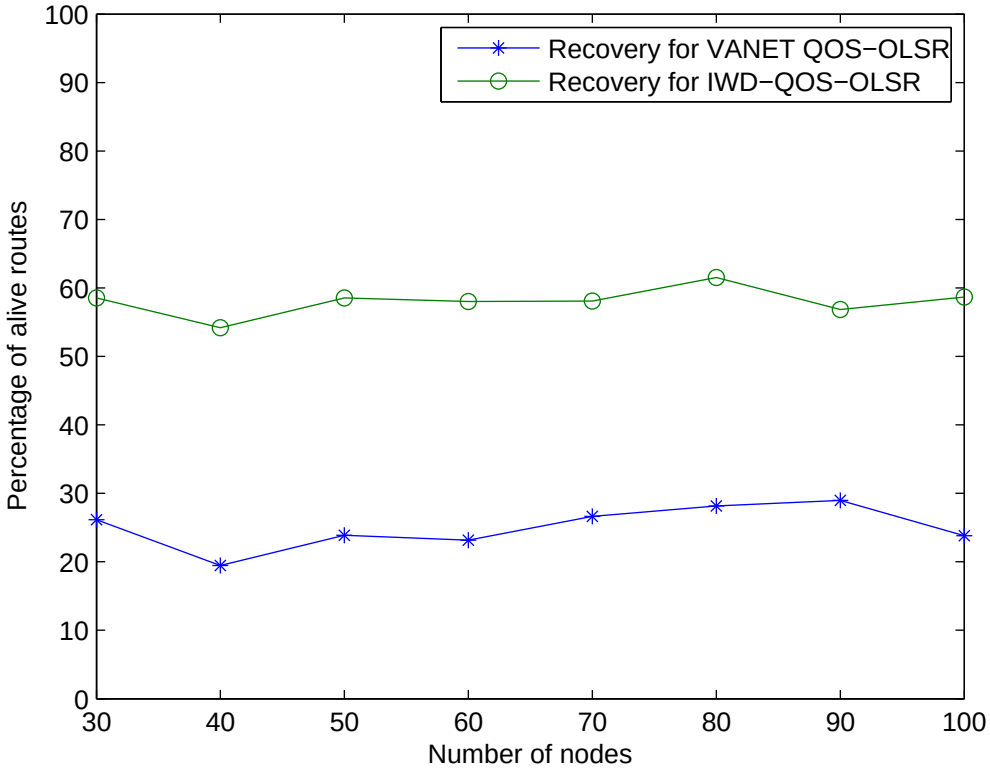


FIGURE 3.9: Percentage of alive routes for the Recovery algorithm based on ant colony and by the Recovery algorithm based on IWD

3.6 Limitation

Although the IWD-QoS-OLSR outer-performed the QoS-OLSR protocol, it suffers from few limitations. The main limitation of IWD-QoS-OLSR is the overhead increase when it is applied to a dense network since it will require a huge increase number of exchanged messages for finding the routes.

3.7 Conclusions

In this chapter, IWD-QoS-OLSR protocol is proposed to improve the connectivity and the stability of the vehicular network. In addition, the proposed protocol is resilient to the presence of the misbehavior nodes which refrain from cooperating in forwarding the traffic. The protocol consists of three main parts:

1) cluster formation 2) MPR selection and 3) MPR failure management. The protocol elects a set of cluster heads based on the local maximum QoS value. Then, by using intelligent water drop algorithm, the MPR nodes that connect the clusters with each other are selected. Finally, a failure management algorithm is proposed in order to compensate for any link failure situation caused by the selfish nodes.

However, mitigating the effect of misbehavior nodes at the Network layer is not sufficient to guarantee a proper implementation of a routing protocol. In fact, a greedy behavior can be adopted by some nodes at the MAC layer resulting in denying the normal nodes to access the channel and hence, disrupting the routing mechanism. Therefore, we propose in Chapter 4 a game theoretical motivational mechanism that can regulate the cooperation among the nodes.

Chapter 4

Cooperative Based Tit-for-Tat Strategies to Retaliate Against the MAC-Layer Selfish Behavior

4.1 Introduction

This chapter discusses the problem of greediness in IEEE 802.11 CSMA/CA protocol in VANETs. CSMA/CA protocol is the primary medium access mechanism of IEEE 802.11 [71] [72]. This protocol is based on a backoff (BO) rule to handle the retransmissions of collided packets. The backoff procedure is implemented as follows. Firstly, the node senses the medium before sending any packet to check whether it is idle or not. If the medium is busy, the node defers its transmission by selecting a random backoff value. The backoff value is selected from the interval [0, CW] where the initial value of the Contention Window (CW) is equal to CW_{min}. At any time a collision occurs, the CW is increased by doubling its value until it reaches CW_{max}.Upon a successful transmission, the CW is reset to its minimum value. After that, when the value of BO reaches zero, the node transmits immediately.

Achieving cooperation in non-managed and decentralized networks such as VANETs is very challenging. Some nodes may purposefully choose to deviate

and show misbehavior at MAC layer in order to obtain more bandwidth [12] [73]. Such a behavior has a catastrophic impact on the network performance as it causes a denial of service for cooperative nodes.

To highlight the aforementioned problem, a simulation is conducted using *NS*-3 in a network comprising 30 nodes. The simulation lasts for 100 seconds. Figure 4.1 plots the received throughput by the selfish nodes for different values of CW along with different percentage of selfish nodes in the network. It is obvious from Figure 4.1 that the selfish nodes can increase their throughput by modifying the contention window to a small value. In addition, the throughput of the selfish nodes increases monotonically as the CW value decreases. Thus, the temptation of transgressing the CSMA/CA protocol is very high for a node in the network.

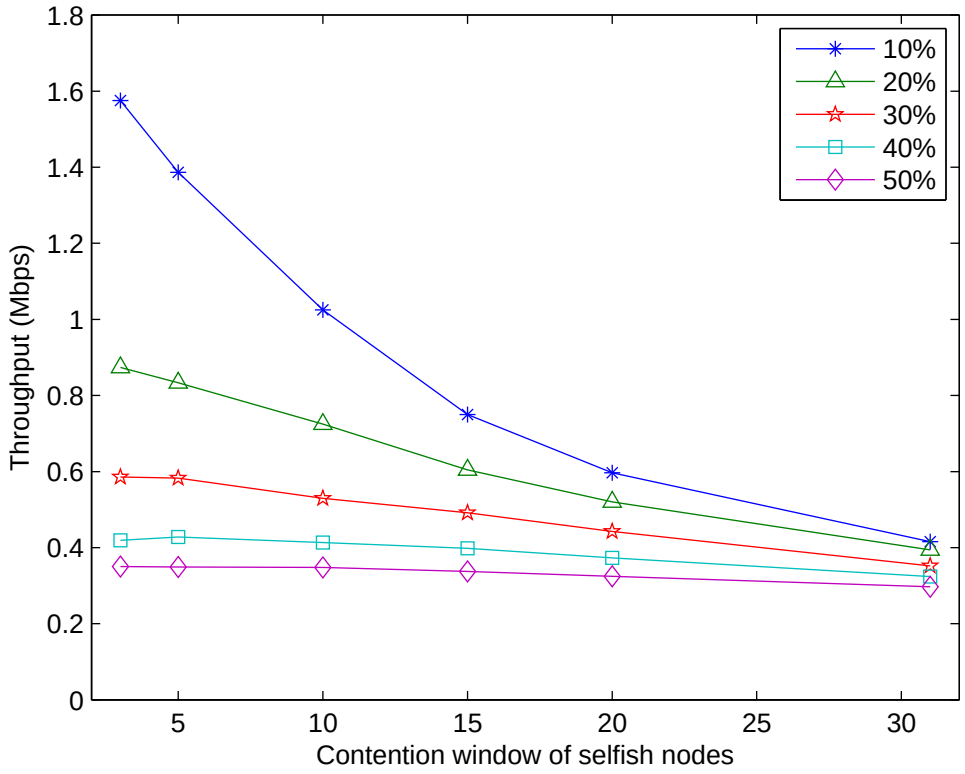


FIGURE 4.1: Different contention window for selfish nodes

Furthermore, an example of the impact of greedy behavior on the network

throughput is demonstrated as depicted in Figure 4.2. This figure compares the obtained throughput for both selfish and normal nodes as the percentage of selfish nodes in the network increases. At time t= 200s, 20% of the nodes starts to misbehave by manipulating their contention window value to be equal to 2. These nodes increase their access probability and gain more throughput as depicted in Figure 4.2 from t = 200s to t = 400s. In this figure, the percentage of selfishness increases by 20% every 200s. Thus, it can be deduced from the figure that as the percentage of selfish nodes increases, the BW share for the normal nodes deteriorates. In addition, as the number of selfish nodes increases, the selfish nodes end up with less throughput (from t= 200s to t= 1000s). As a conclusion, when all the nodes misbehave in the interval [t= 1000s to t= 1200s], they obtain the same throughput but with less value compare to the interval where all of them were behaving normally.

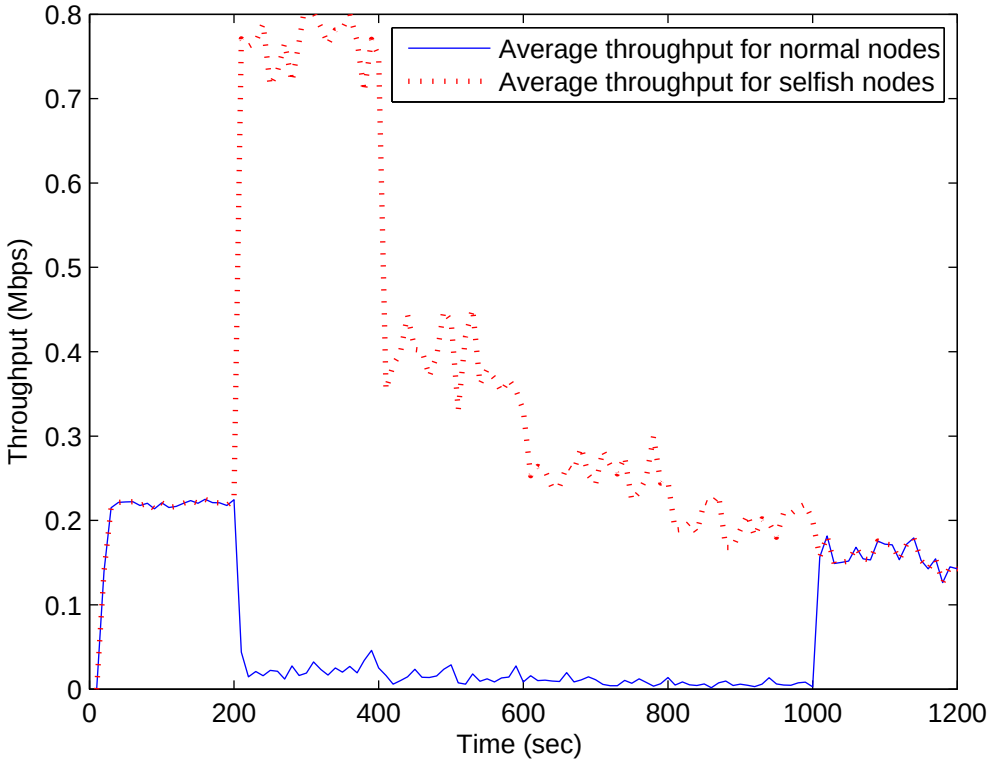


FIGURE 4.2: Impact of selfish nodes on the network throughput

To overcome such a problem, a motivational mechanism is needed to incentivize vehicles to comply with the protocol design in such distributed networks. Classical, Generous and Reputation based tit-for-tat strategies have been proposed to study the interaction among vehicles. However, these strategies are not able to enforce the cooperation since they rely on one-to-one monitoring that is not resilient to collisions. Thus, we propose two collaborative-based tit-for-tat strategies that are identified as Group Reputation and Cooperative Detection tit-for-tat strategies. Both strategies are able to improve the misbehavior detection decision and thus impose the MAC-layer cooperation in VANETs.

In summery, our contributions are two new collaborative based tit-for-tat strategies that are:

- Immune to the ambiguous monitoring caused by collisions.
- Able to motivate the selfish nodes to cooperate under the threat of retaliation.

The remainder of this chapter is organized as follows. Section 4.2 describes the proposed game model. Moving to Section 4.3, the proposed tit-for-tat CSMA/CA protocol is elaborated along with the simulation scenarios. Finally, Section 4.6 concludes the chapter.

4.2 Game theoretical selfishness prevention model

Game theory [64] [60] is a formal study of the interaction between multiple participants making individual decisions. Each game consists of three main elements: players, strategies, and payoff function. In VANETs, the vehicles are independent nodes that make decisions whether to cooperate or not. Hence, the issue of the cooperation among the nodes at the MAC layer can be modeled using game theory. In this research, we use a repeated game model to address

the issue of selfishness. Selfish users seek to increase their payoff at the expense of normal nodes by reducing their ability to access the shared medium. The game model description is presented in the following sub-section. Note that the symbols used in the game model are provided in Table 4.1.

TABLE 4.1: Game model Notations

Symbol		Meaning
N	:	The players set
a_i	:	is the strategy chosen by node i
P_i	:	is the payoff of node i
R_i	:	The reputation of node i (level of selfishness)
Th	:	Reputation threshold value
$R_{final}(i)$	:	is the aggregated reputation for node i
W_i	:	is the set of observers monitoring i
R_{w_j}	:	is the reputation calculated from observer w_j.
AK_i	:	is the number of ACK frames received by node i
T	:	is the selfishness threshold value

4.2.1 Game model

The MAC layer cooperation among the nodes in VANET can be modeled using a non-zero non-cooperative repeated game model. It is a non-zero model because the benefits are shared among the players rather than transferred to one player, and it is modeled as non-cooperative because each player is considered as rational.

The collection of players set is denoted by N where a single player is indicated by i, where $i \in N$. The set of all potential actions adopted by a player i is denoted by A_i where a_i represents a single action made by player i. The interaction between k players can be modeled as 2-alternative game in which each player can either cooperate C_i or defect D_i; thus, $A_i = \{Cooperate, Defect\}$. The symbolic representation of the game model proposed in this chapter is as follows.

Definition *A MAC game in VANET is*

$$G = \langle N, \{a_i\}, \{P_i\} \rangle$$

where:

– N is the the neighboring nodes in the network.

– a_i is the strategy chosen by each node i, either to cooperate or defect.

– P_i is the payoff of node i.

The game can be depicted as k-Prisoners' dilemma [74] as shown in Table 4.2. The payoff of each player depends on its strategy as well as the strategy adopted by the rest of opponents. The payoff of prisoners' dilemma is proportional to the throughput enjoyed by node i where it should satisfy these conditions:

$$C_i > C_{i-1} \quad \textit{and} \quad D_i > D_{i-1} \quad \forall i \tag{4.1}$$

$$D_i > C_i \quad \forall i \tag{4.2}$$

$$C_{n-1} > D_0 \tag{4.3}$$

The first condition 4.1 indicates that the payoff of player i cannot be increased while more opponents are selfish regardless of its decision. For the second condition 4.2, any player i can have a higher payoff if i follows the defect strategy, assuming that other players maintain their strategies. Moving to the third condition 4.3, the obtained payoff when all players are cooperative would be higher than the payoff when all of them adopt the defect strategy.

According to the above conditions, the optimal strategy for any given node in one-shot game is to defect and behave selfishly. Therefore, if the game is played only once, it will have a single Nash Equilibrium [50] where any rational node will select the defect strategy in order to obtain the maximal payoff. Consequently, all of the nodes would end up with smaller bandwidth shares as

TABLE 4.2: The *k*-Prisoners' dilemma payoff matrix

Opponents	$k-1$	...	1	0
Cooperate	C_{k-1}		C_1	C_0
Defect	D_{k-1}		D_1	D_0

previously illustrated in Figure 4.2. In order to impose cooperation among the nodes, the game should be repeated unknown number of times. As a result, rational nodes will be forced to cooperate under the threat of potential retaliation in the repeated game.

4.2.2 Game model analysis

Consider that there is a group of N players (nodes) in the MAC game. The nodes' strategies are characterized by the selection of the size of the contention window. For instance, the selfish nodes in this model use smaller CW value in order to have the priority to send the packet before any other node. Whereas, the cooperative nodes select the contention window as specified in the CSMA/CA protocol. There are several assumptions that should be considered while modeling the game:

- The network is at the saturation condition. Otherwise, the selfish nodes are using the channel available bandwidth.
- The available bandwidth is shared evenly among the nodes in relatively long term period.
- The nodes do not collude with each other.

The next step in forming the game model is to define how the interaction between the nodes in the network takes place. The observation time is divided into intervals of Δ seconds called monitoring intervals. During a monitoring interval, each node will be monitoring the behavior of its neighboring nodes. For any defined interval, a node may be cooperative or non-cooperative (defect).

Each node determines its move for the next monitoring interval based on the behavior of its opposing node.

Each node can monitor the number of ACK frames (AK) received by its opposing nodes. This represents the number of successful transmissions made by a node since receiving an ACK means that the transmission has been successfully accomplished. It is assumed that all the nodes should have the same bandwidth share. Hence, all of them have approximately the same number of transmissions. This step is crucial for any given node in order to determine the moves of its opponents. This can be summarized as follows:

- If $AK_i > T$: Node i is assumed to be selfish.

where:

  - AK_i: is the number of ACK frames received by node i.
  - T: is the threshold value that defines the existence of a selfish node.

T is calculated using the mean and the variance of the AK_i variables as follows:

$$T = \mu + b\sigma \tag{4.4}$$

where: $\mu = Up$, $\sigma^2 = Up(1-p)$, p is the probability of transmission and U is the number of transmissions occurred during the monitoring interval. According to experimental measurements, the constant value b is affected by the number of neighboring nodes contending to the channel and the monitoring interval Δ. For example, if Δ = 5 sec and the number of neighboring nodes is four, a value of three can lead to a low ratio of false alarms.

4.3 Adaptive CSMA/CA protocol

In this section, the adaptive tit-for-tat CSMA/CA protocol is demonstrated. The new protocol allows the cooperative nodes to obtain a fair bandwidth share

under the presence of selfish nodes. Once the cooperative nodes encounter selfish behavior, they can accept this behavior or they can retaliate against the selfish nodes. The retaliation can be done by adopting the selfish behavior in which all the nodes can share the medium fairly. The parameters (CW_{min}, CW_{max}) reflects the contention window boundary. Thus, adaptive CSMA/CA protocol allows the cooperative nodes to retaliate by using ($CW_{min} = 2$, $CW_{max} = 2$) in order to gain the same payoff (throughput) as the selfish nodes. Algorithm 5 presents the proposed protocol.

The protocol consist of two main phases, monitoring phase and reaction phase. In monitoring phase, each node observes the behavior of its neighbors where at the end of each monitoring interval each node determines whether a neighboring node is selfish or not. As it is stated previously, the ACK packets are used to determine the behavior of any node as they reflects the effective bandwidth utilized by each node. After that, in the reaction phase the node retaliates against the node marked as selfish. The retaliation is performed in order to make the cooperative node gains the same throughput as the selfish nodes (see Line 28 in Algorithm 5). The key point in developing the new CSMA/CA protocol is to find powerful mechanisms that can (i) assure a fair utilization of the channel bandwidth and (ii) ensure that the selfish nodes do not have any other choice than cooperation. These mechanisms should be immune to the misinterpretation errors that are caused when a node determines the behavior of its neighboring node. The errors are mainly caused by the collisions and by the short term unfairness of the MAC protocol [75].

4.4 Tit-for-tat CSMA/CA strategies

In this section, the behavior of the nodes according to different tit-for-tat strategies is analyzed. Through the use of simulations, each strategy is evaluated

Algorithm 5 Tit-for-Tat CSMA/CA

Input: N_ACK_i, T
Output: P_i

```
1: c = set of cooperative nodes.
2: let N_ACK_i = Number of ACK frames received by node i.
3: let T = Threshold value that determine the selfishness behavior.
4: Δ = the monitoring interval.
5: let P_i = the payoff (throughput) obtained by node i.
```

Phase 1 – Monitoring phase

```
6: procedure MONITORING-PHASE
7:     for each Δ do
8:         for each node j ∈ c do
9:             Monitor N_ ACK_i for neighbor i.
10:            if N _ AK_i > T then
11:                j marks i as "Selfish"
12:            else
13:                j marks i as "Cooperative"
14:            end if
15:        end for
16:    end for
17: end procedure
```

Phase 2 – Reaction Phase

```
18: procedure REACTION-PHASE
19:    for Each node j ∈ c do
20:        if Neighbor i == Selfish then
21:            The cooperative node j should react against this neighbor s.t:
22:            (CW_min,CW_max) ← (2,2)
23:        else
24:            All the nodes share the bandwidth fairly s.t:
25:            (CW_min,CW_max) ← as defined by CSMA/CA
26:        end if
27:    end for
28:    P_Cooperative_Nodes = P_Selfish_Nodes
29: end procedure
```

and hence, the best strategy is selected. The aim of the best strategy is to maximize the bandwidth share of the cooperative nodes and retaliate against selfish nodes to impose cooperation among the nodes.

4.4.1 Setup and simulation scenarios

The Network Simulator *NS*-3[76] has been used to simulate the game model with the multiple tit-for-tat strategies. The simulated network consists of 30 nodes generating data traffic using Constant Bit Rate (CBR) model and saturating the channel. The simulation parameters are depicted in Table 4.3. The IEEE 802.11*p* has been used as the MAC protocol since it is specifically designed for the vehicular environment [11]. The IEEE 802.11*p* supports transmission rates ranging from 3 to 27 Mb/s over a bandwidth of 10 MHz. The simulation has been conducted on the Optimized Link State Routing (OLSR) protocol and the data rate was selected to be 12 Mb/s. The results are obtained from the average of 10 simulation runs in which each run used different seed value. Referring to section 2.8, there are several techniques that a node can adopt to the selfish behavior such as: *CW Cheating Strategy*, *Fixing CW Strategy*, and *BO Cheating Strategy* [44] [45]. In this research, fixing CW Strategy is selected to simulate the behavior of a selfish node.

Referring to the game model demonstrated in the previous section, if the nodes detect a greedy behavior in the network then a retaliation decision will take place. As a result, all the nodes will have a fair channel utilization.

Example 1. Consider the scenario illustrated in Figure 4.3. At t= 100s, 20% of the nodes behave selfishly to gain more throughput. At t=200s, normal nodes retaliate in which they use the same parameters used by the selfish node. As a result, all the nodes obtain the same bandwidth share. The key question is: *What action should be taken by the cooperative nodes for the next interval?* To answer this question, the behavior of the cooperative nodes for the next monitoring

TABLE 4.3: Simulation Parameters

Parameter	Value
Number of nodes	30
Topology	Freeway
Transmission Range	100 m
Speed Range	60 km/h - 120 km/h
Channel capacity	12 Mbps
MAC protocol	802.11*p*
PHY	OFDM
Packet size	2000 bytes
Traffic source	CBR/UDP
Access method	Basic scheme
Routing Protocol	OLSR
Propagation model	Free space
Number of simulation runs	10

interval is determined by each tit-for-tat strategy differently which is further discussed in the following sub-sections.

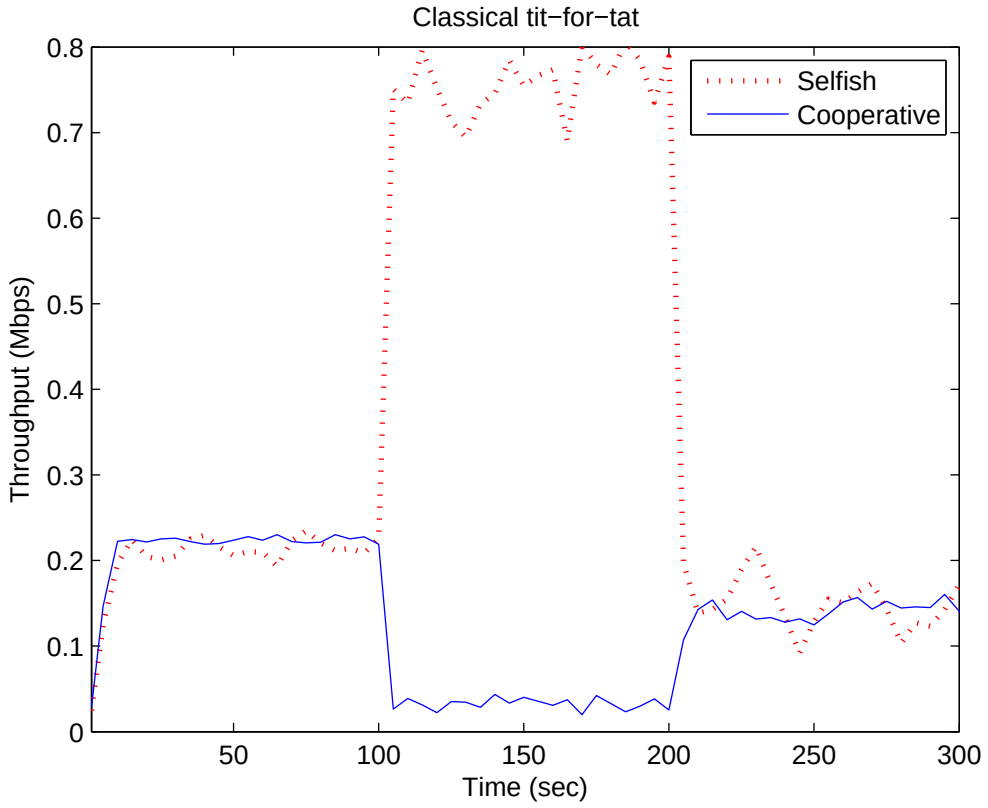


FIGURE 4.3: Classical tit-for-tat

4.4.2 Classical tit-for-tat

Classical tit-for-tat suggests that the nodes start with cooperation and then mirror the behavior of their opponents. Referring to Example 1, nodes following the classical approach will continue to defect after retaliation. This approach negatively affects the overall performance of the network as all the nodes will eventually end up with lesser bandwidth as illustrated in Figure 4.3 in the interval from t= 200s to t=300s. According to this, the Classical approach has several drawbacks. Firstly, nodes will suffer from mutual selfishness where no node cooperates with any other node. Secondly, this strategy does not perform well in the case of interpretation errors (i.e. detection of a normal node as a selfish one). Figure 4.4 clarifies the performance of classical approach in the case of interpretation error (solid curve). It is obvious from the figure that having multiple interpretation errors make all the nodes eventually stay in mutual selfishness and hence, with limited bandwidth share. This is due to the fact that the remaining nodes refrain from cooperating with the node detected as selfish.

4.4.3 Generous tit-for-tat

Generous tit-for-tat is a variation of the classical approach in which it forgives the periodic defection. To avoid the mutual selfishness introduced in the Classical approach, Generous tit-for-tat proposes that for the next interval nodes forgive the misbehaved node. However, this strategy is not efficient in a network where the selfish node follows the always defect (AD) strategy as it will always have an advantage over the other nodes. In Figure 4.5, at t= 100s, 20% of the nodes behave selfishly, then the cooperative nodes are altering between the retaliation and forgiveness states. Following this strategy the selfish node is partially punished and thus, more nodes are motivated to defect. According to this, the classical approach will be more effective in the case of having a selfish node that follows the AD strategy since the cooperative nodes will not lose

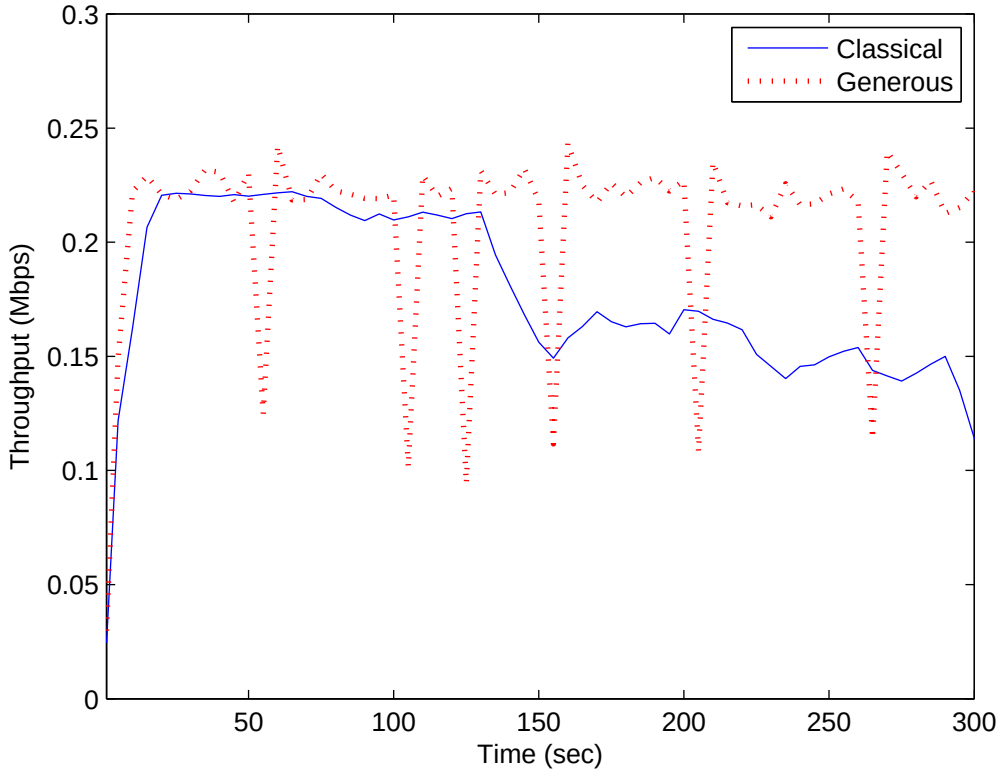


FIGURE 4.4: Interpretation errors

and the selfish nodes will not gain bandwidth at the expense of the cooperative nodes.

Now, assume that a node makes a wrong decision about another node (i.e interpretation error case). According to the Generous approach, having an interpretation error leads to a momentary degradation of the throughput as clarified in Figure 4.4 (dotted curve). This is due to the fact that the normal nodes forgive and cooperate with the misbehaving nodes. It is obvious from Figure 4.4, that the Generous approach performs better in the case of errors. However, following such approach is advantageous for the nodes that always defect and encourages more nodes to behave selfishly. Figure 4.6 compares the obtained throughput for the selfish nodes according to the Classical and the Generous approaches as the number of the nodes varies between 30 and 70. It can be

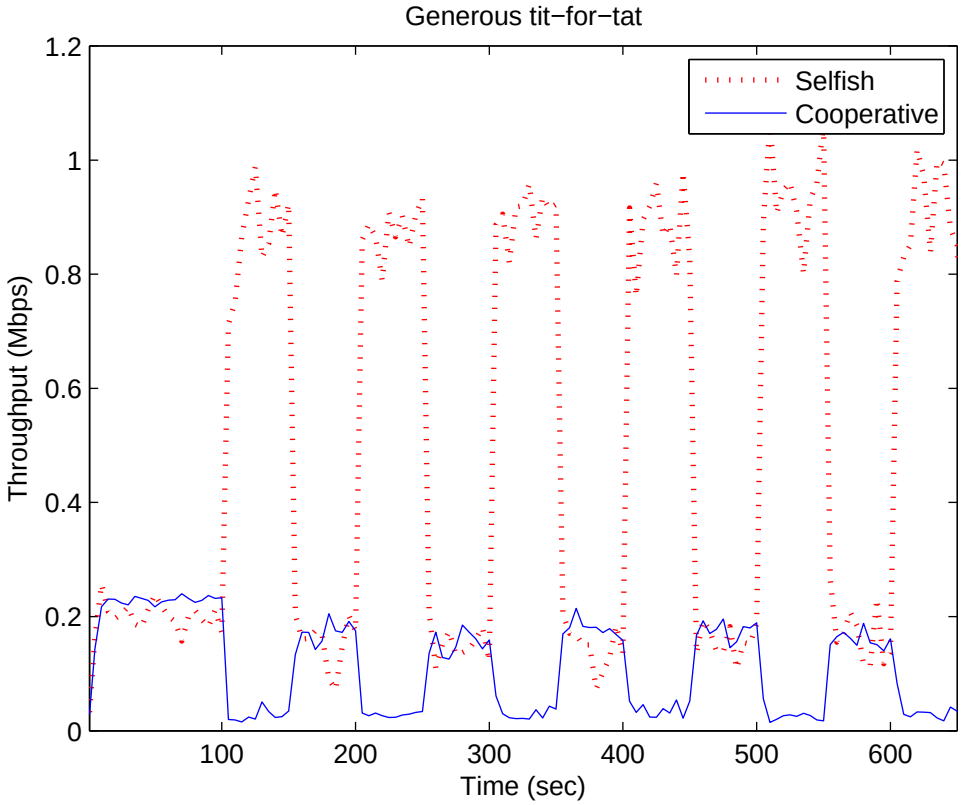


FIGURE 4.5: Generous tit-for-tat

deduced that the selfish nodes always have more bandwidth share in the Generous approach than in the classical one. Given this dilemma, a good strategy should have an immunity against interpretation errors and does not give advantage for the nodes that always defect.

4.4.4 Reputation based tit-for-tat

In order to overcome the limitations that exist in the Classical and the Generous tit-for-tat, there should be a strategy that cannot be exploited by a node that follows the AD strategy and can tolerate the interpretation errors. The Reputation based tit-for-tat suggests that the nodes retaliate against the selfish node if its reputation is higher than a threshold value (Th). The reputation of node i is calculated using (4.5). Here the Reputation refers to the selfishness level of a

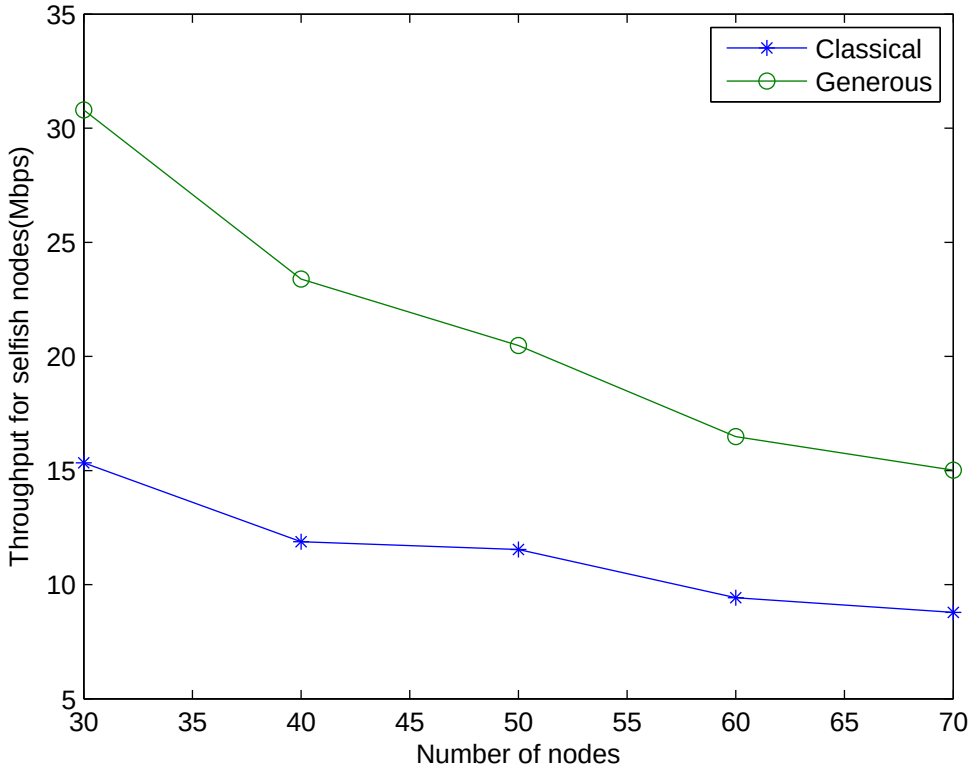


FIGURE 4.6: Classical Vs Generous

node.

$$R_i = \frac{S_i}{O_i} \tag{4.5}$$

where:

- S_i: is the number of times the node behaves selfishly.
- O_i: is the number of observation intervals.

Then, if Reputation (i) $> Th$, the node retaliates against this node otherwise they will cooperate with it. In this way, this strategy can react very well against the existence of a selfish node, as it will not continue defecting without affecting its reputation. This strategy is summarized in Algorithm 6.

Selecting the threshold value is very crucial for the performance of the strategy. Having a high value of Th will reduce the effect of local interpretation

Algorithm 6 Reputation Tit-for-Tat

Input: N_ACK_i, T
Output: P_i

let Δ = The length of the monitoring interval
let O be number of current monitoring interval
let N_O Total Number of monitoring intervals
let S = The number of times the node behaves selfishly
let P_i = the payoff obtained by node i

Phase 1 – Monitoring phase

procedure MONITORING-PHASE
  while $O < N\_O$ **do**
    for each Δ **do**
      for each node $j \in c$ **do**
        Monitor N_ ACK_i for neighbor i.
        if N _ $AK_i > T$ **then**
          j marks i as "Selfish"
          S $\leftarrow$ S+1
        else
          j marks i as "Cooperative"
        end if
      end for
    end for
    O $\leftarrow$ O+1
    Compute $R \leftarrow \frac{S}{O}$
  end while
end procedure

Phase 2 – Reaction Phase

procedure REACTION-PHASE
  for each node $j \in c$ **do**
    if $R_i > Th$ **then**
      $(CW_{min}, CW_{max}) \leftarrow (2, 2)$
    end if
  end for
  $P_{Cooperative\_Nodes} = P_{Selfish\_Nodes}$
end procedure

errors. On the other hand, this will give incentive for the selfish node to deviate as it can deviate once every $\frac{1}{Th}$ without being punished. Therefore, a low value of Th will reduce the incentive for the selfish node to deviate. Table 4.4 studies the impact of the Th on the average bandwidth of a selfish node. This study has been conducted on four nodes in which one of them is a selfish node. The values are based on 25 monitoring intervals.

TABLE 4.4: Impact of the Threshold (Th) on the Average Bandwidth Shares

Th	Selfish shares(Mbps)	Cooperative shares(Mbps)
0.05	2.42	2.15
0.10	3.07	1.96
0.15	3.28	1.88
0.20	3.36	1.87

As mentioned previously, this strategy is immune to infrequent interpretation errors (i.e., false detection). However, having multiple interpretation errors in the detection process makes this strategy unable to maximize the throughput of the cooperative nodes. Thus, a node which makes an incorrect decision about another node will retaliate. Therefore, more efficient strategy should be proposed in a way that the reputation of any given node is aggregated from multiple nodes to avoid the frequent misinterpretations.

4.4.5 Group Reputation based tit-for-tat

In order to overcome the issues in the Reputation based tit-for-tat strategy, we propose a new collaborative strategy which is the Group Reputation based tit-for-tat. In this strategy, the nodes retaliate against the selfish node if its aggregated reputation is higher than a predefined threshold value. The aggregated reputation is calculated by averaging the individual reputation from all neighboring nodes as in (4.6):

$$R_{final}(i) = \frac{\sum_{w_j \in W_i} R_{w_j}}{|W_i|} \tag{4.6}$$

where:

- $R_{final}(i)$: is the aggregated reputation for node i.
- R_{w_j}: is the reputation calculated from observer w_j.
- W_i: is the set of observers monitoring i.
- $|W_i|$: is the number of elements in W_i.

Therefore, if $R_{final}(\text{i}) > Th$, then the nodes will retaliate against node i. Referring to Algorithm 6, this strategy has the same Monitoring phase as the Reputation based strategy. However, the Reaction phase has been modified in a way that the reputation of a node is computed collaboratively from the neighbors as it is depicted in Algorithm 7.

Algorithm 7 Reaction Phase

Input: R_{w_j}
Output: $R_{final}(i)$

let W_i is the set of observers monitoring i
let R_{w_j}: is the reputation calculated from observer w_j in W_i
let $Th =$ Reputation threshold value
procedure GROUPREPUTATIONTIT-FOR-TAT
    for each node $w_j \in W_i$ **do**
        Compute R_{w_j}
        Compute $R_{final}(i)$ as shown in (4.6)
        if $R_{final}(i) > Th$ **then**
            $(CW_{min}, CW_{max}) \leftarrow (2, 2)$
        end if
    end for
end procedure

This strategy shows better performance than the reputation based tit-for-tat because it can combact the problems encountered from the frequent interpretation errors. This is due to the fact that the reputation of a node is aggregated from the neighboring nodes. Table 4.5 compares the performance of the Reputation based tit-for-tat and the Group Reputation based tit-for-tat strategies in the case of interpretation errors. It is clearly shown that the throughput of

the cooperative nodes is higher in the Group Reputation strategy compared to the throughput of the cooperative nodes in the Reputation strategy. This is because the reputation is aggregated from the neighbor and hence the decision of reaction will be done collaboratively.

However, the problem in the Group based approach is that the reputation value is calculated after the individual observer made the decision about a given node. Thus, the group approach can handle certain frequent interpretation errors without affecting the reputation of the cooperative node but eventually, the reputation of a cooperative node gets affected. This problem has been solved using the following strategy.

TABLE 4.5: Throughput of cooperative node in Reputation Vs Group Reputation

Th	Throughout in Reputation (Mbps)	Throughout in Group Reputation (Mbps)
0.05	1.9	2.2
0.10	2.07	2.4
0.15	2.095	2.63
0.20	2.12	3.01

4.4.6 Cooperative Detection based tit-for-tat

In order to overcome the limitations of the previously mentioned strategies, we propose a Cooperative tit-for-tat strategy which consists of three phases: monitoring, aggregation, and cooperation decision. This strategy improves the detection ratio and reduces the impact of any interpretation error. This is due to that fact that the final detection decision is based on weighted sum of the individual detector decision where the weight represents the reputation of a node. The phases are outlined below:

Monitoring

In this phase, each node monitors its neighboring nodes to determine their behavior. According to the detection model described previously, if a node detects a neighboring node as a selfish node the detector outcome is 1 otherwise it is 0. The phase can be summarized as illustrated in equation (4.7). It should be indicated, that this is the outcome of an individual detector.

$$f(w_j, i) = \begin{cases} 1 & w_j \text{ detect } i \text{ as selfish} \\ 0 & w_j \text{ detect } i \text{ as normal} \end{cases} \tag{4.7}$$

where: w_j: is the monitoring node for node i, and i: is the monitored node.

Aggregation

The evaluation about a monitored node is based on a weighted decision from all neighboring node. Thus, a decision made by a particular observer is weighted according to its reputation. Here, we use the node's reputation for behaving normally which is $1 - R(i)$. The aggregation function is given in equation (4.8).

$$F(W_i, i) = \frac{\sum_{w_j \in W_j} (1 - R(i))(w_j) \times f(w_j, i)}{\sum (1 - R(i))(w_j)} \tag{4.8}$$

where: W_i is the set of observers monitoring node i.

Cooperation decision

In this phase, the decision of cooperation is determined by the aggregated decision; that is, nodes will cooperate with the monitored node if $F(i)$ is greater than a threshold value. In the simulations, the threshold value has been selected to

be 0.5. The outcome of this phase is given by equation(4.9).

$$FinalDecision = \begin{cases} if F(i) > Threshold & Cooperate \\ otherwise & Retaliate \end{cases} \tag{4.9}$$

This strategy can be summarized in Algorithm 8. The most significant difference between this strategy and the Group Reputation tit-for-tat strategy is in the tolerance level of mis-detection. This strategy can tolerate better in terms of misinterpretation especially when we have more monitors.

Algorithm 8 Cooperative Detection based tit-for-tat

Input: R_{w_j}
Output: $F(i)$

let c = set of cooperative nodes
let P_i = the payoff obtained by node i
let W_i be the set of observers monitoring node i
let w_j be the monitoring node for node i
let R_{w_j} be the reputation of observer w_j

Phase 1 – Monitoring phase

procedure MONITORING-PHASE
    for each node $w_j \in W_i$ **do**
        Compute $f(w_j, i)$ as in (4.7)
        Compute $F(i) = \frac{\sum_{w_j \in W_j}(1-R_{w_j})(w_j) \times f(w_j,i)}{\sum(1-R_{w_j})(w_j)}$
    end for
end procedure

Phase 2 – Reaction Phase

procedure REACTION-PHASE
    for each node $j \in c$ **do**
        if F(i)> H **then**
            $(CW_{min}, CW_{max}) \leftarrow (2, 2)$
        end if
    end for
    $P_{Cooperative\_Nodes} = P_{Selfish\_Nodes}$
end procedure

4.4.7 Discussion

In this sub-section, we evaluate the performance of the five models namely Classical, Generous, Reputation, Group Reputation and Cooperative Detection tit-for-tat strategies with different scenarios. The Classical, Generous and Reputation strategies are introduced in the literature whereas the Group Reputation and Cooperative Detection are proposed in this research.

Figure 4.7 shows the throughput of the cooperative nodes as the network density changes. The number of nodes varies between 30 and 70 and the percentage of selfishness is chosen to be 20%. The simulation lasts for 500 seconds. The main purpose of such simulation is to determine the strategy that can maximize the throughput of the cooperative nodes under the case of multiple interpretation errors. It can be clearly noticed that the cooperative nodes can obtain high throughput if the Cooperative Detection strategy is adopted while they obtain a very low bandwidth share if the Classical approach is adopted. In fact, the Cooperative Detection strategy is able to maximize the payoff of the cooperative nodes by 20% compared to the Classical strategy and by 9% compared to the Reputation based strategy.

Figure 4.8 studies the performance of the five models with respect to the percentage of selfishness in the network. The percentage of selfish nodes varies between 10% to 50% where the number of nodes in the network is 50. Figure 4.8 illustrates the payoff (throughput) obtained by the selfish nodes where the simulations last for 500 seconds. From the Figure 4.8, it is clear that when the nodes follow the Cooperative Detection strategy, selfish nodes gain low throughput in comparison with the Generous strategy. In fact, the Group Reputation and the Cooperative Detection strategies provide significantly better results even with 10% of the nodes being selfish as they minimize the throughput of selfish nodes by 67% and 76% respectively compared with the Generous strategy. In addition, the variation in the throughput for Group strategy and Cooperative Detection

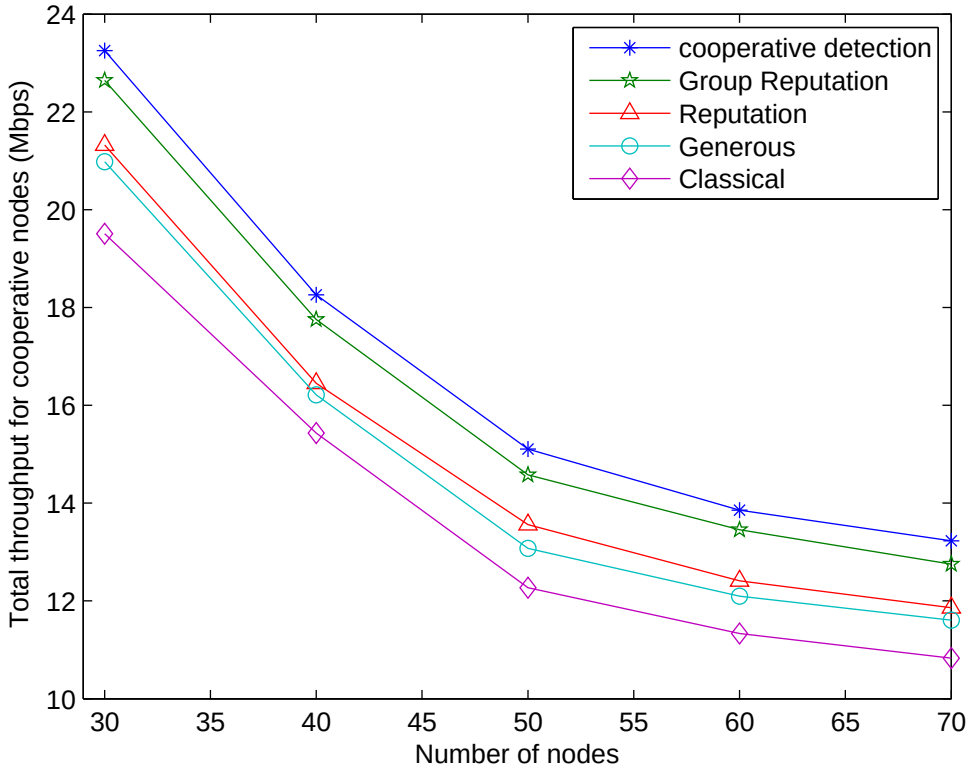


FIGURE 4.7: All models with different network density

strategies is limited as the percentage of selfish nodes increases. This indicates that both strategies are robust and nearly independent of the percentage of selfish nodes.

In Figure 4.9, we examine the performance of the five strategies with respect to an incremental number of monitoring nodes. This number varies between 20% to 100%. This percentage refers to the number of monitors that can observe the behavior of other nodes to provide a cooperation decision about their neighbors. The percentage of the selfish nodes is selected to be 20% and the number of nodes simulating this scenario is 50. It can be deduced that the cooperative model performs well with a low number of monitors and does not allow the selfish nodes to take advantage over the normal nodes. As as result, the selfish node is motivated to cooperate and can still be punished using such strategy. At the same time, the Generous strategy gives the selfish nodes tremendous

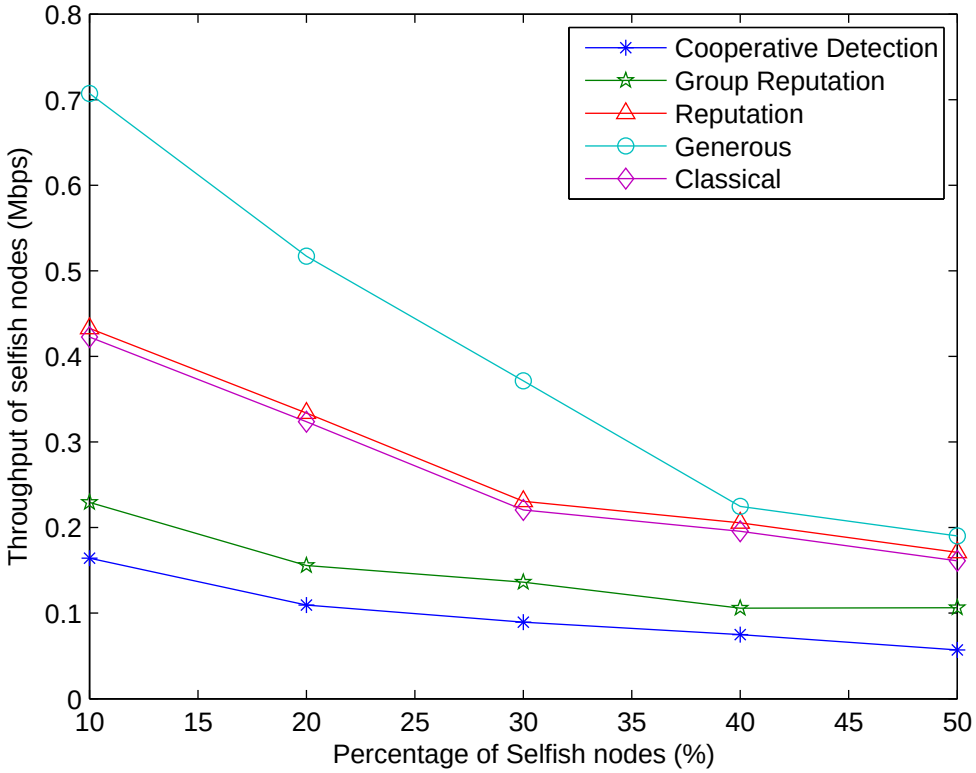


FIGURE 4.8: The five models with different selfishness percentage

gain out of all the strategies. This makes the Generous strategy vulnerable to cope with the presence of the selfishness in the network. For example, when the percentage of monitors is 40% the average throughput for a selfish node is 0.3Mbps when the Cooperative Detection strategy is applied whereas this value is around 0.5Mbps when the Generous strategy is adopted.

According to Figures 4.7, 4.8 and 4.9, it has been proved that the cooperative based tit-for-tat model outperforms the rest of the strategies. This can be justified due to the fact that this strategy has a high detection probability for the selfish nodes and a low percentage of interpretation errors resulting from the use of the aggregated detection algorithm. Thus, the cooperative nodes are able to maximize their throughput and at the same time retaliate against any selfish behavior. In this way, the selfish nodes are motivated to cooperate and comply with the protocol.

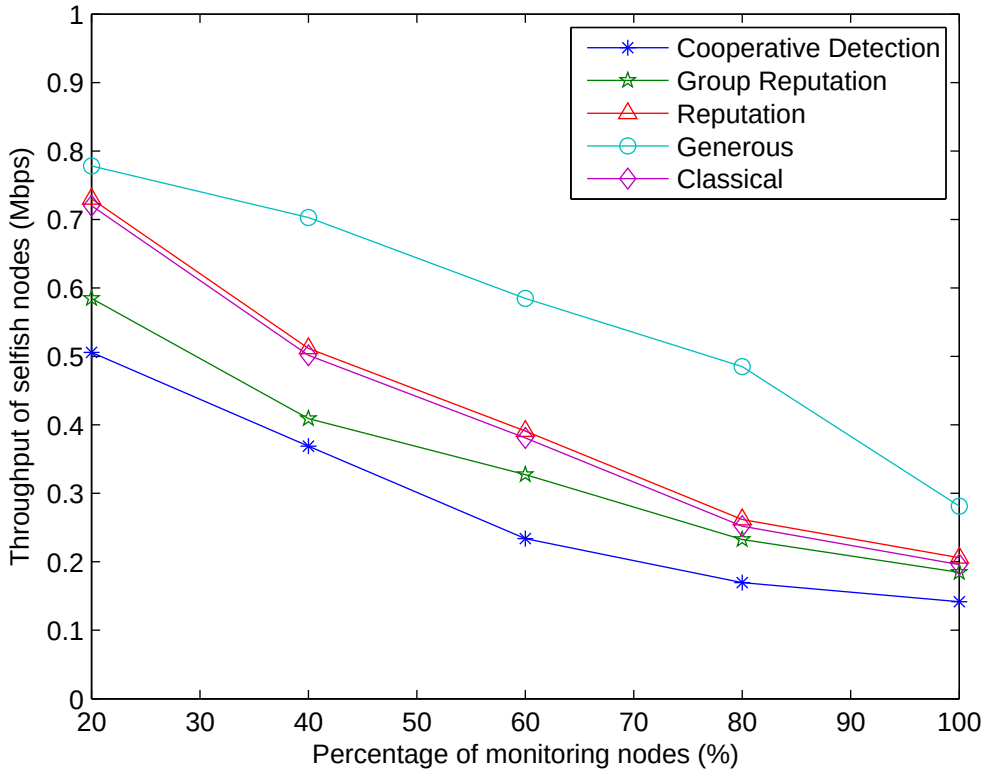


FIGURE 4.9: The five models with different monitoring percentage

4.5 Limitation

Although the two new collaborative strategies outer-performed the existing strategies, it suffers from the trustworthy issue. The main limitation of the new collaborative strategies that they are based on trusting the detection report results from the neighbors. They do not consider the situations where some nodes may act maliciously and provide fake reports for other nodes.

4.6 Conclusions

CSMA/CA protocol can be exploited by some nodes in order to increase their bandwidth share. This simply can be achieved by modifying the size of the contention window. Therefore, such nodes are able to increase their access

to the channel and increase their throughput as well. Consequently, the overall performance of the network is severely degraded as transmission collision probability is increased. As a result, the normal nodes are denied from accessing the channel. In this chapter, we proposed a motivational mechanism that emphasises the cooperation among the nodes based on tit-for-tat strategy. We presented five strategies built on top of tit-for-tat strategy to deal with the problem of selfish nodes which are: (1) Classical tit-for-tat (2) Generous tit-for-tat (3) Reputation based tit-for-tat (4) Group Reputation based tit-for-tat and (5) Cooperative Detection based tit-for-tat strategies. Classical, Generous, and Reputation tit-for-tat strategies were proposed in the literature and due to their limitations in imposing the cooperation among the nodes, the Group Reputation and the Cooperative Detection strategies were proposed in this chapter.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, we mitigated the effect of selfish nodes at the network layer. Such nodes refrain from cooperating in the routing protocol by not relaying the traffic of others. This was achieved by proposing a new cluster-based routing protocol referred to as IWD-QoS-OLSR. The proposed protocol is an enhanced version of the QoS-OLSR protocol that is based on Intelligent Water Drop algorithm to establish the communication among the clusters. This algorithm introduces a recovery mechanism that can compensate for any link failure between the clusters. In addition, the performance of the introduced MPR recovery algorithm was compared with the performance of a recovery algorithm based on ant-colony algorithm. According to the conducted simulations, the proposed recovery algorithm based on IWD proved to be more reliable than the recovery algorithm based on ant colony. This is due to the ability of our proposed algorithm to increase the percentage of alive routes by 75% when the network density equals to 90 nodes. Simulation results also demonstrated that the new proposed protocol is able to improve the connectivity and packet delivery ratio, as well as reduce the path length and probability of packet loss. More specifically, the proposed protocol reduced the path length by 2 hops, the percentage of disconnected clusters by 59.3%, and improved the packet delivery ratio by

19% compared to the standard QoS-OLSR. The results also revealed that our model presented an acceptable percentage of bandwidth average difference.

Moreover, we addressed the problem of greedy behavior at the MAC layer as some nodes have the temptation to transgress the channel access mechanism to acquire the channel for a longer time. Thus, we developed a MAC-layer cooperation model based on tit-for-tat strategies to motivate the nodes to be cooperative. The proposed model is based on two new collaborative tit-for-tat strategies which are: (1) Group Reputation based tit-for-tat and (2) Cooperative Detection based tit-for-tat. In the former strategy, the reputation of a node is aggregated among its neighbors. Based on the aggregated value, the behavior of a node is determined. In the latter strategy, the detection decision about a node is collected and weighted based on the nodes' reputations in order to decide the cooperation level of a node. A retaliation decision is taken based on the computed aggregated detection level. Through the use of simulations, Cooperative Detection and Group Reputation tit-for-tat strategies outperformed the rest of the strategies as they rely on collaborative schemes to determine the cooperation decision.

Furthermore, the Cooperative Detection based tit-for-tat strategy outperformed the Group Reputation tit-for-tat strategy since it relies on cooperative technique to build the detection decisions instead of one-to-one decision mechanism. This strategy is immune to frequent interpretation errors and it enforces the selfish nodes to cooperate under the threat of retaliation. This strategy is also considered to be the most reliable, robust and powerful among all of the aforementioned strategies. It is able to maximize the payoff of the cooperative nodes and minimize the payoff of the selfish nodes compared to the other models. Its is able to maximize the payoff of the cooperative nodes by 20% compared to the Classical strategy, and by 9% compared to the Reputation based strategy. It can also minimize the throughput of selfish nodes by 76% compared

with the Generous strategy.

5.2 Future work

The research presented in this thesis tackled the problem of selfishness in Vehicular ad-hoc Networks. There are several research topics that can be pursued in the future based on the work in this thesis:

- **Designing a disconnection prediction algorithm.** The MPR recovery algorithm proposed in the thesis can be improved. This can be achieved by implementing a prediction algorithm that can predict any possible link failure between any MPR and its cluster head. To achieve such a goal, a continuous monitoring on the received power cluster head with its corresponding MPR should be carried out.

- **Preventing the presence of selfish nodes at the Network layer.** In this research, we mitigated the effect of the selfish nodes that refuse to cooperate in the routing process. However, in order to prevent such behavior, a penalty scheme should be introduced to penalize the misbehaving nodes. In this way, the nodes can be forced to cooperate.

- **Integrating the MAC-layer cooperation model with the QoS-OLSR.** In this research, we developed the MAC-layer cooperation model for the standard OLSR protocol. This model can be applied to the QoS-OLSR protocol as well.

- **Developing a Cross layer based routing protocol.** We can exploit the reputation value introduced in the game model to design a MAC-layer reputation system. In this way, the reputation of a node aggregated from the reputation system can be propagated to the upper network layer in order to be incorporated in the route selection mechanism. In this way, the

nodes who maintain a good reputation can have a higher probability to be selected to route the traffic. This can highly improve the performance of the routing protocol.

Bibliography

[1] J. Das and P. Das. "An Overview of Wireless Ad hoc Networks". In: *International Journal of Advanced Research in Computer Science* 6.1 (2015).

[2] H. Lee and D. Jeon. "A mobile ad-hoc network multi-path routing protocol based on biological attractor selection for disaster recovery communication". In: *ICT Express* (2015).

[3] H. Hartenstein and K. P. Laberteaux. "A tutorial survey on vehicular ad hoc networks". In: *Communications Magazine, IEEE* 46.6 (2008), pp. 164–171.

[4] F. J. Martinez et al. "Emergency services in future intelligent transportation systems based on vehicular communication networks". In: *Intelligent Transportation Systems Magazine, IEEE* 2.2 (2010), pp. 6–20.

[5] B. T. Sharef, R. A. Alsaqour, and M. Ismail. "Vehicular communication ad hoc routing protocols: A survey". In: *Journal of Network and Computer Applications* 40 (2014), pp. 363–396.

[6] F. Li and Y. Wang. "Routing in vehicular ad hoc networks: A survey". In: *Vehicular Technology Magazine, IEEE* 2.2 (2007), pp. 12–22.

[7] J. Toutouh, J. García-Nieto, and E. Alba. "Intelligent OLSR routing protocol optimization for VANETs". In: *Vehicular Technology, IEEE Transactions on* 61.4 (2012), pp. 1884–1894.

[8] R. F. Atallah, M. J. Khabbaz, and C. M. Assi. "Vehicular networking: A survey on spectrum access technologies and persisting challenges". In: *Vehicular Communications* 2.3 (2015), pp. 125–149.

[9] Z. Y. Rawshdeh and S. M. Mahmud. "Toward strongley connected clustering structure in vehicular ad hoc networks". In: *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*. IEEE. 2009, pp. 1–5.

[10] C. Shea, B. Hassanabadi, and S. Valaee. "Mobility-based clustering in VANETs using affinity propagation". In: *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*. IEEE. 2009, pp. 1–6.

[11] C. Han et al. "Analytical study of the IEEE 802.11 p MAC sublayer in vehicular networks". In: *Intelligent Transportation Systems, IEEE Transactions on* 13.2 (2012), pp. 873–886.

[12] L. Mokdad, J. Ben-Othman, and A. T. Nguyen. "DJAVAN: Detecting jamming attacks in Vehicle Ad hoc Networks". In: *Performance Evaluation* 87 (2015), pp. 47–59.

[13] M. Raya et al. "DOMINO: Detecting MAC layer greedy behavior in IEEE 802.11 hotspots". In: *IEEE Transactions on Mobile Computing* 5.12 (2006), pp. 1691–1705.

[14] J. Tang, Y. Cheng, and W. Zhuang. "An analytical approach to real-time misbehavior detection in IEEE 802.11 based wireless networks". In: *INFOCOM*. IEEE. 2011, pp. 1638–1646.

[15] S. Boyer et al. "An adaptive tit–for–tat strategy for IEEE 802.11 CSMA/CA protocol". In: *International Journal of Security and Networks* 7.2 (2012), pp. 95–106.

[16] D. T. Hoang et al. "Applications of Repeated Games in Wireless Networks: A Survey". In: *Communications Surveys & Tutorials, IEEE* 17.4 (2015), pp. 2102–2135.

[17] C. Kopp. "Ad Hoc Networking". In: *Systems Journal* (1999), pp. 33–40.

[18] F. J. Martinez et al. “A survey and comparative study of simulators for vehicular ad hoc networks (VANETs)”. In: *Wireless Communications and Mobile Computing* 11.7 (2011), pp. 813–828.

[19] F. Qu, F.-Y. Wang, and L. Yang. “Intelligent transportation spaces: vehicles, traffic, communications, and beyond”. In: *Communications Magazine, IEEE* 48.11 (2010), pp. 136–142.

[20] M. Gerla and L. Kleinrock. “Vehicular networks and the future of the mobile internet”. In: *Computer Networks* 55.2 (2011), pp. 457–469.

[21] A. Baiocchi and F. Cuomo. “Infotainment services based on push-mode dissemination in an integrated VANET and 3G architecture”. In: *Communications and Networks, Journal of* 15.2 (2013), pp. 179–190.

[22] S. Zeadally et al. “Vehicular ad hoc networks (VANETS): status, results, and challenges”. In: *Telecommunication Systems* 50.4 (2012), pp. 217–241.

[23] S. Al-Sultan et al. “A comprehensive survey on vehicular Ad Hoc network”. In: *Journal of network and computer applications* 37 (2014), pp. 380–392.

[24] H Ghaffarian, M. Fathy, and M. Soryani. “Vehicular ad hoc networks enabled traffic controller for removing traffic lights in isolated intersections based on integer linear programming”. In: *Intelligent Transport Systems, IET* 6.2 (2012), pp. 115–123.

[25] A. M. Vegni, M. Biagi, and R. Cusani. *Smart vehicles, technologies and main applications in vehicular ad hoc networks*. INTECH Open Access Publisher, 2013.

[26] E. Schoch et al. “Communication patterns in VANETs”. In: *Communications Magazine, IEEE* 46.11 (2008), pp. 119–125.

[27] B. Paul et al. “VANET Routing Protocols: Pros and Cons”. In: *arXiv preprint arXiv:1204.1201* (2012).

[28] A. Kumar, P. K. R. Sreenivasulu, and A. Lakkshmanan. "An Enhancement of Dynamic Source Routing by Efficient Load Balancing in Wireless HOC Networks". In: *International Journal of Applied Engineering Research* 8.19 (2013), p. 2013.

[29] D. Al-Terri et al. "Q-DSR protocol in vehicular ad-hoc networks". In: *Innovations in Information Technology (IIT), 2015 11th International Conference on*. IEEE. 2015, pp. 162–165.

[30] H. Otrok et al. "A cluster-based model for QoS-OLSR protocol". In: *IWCMC*. IEEE. 2011, pp. 1099–1104.

[31] T. Clausen and P. Jacquet. "Optimized link state routing protocol(OLSR)". In: (RFC 3626, Internet Engineering Task Force, October 2003).

[32] H. Shah-Hosseini. *Optimization with the nature-inspired intelligent water drops algorithm*. INTECH Open Access Publisher, 2009.

[33] D. Sensarma and K. Majumder. "IWDRA: An Intelligent Water Drop Based QoS-Aware Routing Algorithm for MANETs". In: *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2013*. Springer. 2014, pp. 329–336.

[34] H. Shah-Hosseini. "Problem solving by intelligent water drops". In: *CEC*. IEEE. 2007, pp. 3226–3231.

[35] Z. Li, F. Zhao, and H. Liu. "Intelligent water drops algorithm for vehicle routing problem with time windows". In: *Service Systems and Service Management (ICSSSM), 2014 11th International Conference on*. IEEE. 2014, pp. 1–6.

[36] H. Shah-Hosseini. "The intelligent water drops algorithm: a nature-inspired swarm-based optimization algorithm". In: *International Journal of Bio-Inspired Computation* 1.1 (2009), pp. 71–79.

[37] K. Socha and C. Blum. "Ant colony optimization". In: *Metaheuristic procedures for training neutral networks*. Springer, 2006, pp. 153–180.

[38] M. H. Manshaei and J.-P. Hubaux. "Performance analysis of the ieee 802.11 distributed coordination function: Bianchi model". In: *Mobile Networks: http://mobnet. epfl. ch Ni W., Romdhani L., Turletti T.(2004), A Survey of QoS Enhancements for IEEE* 802 (2007).

[39] P. Roshan and J. Leary. *802.11 Wireless LAN fundamentals*. Cisco press, 2004.

[40] I. S. Association et al. *IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE, 2001.

[41] M. Ilyas. *The handbook of ad hoc wireless networks*. CRC press, 2002.

[42] G. Bianchi. "Performance analysis of the IEEE 802.11 distributed coordination function". In: *Selected Areas in Communications, IEEE Journal on* 18.3 (2000), pp. 535–547.

[43] M. N. Mejri and J. Ben-Othman. "Detecting greedy behavior by linear regression and watchdog in vehicular ad hoc networks". In: *Global Communications Conference (GLOBECOM), 2014 IEEE*. IEEE. 2014, pp. 5032–5037.

[44] M. Li et al. "Mac-layer selfish misbehavior in ieee 802.11 ad hoc networks: Detection and defense". In: *IEEE Transactions on Mobile Computing* 14.6 (2015), pp. 1203–1217.

[45] Z. Lu, W. Wang, and C. Wang. "Modeling and Performance Evaluation of Backoff Misbehaving Nodes in CSMA/CA Networks". In: *IEEE Transactions on Mobile Computing* 11.8 (2012), pp. 1331–1344.

[46] A. Al-Dhanhani et al. "A game theoretical model for collaborative groups in social applications". In: *Expert Systems with Applications* 41.11 (2014), pp. 5056–5065.

[47] H. Marshoud, H. Otrok, and H. Barada. "Macrocell–femtocells resource allocation with hybrid access motivational model". In: *Physical Communication* 11 (2014), pp. 3–14.

[48] D. T. Tran, Z. Chen, and A. Farago. "On selfish behavior in TDMA-based bandwidth sharing protocols in wireless networks". In: *Journal of Telecommunications* 4.1 (2010), pp. 1–9.

[49] B. Niu, H. V. Zhao, and H. Jiang. "A cooperation stimulation strategy in wireless multicast networks". In: *IEEE Transactions on Signal Processing* 59.5 (2011), pp. 2355–2369.

[50] D. E. Charilas and A. D. Panagopoulos. "A survey on game theory applications in wireless networks". In: *Computer Networks* 54.18 (2010), pp. 3421–3430.

[51] C. Wedekind and M. Milinski. "Human cooperation in the simultaneous and the alternating Prisoner's Dilemma: Pavlov versus Generous Tit-for-Tat". In: *Proceedings of the National Academy of Sciences* 93.7 (1996), pp. 2686–2689.

[52] A. Chriqi, H. Otrok, and J.-M. Robert. "SC-OLSR: Secure clustering-based OLSR model for ad hoc networks". In: *WiMob*. IEEE. 2009, pp. 239–245.

[53] A. Munaretto et al. "A link-state QoS routing protocol for ad hoc networks." In: *MWCN*. 2002, pp. 222–226.

[54] O. A. Wahab, H. Otrok, and A. Mourad. "VANET QoS-OLSR: QoS-based clustering protocol for Vehicular Ad hoc Networks". In: *Computer Communications* 36.13 (2013), pp. 1422–1435.

[55] T. A. Khaleel and M. Y. Ahmed. "Using intelligent water drops algorithm for optimisation routing protocol in mobile ad–hoc networks". In: *International Journal of Reasoning-based Intelligent Systems* 4.4 (2012), pp. 227–234.

[56] F. Shi et al. "A novel scheme to prevent MAC layer misbehavior in IEEE 802.11 ad hoc networks". In: *Telecommunication Systems* (2013), pp. 1–10.

[57] J. Tang and Y. Cheng. "Adaptive Misbehavior Detection in IEEE 802.11 TM Based on Markov Decision Process". In: *Intrusion Detection for IP-Based Multimedia Communications over Wireless Networks*. Springer, 2013, pp. 35–51.

[58] R. Axelrod. "The evolution of cooperation: revised edition". In: (2006).

[59] G. Quer et al. "Inter-network cooperation exploiting game theory and bayesian networks". In: *IEEE Transactions on Communications* 61.10 (2013), pp. 4310–4321.

[60] O. A. Wahab, H. Otrok, and A. Mourad. "A Dempster–Shafer Based Tit-for-Tat Strategy to Regulate the Cooperation in VANET Using QoS-OLSR Protocol". In: *Wireless Personal Communications* 75.3 (2014), pp. 1635–1667.

[61] J. Konorski. "A game-theoretic study of CSMA/CA under a backoff attack". In: *IEEE/ACM Transactions on Networking (TON)* 14.6 (2006), pp. 1167–1178.

[62] I. Tinnirello, L. Giarré, and G. Neglia. "A game theoretic approach to MAC design for infrastructure networks". In: *49th IEEE Conference on Decision and Control (CDC)*. IEEE. 2010, pp. 1933–1938.

[63] I. Tinnirello, L. Giarré, and G. Neglia. "MAC design for WiFi infrastructure networks: A game-theoretic approach". In: *IEEE Transactions on Wireless Communications* 10.8 (2011), pp. 2510–2522.

[64] M. Ghazvini, N. Movahhedinia, and K. Jamshidi. "GCW: A Game Theoretic Contention Window Adjustment Approach for IEEE 802.11 WLANs". In: *Wireless Personal Communications* 83.2 (2015), pp. 1101–1130.

[65] O. A. Wahab, H. Otrok, and A. Mourad. "A cooperative watchdog model based on Dempster–Shafer for detecting misbehaving vehicles". In: *Computer Communications* 41 (2014), pp. 43–54.

[66] D. Al-Terri et al. "QoS-OLSR protocol based on intelligent water drop for Vehicular ad-hoc networks". In: *International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE. 2015, pp. 1352–1357.

[67] Y. Wang and F. Li. "Vehicular ad hoc networks". In: *Guide to wireless ad hoc networks*. Springer, 2009, pp. 503–525.

[68] S. M. Mousavi et al. "Mobisim: A framework for simulation of mobility models in mobile ad-hoc networks". In: *WiMob*. IEEE. 2007, pp. 82–82.

[69] J. Harri, F. Filali, and C. Bonnet. "Mobility models for vehicular ad hoc networks: a survey and taxonomy". In: *Communications Surveys and Tutorials, IEEE* 11.4 (2009), pp. 19–41.

[70] M. Dorigo et al. *Ant Colony Optimization and Swarm Intelligence: 6th International Conference, ANTS 2008, Brussels, Belgium, September 22-24, 2008, Proceedings*. Vol. 5217. Springer, 2008.

[71] S.-L. Wu, S. Fan-Jiang, and Z.-T. Chou. "An efficient quality-of-service MAC protocol for infrastructure WLANs". In: *Journal of network and computer applications* 29.4 (2006), pp. 235–261.

[72] H. Zhao et al. "E-MAC: An evolutionary solution for collision avoidance in wireless ad hoc networks". In: *Journal of Network and Computer Applications* 65 (2016), pp. 1–11.

[73] J. Tang et al. “Real-time detection of selfish behavior in IEEE 802.11 wireless networks”. In: *72nd Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE*. IEEE. 2010, pp. 1–5.

[74] J. J. Jaramillo and R Srikant. “A game theory based reputation mechanism to incentivize cooperation in wireless ad hoc networks”. In: *Ad Hoc Networks* 8.4 (2010), pp. 416–429.

[75] C. E. Koksal, H. Kassab, and H. Balakrishnan. “An analysis of short-term fairness in wireless media access protocols (poster session)”. In: *ACM SIGMETRICS Performance Evaluation Review*. Vol. 28. 1. ACM. 2000, pp. 118–119.

[76] *NS-3*. URL: https://www.nsnam.org/.