LEIBNIZ UNIVERSITÄT HANNOVER

Geo-Routing in Urban Car-2-X Communication Networks

Von der Fakultät für Elektrotechnik und Informatik
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des akademischen Grades

Doktor-Ingenieur

genehmigte

Dissertation

von

M.Sc. MEHDI TAVAKOLI GARROSI

geboren am 04. Juni 1983 in Gorgan, Iran

2018

REFERENT: Prof. Dr.-Ing. Markus Fidler
KORREFERENTIN: Prof. Yusheng Ji

TAG DER PROMOTION: 18. Juni 2018

ABSTRACT

As reported by the European Telecommunications Standards Institute (ETSI), road traffic is constantly increasing, that causes serious problems, e.g., congested roads, road-safety and environmental effects. Stand-alone driver assistance systems are beneficial but the development of Intelligent Transportation Systems (ITS) enables the Vehicle-to-X (V2X) communications and the cooperation between vehicles to provide the traffic management, road-safety and comfort applications. ETSI has developed Cooperative ITS (C-ITS) standards for ITS in Europe.

In Vehicular Ad-hoc Networks (VANETs), vehicles dynamically set up an ad-hoc network without any aid of infrastructure. Vehicles move fast and are constrained within the layout of the roads, which leads to frequent network disconnections. Moreover, in urban scenarios, vehicles are facing the shadowing effects of buildings. When, the source and destination of a data packet are located outside of each other's communication range, other vehicles in between should receive the packet from the source and relay it through the network towards the destination. Therefore, vehicles need routing protocols, that help them to find short, robust and reliable routes to deliver the data packets.

Geo-routing is appropriate for the networks with high mobility and frequent topology changes. Moreover, geo-routing protocols rely only on the geographic position information of vehicles. Therefore, they scale better in large networks. VANET routing protocols inherit the problems of traditional routing protocols of Mobile Ad-hoc Networks (MANETs) and face new problems because of the aforementioned unique characteristics of urban VANETs. Thus, the research issues of VANET routing protocols should be identified and appropriate solutions should be introduced.

The focus of this work is on the geo-routing protocols for urban VANETs. My methodology is to analyze and evaluate the VANET geo-routing protocols based on simulations employing the network simulator Objective Modular Network Testbed in C++ (OMNeT++) and the road traffic simulator Simulation of Urban MObility (SUMO).

This work studies the ETSI standards regarding ITS and also studies and classifies the state of the art VANET routing protocols to identify their research issues. The most outstanding VANET geo-routing protocols are implemented in order to be evaluated. Afterwards, it introduces the Enhanced Intersection-based Perimeter Forwarding (EIPG) and subsequently the EIPG2 geoUnicast routing protocols to address the problem of Wrong Street Estimation (WSE), routing loop and partitioned networks. Finally, it introduces the Unicast-Assisted GeoBroadcast (UAG) geoBroadcast routing protocol to address the problem of broadcast storming.

Keywords: Geo-routing Protocols, Car-2-X Communication, GeoNetworking, VANET

ZUSAMMENFASSUNG

Wie von dem ETSI berichtet, nimmt der Straßenverkehr ständig zu, was ernsthafte Probleme verursacht, zum Beispiel Straßenstau, Verkehrssicherheit und Umwelteinflüsse. Unabhängige Fahrerassistenzsysteme sind vorteilhaft, aber die Entwicklung von ITS zielt darauf ab, die Kommunikation und die Zusammenarbeit zwischen Fahrzeugen zu ermöglichen, um die Verkehrsmanagement-, Verkehrssicherheits- und Komfortanwendungen bereitzustellen. ETSI hat C-ITS Standards für ITS in Europa entwickelt.

In VANETs bauen Fahrzeuge dynamisch ein Ad-hoc-Netzwerk ohne Infrastruktur auf. Fahrzeuge bewegen sich schnell und sind innerhalb des Straßenlayouts eingeschränkt, was zu häufigen Netzunterbrechungen führt. Darüber hinaus leiden Fahrzeuge in urbanen Szenarien von der Wirkung der Abschattung durch Gebäude. Wenn die Quelle und das Ziel eines Datenpakets ausserhalb des Kommunikationsbereichs von einander liegen, sollten andere Fahrzeuge dazwischen das Paket von der Quelle empfangen und es durch das Netzwerk zum Ziel weiterleiten. Daher benötigen Fahrzeuge Routing Protokolle, die ihnen helfen, kurze, robuste und zuverlässige Routen zur Übermittlung der Datenpakete zu finden.

Geo-Routing ist für Netzwerke mit hoher Mobilität und häufigen Topologieänderungen geeignet. Darüber hinaus verlassen sich Geo-Routing Protokolle nur auf die geographischen Positionsinformationen von Fahrzeugen. Daher skalieren sie besser in großen Netzwerken. VANET Routing Protokolle erben die Probleme herkömmlicher Routing Protokolle von MANETs und stehen aufgrund der oben erwähnten einzigartigen Merkmale von urbanen VANETs neuen Problemen gegenüber. Daher sollten die Forschungsprobleme der VANET Routing-Protokolle identifiziert und geeignete Lösungen eingeführt werden.

Der Schwerpunkt dieser Arbeit liegt auf den Geo-Routing Protokollen für urbane VANETs. Meine Methodik besteht darin, die VANET Geo-Routing Protokolle basierend auf Simulationen mit dem Netzwerk-Simulator OMNeT++ und dem Straßenverkehrs-Simulator SUMO zu analysieren und zu bewerten.

Diese Arbeit untersucht die ETSI Standards in Bezug auf ITS und untersucht und klassifiziert die VANET Routing Protokolle, die zum Stand der Technik gehören, um ihre Forschungsprobleme zu identifizieren. Die ausstehendeste VANET Geo-Routing Protokolle wurden implementiert und evaluiert. Danach werden die EIPG und anschliessend EIPG2 geoUnicast Routing Protokolle eingeführt, um das Problem von WSE, Routing-Schleifen und partitionierten Netzwerken zu lösen. Abschließend führt diese Arbeit das UAG geoBroadcast Routing Protokoll ein, um das Problem des "broadcast storming" anzugehen.

Schlagwörter: Fahrzeug-Ad-hoc-Netzwerke, Geo-Routing Protokolle, Car-to-X Kommunikation

CONTENTS

i DISSERTATION 1

1 INTRODUCTION 3

1.1 Motivation . . . 4

1.2 Approach . . . 4

1.3 Contribution . . . 5

1.4 Methodology . . . 6

1.5 Outline . . . 6

2 STANDARDIZATION 9

2.1 C-ITS standards in Europe . . . 9

2.1.1 C-ITS Application Layer . . . 14

2.1.2 C-ITS Facilities Layer . . . 17

2.1.3 C-ITS Networking and Transport Layer . . . 26

2.1.4 C-ITS Access Layer . . . 33

2.1.5 C-ITS Management Entity . . . 36

2.1.6 C-ITS Security Entity . . . 37

3 STATE OF THE ART AND CLASSIFICATION OF PROTOCOLS 39

3.1 GeoBroadcast Protocols . . . 40

3.2 GeoUnicast Protocols . . . 44

3.2.1 Research Issues . . . 44

3.2.2 Input Information of Geo-routing Protocols . . . 45

4 GEOUNICAST 59

4.1 Simulation Environment . . . 59

4.1.1 Network simulator . . . 59

4.1.2 Traffic simulator . . . 60

4.1.3 Simulation Scenarios . . . 60

4.2 Outstanding GeoUnicast Routing Protocols . . . 62

4.3 EIPG . . . 79

4.3.1 Wrong Street Estimation . . . 80

4.3.2 Simulation Evaluations . . . 82

4.4 EIPG2 . . . 85

4.4.1 Greedy Forwarding Mode . . . 86

4.4.2 Perimeter Forwarding Mode . . . 87

4.4.3 Carry Forwarding Mode . . . 88

4.4.4 Cross-link Detection . . . 89

4.4.5 Loop Detection . . . 90

4.4.6 Simulation Evaluations . . . 92

5 GEOBROADCAST 99

5.1 GeoBroadcast Destination Area . . . 99

5.2 Outstanding GeoBroadcast Routing Protocols . . . 100

5.3 Unicast-assisted GeoBroadcast . . . 103

5.3.1 Target Position Selection . . . 104

5.3.2 Forwarding-zone Breathing . . . 104

5.3.3 EIPG2 106
5.3.4 Perimeter Forwarding with Left-hand Rule 107
5.3.5 Broadcaster and Rebroadcaster Selection 109
5.4 Simulation Evaluations 109
6 CONCLUSION AND OUTLOOK 113

ii APPENDICES 115
A TABLES OF VANET ROUTING PROTOCOLS 117
B PUBLICATIONS 121
C SCIENTIFIC CAREER 123

BIBLIOGRAPHY 125

LIST OF FIGURES

Figure 1.1 Annual road traffic accidents in Germany 3
Figure 1.2 Architecture of the simulation environment 6
Figure 2.1 External networks involved in the ITS architecture . . 10
Figure 2.2 GeoUnicast 12
Figure 2.3 GeoBroadcast 12
Figure 2.4 Emergency vehicle warning 14
Figure 2.5 C-ITS protocol stack 15
Figure 2.6 Different road safety applications 16
Figure 2.7 Age of data at the target vehicle application 17
Figure 2.8 C-ITS facilities layer 18
Figure 2.9 Cooperative awareness basic service 20
Figure 2.10 General structure of a CAM 23
Figure 2.11 General structure of a DENM 26
Figure 2.12 C-ITS networking and transport layer 27
Figure 2.13 Sectorial contention area 32
Figure 2.14 C-ITS access layer 34
Figure 2.15 C-ITS management entity 37
Figure 2.16 C-ITS security entity 38
Figure 3.1 Usage of the direction and velocity information 48
Figure 4.1 Grid street plan (1800 m by 1800 m) 59
Figure 4.2 NLoS path-loss 61
Figure 4.3 Greedy forwarding 62
Figure 4.4 Local optimum 63
Figure 4.5 Perimeter forwarding 64
Figure 4.6 Planar graph 64
Figure 4.7 PDR of GPSR 65
Figure 4.8 Restricted greedy forwarding 65
Figure 4.9 Perimeter forwarding 66
Figure 4.10 PDR of GPCR 67
Figure 4.11 Prediction in GpsrJ+ 68
Figure 4.12 PDR of GpsrJ+ 68
Figure 4.13 Cross-link 69
Figure 4.14 Cross-link frequency [1] 69
Figure 4.15 Cross-link at a road segment 70
Figure 4.16 An example of perimeter forwarding in GeoCross . . . 72
Figure 4.17 New path after removing the cross-link 73
Figure 4.18 PDR of GeoCross 74
Figure 4.19 Cross-link frequency of GeoCross 74
Figure 4.20 Relaying CDP between intersections 76
Figure 4.21 Selection the next intersection 77
Figure 4.22 Forwarding a packet between two intersections 78
Figure 4.23 Recovery strategy of GyTAR 78

Figure 4.24 PDR of basic GyTAR for different α and β 79
Figure 4.25 PDR of basic GyTAR 80
Figure 4.26 Illustration of the WSE problem 81
Figure 4.27 Intersection's node point (from OpenStreetMap & Bing) 81
Figure 4.28 Perimeter forwarding with the help of angle bisectors 82
Figure 4.29 PDR for different number of vehicles 83
Figure 4.30 CDF of end-to-end latency, aggregating all packets . . 84
Figure 4.31 Routing overhead 85
Figure 4.32 Cross-link frequency of EIPG 90
Figure 4.33 An example of perimeter forwarding mode in EIPG2 . 91
Figure 4.34 Loop frequency 92
Figure 4.35 Cross-link frequency in EIPG and GeoCross 93
Figure 4.36 PDR for different number of vehicles 93
Figure 4.37 Loop frequency in EIPG and EIPG2 94
Figure 4.38 PDR for different number of vehicles 95
Figure 4.39 CDF of end-to-end latency, aggregating all packets . . 96
Figure 4.40 Routing overhead 96
Figure 5.1 Geometric shapes defining geographical areas [2] . . . 100
Figure 5.2 Intersection-based packet forwarding in UGAD 102
Figure 5.3 Calculation of the angle θ 102
Figure 5.4 Intersection-based approach 105
Figure 5.5 Road-based approach 105
Figure 5.6 Forwarding-zone breathing 106
Figure 5.7 Left-hand rule or right-hand rule 109
Figure 5.8 Reachability 110
Figure 5.9 Scalability 111

LIST OF TABLES

Table 2.1 Basic set of applications 13
Table 2.2 Frequency allocation in the European union 34
Table 2.4 European channel allocation 35
Table 2.3 Data rates and channel spacing 36
Table 3.1 Urban geoBroadcast routing protocols 43
Table 3.2 Contributions of geo-routing protocols 47
Table 3.3 Input information of geo-routing protocols 54
Table 3.4 Map and traffic information of geo-routing protocols . 55
Table 4.1 Simulation parameters 62
Table A.1 Cluster-based VANET routing protocols 117
Table A.2 Topology-based VANET routing protocols 118
Table A.3 Position-based VANET routing protocols 119

ACRONYMS

A-STAR	Anchor-Based Street and Traffic Aware Routing
ACAR	Adaptive Connectivity Aware Routing
ANSI	American National Standards Institute
AODV	Ad-hoc On-demand Distance Vector
ARIB	Association of Radio Industries and Businesses
ARM	Application-level Role Mobility
ASTM	American Society for Testing and Materials
BRAN	Broadband Radio Access Network
BSA	Basic Set of Application
BSS	Basic Service Set
BTP	Basic Transport Protocol
C-ITS	Cooperative ITS
CAL	Communication Adaptation sub-Layer
CA	Cooperative Awareness
CaF	Carry and Forward
CAG	Cache Agent-based Geocasting
CALM	Communications Access for Land Mobiles
CAM	Cooperative Awareness Message
CAN	Controller Area Network
CAR	Connectivity-Aware Routing
CBF	Contention-Based Forwarding
CDF	Cumulative Distribution Function
CDP	Cell-Density Packet
CLD	Cross-Link Detection
ComS	Communities Service
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance

CTS	Clear To Send
DCC	Decentralized Congestion Control
CEN	Committee for Standardization
DFS	Dynamic Frequency Selection
DI	Delay Intolerant
DLL	Data Link Layer
DEN	Decentralized Environmental Notification
DENM	Decentralized Environmental Notification Message
DPD	Duplicate Packet Detection
DRG	Distributed Robust Geocast
DSRC	Dedicated Short Range Communications
DT	Delay Tolerant
EDCA	Enhanced Distributed Channel Access
EHTAR	Enhanced Hybrid Traffic-Aware Routing
EIPG	Enhanced Intersection-based Perimeter Forwarding
EN	European Norm
ETSI	European Telecommunications Standards Institute
FBC	Forwarding-zone Breathing Coefficient
Fcar	Fuzzy Control AODV-based Routing
G5CC	ITS-G5 Control Channel
G5SC	ITS-G5 Service Channel
GeoDTN+Nav	Geographic DTN Routing with Navigator prediction
GeOpps	Geographical Opportunistic
GeoSVR	Geographic Stateless VANET Routing
GN6ASL	GeoNetworking to IPv6 Adaptation Sub-Layer
GSR	Geographical Source Routing
GLS	Geo-Location Service
GPCR	Greedy Perimeter Coordinator Routing
GPS	Global Positioning System
GPSR	Greedy Perimeter Stateless Routing

GRANT	Greedy Routing with Abstract Neighbor Table
GSPR-MV	Greedy Simplified Perimeter Routing with Moving Vector
GRUV	Geocast Routing in Urban VANET
GyTAR	Greedy Traffic Aware Routing Protocol
HCCA	Hybrid coordination function Controlled Channel Access
HF	High Frequency
HLAR	Hybrid Location-based Routing protocol
HMI	Human Machine Interface
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
I2V	Infrastructure-to-Vehicle
ICRW	Intersection Collision Risk Warning
IEC	International Electrotechnical Commission
IMT	International Mobile Telecommunication
IP	Internet Protocol
ISO	International Organization for Standardization
ITU	International Telecommunication Union
IVI	In-Vehicle Information
ITS	Intelligent Transportation Systems
JARR	Junction-based Adaptive Reactive Routing
KAF	Keep-Alive Forwarding
LCM	Life Cycle Management
LCRW	Longitudinal Collision Risk Warning
LD	Loop Detection
LDM	Local Dynamic Map
LF	Low Frequency
LocT	Location Table
LocTE	Location Table Entry
LoS	Line-of-Sight
LLC	Logical Link Control

LPRV	Localization Prediction-based Routing for VANET
LPV	Local Position Vector
MAC	Medium Access Control
MAE	Management Adaptation Entity
MCS	Modulation and Coding Scheme
MDDV	Mobility-centric Data Dissemination algorithm for Vehicular network
MFR	Most Forward within Radius
MANET	Mobile Ad-hoc Network
MAP	Road Map Message
MBMPR	Map-Based Multi-Path Routing
MoVe	Motion Vector
MOPR	Movement Prediction-based Routing
MORA	Movement-based Routing Algorithm
MSDU	MAC Service Data Unit
NLoS	Non-Line-of-Sight
OFDM	Orthogonal Frequency Division Multiplexing
OMNeT++	Objective Modular Network Testbed in C++
OSI Model	Open Systems Interconnection Model
PCF	Point Coordination Function
PGB	Preferred Group Broadcasting
PDR	Packet Delivery Ratio
PDU	Protocol Data Unit
POTI	Position and Time management
PHY	Physical
QoS	Quality of Service
R2V	Road-to-Vehicle
REAR	Receipt Estimation Alarm Routing
RFC	Requests for Comment
RHS	Road Hazard Signaling

RIVER	Reliable Inter-Vehicular Routing
RLAN	Radio Local Area Network
RREP	Routing Reply
RREQ	Routing Request
RLS	Reactive Location Service
SAE	Security Adaptation Entity
SAM	Service Announcement Message
SCRP	Stable and reliable CDS-based Routing Protocol
SADV	Static-node Assisted Adaptive Data Dissemination protocol for Vehicular networks
SN	Sequence Number
SRR	Stability and Reliability aware Routing
SUMO	Simulation of Urban MObility
RBVT	Road-Based using Vehicular Traffic
RHW	Road Hazard Warning
RSU	Road Side Unit
RTS	Request To Send
RPGR	Road Perception based Geographical Routing
STAR	Spatial and Traffic-Aware Routing
T-TSG	Traffic light based Time Stable Geocast
TCP	Transmission Control Protocol
TIC	Transmission Interval Control
To-Go	Topology-assisted Geo-opportunistic routing
TOPO	Road Topology Message
TPC	Transmit Power Control
TRaCI	Traffic Control Interface
TRC	Transmit Rate Control
TC	Technical Committee
TR	Technical Report
TS	Technical Specification

UP	User Priority
UR	Unroutable Road
UDP	User Datagram Protocol
UAG	Unicast-Assisted GeoBroadcast
UGAD	Urban Geocast based on Adaptive Delay
V2V	Vehicle-to-Vehicle
V2I	Vehicle-to-Infrastructure
V2R	Vehicle-to-Road
V2X	Vehicle-to-X
VADD	Vehicle-Assisted Data Delivery
VANET	Vehicular Ad-hoc Network
VDP	Vehicle Data Provider
WAVE	Wireless Access in Vehicular Environments
WLAN	Wireless Local Area Network
RSU	Road-Side Unit
WG	Working Group
WNPRP	Wagon Next Point Routing Protocol
WSE	Wrong Street Estimation
WSN	Wireless Sensor Network

Part I

DISSERTATION

1

INTRODUCTION

According to the ETSI, road traffic is constantly increasing, which yields to serious problems, e.g., congested roads, road-safety and environmental effects. Figure 1.1 shows the total annual number of the road traffic accidents in Germany from 2012 until 2017 [3]. According to this diagram, number of the annual road traffic accidents in Germany increases every year since 2014.

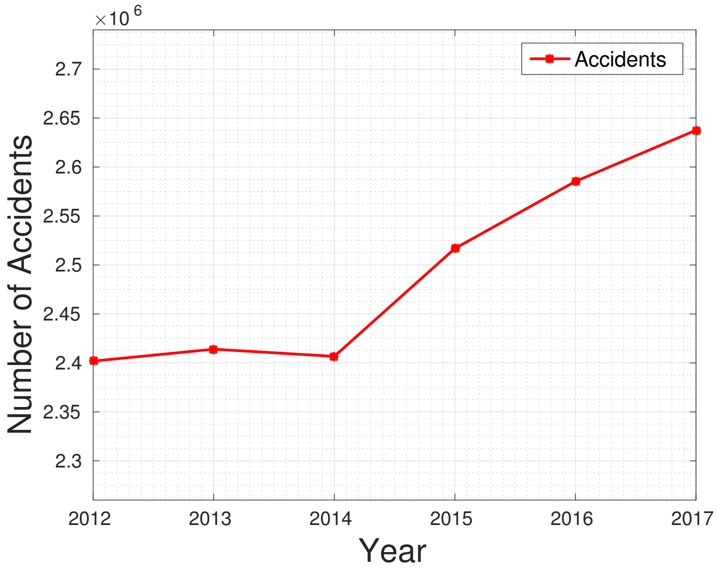


Figure 1.1: Annual road traffic accidents in Germany

Stand-alone driver assistance systems have several benefits, e.g., maintaining a safe speed and distance. These advantages can be boosted by means of cooperation between vehicles. The development of ITS aims to enable the Vehicle-to-Vehicle (V2V), Infrastructure-to-Vehicle (I2V) and Vehicle-to-Infrastructure (V2I) communications to reduce the number of the accidents and to provide a safer traffic environment, exchanging traffic information among the ITS stations. This way, vehicles can exchange information in a VANET, in order to provide the ITS's applications, e.g., traffic management, road-safety and comfort applications.

A series of standards for ITS have been established in Europe, US and Japan. The ITS Info-communications Forum of Association of Radio Industries and Businesses (ARIB) promotes the R&D and standardization of communication technologies in order to the successful introduction of ITS in Japan. American Society for Testing and Materials (ASTM) has developed Dedicated Short Range Communications (DSRC) standards for ITS in the United States. ETSI has developed C-ITS standards for ITS in Europe. The Institute of Electrical and Electronics Engineers (IEEE) 802.11p equivalent in the C-ITS stack covering physical transmission and Medium Access Control (MAC) is called ITS-G5. ITS-G5 is the access technology used in the frequency range from 5.875 GHz to 5.925 GHz that are dedicated for European ITS.

1.1 MOTIVATION

In MANETs, mobile nodes dynamically set up an ad-hoc network without any aid of infrastructure. VANET is a type of MANET, that has two major scenarios, i.e., highway and urban areas. In both of them, vehicles move fast and are constrained within the layout of the roads, which leads to frequent reception failure and network disconnections. Moreover, in urban scenarios, vehicles are facing the shadowing effects of buildings and are suffering from fading phenomena. Besides that, sources and destinations of a data packet may be located outside of each other's communication range, e.g., several intersections away from each other. Therefore, vehicles should work as router and terminal, so that they can receive the packet from the source and relay it through the network towards the destination.

Having several vehicles between the source and destination of a packet, it might result in having more than one possible sequence of vehicles that can relay the packet through the network towards the destination. Thus, the best route to the destination should be selected based on the Quality of Service (QoS) requirements of ITS, which are normally assessed based on three main metrics, i.e., packet delivery ratio, end-to-end latency, and network over-head. Therefore, successful establishment of VANETs depends on routing protocols, which help vehicles to find short, robust and reliable routing paths to deliver the data packets.

VANET routing protocols inherit the problems of traditional routing protocols of MANETs. Moreover, because of the aforementioned unique characteristics of urban VANETs, they are facing new problems. Therefore, the research issues of VANET routing protocols should be identified and appropriate solutions should be introduced.

1.2 APPROACH

While topology-based and cluster-based routing approaches are not suitable for the networks with high mobility and frequent topology changes, geo-routing relies on the geographic position information of vehicles. It works essentially based on the greedy forwarding, i.e., each vehicle tries to forward the data packet to a neighbor vehicle that is closer to the destination. To do so, geo-routing employs the position of the source vehicle, direct neighbor vehicles and the final destination, and as a result of this simplicity, geo-routing protocols scale better in large networks.

According to [4, 5], ETSI has introduced geoNetworking as a family of network protocols that employ the geographical positions for addressing and transport of data packets in VANETs. It works connectionless and fully distributed, and can provide wireless communication between vehicles and/or between vehicles and fixed infrastructures along the roads. GeoNetworking employs geographical positions information to disseminate and transport data packets and to provide multihop communications in order to extend the telematics horizon.

GeoNetworking is appropriate for the networks with high mobility and frequent changes in the network topology, e.g., VANETs. Moreover, it is flexible in supporting different applications and their heterogeneous requirements. In case of road safety and traffic efficiency, geoNetworking provides periodic transmission of safety status messages and fast multi-hop dissemination of emergency warning packets in geographical regions. In case of infotainment, geoNetworking provides the transport of the unicast packets for internet applications.

ETSI European Norm (EN) 302 636-4-1 [6] has introduced geoUnicast and geoBroadcast forwarding algorithms for geoNetworking. But these algorithms does not address the research issues of urban VANETs, e.g., local optimum, and are only applicable in highway scenarios. Therefore, the focus of this work is on the geo-routing protocols for urban VANET.

1.3 CONTRIBUTION

According to ETSI Technical Specification (TS) 102 636-2 [7], two main types of connection in VANETs are defined as geoUnicast, i.e., one ITS station sends a packet to one ITS station, and geoBroadcast, i.e., one ITS station sends a packet to all the ITS stations located in a geographical target area.

The family of greedy perimeter geo-routing protocols became a considerable solution for VANETs. But, there are still shortcomings and issues, that need to be addressed. Also, the existing geoBroadcast protocols have some drawbacks that can be critical, e.g., they are optimized for a specific scenario or they perform worse in urban VANETs.

In this work, I study the relevant ETSI standards regarding ITS and also study the state of the art VANET routing protocols, especially the VANET geo-routing protocols. Because of the high number of the proposed VANET geo-routing protocols, it is necessary to have a comprehensive collection of the materials on this topic to provide a solid background for a research work's investigation. Therefore, I identify the research issues of VANET geo-routing protocols and classify them based on their specifications, approaches and also the research issues that they have addressed. Finally, I have implemented the most outstanding VANET geo-routing protocols in order to evaluate them.

Afterwards, I introduce the EIPG geoUnicast routing protocol that employs a new intersection-based perimeter forwarding in order to avoid the problem of WSE. EIPG shows a significant improvement in comparison to its predecessors, in terms of Packet Delivery Ratio (PDR), end-to-end delay, and network overhead. Subsequently, I introduce the EIPG2 geoUnicast routing protocol, that employs a new preferential unrestricted greedy forwarding and applies a new preferential intersection-based perimeter forwarding based on the intersection-based perimeter forwarding of EIPG. EIPG2 also adopts the Cross-Link Detection (CLD) mechanism and takes advantage of new Loop Detection (LD) and Carry and Forward (CaF) approaches to show a significant improvement in comparison to EIPG, in terms of PDR, end-to-end delay, and network overhead.

Finally, I introduce the UAG geoBroadcast routing protocol to address the problem of broadcast storming. UAG proposes to send the message from the source vehicle to different target positions within the geo-region with the help of geoUnicast, e.g., EIPG2. Afterwards, some of the vehicles located in the geo-region are selected to broadcast the message within the geo-region. I proposed intersection-based and road-based approaches to select the target positions. Moreover, I propose the forwarding-zone breathing based on the road topology of the geo-region and its neighborhood in order to increase the reachability. Also, I propose that vehicles employ both right-hand and left-hand rule in perimeter forwarding so that these message copies take different routes towards the geo-region in order to increase the chance of reaching the geo-region. UAG shows a better performance in terms of reachability and scalability, comparing to simple flooding and Urban Geocast based on Adaptive Delay (UGAD).

1.4 METHODOLOGY

The methodology of this work is to analyze and evaluate the VANET routing protocols based on simulations. For this purpose and as depicted in Figure 1.2, the network simulator OMNeT++ and the road traffic simulator SUMO are coupled employing Traffic Control Interface (TRaCI). In my simulations, OMNeT++ works based on the ETSI documentations regarding ITS, and SUMO generates vehicles with random trips and random mobility routes.

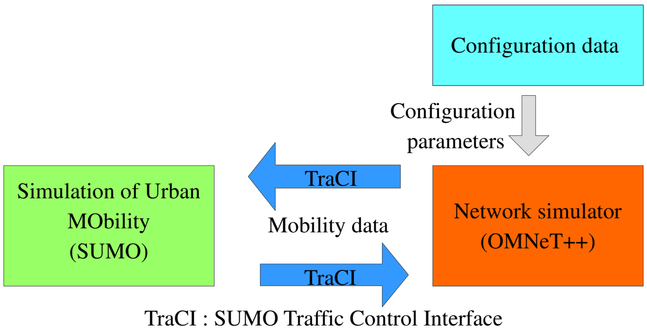


Figure 1.2: Architecture of the simulation environment

1.5 OUTLINE

Chapter 2 discusses the state of the art ETSI standards regarding ITS and explains the necessary technological background, i.e., different types of networks, different types of ITS stations, different types of communications and the different layers of the C-ITS and their most important entities, e.g., Cooperative Awareness Message (CAM), Decentralized Environmental Notification Message (DENM) and geoNetworking.

Chapter 3 discusses the state of the art geoBroadcast and geoUnicast routing protocols and identifies their research issues and classifies these routing protocols based on the research issues they have addressed and their approaches.

Chapter 4 introduces my first proposed geoUnicast routing protocol, i.e., EIPG, that addresses the WSE problem in the implementation of the right-hand rule during the permiter forwarding when a local optimum happens. Subsequently, this chapter introduces my second proposed geoUnicast routing protocol, i.e., EIPG2, that addresses the problem of partitioned networks, routing loops and cross-links at empty intersections. This chapter discusses their algorithms and the research issues that they address. Moreover, it discusses the most outstanding geoUnicast routing protocols and shows their simulation evaluations and results, comparing them with EIPG and EIPG2.

Chapter 5 introduces my proposed geoBroadcast routing protocol, i.e., UAG, that addresses the broadcast storming problem. This chapter discusses the algorithm of UAG and the most outstanding geoBroadcast routing protocols and shows UAG's simulation evaluations and results, comparing it with other geoBroadcast routing protocols.

2

STANDARDIZATION

The development of ITS aims to enable the V2V, I2V and V2I communications to reduce the number of accidents and to provide a safer traffic environment, exchanging traffic information among the ITS stations. V2I and I2V are defined as the "direct vehicle to road infrastructure communication using a wireless local area network such as standardized in EN 302 571" [8]. V2V is defined as the "direct vehicle(s) to vehicle(s) communication using a wireless local area network such as standardized in EN 302 571" [8].

A key concept in ITS is the "co-operative awareness", i.e., the transport entities, e.g., vehicles, roadside infrastructure, pedestrians, etc. should be able to collect knowledge of their local environment and also from a range of sensor equipment, and to share it in order to use the transport infrastructure in a more intelligent way.

A series of standards for ITS have been established in Europe, US and Japan. The ITS Info-communications Forum of ARIB promotes the R&D and standardization of communications technologies in order to the successful introduction of ITS in the frequency range from 755 MHz to 765 MHz in Japan.

ASTM has developed DSRC standards for ITS in US. The access technology in US operates in the frequency range from 5.850 GHz to 5.925 GHz. IEEE 802.11p defines the physical transmission and MAC. Besides the well known networking protocols, e.g., Internet Protocol (IP), User Datagram Protocol (UDP) and Transmission Control Protocol (TCP), many V2V and V2I applications apply direct communication among ITS stations. Thus, the IEEE 1609 Wireless Access in Vehicular Environments (WAVE) series of standards has been developed [9].

ETSI has developed C-ITS standards for ITS in Europe. The IEEE 802.11p equivalent in the C-ITS stack covering physical transmission and MAC is called ITS-G5. ITS-G5 is the access technology used in the frequency range from 5.875 GHz to 5.925 GHz that are dedicated for European ITS [9].

However, the parallel development of V2X in U.S. and Europe resulted in different protocol stacks, IEEE 802.11p and ITS-G5 have the same key technology features. In Physical (PHY) layer, they apply Orthogonal Frequency Division Multiplexing (OFDM) with the same parameter sets. In MAC layer, ITS-G5 also considers Enhanced Distributed Channel Access (EDCA) with Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA) and data traffic prioritization is possible with the help of access categories.

2.1 C-ITS STANDARDS IN EUROPE

According to [10, 11], the ITS network architecture is composed of the internal and external networks. An internal network interconnects the components of an ITS station. External networks interconnect ITS stations to each other. They can also connect ITS stations to other network entities. Furthermore, the external networks can be categorized into an ITS domain and a generic domain as specified in ETSI EN 302 665 [12]. Figure 2.1 illustrates the abstraction of the ITS network architecture in the highest level. An ITS station can also be connected to proprietary local networks of the vehicle ITS sub-systems, e.g., Controller Area Network (CAN). The external networks are classified as follows:

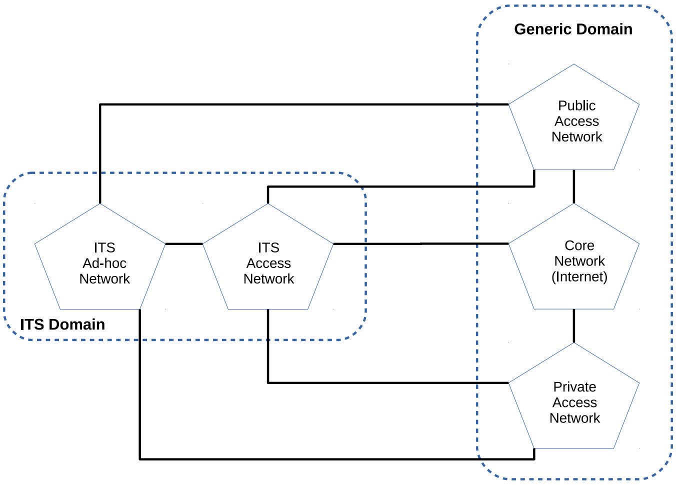


Figure 2.1: External networks involved in the ITS architecture

- *ITS ad-hoc network*: A special type of MANET enabling self-organized communication between ITS stations without any coordinating communication infrastructure.
- *ITS access network*: A communication network in order to interconnect the roadside ITS stations according to the ITS specifications. It can optionally interconnects the roadside ITS stations to the core network.
- *Public access network*: A communication network in order to provide access to the publicly accessible general purpose networks, e.g., an International Mobile Telecommunication (IMT)-2000 network as defined in International Telecommunication Union (ITU)-R M.687-2 [13] which provides mobile internet access and connects the vehicle ITS stations to the internet.
- *Private access network*: A communication network which provides data services to a closed user group in order to have a secured access to another system, e.g., a private access network to connect the vehicle ITS stations to a company's intranet.
- *Core network*: For example the internet.

Considering the ITS domain and based on ETSI EN 302 665 [12], there are four types of ITS stations:

- A central ITS station that plays the role of an operator and provides the centralized Basic Set of Application (BSA) of ITS applications,
- A roadside ITS station that operates independently or co-operatively with the central ITS station or other roadside ITS stations to provide the ITS applications for roadside,
- A vehicle ITS station that provides ITS applications to drivers and passengers, and
- A personal ITS station that provides the ITS application to personal and their mobile devices

These four types of ITS stations, can be called in the following ways depending on their role in the VANET:

- *Sender ITS station*: An ITS station that has sent the geoNetworking packet.
- *Source ITS station*: A sender ITS station that originates a geoNetworking packet.
- *Forwarder ITS station*: An ITS station that processes a packet and relays it to other ITS stations.
- *Receiver ITS station*: An ITS station that processes a packet and delivers its data to the upper protocol entities.
- *Destination ITS station*: A receiver ITS station that does not relay the packet to other ITS stations.
- *Neighbor ITS station*: An ITS station within the direct (single-hop) communication range.

According to [7, 14], and considering the different types of communication endpoints, the focus of the geoNetworking is on the following scenarios:

- *V2V communication*: an ITS vehicle station communicates with other ITS vehicle stations
- *Road-to-Vehicle (R2V) communication*: an ITS roadside station communicates with ITS vehicle stations
- *Vehicle-to-Road (V2R) communication*: an ITS vehicle station communicates with ITS roadside stations

Based on ETSI TS 102 636-2 [7], four types of connection can be assumed between the aforementioned ITS stations:

- *Point-to-point*: An ITS station sends a packet to another ITS station. (Figure 2.2)
- *Point-to-multipoint*: An ITS station sends a packet to multiple ITS stations.
- *GeoAnycast*: An ITS station sends a packet to an arbitrary ITS station within a geographical target area.

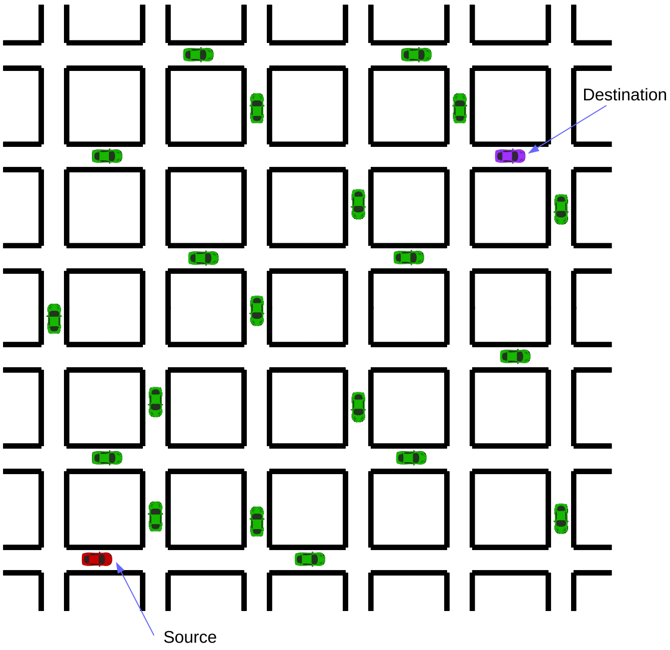


Figure 2.2: GeoUnicast

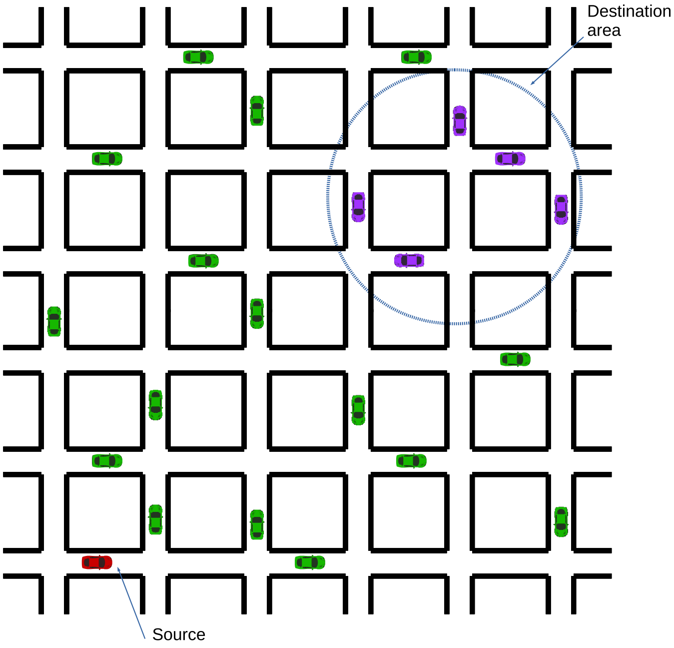


Figure 2.3: GeoBroadcast

- *GeoBroadcast*: An ITS station sends a packet to all the ITS stations within a geographical target area. (Figure 2.3)

The ETSI Technical Report (TR) 102 638 [8] has defined the BSAs within the scope of the European Mandate M/453 [15] to identify different application classes, and their applications and use cases. An ITS application is defined as a "system that defines and implements an ITS service to users of the system [8]". An ITS use case is defined as a "procedure of executing an application in a particular situation with a specific purpose [8]". Table 2.1 shows the defined application classes.

Table 2.1: Basic set of applications

Applications Class	**Application**
Active road safety	Driving assistance - Co-operative awareness
	Driving assistance - Road Hazard Warning
Cooperative traffic efficiency	Speed management
	Co-operative navigation
Co-operative local services	Location based services
Global internet services	Communities services
	ITS station life cycle management

According to ETSI TS 102 637-1 [16], active road safety applications employ CAMs and DENMs which are explained in Sections 2.1.2.4 and 2.1.2.5 respectively. Driving assistance - co-operative awareness applications assist drivers to be aware of the presence of other vehicles or situations in their vicinity, e.g., slow vehicle approaching. This application employ the periodically broadcasted CAMs sent from ITS stations and might send complementary DENMs to send the situation information to a longer distance or to provide additional information. Road Hazard Warning (RHW) applications assist drivers providing them information on the road hazard events. Moreover, RHW applications can provide information for the traffic management purposes related to hazardous situations. These applications employ the Decentralized Environmental Notification (DEN) basic service and the dissemination of the DENMs.

Co-operative traffic efficiency applications mainly provide traffic information from a roadside ITS station to vehicle ITS stations or personal ITS stations. Also, they might need communication between a roadside ITS station and a central ITS station.

Co-operative local services and global internet services classes have the role of advertising and providing on-demand information to the vehicles. This can be done on a commercial or non-commercial basis. Communities Services (ComSs) are mainly services that community service providers, e.g., logistic companies, give to multiple ITS stations that belong the community. When a roadside ITS station sends a Life Cycle Management (LCM) announcement to vehicle ITS stations to inform them of the service availability, a point-to-point session is established with the roadside ITS station to download and update the software and data of the vehicle ITS.

"Emergency vehicle warning" is an example of a use case from the "driving assistance - Co-operative awareness" applications. It is a periodic (10

Hz) broadcast V2X co-operative awareness message that is triggered by the emergency vehicle. As depicted in Figure 2.4, all the relevant ITS stations should receive it in less than 100 ms and process it. This way an active emergency vehicle can indicate its presence so that the other vehicles on the way of the emergency vehicle can drive accordingly and make an emergency corridor.

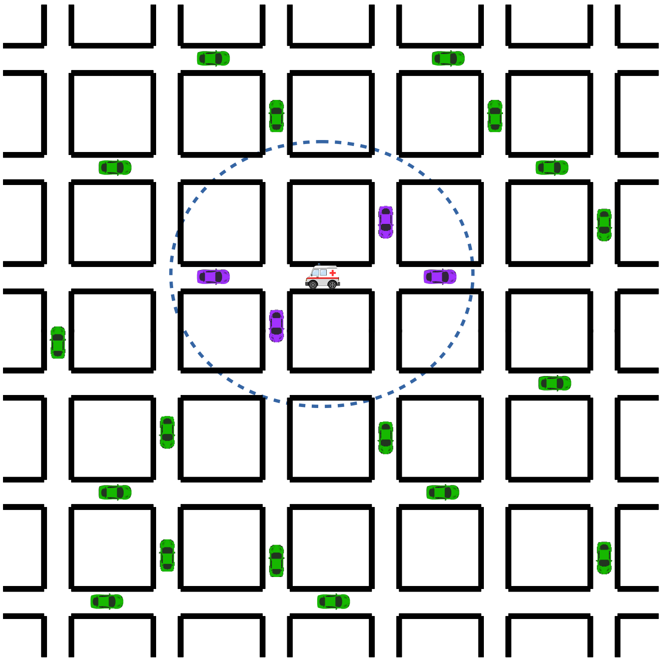


Figure 2.4: Emergency vehicle warning

Another example of a use case is the ITS local electronic commerce from the location based services. This use case is associated to other applications, e.g., parking management. ITS local electronic commerce sometimes imply a transaction with a financial service that requires a point-to-point communication session between the client ITS station and the local service provider that has the authorization to receive the payment.

According to [12, 10], the protocol stack of an ITS station basically follows the International Organization for Standardization (ISO)/Open Systems Interconnection Model (OSI Model). As specified in Figure 2.5, it defines four horizontal protocol layers and two vertical protocol entities.

2.1.1 *C-ITS Application Layer*

According to [8, 10], the applications layer provides the ITS services. ITS application is defined as a "system that defines and implements an ITS service to users of the system" [8]. ITS use cases are defined as the "procedure of executing an application in a particular situation with a specific purpose" [8]. Three classes of applications have been distinguished at this level:

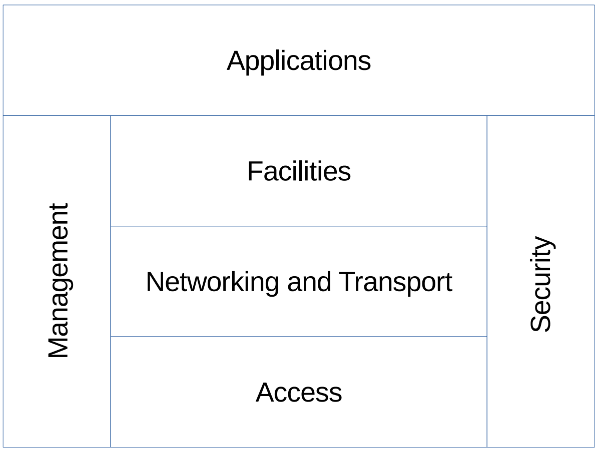


Figure 2.5: C-ITS protocol stack

Road safety, traffic efficiency and other applications, e.g., infotainment and business.

2.1.1.1 *Primary Road Safety Applications*

According to [17, 18, 19], in order to reduce the risk of collision and to improve the road safety, ITS introduced the primary road safety applications to provide the following services:

- *The "driver information":* that can be done through the digital radio broadcast channels.
- *The In-Vehicle Information (IVI):* that provides the static road signage or variable message sign information.
- *The "driver awareness":* that can be done by the Road Hazard Signaling (RHS) applications.
- *The "driver warning":* that can be done by the Intersection Collision Risk Warning (ICRW) and the Longitudinal Collision Risk Warning (LCRW) applications.
- *The "direct action":* that can be taken by the vehicle's system to avoid a possible collision without warning the user.

RHS application is an application layer entity of the ITS stations that provides road hazard informations to users by triggering the transmission of DENMs and processing the received DENMs. The originating ITS station detects a road hazard and triggers the related DENMs transmission in order to signal to all the vehicle ITS stations that are moving in the relevance area. The DEN basic service determines the relevance area based on the event position, heading direction and trace data elements.

The RHS application might also request adjustments to the CAM interval to enhance the confidence level for the receiving ITS stations. On the side of the receiving ITS stations, RHS application should allow the driver awareness that does not require an immediate action.

The specifications of the ITS RHS application based on the Cooperative Awareness (CA) basic service and the DEN basic service is specified in ETSI TS 101 539-1 [17]. It also considers the performance requirements in case of the generation and transmission of CAMs and DENMs in order to enable the different levels of RHS, i.e., classes A and B.

The ETSI Technical Committee (TC) ITS Working Group (WG)1 and Committee for Standardization (CEN) TC 278 WG16 develop standards in the scope of the M/453 mandate [15]. As depicted in Figure 2.6, an originating ITS station might serve any application categories, i.e., driving assistance or direct control, but does not necessarily know which one will be implemented on the side of the receiving ITS stations. (IVS stands for In Vehicle Signage)

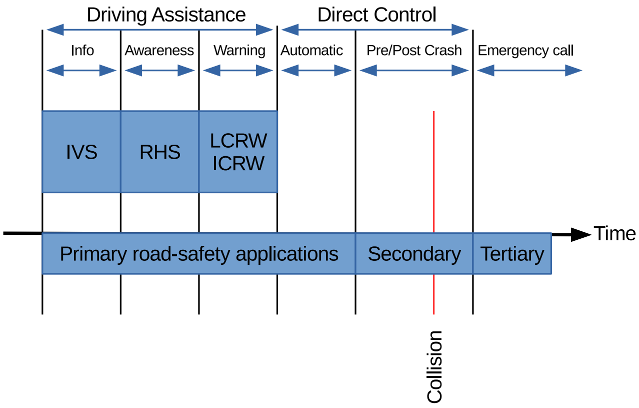


Figure 2.6: Different road safety applications

In such applications, an important quality parameter is the *age of data*, i.e., the "difference between the time of a data element value setting at the originating ITS station and the time the same data element value is used to undertake an appropriate action at a receiving ITS station level [17]". As depicted in Figure 2.7, the age of data depends on the distributed application end to end latency time that can be calculated on the side of the receiving ITS stations. Therefore, it is possible to consider the system performance in the originating ITS station, in the receiving ITS station, and in the wireless network. Where, the wireless network performance might vary depending on the network characteristics, available radio obstacles and the network load.

Figure 2.7 shows the different time stamps, i.e., T_0 to T_6. T_0 is measured at the electronic sensor of the vehicle ITS station. T_1 gives the time-stamp of CAM and DENM. T_2 is measured when the messages are transmitted on the

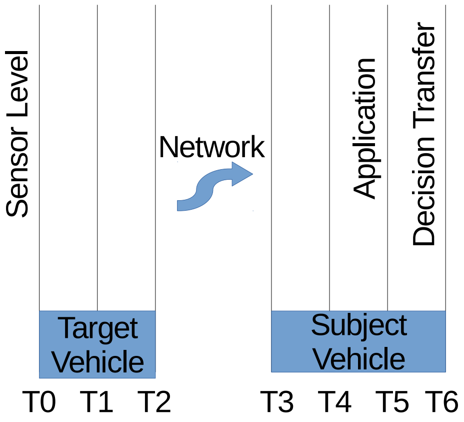


Figure 2.7: Age of data at the target vehicle application

air. T_3 is measured when the messages are received by the access layer of the receiving ITS station. T_4 is measured when the message data is received by the facilities layer of the receiving ITS station. T_5 is measured when the application is finished with processing the message data. T_6 is measured when a warning is presented on the Human Machine Interface (HMI) of the vehicle ITS station or a direct action is requested to the electronic system of the vehicle ITS station.

It is possible to calculate the the age of data on the side of the receiving ITS station as T_5 to T_1. Although, T_0 to T_1 is not included in this calculation, it is possible to estimate this duration and add it to the final calculations. Thus, considering an uncontested network and assuming a confidence interval of 95%, two performance classes for the ITS stations are specified: Class A should be able to guarantee a $T_1 - T_0 < 150$ ms, corresponding to the minimum CAM and DENM time interval, i.e., 100 ms. Class B should be able to guarantee a $T_1 - T_0 < 1.5$ s, corresponding to the fact that the CAM and DENM time interval should be set between 100 ms and 1 s.

As stated in ETSI TS 101 539-1 [17] and ETSI TS 101 539-3 [19], an estimated end to end latency time of 300 ms should be required to avoid the false decisions made by the critical road safety applications, e.g., collision avoidance and pre-crash applications, assuming that the speed of the involved vehicle ITS stations is less than 130 km/h. The latency time of $T_6 - T_4$ depends on the implementation of the receiving ITS station and in a critical road safety situation should be less than 80 ms.

2.1.2 *C-ITS Facilities Layer*

According to [8, 12], the facilities layer contains functionality from the OSI Model application layer and the OSI Model presentation layer, e.g., encoding, decoding, and encryption, and the OSI Model session layer, e.g., inter-host communication. It also has some amendments dedicated to ITS communications to provide generic support facilities to applications. This layer is further composed of three main components: application support, information support and communication support.

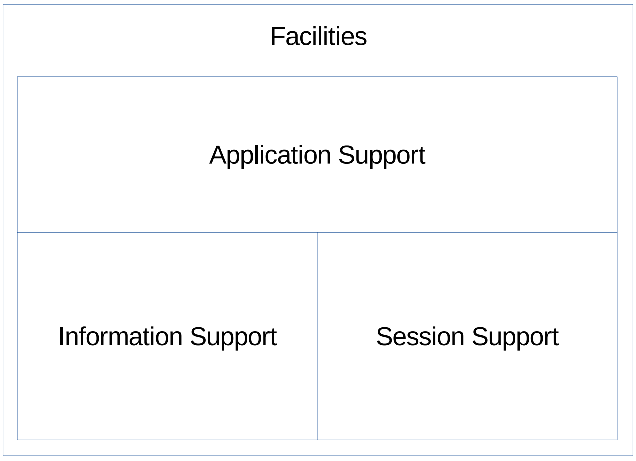


Figure 2.8: C-ITS facilities layer

2.1.2.1 *Application Support*

Application support is the kernel for the common functions that support the applications.

Station positioning is one of the main facilities of the application support that provides the absolute 3D position information of the ITS stations, i.e., latitude, longitude, and altitude. These position information can be (near) real time, e.g., from the satellite position systems, or can be pre-configured, e.g., for the fixed infrastructure elements. Some of the road safety applications may need a specific confidence level, i.e., a high accuracy of positioning, e.g., less than 0.5 m. This can be obtained combining several means, e.g., GNSS[1], odometer[2], gyro[3], etc.

Mobile station dynamic monitoring is another support facility that permanently monitors the real time evolution of the relevant vehicle electronic information, e.g., braking system, steering system, acceleration control, speed control, etc. and is able to update the parameters that are used for CAM.

Station state monitoring permanently monitors the current static state of the ITS station. This monitoring can be performed directly, e.g., the light bar and siren states of a service vehicle, or through some vehicle in the network. It is also able to update the parameters that are used for CAM.

The Local Dynamic Map (LDM) management facility is responsible for permanently updating the LDM data base. LDM is updated according to the

---

[1] Global Navigation Satellite System
[2] An instrument to measure the distance traveled by a vehicle
[3] An instrument to measure orientation and angular velocity

local information and the remote information that the ITS station obtains from V2X communication messages. This LDM might include lane-specific information including curves, and road furniture, e.g., traffic signs and traffic lights. Moreover, all the dynamic objects should have a reference to the LDM. These objects might be sensed directly or indirectly (co-operative awareness). LDM can be beneficial for safety critical applications.

2.1.2.2 *Information Support*

Information support is mainly consist of repositories, e.g., LDM data base, and associated processing capabilities and has the role of data management.

2.1.2.3 *Communication Support*

Communication support cooperates with the networking and transport layer to gain the various communication modes, e.g., broadcasting, geocasting, and unicasting, and to support functions such as addressing of vehicles, areas and other multicast groups, and transparent routing in ad-hoc fashion.

2.1.2.4 *Cooperative Awareness Basic Service*

According to ETSI EN 302 637-2 [20], cooperative awareness in road traffic suggests that the road users, e.g., cars, trucks, motorcycles, bicycles and pedestrians, also roadside infrastructure, e.g., road signs, traffic lights or barriers, and gates, are informed about the position, dynamics and attributes of each other. This cooperative awareness is the basis for the road safety and traffic efficiency applications with several use cases as described in ETSI TR 102 638 [8].

At startup each ITS station sends an initial CAM to announce its presence to other ITS stations. CAMs should be sent periodically excepting that the ITS station sends another geoNetworking packet which includes the Local Position Vector (LPV) of the ITS station.

The periodically transmitted CAMs contain the information to be exchanged for cooperative awareness. The cooperative awareness basic service constructs, manages and processes the CAMs. It is a part of the facilities layer of the ITS communication architecture as described in ETSI EN 302 665 [12]. All types of ITS stations taking part in the road traffic, have the cooperative awareness basic service as a compulsory facility. The BSA as defined in ETSI TR 102 638 [8], specifies the requirements of the performance of the cooperative awareness basic service and the contents of the CAMs.

Depending on the type of ITS station, e.g., vehicle ITS station, road side ITS station and personal ITS station, the cooperative awareness basic service activation might be different. The cooperative awareness basic service should manage and trigger the CAM generation, as long as it is active. In case of a vehicle ITS station, the cooperative awareness basic service should be activated and terminated with the activation and deactivation of the ITS station.

As defined in ETSI EN 302 665 [12], the cooperative awareness basic service is an entity in the facilities layer of the ITS station architecture. In order to collect the related information for CAM generation and also in order to forward the contents of the received CAM for further processing, the cooperative awareness basic service might have interfaces with other entities in the facilities layer and also with the application layer. It is depicted in Figure 2.9.

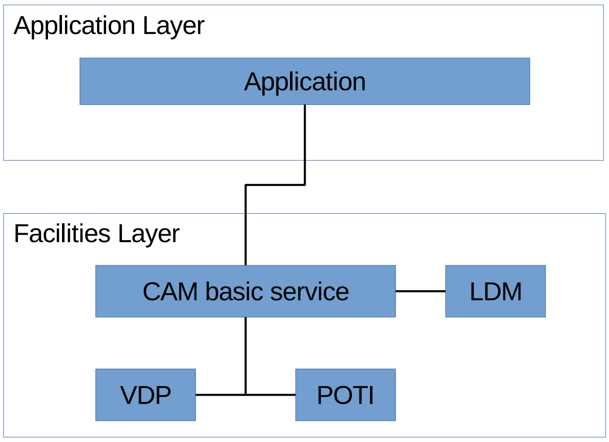


Figure 2.9: Cooperative awareness basic service

Data collection entities of a vehicle ITS station might be the Vehicle Data Provider (VDP), the Position and Time management (POTI) and the LDM. The VDP provides the status information of vehicle and it is connected to the vehicle network. The POTI provides the position and time information of the ITS station, as specified in ETSI TS 102 890-3 [21]. The LDM is a database in the ITS station, that might be updated by the information included in the received CAMs as described in ETSI TR 102 863 [22].

CAMs are "messages exchanged in the ITS network between ITS stations to create and maintain awareness of each other and to support cooperative performance of vehicles using the road network [20]". The status and attribute information of the originating ITS station is included in a CAM. These information differ based on the type of the ITS station. For example, in case of a vehicle ITS station, the status information can be the time, position, motion state, and activated systems of the originating vehicle. The attribute information can be about the dimensions, type and the role of the vehicle in the road traffic.

The cooperative awareness basic service of the receiving ITS station makes the content of the CAMs accessible for its ITS applications and other facilities, e.g., LDM. Thus, the ITS stations that receive the CAM become informed about the type, presence, and the status of the originating ITS station. The receiving ITS stations can employ these information to support several ITS applications. The cooperative awareness of other ITS stations can also be used in the networking and transport layer for the purpose of the position dependent dissemination of messages, e.g., geoBroadcasting of DENMs as specified in ETSI EN 302 636-4-1 [6].

After generating a CAM, the originating ITS station delivers it to the ITS networking and transport layer for dissemination, that might differ depending on the applied communication system. As specified in ETSI TS 102 636-3 [10], the originating ITS station should employ the point-to-multipoint communication to transmit CAMs. As defined in ETSI EN 302 663 [23], in case of ITS-G5, ITS-G5 Control Channel (G5CC) should be used and the originating ITS station sends the CAMs in a single hop to all the ITS stations within the direct communication range, i.e., the direct neighbor ITS stations. This communication range of the ITS stations might, inter alia, change by changing the transmit power of the originating ITS station. A received CAM shall not be forwarded to other ITS stations.

The originating ITS station periodically generates CAMs with a frequency that is controlled by its cooperative awareness basic service. Decentralized Congestion Control (DCC) determines the generation frequency based on the status change of the originating ITS station, e.g., change of position or speed, and the radio channel load. DCC assures the network stability, throughput efficiency and fair resource allocation to ITS stations by limiting the channel load.

The cooperative awareness basic service manages the CAM generation frequency which is the time interval between the generations of two consecutive CAMs. It sets the upper and lower limits of the transmission interval of the CAM generation according to the requirements as specified in ETSI TS 101 539-1 [17], ETSI TS 101 539-2 [18] and ETSI TS 101 539-3 [19]. "The CAM generation interval shall not be inferior to T_GenCamMin = 100 ms. This corresponds to the CAM generation rate of 10 Hz [20]". "The CAM generation interval shall not be superior to T_GenCamMax = 1000 ms. This corresponds to the CAM generation rate of 1 Hz [20]". Subject to the dynamics of the originating ITS station and the channel congestion status, the CAM generation should be triggered within these limits.

According to the channel usage requirements of DCC as specified in ETSI TS 102 724 [24], in order to reduce the CAM generation, T_GenCam_Dcc should be the minimum time interval between two consecutive CAM generations. This way, in case of channel congestion, it is possible to adjust the CAM generation rate to the remaining capacity of the radio channel. The management entity provides T_GenCam_Dcc in the unit of milliseconds in the range of:

$$T\_GenCamMin \leqslant T\_GenCam\_DCC \leqslant T\_GenCamMax$$

In a more precise way, there are two conditions to be followed for the purpose of the CAM generation trigger. If one of the following two conditions is met, a CAM should be generated immediately:

1) The time elapsed since the last CAM generation is greater than or equal to T_GenCam_Dcc and one of the following conditions regarding the dynamics of the ITS station is satisfied:

- "The absolute difference between the current heading of the originating ITS station and the heading included in the CAM previously transmitted by the originating ITS station exceeds 4° [20]"
- "The distance between the current position of the originating ITS station and the position included in the CAM previously transmitted by the originating ITS station exceeds 4 m [20]"
- "The absolute difference between the current speed of the originating ITS station and the speed included in the CAM previously transmitted by the originating ITS station exceeds 0.5 m/s [20]"

2) The time elapsed since the last CAM generation is greater than or equal to T_GenCam and greater than or equal to T_GenCam_Dcc.

In case of the Road-Side Unit (RSU) ITS stations, the CAM generation frequency should be determined in a way that the RSU transmits at least one CAM, while a vehicle ITS station is in its communication zone. Again, the time interval should be equal to or smaller than 1000 ms which matches a minimum CAM generation frequency of 1 Hz.

All the CAMs should be time-stamped so that it is possible to ensure the useful interpretation of the received CAMs. The time-stamp should correspond to the time at which the originating ITS station has determined its reference position. Also, the required time for a CAM generation, i.e., the time difference between triggering the CAM generation and the time at which the CAM is delivered to networking transport layer, should be less than 50 ms.

As depicted in Figure 2.10, A common ITS Protocol Data Unit (PDU) header and multiple containers together compose a CAM. The ITS PDU header is a common header including information about the version of the protocol, the type of the message and the ID of the originating ITS station. It should be generated as specified in ETSI TS 102 894-2 [25].

All the RSU ITS stations should generate CAMs including a basic container and optionally more containers. All the vehicle ITS stations should generate CAMs including one basic container, at least a High Frequency (HF) vehicle container, and optionally Low Frequency (LF) vehicle containers or special containers. The containers are specified as follows:

- The basic container consists of the basic information, type and the latest geographic position of the originating ITS station.
- The high frequency container consists of the highly dynamic information of the originating ITS station, e.g., heading and speed.
- The low frequency container consists of the static information of the originating ITS station, e.g., the status of the exterior lights.
- The special vehicle container consists of the information about the role of the originating vehicle ITS station.

Vehicle ITS stations with a specific role in road traffic, e.g., public transport, should add more status information in the special vehicle containers according to the specification in ETSI EN 302 637-2[20].

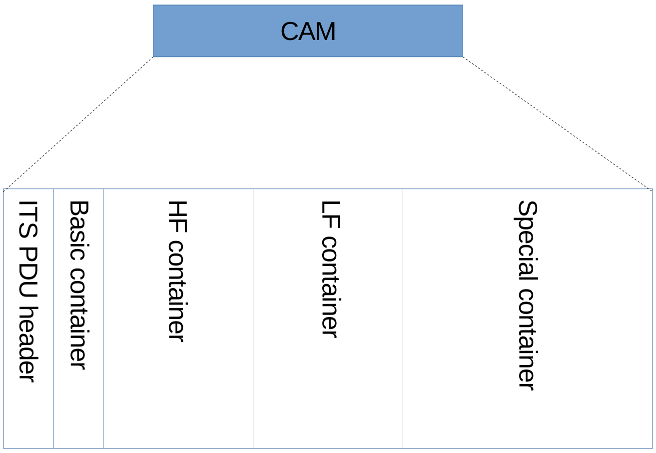


Figure 2.10: General structure of a CAM

2.1.2.5 *Decentralized Environmental Notification Basic Service*

According to [26, 16], ETSI TC ITS defines the DEN basic service to support the RHW applications of BSA. The DEN basic service is an ITS facility layer entity that provides services to the ITS application layer entities. The DEN basic service constructs, manages and processes the DENMs. The trigger to construct a DENM comes from an ITS station application. The information of a road hazard or an abnormal traffic condition, e.g., its type, position, detection time and a time duration, is contained in the DENM.

A DENM should normally be disseminated through V2X communications to ITS stations located in an specific geographic area. For the message dissemination, the DENM is delivered as a payload to the ITS networking and transport layer. The packet centric forwarding functionality of the ITS networking and transport layer forwards a DENM from the originating ITS station to the destination area. The destination area is a geographical area used by the ITS networking and transport layer for the DENM dissemination.

As many ITS stations as possible that are located in the relevance area or entering the relevance area within the validity duration, should receive the DENM. The relevance area is defined as a "geographic area in which information concerning the event is identified as relevant for use or for further distribution [26]". The ITS station application of the originating ITS station sets the relevance area and should include it in the DENM, when the information is available. The received DENM is processed by the DEN basic service of a receiving ITS station in order to provide the content of the DENM to an ITS station application. This information might be presented to the driver if relevant, so that the driver can react to the situation appropriately.

A DENM should at least include one trace, i.e., location referencing information of the event position, as complementary to the relevance area. "A trace contains a list of well-ordered waypoints that forms an itinerary approaching towards the event position [26]". The relevance area determines the geographical area in that DENMs should be transmitted, while the trace is used by the receiving ITS stations to fine tune the notification triggering of the driver awareness.

The DENM reception management is a sub-function of the DEN basic service, that updates the receiving ITS station message table, discards the received invalid DENMs and provides the data of the received DENM to the applications and/or other entities of the facilities layer of the receiving ITS station.

The DENM Keep-Alive Forwarding (KAF) is an optional sub-function of the DEN basic service, that implements the DENM protocol operation of the forwarding ITS station, e.g., storing a received DENM as long as it is still valid, i.e., validity duration is not expired, and forwards the DENM when applicable. Either the ITS applications requirements or a cross-layer functionality of the management entity might define the usage conditions of the KAF.

The general procedure of an ITS use case supported by the DEN basic service is like this:

- When an ITS station detects an event, it transmits a DENM as the originating ITS station. This DENM should be disseminated to the ITS stations located inside the relevance area.

- The transmission of a DENM should be initiated and terminated at the ITS application layer of an ITS station. Examples are given in ETSI TS 101 539-1 [17], ETSI TS 101 539-2 [18] and ETSI TS 101 539-3 [19].

- The transmission of a DENM might be repeated and continued as long as the event is present.

- As long as the the DENM is still valid, the ITS stations should keep the DENM alive inside the relevance area, even if the originating ITS station has moved far away from the event position or has stopped sending DENMs.

- An ITS station application of the originating ITS station might update the transmission of DENMs.

- If the destination ITS stations are not located in the direct communication range of the originating ITS station, the intermediate ITS stations might forward the DENM. The ITS networking and transport layer of the intermediate ITS stations performs this forwarding.

- The termination of DENM transmission can be done in two ways. Either a predefined expiry time is reached, and the termination is automatically achieved by the DEN basic service of the originating ITS station. Or an ITS station application recognizes that the event is not present anymore and requests the generation of a DENM to announce the event termination.

- The receiving ITS station, processes the information contained in a DENM and might present an appropriate warning or information to user, if relevant.

In some cases, more than one originating ITS stations might transmit DENMs related to the same event. Moreover, a mobile originating ITS station, e.g. a vehicle detecting black ice on the road surface, can move to a position far from the event position, although the event persists. In such a case, other ITS stations relay the DENMs. Therefore, the DENM transmission and the originating ITS station can be independent.

The DENM protocol defines four types of DENMs as follows:

- New DENM: "A DENM generated by the DEN basic service when an event is detected by an originating ITS station for the first time [26]". The originating ITS station assigns each new DENM a new actionID as an identifier of a detected event.

- Update DENM: "A DENM generated by the DEN basic service that includes update information of an event [26]". The same originating ITS station that generated the new DENM for an event, might transmit an update DENM for the same event.

- Cancellation DENM: "A DENM that informs the termination of an event [26]". The same originating ITS station that generated the new DENM for an event, transmits a cancellation DENM for the same event.

- Negation DENM: "A DENM that informs the termination of an event for which the new DENM has been received by the originating ITS station from another ITS station [26]". The transmitter ITS stations of the new DENM and the negation DENM are different.

The DENM termination, i.e., the end of the detected event, can be indicated either by a cancellation DENM or a negation DENM.

According to ETSI EN 302 931 [2], three geometric shapes of destination area are defined, i.e., circular, rectangular and elliptical shapes. Each of these shapes is represented by one or several geographical points and distance information.

As depicted in Figure 2.11, a DENM is consisted of a common ITS PDU header and multiple containers, containing the DENM payload. The information about the version of the protocol, type of the message and the ID of the originating ITS station are included in the ITS PDU header. The DENM payload consists of a management container and three optional containers, i.e., the situation, the location and the à la carte container. The DENM protocol and management information are contained in the management container. The information regarding the type of the detected event are contained in the situation container. The information about the location of

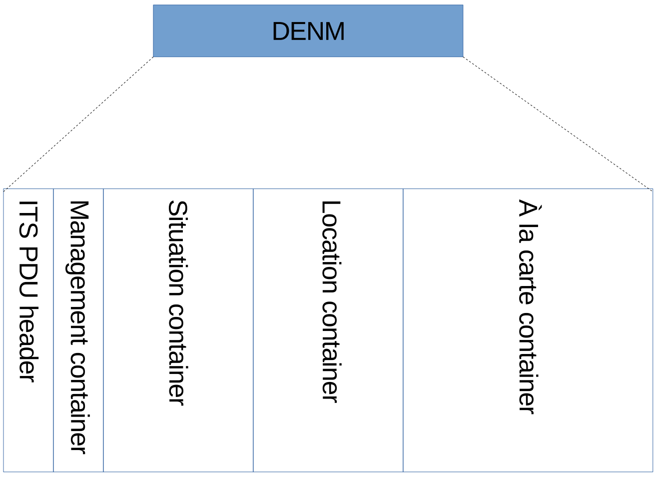


Figure 2.11: General structure of a DENM

the event and the location referencing are contained in the location container. Additional use case specific information which is not included in the three other containers, is included in the à la carte container. The ITS PDU header and the management container should be always included in all types of DENM.

2.1.3 *C-ITS Networking and Transport Layer*

According to [12, 10], the C-ITS networking and transport layer has functionalities from the OSI Model network layer and the OSI Model transport layer including amendments in order to be suitable for C-ITS. The C-ITS networking and transport layer consists of one or several networking protocols, one or several transport protocols, and a network and transport layer management entity.

ITS network protocols especially include the data routing from a source ITS station to a destination ITS station through intermediate ITS stations. They also include the data dissemination in geographical regions. A few different networking modes are specified for C-ITS, e.g., geoNetworking protocol in ETSI TS 102 636 [27], IPv6 networking with mobility support in ISO/International Electrotechnical Commission (IEC) 21210 [28], IPv6 over geoNetworking in ETSI TS 102 636-6-1 [29], Communications Access for Land Mobiles (CALM) protocol in ISO/IEC 29281 [30], etc. Each networking protocol can be connected to a dedicated C-ITS transport protocol or it can be connected to an already existing transport protocol, e.g., UDP and TCP.

In order to be able to keep the network load at an acceptable level and to avoid unstable behavior of the system due to its ad-hoc topology, the ITS network and transport layer should support DCC functions. This is done by means of Transmission Interval Control (TIC), Transmit Power Control (TPC), efficient routing and forwarding protocols, etc. The DCC functions of geoNetworking protocols should be compliant with ETSI TS 102 687 [31].

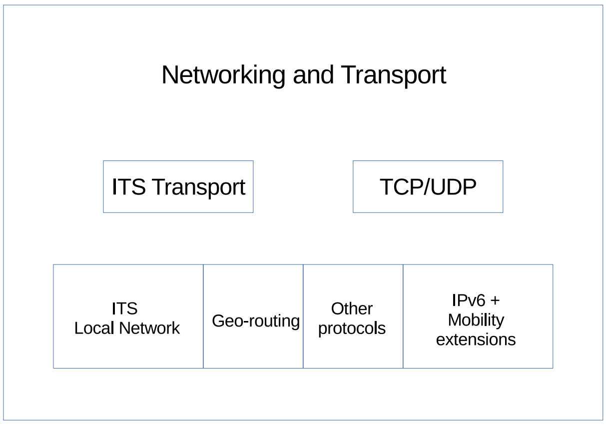


Figure 2.12: C-ITS networking and transport layer

ITS transport protocols accommodate the end-to-end delivery of data. Also, they provide additional services, e.g., reliable data transfer, flow control and congestion avoidance, depending on the requirements of ITS facilities and applications. As indicated in Figure 2.12, a few different transport protocols are specified for C-ITS, e.g., UDP/TCP, dedicated ITS transport protocols, etc.

Based on [4, 5], in some cases, e.g., when there is no direct neighbor that forwards the packet to the destination, the ITS network and transport layer should provide buffering functions in order to send the packet to the lower layer later on.

According to [32, 33], Basic Transport Protocol (BTP) provides a connectionless, end-to-end transport service for the ITS ad-hoc networks. BTP offers the non-guaranteed delivery of its PDUs between its entities, similar to the UDP Internet Engineering Task Force (IETF) Requests for Comment (RFC) 768 [34]. BTP provides services to the protocol entities of the ITS facilities layer, e.g., CA basic service [20] and DEN basic service [26]. The main goal of BTP is to multiplex and de-multiplex the messages from different ITS facilities layer processes in order to send the packets with geoNetworking protocol. Therefore, the protocol entities of the ITS facilities layer can access the services of the geoNetworking protocol.

In general, the ITS network and transport layer should be able to fulfill the followings:

- It should provide low-latency communications.
- It should provide reliable communications giving the highest reliability to the safety messages.
- It should provide a low overhead signaling, routing and packet forwarding.

- It should provide fairness concerning bandwidth usage among different ITS stations.
- It should be robust against the security attacks and mal-functions in ITS stations.
- It should be able to perform in scenarios with different density of geoNetworking-enabled ITS stations.

2.1.3.1 *GeoNetworking*

As stated in [4, 5], geoNetworking is a network-layer protocol that works based on the ITS-G5 wireless technology such as in ad-hoc communication networks. It is a family of network protocols that employ the geographical positions for addressing and transport of data packets in Wireless Sensor Networks (WSNs), MANETs and VANETs. GeoNetworking employs geographical position information to disseminate and transport data packets and to provide multihop communications in order to extend the telematics horizon.

GeoNetworking is a network service that works connectionless and fully distributed, and can provide wireless communication between vehicles and/or between vehicles and fixed infrastructures along the roads. GeoNetworking is appropriate for the networks with high mobility and frequent changes in the network topology, e.g., VANETs. Moreover, it is flexible in supporting different applications and their heterogeneous requirements. In case of road safety and traffic efficiency, geoNetworking provides periodic transmission of safety status messages and fast multi-hop dissemination of emergency warning packets in geographical regions. In case of infotainment, geoNetworking provides the transport of the unicast packets for internet applications.

Essentially, geoNetworking provides two functions, i.e., geographical addressing and geographical forwarding, that are strongly coupled. In conventional networks, each node has an IP address linked to its identity. But in geoNetworking, packets can be sent to a node by its position or to multiple nodes in a geographical area. To perform the multihop communications and in order to forward the packets from the source vehicle to the destination vehicle, geoNetworking assumes that each vehicle has a partial view of the network topology in its neighborhood. It also assumes that each packet includes the geographical address of the destination, e.g., a geographical position or a geographical area. The vehicles between the source and destination, receive the data packet, compare the network topology in the neighborhood with the geo-address included in the data packet, and make a forwarding decision autonomously. Thus, there is no need to setup and maintain routing tables in each vehicle, i.e., packets are forwarded *on the fly*.

Basically, the following forwarding schemes are included in geographical routing:

- *GeoUnicast*: When a vehicle wants to send a unicast packet to another vehicle, it determines the position of the destination vehicle and then forwards the data packet to a neighbor vehicle towards the destination. This neighbor vehicle reforwards the packet along the route until it reaches the destination vehicle.

- *GeoBroadcast*: When a vehicle wants to send a packet to all the vehicles located in a geographical area, the packet is forwarded hop-by-hop until it reaches the destination area. Afterwards, the vehicles located within the destination area rebroadcast the packet. GeoAnycast is a special form of sending a packet to an arbitrary vehicle located in the destination area, in which this vehicle will not re-broadcast the received packet.
- *Topologically-scoped broadcast*: Means broadcasting a packet from a vehicle to all the vehicles in the n-hop neighborhood. It is called single-hop broadcast, if the packet is only sent to the direct neighbor vehicles.

GeoBroadcast is the most innovative method to distribute the information in an specific geographical area by means of geo-routing. The source vehicle specifies a well-delimited geographic area for a packet and the intermediate vehicles work as relays to forward the packet towards the destination area. Afterwards, vehicles located inside the destination area process the packet and send its information to the corresponding applications. This way, it is possible to only notify the vehicles that are concerned with a dangerous situation or a traffic notification.

- **Geographical Addressing:**

According to [35, 6, 36], geoNetworking is a network protocol in the C-ITS network and transport layer. GeoAdhoc routers implement the geoNetworking protocol to provide services to the upper protocol entities, e.g., BTP and GeoNetworking to IPv6 Adaptation Sub-Layer (GN6ASL).

The network protocol of an ITS station should be addressed by a network address in order to transport a packet. The type of this network address is protocol-specific and it should comprise geoNetworking, IPv6 and IPv4 addresses. In order to communicate with an ITS station, geoNetworking implements a particular concept of geographical addressing by the network address and geographical position of the ITS station in the VANET. Depending on the network scenario, IP addresses are assigned to ITS stations based on the existing approaches, e.g., auto-configuration. Basically, IP version 6 is employed for the communication in ITS that uses the Internet protocol. In order to support the legacy Internet applications that require IPv4 and public access networks that are only capable of IPv4, backward compatibility from IPv6 to IPv4 should be provided.

In order to identify the communicating geoNetworking entities, each geoAdhoc router has a unique geoNetworking address which should be used in the header of a geoNetworking packet. Each geoAdhoc router has a self-assigned initial geoNetworking address at start-up that is specified in ETSI EN 302 636-4-1 [6].

- **Geographical Forwarding:**

Based on [35, 6, 36], geoNetworking is a family of network protocols that employ the geographical positions for addressing and transport of data packets in VANETs. ETSI EN 302 636-4-1 [6] has introduced geoUnicast and geoBroadcast forwarding algorithms. But these algorithms does not address the research issues of urban VANETs, e.g., local optimum, and are only applicable in highway scenarios. Two geoUnicast forwarding algorithms are defined: Greedy forwarding algorithm and Contention-Based Forwarding (CBF) algorithm.

The greedy forwarding algorithm employs the Most Forward within Radius (MFR) policy, i.e., it chooses the geographically closest neighbor ITS station to the destination, to provide the most progress, forwarding a data packet. If such a neighbor ITS station does not exist, it is called a local optimum and the greedy forwarding algorithm does not help anymore.

In contrast to the sender-based forwarding schemes, in CBF algorithm a receiver decides whether to forward a data packet or not. In order to implicitly perform an optimal forwarding of a data packet, the CBF algorithm employs a timer-based re-broadcasting and overhears for duplicated packets sent. In other words, after receiving a data packet, the ITS stations with a positive progress, buffer the packet and start a timer which duration is inversely proportional to the forwarding progress of the ITS station towards the destination. The ITS stations calculate the delay using the Equation 2.1.

$$\mathrm{TO\_CBF\_GUC} = \begin{cases} \mathrm{TO\_CBF\_MAX} + \frac{\mathrm{TO\_CBF\_MIN} - \mathrm{TO\_CBF\_MAX}}{\mathrm{DIST\_MAX}} \times \mathrm{PROG} & \mathrm{PROG} \leqslant \mathrm{DIST\_MAX} \\ \mathrm{TO\_CBF\_MIN} & \mathrm{PROG} > \mathrm{DIST\_MAX} \end{cases} \tag{2.1}$$

Here, TO_CBF_MIN gives the minimum duration that the data packet should be queued in the buffer. TO_CBF_MAX gives the maximum duration that the data packet is allowed to be queued in the buffer. PROG gives the difference between the senders distance from the destination and the ITS station's distance from the destination, i.e., the forwarding progress of the ITS station. DIST_MAX gives the theoretical maximum communication range of the ITS stations.

While the timer is running, the ITS station might receive a duplicate of the data packet from another ITS station that had a shorter timeout because of its shorter distance to the destination. If this is the case, then the ITS station stops its timer and removes the data packet from the buffer. Otherwise, as soon as the timer is expired, the ITS station re-broadcasts the data packet.

Comparing to the greedy forwarding algorithm, CBF implicitly makes a trade-off between reliability and forwarding delay. At the cost of the additional processing and greater forwarding delay, CBF ensures the reliability, i.e., if the theoretically optimal forwarder is not reachable, the data packet can be re-forwarded by another forwarder.

According to [35, 6, 36], three geoBroadcast forwarding algorithms are defined: Simple geoBroadcast forwarding algorithm, contention-based forwarding algorithm for geoBroadcast and advanced geoBroadcast forwarding algorithm.

In the simple geoBroadcast forwarding algorithm, if the ITS station is located outside of the destination area, it forwards the packet using the greedy forwarding algorithm. If the ITS station is located inside or at the border of the area, it re-broadcasts the packet.

In the CBF algorithm for geoBroadcast, a receiver decides whether to forward the packet or not. When an ITS station broadcasts a packet, the receiver ITS stations with a positive progress, buffer the packet and start a timer which duration is inversely proportional to the forwarding progress of the ITS station towards the destination. The ITS stations calculate the delay using the Equation 2.2. It shows that the ITS stations with the maximum forwarding progress towards the destination, will have the smallest timeout.

$$\mathrm{TO\_CBF\_GBC} = \begin{cases} \mathrm{TO\_CBF\_MAX} + \frac{\mathrm{TO\_CBF\_MIN} - \mathrm{TO\_CBF\_MAX}}{\mathrm{DIST\_MAX}} \times \mathrm{DIST} & \mathrm{DIST} \leqslant \mathrm{DIST\_MAX} \\ \mathrm{TO\_CBF\_MIN} & \mathrm{DIST} > \mathrm{DIST\_MAX} \end{cases} \tag{2.2}$$

Here, TO_CBF_MIN gives the minimum duration that the data packet should be queued in the buffer. TO_CBF_MAX gives the maximum duration that the data packet is allowed to be queued in the buffer. DIST gives the distance of the sender from the ITS station. DIST_MAX gives the theoretical maximum communication range of the ITS stations.

The CBF algorithm for geoBroadcast is similar to the geoUnicast CBF algorithm, but the definition of the distance is different, i.e., the definition of the parameter DIST and PROG. During the time that the timer is running, the ITS station might receive a duplicated data packet from another ITS station that had a shorter distance to the destination. If so, then the ITS station stops its timer and removes the data packet from its buffer. Otherwise, as soon as the timer is expired, the ITS station re-broadcasts the data packet.

The advanced geoBroadcast forwarding algorithm consists of the greedy forwarding algorithm and the CBF algorithm. Therefore, it is both a sender-based and receiver-based algorithm. Moreover, it applies enhancements to the CBF algorithm to improve the efficiency and reliability of the geoBroadcast forwarding algorithm. The advanced geoBroadcast forwarding algorithm includes four main mechanisms:

- *CBF algorithm:* copes with the reception uncertainty issues resulted by the mobility of the ITS stations, fading phenomena and the collisions on the wireless channels.

- *Enhancements CBF:* copes with the additional forwarding delay caused by the traditional CBF. Enhancements CBF additionally selects one specific forwarder ITS station that forwards the message immediately upon the correct reception of the packet.

- *Controlled packet retransmission scheme:* applies inside the geographical target area and increases the reliability of the dissemination process. The ITS stations in CBF mode, provide a counter for the number of retransmissions of a packet, i.e., number of the times that this packet is received. When this counter reaches a threshold, the ITS station stops contending for this packet. This way, the packet can be re-transmitted several times but the data overhead is controlled.

- *Sectorial contention area:* improves the efficiency of the CBF, choosing the potential forwarder ITS stations solely from a limited sector of the circular forwarding area, as shown in Equation 2.3. Basically, if the ITS station is located outside of the sectorial area, it schedules the packet for re-broadcasting. Otherwise, it refrains from contending. This sectorial contention area is defined by a threshold angle and the maximum communication range of the ITS station, as shown in Equation 2.4.

$$G = \begin{cases} +1 & \mathit{inside\ or\ at\ border\ of\ sectorial\ area} \\ -1 & \mathit{outside\ sectorial\ area} \end{cases} \tag{2.3}$$

$$G = \begin{cases} +1 & (\mathit{DIST\_R} < \mathit{DIST\_F} < \mathit{DIST\_MAX}) \& (\angle \mathit{FSR} \leqslant \mathit{ANGLE\_TH}) \\ -1 & \mathit{Otherwise} \end{cases} \tag{2.4}$$

Here, DIST_R gives the distance between the receiver ITS station's position and the sender's position. DIST_F gives the distance between the forwarder ITS station's position and the sender's position. DIST_MAX gives the theoretical maximum communication range of the ITS stations. ∠FSR gives the angle between the positions of the forwarder ITS station, the sender ITS station and the receiver ITS station. ANGLE_TH is a threshold value for ∠FSR and has a minimum value of 30° and a maximum value of 60°. ANGLE_TH varies based on the traffic density in the neighborhood. The default value of ANGLE_TH is given by the geoNetworking protocol.

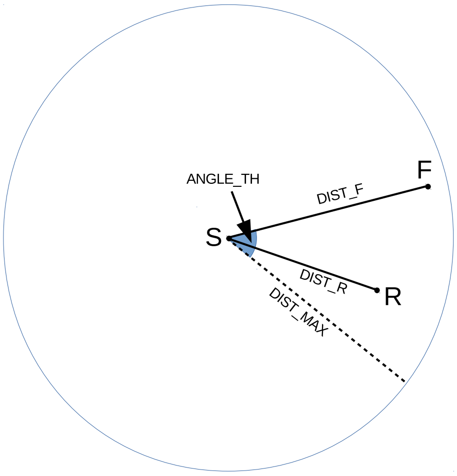


Figure 2.13: Sectorial contention area

- **Data Structure:**

As stated in [35, 6, 36], all the ITS stations should maintain a Location Table (LocT) that includes the information of the other ITS stations executing the geoNetworking protocol. The data elements of the Location Table

Entrys (LocTEs) are the geoNetwork address of the ITS station, Link-layer address of the ITS station, type of the ITS station, position vector of the ITS station and the time-stamp of the last packet received from the ITS station. The LocTEs should be soft-state, meaning that all the entries have a lifetime set by the geoNetworking protocol and should be deleted as soon as the lifetime expires.

Also, all the ITS stations should maintain a local data structure holding its LPV. The LPV should include the geographical position and its accuracy, also speed and heading of the ITS station. It should as well include a time-stamp indicating the generation time of the geographical position.

Moreover, all the ITS stations should locally maintain the Sequence Number (SN) field of the next geoNetworking packet that is going to be transmitted. For each geoNetworking packet, SN should start from zero and should be incremented following Equation 2.5. The resulting SN will be included in the multi-hop geoNetworking packet. CAMs do not have a SN field.

$$SN(P) \;=\; (SN(P) + 1) \bmod SN\_MAX \tag{2.5}$$

Here, $SN(P)$ gives the SN of the geoNetworking packet. SN_MAX gives the largest possible SN.

- **Duplicate Packet Detection:**

According to [35, 6, 36], it is possible that an ITS station receive multiple copies of the same geoNetworking packet as a result of the routing loops, multiple forwarding ITS station and etc. Therefore, geoNetworking protocol has mechanisms for Duplicate Packet Detection (DPD) to control the forwarding of duplicated packets. The geoNetworking protocol applies SN-based methods for DPD to multi-hop packets, e.g., DENMs. Moreover, for single-hop packets, e.g., CAMs, that do not carry a SN field, the geoNetworking protocol applies time-stamp-based methods.

2.1.4 *C-ITS Access Layer*

Based on [12, 23], the two lowest layers of OSI Model, i.e., PHY layer and Data Link Layer (DLL), are termed as access layer in C-ITS. The technology that is specified for the access layer is conjointly called ITS-G5. The ITS-G5 standard is based on the existing standards for communications. The PHY layer physically connects to the communication medium. The DLL is divided into two sub-layers, i.e., MAC layer and Logical Link Control (LLC) layer. The MAC sub-layer manages the access to the communication medium. The LLC sub-layer makes it possible for different network protocols to exist side-by-side, providing multiplexing mechanisms.

IEEE 802.11 [37] covers the PHY layer and the MAC layer. The LLC is found on the American National Standards Institute (ANSI)/IEEE Std 802.2 [38]. As shown in Figure 2.14, the C-ITS supports different access technologies including the legacy technologies. Thus, an adaptation of legacy technologies might be necessary. To do so, access layer includes a Communication Adaptation sub-Layer (CAL), a Management Adaptation Entity (MAE), and

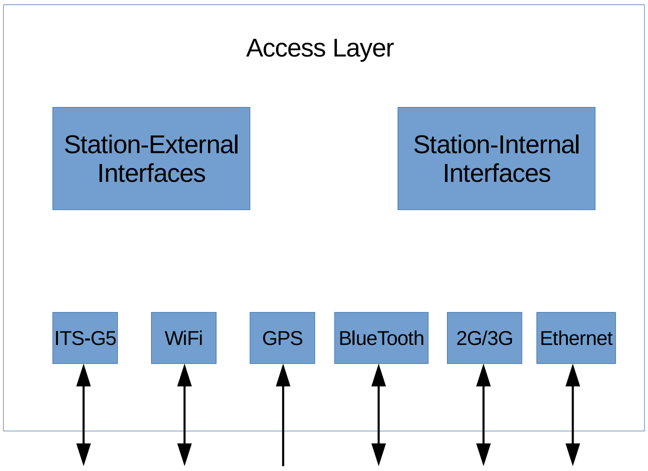


Figure 2.14: C-ITS access layer

a Security Adaptation Entity (SAE).

According to [23, 9, 39], the European frequency spectrum allocation is divided into four parts. As shown in Table 2.2, the primary frequency band is the ITS-G5A with 30 MHz. It is dedicated for active road safety and cooperative traffic efficiency applications. ITS-G5B with 20 MHz is dedicated for non-safety applications. ITS-G5C is shared with the Radio Local Area Network (RLAN) frequency band. It is also referred to as the Broadband Radio Access Network (BRAN), RLAN and Wireless Local Area Network (WLAN).

Table 2.2: Frequency allocation in the European union

Channel	Frequency range [MHz]	Usage
ITS-G5A	5875 to 5905	ITS road safety related
ITS-G5B	5855 to 5875	ITS non-safety
ITS-G5C	5470 to 5725	RLAN (BRAN, WLAN)
ITS-G5D	5905 to 5925	Future ITS applications

Only the ITS-G5 compliant stations are allowed to operate on the ITS-G5A and ITS-G5B frequency bands. All the ITS stations operating on ITS-G5A and ITS-G5B are treated equally, no matter they are fixed or mobile. Operation in the ITS-G5C frequency band requires TPC and a procedure for Dynamic Frequency Selection (DFS). It also requires uniform spreading, to be able to detect signals from radar systems and in order to avoid co-channel interference. In ITS-G5C frequency band, in order to apply the spectrum management based on DFS, mobile ITS stations act as DFS slaves and fixed ITS stations as DFS masters. Therefore, communication between mobile ITS stations in ITS-G5C is not possible. ITS-G5 compliant stations are allowed to use the ITS-G5D frequency band.

In order to maintain the network stability and throughput efficiency, and

Table 2.4: European channel allocation

Channel type	Center frequency	IEEE channel number	Channel spacing	Default data rate	TX power limit	TX power density limit
G5CC	5900 MHz	180	10 MHz	6 Mbit/s	33 dBm EIRP	23 dBm/MHz
G5SC2	5890 MHz	178	10 MHz	12 Mbit/s	23 dBm EIRP	13 dBm/MHz
G5SC1	5880 MHz	176	10 MHz	6 Mbit/s	33 dBm EIRP	23 dBm/MHz
G5SC3	5870 MHz	174	10 MHz	6 Mbit/s	23 dBm EIRP	13 dBm/MHz
G5SC4	5860 MHz	172	10 MHz	6 Mbit/s	0 dBm EIRP	-10 dBm/MHz
G5SC5	5850 MHz	182	10 MHz	6 Mbit/s	0 dBm EIRP	-10 dBm/MHz

to have a fair resource allocation to ITS-G5 stations, DCC, e.g., CSMA/CA in MAC layer and TPC and Transmit Rate Control (TRC) in network layer, is introduced and is specified in ETSI TS 102 687 [31].

Table 2.3: Data rates and channel spacing

Modulation Coding Scheme (MCS)	0	1	2	3	4	5	6	7
40 MHz channel (Data rate in Mbit/s)	12	18	24	36	48	72	96	108
20 MHz channel (Data rate in Mbit/s)	6	9	12	18	24	36	48	54
10 MHz channel (Data rate in Mbit/s)	3	4.5	6	9	12	18	24	27
Modulation scheme	BPSK	BPSK	QPSK	QPSK	16-QAM	16-QAM	64-QAM	64-QAM
Coding rate R	1/2	3/4	1/2	3/4	1/2	3/4	2/3	3/4

The PHY layer of ITS-G5 should be compliant with the PHY layer OFDM specification of IEEE 802.11 [37] for the 5 GHz band. Because ITS-G5 stations work outside the context of a Basic Service Set (BSS), mechanisms like Point Coordination Function (PCF) and Hybrid coordination function Controlled Channel Access (HCCA) are not suitable, therefore EDCA should be applied. To contend for medium access, ITS stations employ the EDCA as specified in IEEE 802.11 [37]. The set of EDCA parameters should be determined statically. To set the parameters of EDCA, ITS-G5 supports MAC Service Data Units (MSDUs) with up to 8 levels of User Priority (UP) as specified in IEEE 802.11 [37].

The specified channel allocation is shown in Table 2.4. One physical channel is G5CC that shall be used for road safety and traffic efficiency applications. It may also be used for ITS service announcements of services operated on other channel. Furthermore, five physical channels are allocated as ITS-G5 Service Channels (G5SCs). G5SC1 and G5SC2 shall be also employed for ITS road safety and traffic efficiency applications. G5SC3, G5SC4 and G5SC5 shall be used for other ITS user applications.

For all the ITS stations that are operating in a safety-related context, the G5CC is the reference channel and CAMs, DENMs, Road Topology Messages (TOPOs) and Road Map Messages (MAPs) are transmitted there. Other messages will be sent randomly on other G5SCs. The ITS stations that send a message on a G5SC should operate it through a *service*, i.e., announce it with a Service Announcement Message (SAM) on a reference channel that the relevant ITS stations are tuned on it.

Table 2.3 lists the Modulation and Coding Schemes (MCSs) and corresponding data rates as described in IEEE 802.11 [37].

TPC limits for ITS-G5A and ITS-G5B are described in ETSI EN 302 571 V1.1.1 [40], and for ITS-G5C are described in ETSI EN 301 893 V1.5.1 [41]. Also, DCC provides further requirements for TPC. The minimum receiver sensitivity is described in IEEE 802.11 [37].

2.1.5 *C-ITS Management Entity*

According to [12, 10], the management entity of C-ITS is responsible for the configuration of an ITS station, cross-layer information exchange among the different layers of C-ITS and etc. The management elements of the management entity of C-ITS are grouped as shown in Figure 2.15.

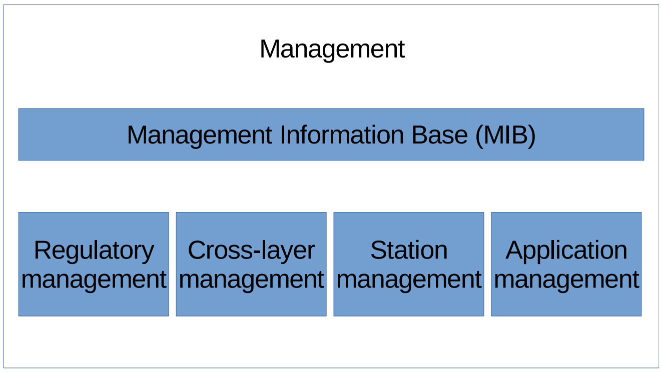


Figure 2.15: C-ITS management entity

2.1.5.1 *ITS Service Advertisement*

In order to make it possible for an ITS station to identify the existence of an ITS service, C-ITS might support push and pull mechanisms. The pull mechanism is known from internet protocols, in which the initial request originates from the client. The reverse, i.e., push mechanism, where the server pushes data to clients, is called *ITS service advertisement*.

2.1.5.2 *General Congestion Control*

In reality, a large number of ITS stations simultaneously try to access the physical communication channel. Considering that the physical communication channels have limited bandwidth, it is necessary to provide special means to avoid excessive load on the physical communication channel. This dynamic modification of access layer parameters have an impact on all the communication layers of C-ITS.

2.1.5.3 *Local Node Map*

The C-ITS station management may maintain information of the neighboring stations, combining their communication parameters, e.g., networking addresses, with their kinematics, e.g., position and speed. It is called the local node map information.

2.1.6 *C-ITS Security Entity*

According to [12, 10], the C-ITS security entity provides security and privacy services. As shown in Figure 2.16, this includes secure messages at different layers of C-ITS, firewall and intrusion management, authentication, authorization and profile management, and identity, crypto key and certificate management.

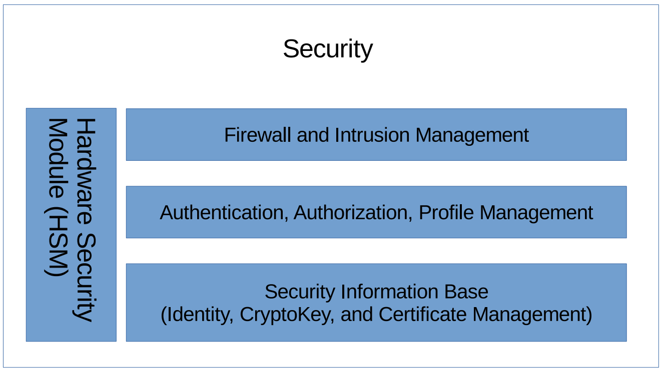


Figure 2.16: C-ITS security entity

3

STATE OF THE ART AND CLASSIFICATION OF PROTOCOLS

Routing performs the task of carrying data packets from a source across a network to a destination, typically involving at least one intermediate forwarder. In MANETs, routing is done in the absence of any fixed infrastructure or any centralized administration, and the topology of the network changes dynamically. VANET is a type of MANET, and therefore they have similar features, e.g., self-organization. However, because of the different characteristics of their network members and their different environments, they show different characteristics. The unique characteristics of urban VANETs are explained in the following:[1]

- *Highly dynamic topology:* The high speed of vehicles leads to a rapidly changing network topology in VANETs.
- *Patterned mobility:* The layout of the street networks, the traffic condition, the speed limits in streets, traffic lights and the traffic behavior of the drivers, lead to a certain mobility pattern [44].
- *Propagation model:* Because of the presence of obstacles in the urban traffic environments, e.g., buildings, and the layout of the road networks, the free space propagation model is usually assumed to be unrealistic [45].
- *Longer network life-time:* In contrast to the WSNs, the network members in VANETs, i.e., vehicles, are not as limited in terms of energy [46].

Routing protocols in VANETs can be divided into three main categories:

- *Topology-based:* routing protocols are traditional MANET routing protocols that work based on the topology information of the network. Topology-based routing protocols perform the *link-state routing*, in which routers make a graph of the connected network members and make the routing decisions based on this graph [47]. Considering the highly dynamic topology of VANETs, topology-based routing protocols are not suitable for VANETs. Table A.2 shows a chronological overview of the state of the art topology-based VANET routing protocols.
- *Cluster-based:* routing protocols organize the network members into smaller groups called *cluster*. Each cluster has a coordinator called *cluster-head* and some *cluster-members*. Each cluster-head is responsible for communication with the cluster-members, or for communication with other cluster-heads. Although, the two-layer hierarchy results in a better scalability for large networks, establishment and maintenance

---

[1] Parts of this chapter have been published in Garrosi et al. [42] and Garrosi et al. [43].

of clusters yields in higher delay and overhead. Table A.1 shows a chronological overview of the state of the art cluster-based VANET routing protocols.

- *Position-based:* routing protocols better known as *geo-routing* protocols, work essentially based on the *greedy forwarding*, i.e., each network member sends the packets to another member that is geographically closer to the destination. To do so, they only need to be informed of their position, position of the destination and the position of the their neighbors, which yields in a good scalability. Table A.3 shows a chronological overview of the state of the art position-based VANET routing protocols.

As mentioned, geo-routing protocols only employ the geographic position information of vehicles to route the data packets from the source ITS station to the destination ITS station in VANETs. This simplicity and efficiency of geo-routing protocols, make them suitable for VANETs. Also, Global Positioning System (GPS) devices cost less and vehicles equipped with GPS devices are getting more common. On the other hand, Geo-Location Service (GLS) have been addressed by a considerable number of research work. Thus, geo-routing protocols became a reasonable solution for VANETs. Although several geo-routing protocols have been proposed, there are still shortcomings, which need to be eliminated to have geo-routing at its best.

3.1 GEOBROADCAST PROTOCOLS

According to ETSI TS 102 636-2 [7], communication between different communication endpoints may be realized by geoBroadcast, i.e., communication from a single ITS station to all the ITS stations within a geographical target area. ETSI TC ITS defines the specifications of decentralized environmental notification basic service in ETSI EN 302 637-3 [26], that supports the RHW applications by constructing, managing and processing DENMs. Therefore, designing an efficient multi-hop geoBroadcast protocol is very important to avoid the broadcast storm problem.

Based on Akamatsu et al. [48] and Ko et al. [49], these routing decisions can be made either as *sender-based decision making* or *distributed decision making*. *Neighbor knowledge* method is a well-known sender-based decision making method that requires information of one-hop neighbors employing CAMs. The distributed decision making methods can be divided into:

- *Simple flooding* method, i.e., all nodes rebroadcast the messages

- *Probability-based* method, in which receivers determine whether they rebroadcast or not, using probabilistic parameters, e.g., *counter-based* method using random back-off

- *Area-based* method, in which receivers determine whether they rebroadcast or not, based on their positions, e.g., *delay-based* scheme using the distance to sender or *directed flooding* defining a forwarding zone.

Table 3.1 includes the state of the art geoBroadcast routing protocols for urban VANETs and lists their objectives, metrics and features.

Joshi et al. [50] proposed Distributed Robust Geocast (DRG) as a multicast routing for inter-vehicle communications to send the packets to the vehicle ITS stations located in a geo-region of a highway or urban scenario. It employs a distance-based back-off timer to favor the farthest vehicle ITS station from the sender vehicle ITS station to be the first to rebroadcast the packet. Any other vehicle ITS station that receives this rebroadcast and loses the back-off contention to this vehicle ITS station, cancels the transmission. To overcome the network fragmentation issues, it proposed a retransmission after a long interval. Also, to overcome the communication losses, it proposed a burst of retransmissions with short interval. DRG defines the ratio of the area of overlap of coverage area of two vehicle ITS stations with respect to their average coverage area as the *coverage ratio*. In urban scenarios, the coverage ratio of the neighbor vehicle ITS stations should be greater than a certain threshold. Also, the neighbor vehicle ITS stations should have a wide angular distance to cover substantially new regions of the target geo-region. Hence, its angular distance should be greater than an angular threshold. But the performance of DRG depends on a very accurate estimate of the actual transmission range in order to calculate the coverage ratio threshold.

Zhang et al. [51] proposed Geocast Routing in Urban VANET (GRUV), that categorizes vehicle ITS stations into crossroad vehicles and in-road vehicles and proposes different vehicle selection algorithms respectively. A rectangle forwarding zone is used to route geocast packets. GRUV also dynamically switches between three forwarding approaches, i.e., forwarding zones, to adapt to the current network environment. A mesh in forwarding zone is used to establish multiple paths between the source vehicle ITS station and the geocast region, in order to effectively maintain the network connectivity. GRUV applies source routing, in which the source vehicle ITS station prefers to choose the route recorded in the earlier response packet, although it may not be the stable path. If the BOX forwarding approach fails to create a mesh, i.e., the source vehicle ITS station receives no Routing Reply (RREP), then after its timer expires, the next Routing Request (RREQ) packet is sent via Extended-BOX forwarding approach. If Extended-BOX forwarding approach can not setup a route before its timer expires, then the next RREQ packet is sent via FLOOD forwarding approach.

Kaiwartya et al. [52] proposed Traffic light based Time Stable Geocast (T-TSG) to inform vehicle ITS stations about an accident in the urban vehicular environments. It has three routing approaches based on the traffic light phases, i.e., forwarding, disseminating and re-live. Forwarding phase delivers the message to the geo-region. Disseminating phase continuously distributes the geocast messages in geo-region. Re-Live phase manages the life time of the geocast messages. Based on the moving direction of vehicle ITS stations and the traffic light situation, T-TSG selects the forwarding vehicle ITS station.

Akamatsu et al. [48] proposed UGAD to suppress the unnecessary retransmissions by controlling the packet forwarding. It defines the forwarding zone as a region that is closer to the geo-region than the sender vehicle ITS station. Vehicles at intersections rebroadcast faster considering preferential delay values over in-road vehicles. Vehicles calculate their own back-off time based on the transmission range, distance to the sender, and the forwarding mode when they are located within the forwarding zone or the geo-region. The receiver vehicle ITS stations are not required to rebroadcast the packet, if they receive duplicated packets from other vehicle ITS stations, before the back-off timer expires. The decision to perform the intersection-based forwarding mode or the greedy forwarding mode is made, based on an angle calculated from the position of receiver, position of sender and the geo-region. The performance of UGAD depends on a very accurate estimate of the actual transmission range in order to calculate the back-off time.

Kaiwartya et al. [53] proposed Cache Agent-based Geocasting (CAG) for urban VANET scenarios. It categorizes the vehicle ITS stations into cache user and cache agent vehicles. The vehicle ITS stations that belong to a particular region become cache agents of the intersection points of that region. All the other vehicle ITS stations that belong to different regions are considered as cache users. At intersections, when a cache user vehicle ITS station can not find an appropriate next hop vehicle ITS station in the message forwarding direction, it tries to find some cache agent vehicle ITS stations within its transmission range. CAG also perform re-caching in case a next hop vehicle ITS station drives out of the range in-between transmission and the cached data packets can not be forwarded. Applying the full radio range transmissions integrated with connectivity assurance algorithm, CAG tries to tackle the problem of high packet loss due to the fast movement of vehicles in VANETs.

Zhang et al. [54] proposed GeoMob, a mobility-aware geocast forwarding via taxicabs and buses for delay-tolerant networks in urban VANETs. When a message is generated, GeoMob selects an optimal routing path, i.e., a sequence of regions towards the destination. To do so, Dijkstra algorithm is applied to the weighted graph with the help of macroscopic mobility. GeoMob introduces real-world trace analysis and employs different levels and aspects of vehicle mobility information. This way it is possible to better understand macroscopic mobility, i.e., the overall geographic distribution of the city traffic and to better understand microscopic mobility, i.e., individual vehicle mobility patterns. When two vehicle ITS stations encounter each other, explicit routing decisions are made with the help of microscopic mobility.

However, these geoBroadcast routing protocols have some drawbacks that can be critical. Performance of some of them, e.g., DRG and UGAD, depend on a very accurate estimation of the actual transmission range in order to calculate the coverage ratio threshold or the back-off timer. Some of them, e.g., GRUV, employ the traditional path discovery methods from the on-demand routing and try to find a route sending RREQ packets and waiting for the RREP packets. Some of them, e.g., UGAD, T-TSG and CAG, suffer from the routing issues like local optimum and partitioned networks. Some of them, e.g., GeoMob, are dependent on the mobility traces collected by the

Table 3.1: Urban geoBroadcast routing protocols

Protocols	Year	Scenario	Objectives	Metrics	Features
DRG [50]	2007	General	Geo-region	Distance	Distance-based back-off transmission Long-interval retransmission to overcome network fragmentation
GRUV [51]	2009	Urban	Geo-region	Vehicle position Round-trip time Intersection	Source routing Routing REQ flooding in dynamic forwarding zone Next-hop selection based on crossroad or in-road modes
T-TSG [52]	2013	Urban	Geo-region Alive time	Vehicle direction Traffic light	Three routing approaches based on the traffic light phases
UGAD [48]	2014	Urban	Geo-region	Distance Vehicle position Intersection	Distance-based backoff transmission Two forwarding modes based on vehicle's location
CAG [53]	2014	Urban	Geo-region	Coverage capability	Full radio range transmissions integrated with connectivity assurance algorithm
GeoMob [54]	2014	Urban	Geo-region	Mobility pattern	Forwarding via taxicabs and buses Dijkstra algorithm

taxicabs and buses to evaluate the microscopic and macroscopic mobility patterns. In Chapter 5, I introduce UAG that has none of the aforementioned problems.

3.2 GEOUNICAST PROTOCOLS

Based on ETSI TS 102 636-2 [7], communication between different communication endpoints may be realized by geoUnicast, i.e., communication from a single ITS station to another ITS station. Because of the large number of the proposed geoUnicast routing protocols, it is requisite to have a comprehensive collection of the materials on this topic to provide a solid background for a research work's investigation. This section presents an overview of the state of the art geoUnicast routing protocols for urban VANETs, summarizes the research issues and categorize them based on the input information employed.

3.2.1 *Research Issues*

In VANETs, vehicles are distributed non-uniformly and dynamically set up an ad-hoc network without any aid from infrastructures. Vehicles move fast and therefore traditional geoUnicast routing protocols of MANETs are facing new problems. Moreover, in urban VANETs, buildings shadow the communication signals and thus disconnections happen frequently. Therefore, the highway geoUnicast routing protocols of VANETs are also facing new issues in urban areas. These characteristics of urban VANETs, make geo-routing a challenging task. Each new routing protocol tries to address one or more of these research issues and to contribute in that direction. Based on my studies, these problems can be divided into ten main groups:

- *Information consistency*, *link stability* or *path lifetime* problems happen mainly as a result of the fast unpredictable movement of vehicle ITS stations. Most of the protocols apply different sorts of movement prediction to anticipate the future position of vehicle ITS stations. Others use fuzzy logic, combine reactive and geo-routing, weight vehicle ITS stations or consider communication channel conditions.

- *Network connectivity* problems happen as a result of the non-uniform distribution of vehicle ITS stations in urban areas. The common solution here is to estimate the traffic density. But different protocols propose different ways of obtaining traffic density information.

- *Local optimum* happens when there is no neighbor vehicle ITS station closer to the destination than the forwarder vehicle ITS station itself. The main solution is to apply the perimeter forwarding. Other suggested solutions are considering connected dominating sets, considering the so called *anchor points*, and taking advantage of traffic density information.

- *Route maintenance* is required as a result of the frequent disconnections. The proposed solutions so far are employing so called *guards*, marking or weighting the streets, and sending update messages.

- *Partitioned networks* are observed normally in urban VANETs as a result of the non uniform distribution of vehicles. The common solution is the so called *CaF*.

- *Delay* in delivering data packets to the destination can happen as a result of the unique characteristics of urban VANETs. The common approach here is to decrease the number of hops by applying some modifications, e.g., simplified perimeter forwarding, auto-adjustability, and predictions.

- *Network overhead* is basically caused by the high number of the messages sent in the network. Some protocols try to avoid it employing Preferred Group Broadcasting (PGB) and others try not to use beacon messages, which is not in accordance with the ETSI ITS standards.

- *Load balance* problems happen as a result of the unfair distribution of resources between vehicles. Main ideas are the congestion detection and applying metrics to vehicle ITS stations in order to solve this problem.

- *Malicious vehicles* or *dishonest vehicles* are also an issue in future. In this regard, some protocols apply a filtering process and set trust values for vehicles.

- *Cross-link* or *loop* happens when a message crosses the same junction for a second time but in a different direction. Therefore, some protocols have mechanisms to check for loops and avoid them.

Table 3.2 gives a summary of the contributions of the state of the art geoUnicast routing protocols, considering the aforementioned problems.

3.2.2 *Input Information of Geo-routing Protocols*

From the activity point of view, geo-routing protocols can be divided into proactive, reactive and hybrid routing protocols as follows:

- *Proactive:* routing protocols or *table-driven* routing protocols maintain routing tables in vehicle ITS stations to represent the network topology. The vehicle ITS stations continuously update their routing table and send them around to all the other vehicle ITS stations by periodic *HELLO* packets. Proactive routing protocols determine the routes to various vehicle ITS stations in the network in advance, including vehicle ITS stations to which no packets are being sent. Thus, proactive routing yields in a considerable control overhead. Also, the fast movement of vehicles in VANETs is problematic for proactive methods.

- *Reactive:* routing protocols or *on-demand* routing protocols only send the control data whenever it is required. Thus, reactive routing yields in a reduced control overhead. They can do a route discovery by flooding the RREQ message in the network and waiting for the RREP message. Therefore, reactive routing yields in a higher route discovery time.

- *Hybrid:* routing protocols contain the characteristics of both proactive and reactive routing protocols. They might divide the network into smaller zones to reduce the routing overhead of proactive protocols and to reduce the route discovery time of reactive protocols.

From the application point of view, geo-routing protocols can be classified into two main groups. First group targets the Delay Tolerant (DT) applications, e.g., comfort applications, and the second one targets the Delay Intolerant (DI) applications, e.g., road safety applications. Some of the geo-routing protocols also consider RSUs available and try to take advantage of them.

Geo-routing protocols may employ different CAM-containers as inputs in addition to position information of vehicles, e.g., direction of vehicles, velocity of vehicles, digital map, traffic density information, and channel quality information, to decide about the next forwarder. Some of them only collect the information of the direct neighbors, i.e., single-hop and others collect the information of multi-hop neighbors. Here, some protocols also try to predict the behavior of other vehicles in near future, based on the available inputs. To help the process of exchanging data between vehicles, some protocols employ beacon messages which is also known as hello messages or CAMs. Table 3.3 shows a chronological overview of the state of the art geo-routing protocols for urban VANETs and summarizes the input information of each protocol, in which ’s’ stands for single-hop and ’m’ stands for multi-hop.

3.2.2.1 *Direction and Velocity of Vehicles*

As shown in Table 3.3, mostly geo-routing protocols employ the direction and velocity information of vehicles together. Some of them predict the future location of vehicles, e.g., MoVe [57], GyTAR [61], ARM [86], HLAR [72], LPRV [77] and EHTAR [78]. Some choose the next vehicle or path, e.g., CAR [62], Fcar [68], JARR [69], MBMPR [79], WNPRP [81] and SCRP [82, 83]. And others check the link stability and link-breakage, e.g., MOPR [63], HLAR [72] and EHTAR [78]. Since the direction and velocity information of vehicles are already included in standard CAMs, geo-routing protocols can take advantage of them, without causing any overload for the network. These protocols will be briefly explained in the following:

LeBrun et al. [57] proposed Motion Vector (MoVe), in which two different strategies for routing are presented, that use the movement information to make a decision for forwarding. MoVe assumes that vehicles are equipped with a GPS receiver and use a HELLO RESPONSE technique to identify neighbors. If a vehicle receives a HELLO message, it responds with a RESPONSE message, containing the motion information. The only difference

Table 3.2: Contributions of geo-routing protocols

Protocol	Overhead	Connectivity	Path-life	Route Maintenance	Partitioned Network	Delay	Local Max	Load Balance	Malicious Vehicles	Cross-link
A-STAR [55]	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗
GPCR [56]	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
MoVe [57]	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
STAR [58]	✗	✓	✗	✗	✗	✗	✓	✗	✗	✗
MORA [59]	✗	✗	✓	✗	✗	✗	✗	✓	✗	✗
GpsrJ+ [60]	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗
GyTAR [61]	✗	✓	✓	✗	✓	✗	✗	✗	✗	✗
CAR [62]	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
MOPR [63]	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
GRANT [64]	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
ACAR [65]	✗	✓	✓	✗	✓	✗	✗	✗	✗	✗
RBVT-R [66]	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗
TO-GO [67]	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
Fcar [68]	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
GeoCross [1]	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓
JARR [69]	✗	✓	✗	✗	✓	✗	✗	✗	✗	✗
DIR [70]	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
IDVR [71]	✗	✓	✓	✗	✓	✗	✗	✗	✗	✗
HLAR [72]	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
RIVER [73]	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗
GeoSVR [74]	✗	✓	✓	✗	✗	✗	✓	✗	✗	✗
GSPR-MV [75]	✗	✗	✓	✗	✗	✓	✗	✗	✗	✗
TROUVE [76]	✗	✗	✓	✗	✗	✗	✗	✗	✓	✗
LPRV [77]	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗
EHTAR [78]	✗	✓	✓	✗	✗	✗	✗	✗	✗	✗
MBMPR [79]	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗
RPGR [80]	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
WNPRP [81]	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗
SCRP [82, 83]	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
GSR [84]	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
REAR [85]	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
ARM [86]	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

of the two strategies is the way they choose the next forwarder for the transmission of data. In the first strategy, MoVe, uses the angle between the motion vector and the shortest distance to the destination to calculate the shortest path. Here, a vehicle is selected for forwarding, which has the smallest angle, and moves towards the destination. The second strategy, MoVe-look-ahead, has only a single modification, in which each vehicle checks its way-point, where its trajectory changes. If there is a change of direction before the vehicle reaches the closest point to the destination, then the distance between the destination and the way-point is used instead of the aforementioned angle, to calculate the shortest path.

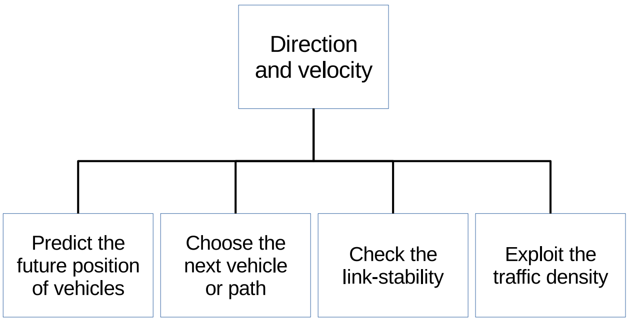


Figure 3.1: Usage of the direction and velocity information

Zhao et al. [87] proposed Vehicle-Assisted Data Delivery (VADD) based on the CaF approach. It uses the predictable vehicle mobility, which is limited by the road layout and the traffic pattern, to forward the packets on the streets with the lowest delay. Based on the techniques used for road selection at the intersection, the Location First Probe (L-VADD), Direction First Probe (D-VADD), Multi-Path Direction First Probe (MD-VADD) and Hybrid Probe (H-VADD) have been proposed.

Jerbi et al. [61] proposed Greedy Traffic Aware Routing Protocol (GyTAR) that dynamically selects the next intersection through which the packet should be forwarded. In contrast to source routing, here the intersections are selected one after each other. Having the pre-loaded digital maps that provide a street-level map, GyTAR can calculate the number of the vehicles located between each two intersections. It also considers that any source vehicle can find out about the position of the destination using the grid location service [88]. Based on these, it assigns a score to each intersection, considering traffic density and the curvemetric distance of the intersection to the destination. Here, vehicles record a table including the velocity, direction and the latest known position of neighbors, based on the periodic hello messages. In order to froward the data packets between intersections, GyTAR employs an improved greedy strategy, in which, each vehicle predicts the new position of the neighbors and forwards the data packet to one of them, based on this prediction. As a recovery strategy, GyTAR employs CaF, in which the vehicle carries the data packet until a suitable forwarder is found or it reaches the intersection itself.

Naumov et al. [62] proposed Connectivity-Aware Routing (CAR) which takes advantage of the information of beacons. CAR employs the PGB to broadcast a path discovery packet at the beginning. Next forwarding vehicle adds an *anchor* point to the path discovery packet, if its velocity vector is not parallel to the previous forwarder's velocity vector. Finally, the destination will receive the path discovery packet having all the route to the source anchored. Afterwards, the destination sends a reply packet as a unicast packet going through the anchors using an advanced greedy forwarding, which tries to find the closest vehicle to the next anchor point. As route maintenance, any end-point activates a *guard*, if it changes its direction. A vehicle within a radius around a guard can estimate the new position of the end-point and might filter, redirect, and add information to a packet. CAR employs two types of guards: standing and traveling. Standing guard gives the temporary state information of a geo-region and is kept alive by the vehicles located in this area. Traveling guard also contains a velocity vector, with which, each vehicle can calculate the new position of the traveling guard before sending it again with the new beacon. As routing error recovery, if there is no vehicle to forward the packet, a timeout algorithm with an active waiting cycle is applied.

Menouar et al. [63] proposed Movement Prediction-based Routing (MOPR), which employs the velocity, position and direction information of vehicles to predict the position of them in the near future to estimate the link stability. Menouar et al. applied MOPR to Greedy Perimeter Stateless Routing (GPSR) in order to show the improvements. MOPR-GPSR selects the next forwarder based on GPSR, which is predicted to stay within the communication range for the next second based on MOPR.

Ding et al. [89] proposed Static-node Assisted Adaptive Data Dissemination protocol for Vehicular networks (SADV), which lets the package to be stored in an static vehicle at an intersection for a while, to see if there will be some vehicle available on the best possible path to the destination. In addition, SADV allows the adjacent vehicles to measure the delay of forwarding data between each other in real time, in order to adapt the routing decisions to the variable density of vehicles.

Wang et al. [68] proposed Fuzzy Control AODV-based Routing (Fcar) to employ the fuzzy logic and fuzzy control method to make routing decisions having multiple selection criteria, e.g., path life-time and the number of vehicles driving in the same direction. Fcar considers the life time of the path, the percentage of vehicles using the same direction, location of vehicles and their speed to evaluate a path.

Tee et al. [69] proposed Junction-based Adaptive Reactive Routing (JARR), which assumes that vehicles are equipped with location service performing adaptive beaconing. JARR tries to estimate the density of paths to be used while sending packets from one intersection to the other one. In order to estimate the density on a certain path, the beaconing rate and the velocity of vehicles are considered. Also the position, direction and velocity of the vehicles are considered with different ratios depending on the traffic density on the path. As recovery strategy, JARR uses the CaF method.

Borsetti et al. [86] proposed Application-level Role Mobility (ARM), in which vehicles take the role of a bearer or collector of information in distributed election procedure, depending on their position, direction and speed. The carrier vehicle can collect data and send data to an assigned location or to provide these data to neighboring vehicles.

Al-Rabayah et al. [72] proposed a new Hybrid Location-based Routing protocol (HLAR) to address the issue of the communication link-breakage, combining the features of reactive routing with location-based geographic routing. It employs small periodic beacon packets and assumes that the source knows about the location of the destination. HLAR combines a modified Ad-hoc On-demand Distance Vector (AODV)-ETX (expected transmission count) protocol with a geographic greedy forwarding in a way that it switches to reactive routing when the location information degrades. HLAR assumes that the only reason for link failure is vehicle mobility. It tries to calculate the probability of the link-breakage, based on the angle between the velocity vectors of two vehicles and the communication range of them.

Tu et al. [75] presented Greedy Simplified Perimeter Routing with Moving Vector (GSPR-MV), which tries to improve the poor performance of GPSR, taking the vehicles' fast moving and forwarding efficiency into account and combining it with a simplified perimeter forwarding. It tries to predict the near future positions of vehicles before forwarding the packet, based on the position, direction and velocity information of vehicles.

Balico et al. [77] proposed the Localization Prediction-based Routing for VANET (LPRV). The key feature of the LPRV algorithm is to exploit the knowledge of vehicles to predict future locations as a metric to forward data packets, without the need of exchanging any extra control messages, since trajectories are sent along with the packets. They consider a vehicle's predicted location, as its direction and speed at a given future time step.

Lo et al. [78] proposed Enhanced Hybrid Traffic-Aware Routing (EHTAR). A dynamically elected functional vehicle called *junc-tracker* located at each intersection is used to explore the real-time vehicular and network traffic information of each road. Having this information, junc-trackers can determine a reliability score for each road and assist with the selection of a more robust and efficient routing path. Also, an enhanced next forwarder selection scheme is adopted to ensure the stability of the next forwarder by adopting a location prediction and link-duration estimation. When two vehicles are traveling at the same velocity and direction, they will maintain a constant connection until one of these vehicles changes its velocity or direction. Here, EHTAR estimates the approximate time that the target vehicle get to the next intersection according to its velocity and position information.

Lin et al. [79] proposed Map-Based Multi-Path Routing (MBMPR), that uses GPS, digital maps, and sensors to find an optimal forwarding path.

Moreover, there exists a congestion detection mechanism to handle the load balance problem at intersections. MBMPR is divided into three parts: Forwarding intersection selection, packet forwarding strategy, and routing recovery strategy. The idea behind forwarding intersection selection is to find a sequence of intersections from the source to destination with the shortest road segments and the highest traffic density using Dijkstra. The packet forwarding strategy executes the actual forwarding taking into account the direction of the vehicles. These information are exchanged by HELLO messages. The routing recovery strategy is used to deal with the problems, where there is no suitable forwarding vehicle available e.g. local optimum.

Togou et al. [82, 83] proposed the Stable and reliable CDS-based Routing Protocol (SCRP) which works based on the connected dominating sets (CDS). It assumes that all the vehicles are equipped with a GPS that provides information about its location, speed, and direction as well as a digital map that includes precise information about the road segments and intersections. Also, source vehicles inquiry location services to acquire the destination location when forwarding the data packets. SCRP is a position-based routing scheme that takes advantage of the topology information stored at bridge vehicles to select the most stable paths (high connectivity and low delivery delay). To achieve this goal, it builds backbones over the road segments. The built backbones are road segments with a low weight and they are connected at junctions via bridge vehicles. They claim that this procedure avoids the local optimum problem.

Chinnasamy et al. [81] proposed the Wagon Next Point Routing Protocol (WNPRP) for VANETs. It assumes that the vehicles are equipped with GPS, navigation system and digital maps to collect information about the position, speed, and moving direction of vehicles, sending Hello messages periodically. WNPRP tries to find the location of the next hop from the source and then applies a filtering process to make sure that the malicious vehicles do not interfere.

Some protocols only employ the direction information of vehicles. STAR [58] exploits the traffic density, MORA [59] checks the link stability, and RPGR [80], GeoDTN+Nav [90] and SRR [91] select the next forwarder vehicle. These protocols will be briefly explained in the following:

Giudici et al. [58] proposed Spatial and Traffic-Aware Routing (STAR), which reduces the number of local optimum considering the traffic density in routing decisions. It assumes that each vehicle has a GPS to determine its own position and to obtain information about the local road map and the vehicle's direction of movement. For this purpose, the protocol is divided into two layers. The lower layer is used for collection and exchange of information about the network status (traffic density). The higher layer creates a weighted graph using street map and traffic information and then calculates the paths applying Dijkstra to the graph. Some *anchor* points on the streets traversed by the computed routes are chosen and packets are forwarded from one anchor point to the successive one with geographic greedy routing.

Granelli et al. [59] proposed a Movement-based Routing Algorithm (MORA), which employs a weighting function for vehicles to have a better link-stability. This function considers a line between source and destination and gives more weights to the vehicle either moving on it or moving towards it. MORA also considers a metric to have a fair distribution of resources between vehicles. Here, the source starts the routing by flooding the network with a route request packet. Afterwards, the destination sends a route reply packet to the source. MORA can be implemented in two modes: unabridged and distributed. U-MORA is similar to source routing and source will have several routes to choose between them, but D-MORA yields in a single path from source to destination.

Cheng et al. [90] proposed Geographic DTN Routing with Navigator prediction (GeoDTN+Nav) for Delay Tolerant Networks (DTNs), which combines the strengths of the geographical routing and delay tolerant routing. This way it tries to address the problem of unconnected network partitions. Between two partitions of a network, where there are too few vehicles, the packet delivery can be done by CaF. If the network is not partitioned, because of the high traffic density, then it changes back to the normal geographic routing.

Ghafoor et al. [91] proposed Stability and Reliability aware Routing (SRR), which employs geographical routing using fuzzy logic to find out about the best forwarder vehicle considering the direction and distance of vehicles. When the network is disconnected, SRR caches the data packets and in case of unpartinioned network it switches back to the geographical routing.

Qureshi et al. [80] introduced the Road Perception based Geographical Routing (RPGR) protocol for VANETs in urban environments. It tries to improve the geographical forwarding assuming the direction of vehicles and traffic density to forward the data packets towards the destination. RPGR can be divided in two different working modes: between intersections and at intersections. If the forwarding vehicle is located between intersections, it computes the mid region vehicle. The mid region vehicle covers the maximum distance and transmission range. When the forwarder vehicle is located at an intersection, it calculates the curvemetric distance, also the traffic density, and the direction towards the destination to select the next forwarder vehicle in the network.

Other protocols only employ the velocity information of vehicles. DIR [70], GeOpps [92] and GeoSpray [93] select the next path, IDVR [71] predicts the future location of vehicles, TROUVE [76] and PRAODV [94] calculate the link-stability and ETAR [95] selects the next forwarder vehicle. These protocols are briefly explained in the following:

Namboodiri et al. [94] proposed PRAODV, which is an improvement of AODV having modified RREP packets. They also include the speed and the location information. Each vehicle receiving the packet, can predict the duration of the connection. This way it tries to determine a new alternate path before the old path collapses.

Leontiadis et al. [92] proposed Geographical Opportunistic (GeOpps), which is a delay tolerant geographical algorithm. GeOpps exploits the availcarries the packet until it finds a suitable vehicle. To ensure fast delivery, the vehicle with highest velocity is selected.

3.2.2.2 *Digital Map and Traffic Information*

Geo-routing protocols might employ the digital map and/or traffic density information of vehicles. Table 3.4 shows these protocols and summarizes the information regarding their weighting functions. Some protocols collect the traffic density data dynamically, e.g., using beacons. Others collect them passively, e.g., collected data, bus lines and type of roads, to weight the graph of roads. Some protocols apply Dijkstra or consider curvemetric distance to the destination to find the shortest path, while others route hop by hop.

Since it is common practice to have access to the navigation systems and digital maps in vehicles, it is helpful to extract the information of streets and intersections from them. Also, with the help of CAMs, vehicles can estimate the real-time traffic density of the neighborhood and make better routing decisions without any extra costs.

Some of the geo-routing protocols employ the digital map and traffic density information of vehicles together, e.g., A-STAR [55], STAR [58], GyTAR [61], ACAR [65], DIR [70], IDVR [71], RIVER [73], GeoSVR [74], TROUVE [76], EHTAR [78], MBMPR [79], RPGR [80], ETAR [95], MDDV [96], VADD [87], SADV [89] and JARR [69].

Seet et al. [55] proposed Anchor-Based Street and Traffic Aware Routing (A-STAR), which chooses the sequence of intersections from the street map information employing Dijkstra. It also uses the information of the bus lines in the city to choose the paths with high connectivity, by weighting the streets based on the number of bus-lines on them. As recovery strategy, A-STAR marks the local optimum streets as out-of-service for a limited time and calculates a new anchor based route.

Yang et al. [65] proposed Adaptive Connectivity Aware Routing (ACAR), which selects an optimal path in an adaptive way based on the statistical traffic and real-time density data that are collected by an on-the-fly collection process. ACAR selects the optimal path taking into account the density of vehicles and the periods of traffic lights. And then, in each of the road segments, it selects the next hop in a way to decrease the packet error rate of the whole path. In partitioned network, ACAR applies CaF.

Bernsen et al. [73] proposed Reliable Inter-Vehicular Routing (RIVER), which is a position-based protocol with an optimized greedy strategy. It performs an active real-time traffic monitoring and employs the other passively collected data to rate the reliability of the streets. Afterwards, the sending vehicle applies the Dijkstra's least weight path to its reliability-weighted street graph to calculate the most reliable path to the destination. As the recovery strategy, RIVER changes the weight of the failed street in a way that it is considered as disconnected. Then it will run the Dijkstra's least weight path algorithm again.

Xiang et al. [74] proposed Geographic Stateless VANET Routing (GeoSVR), in which vehicles are equipped with GPS and digital map. GeoSVR consists of

Table 3.3: Input information of geo-routing protocols

Protocol	Year	Application	Direction	Velocity	Map	Traffic	Prediction	Channel	# of hops	Beacon
GSR [84]	2003	DI	✗	✗	✓	✗	✗	✗	s	✓
A-STAR [55]	2004	DI	✗	✗	✓	✓	✗	✗	m	✓
MDDV [96]	2004	DT	✗	✗	✓	✓	✗	✗	m	✗
PRAODV [94]	2004	DI	✗	✓	✗	✗	✓	✗	m	✗
GPCR [56]	2005	DI	✗	✗	✓	✗	✗	✗	s	✓
MoVe [57]	2005	DI	✓	✓	✗	✗	✓	✗	s	✓
STAR [58]	2005	DI	✓	✗	✓	✓	✗	✗	s	✓
MORA [59]	2006	DI	✓	✗	✗	✗	✗	✗	s	✓
VADD [87]	2006	DT	✓	✓	✓	✓	✓	✗	m	✗
GpsrJ+ [60]	2007	DI	✗	✗	✓	✗	✓	✗	m	✓
GyTAR [61]	2007	DI	✓	✓	✓	✓	✓	✗	m	✓
CAR [62]	2007	DI	✓	✓	✗	✗	✗	✗	s	✓
MOPR [63]	2007	DI	✓	✓	✓	✗	✓	✗	s	✓
GeOpps [92]	2007	DT	✗	✓	✓	✗	✗	✗	s	✓
SADV [89]	2007	DT	✓	✓	✓	✓	✗	✗	s	✗
GRANT [64]	2008	DI	✗	✗	✗	✗	✗	✗	m	✓
ACAR [65]	2008	DI	✗	✗	✓	✓	✗	✗	m	✓
REAR [85]	2008	DI	✗	✗	✗	✗	✗	✓	s	✓
RBVT-R [66]	2009	DI	✗	✗	✓	✗	✗	✗	s	✗
TO-GO [67]	2009	DI	✗	✗	✓	✗	✓	✓	m	✓
Fcar [68]	2009	DI	✓	✓	✗	✗	✗	✗	m	✗
GeoCross [1]	2010	DI	✗	✗	✓	✗	✗	✗	s	✓
JARR [69]	2010	DI	✓	✓	✓	✓	✗	✗	s	✓
DIR [70]	2010	DI	✗	✓	✓	✓	✗	✗	m	✓
GeoDTN+Nav [90]	2010	DT	✓	✗	✓	✗	✗	✗	s	✓
SRR [91]	2011	DT	✓	✗	✗	✗	✗	✗	s	✗
IDVR [71]	2011	DI	✗	✓	✓	✓	✓	✗	s	✓
ARM [86]	2011	DI	✓	✓	✗	✗	✗	✗	s	✓
HLAR [72]	2012	DI	✓	✓	✗	✗	✗	✗	s	✓
RIVER [73]	2012	DI	✗	✗	✓	✓	✗	✗	s	✓
GeoSVR [74]	2013	DI	✗	✗	✓	✓	✗	✓	s	✓
GSPR-MV [75]	2014	DI	✓	✓	✗	✗	✓	✗	s	✓
GeoSpray [93]	2014	DT	✗	✓	✓	✗	✓	✗	m	✗
TROUVE [76]	2015	DI	✗	✓	✓	✓	✗	✗	s	✓
LPRV [77]	2015	DI	✓	✓	✓	✗	✓	✗	s	✗
EHTAR [78]	2015	DI	✓	✓	✓	✓	✓	✓	m	✓
MBMPR [79]	2015	DI	✓	✓	✓	✓	✓	✓	s	✓
SCRP [82, 83]	2015	DI	✓	✓	✓	✗	✗	✗	m	✓
ETAR [95]	2015	DI	✗	✓	✓	✓	✗	✗	m	✓
RPGR [80]	2016	DI	✓	✗	✓	✓	✗	✗	s	✓
WNPRP [81]	2016	DI	✓	✓	✓	✗	✗	✗	s	✓

Table 3.4: Map and traffic information of geo-routing protocols

Routing Protocol	Inputs of Weighting Function		Algorithm
A-STAR [55]	Static traffic	Bus lines	Dijkstra
STAR [58]	Dynamic traffic	-	Dijkstra
GyTAR [61]	Dynamic traffic	-	Curvemetric distance
ACAR [65]	Dynamic/Static traffic	Traffic lights	Hop by hop
DIR [70]	Dynamic traffic	-	Hop by hop
IDVR [71]	Dynamic traffic	-	Hop by hop
RIVER [73]	Dynamic/Static traffic	-	Dijkstra
GeoSVR [74]	Road type (width)	-	Dijkstra
TROUVE [76]	Dynamic traffic	-	Hop by hop
EHTAR [78]	Dynamic traffic	-	Hop by hop
MBMPR [79]	Dynamic traffic	-	Dijkstra
RPGR [80]	Dynamic traffic	-	Curvemetric distance
ETAR [95]	Dynamic traffic	Traffic lights	Hop by hop
MDDV [96]	Static traffic	-	Trajectory
VADD [87]	Dynamic traffic	-	Hop by hop
SADV [89]	Dynamic traffic	-	Hop by hop
JARR [69]	Dynamic traffic	-	Hop by hop
GeoSpray [93]	-	-	Hop by hop
GeoDTN+Nav [90]	-	-	Hop by hop
GeOpps [92]	-	-	Hop by hop
GPCR [56]	-	-	Hop by hop
GpsrJ+ [60]	-	-	Hop by hop
MOPR [63]	-	-	Hop by hop
RBVT-R [66]	-	-	Hop by hop
TO-GO [67]	-	-	Hop by hop
GeoCross [1]	-	-	Hop by hop
LPRV [77]	-	-	Hop by hop
WNPRP [81]	-	-	Hop by hop
SCRP [82], [83]	-	-	Hop by hop
GSR [84]	-	-	Dijkstra

two core algorithms: an optimal forwarding path algorithm to eliminate the problem of local optimum and sparse connectivity, and a limited forwarding algorithm to address the unreliable wireless channel issues. GeoSVR uses the road type (width) to have a weighted graph from the map and then it applies Dijkstra to find the shortest path with a minimum weight.

Wu et al. [96] proposed the Mobility-centric Data Dissemination algorithm for Vehicular network (MDDV) to fix the problem of partitioned VANETs. It assumes that vehicles know the topology of roads, having digital map. MDDV also assumes that vehicles are equipped with a GPS device, but it does not assume that vehicles are aware of the location of their neighbors. MDDV combines the trajectory-based forwarding and geographical forwarding, in which messages are forwarded along a predefined trajectory geographically. Here, the road distance and traffic condition (static road network topology information) are taken into account. Since it assumes that there is no end-to-end connection, the intermediate vehicles have to act as buffer and opportunistically forward the messages.

Some other geo-routing protocols only employ the digital map, e.g., GeoSpray [93], GeoDTN+Nav [90], GeOpps [92], GPCR [56], GpsrJ+ [60], MOPR [63], RBVT-R [66], TO-GO [67], GeoCross [1], LPRV [77], WNPRP [81], SCRP [82, 83] and GSR [84].

Lochert et al. [56] realized that the streets and intersections form a natural planar graph, therefore there is no need for planarization of the graph of urban VANETs. Hence, they have proposed Greedy Perimeter Coordinator Routing (GPCR), which has two parts: a restricted greedy forwarding and a repair strategy. GPCR calls the vehicles located at intersections *coordinators*. As long as no local optimum is encountered, non-coordinator vehicles forward the packets along the street towards the next intersection. Packets should always be forwarded to coordinator vehicles and should not be forwarded across the intersections. Here, coordinators are the vehicles that make the main routing decisions. This requires that all the coordinators inform their neighbors, with the aid of beacon messages, that they are located at intersections. In case of the local optimum problem, GPCR uses a repair strategy consisting of two parts: non-coordinator vehicles again employ the restricted greedy forwarding to forward the packet along the street towards the next intersection. Coordinator vehicles use the right-hand rule to choose the street which is the next one counterclockwise from the street from which the packet has arrived.

Lee et al. [60] realized that in the repair strategy of GPCR, packets are backtracked along the perimeter of the roads to come back to an intersection. They have proposed GpsrJ+, in which it is not requisite to always forward the packets to intersections. It also includes the IDs of the road segments, on which neighbor vehicles are located, in beacon messages, to be able to perform a prediction, taking advantage of on-board topological maps. GpsrJ+ lets a non-coordinator vehicle that has a coordinator neighbor vehicle predict, to which road segment its coordinator neighbor will forward the packet. Therefore, overpasses it, if the next neighbor vehicle has the same x or y coordinate as the coordinates of the predicted neighbor vehicle. Otherwise, simply forward the packet to the coordinator vehicle. This way, it saves one hop where applicable, but causes overhead in beacons.

Nzouonta et al. [66] proposed a proactive and a reactive routing protocol. Considering that the vehicles are equipped with GPS, digital maps and navigation systems to map them, Road-Based using Vehicular Traffic (RBVT)-R is a road-based traffic aware routing protocol, which tries to find the route as the succession of intersections. It employs a beaconless distributed receiver-based election of next hop, having modified Request To Send (RTS) and Clear To Send (CTS) techniques, in which vehicles prioritize themselves, based on the distance to destination, the received power and the distance to sender. Between intersections, geographical routing is applied. RBVT-R does not consider a location service, therefore it applies a flooding route discovery at the beginning. Afterwards, a route reply will be unicasted from the destination towards the source. For the purpose of route maintenance, source or destination sends a route update, if their movement changes the route. In case there is no vehicle available to reach the next intersection, a route error will be sent to the source.

Lee et al. [67] proposed Topology-assisted Geo-opportunistic routing (To-Go), which considers the road topology via two-hop beaconing and uses opportunistic packet forwarding in order to increase the packet delivery ratio. The wireless channel quality is also taken into account in the forwarder selection process. To-Go defines a set of candidate forwarding vehicles between the current sender and the destination vehicle and uses a simple junction prediction algorithm to predict the target vehicle, i.e., either the furthest vehicle or the junction vehicle.

Lee et al. [1] proposed GeoCross, that also consists of greedy forwarding and perimeter forwarding modes like GPSR. GeoCross exploits the natural planar feature of urban maps not to apply the bulky planarization. The idea is to check for loops, i.e., if the packet comes back to the same vehicle, whenever a vehicle wants to forward a packet in perimeter mode. GeoCross focuses on cross-links, which arise at intersections without coordinator vehicles.

Lochert et al. [84] proposed Geographical Source Routing (GSR), which considers the periodic beacon messages to inform direct neighbors about each others positions. Before the source vehicle starts sending the data packet, it needs to find out about the position of the destination. To do so, it employs the Reactive Location Service (RLS), in which it floods the network with a position-request packet. When the destination receives this packet, it will reply to the source with a position-reply packet. When the source receives these packets, it starts to calculate the sequence of intersections employing the Dijkstra algorithm, considering that the vehicles are equipped with digital maps. In between the intersections, the packets are forwarded using greedy forwarding.

3.2.2.3 *Prediction*

As stated in Table 3.3, a group of geo-routing protocols employ the prediction and estimation techniques to improve their performance. Considering that the standard CAM frequency range is between 1 to 10 Hz, vehicles can move a couple of tens of meters after sending each CAM. Therefore, prediction and estimation techniques which are based on the standard information of CAMs, can be really helpful without any extra costs. Most of the protocols try to predict the future position of vehicles, e.g., MoVe [57], GyTAR [61], MOPR [63], IDVR [71], GSPR-MV [75], LPRV [77], MBMPR [79] and GeoSpray [93]. Moreover, EHTAR [78] and CAR [62] try to estimate the link duration. A couple of the geo-routing protocols employ prediction techniques to make an advanced decision on whether to bypass the junction node or not, e.g., GpsrJ+ [60] and To-Go [67].

3.2.2.4 *Channel Quality Information*

As shown in Table 3.3, some protocols weight the road segments based on the traffic density, because they consider that they need a min number of the vehicle on a road segment to be able to forward a packet to the next intersection. But what is not considered is the receipt probability, in case there are too many vehicle in a traffic jam. On the other hand, most of the protocols depend mainly on the vehicles located at intersection which can overload them at some points. Hence, it is possible to improve the routing performance, considering channel load and load balancing, without any extra costs. Some of the geo-routing protocols take the channel load and quality information into account, e.g., TO-GO [67], GeoSVR [74] and EHTAR [78]. Others consider receipt probability, e.g., REAR [85] and load balancing, e.g., MBMPR [79].

Jiang et al. [85] proposed Receipt Estimation Alarm Routing (REAR) to send alarm messages. It uses beaconing to exchange the location information and size of the neighbor vehicles. In REAR, a vehicle does not select the vehicle with the largest distance from itself, but it chooses a vehicle, which most probably can forward the packet, based on the wireless channel conditions.

3.2.2.5 *CAMs and Number of the Hops*

According to ETSI EN 302 637-2 [20], CAMs, beacons or hello messages are exchanged in VANETs between vehicles to make each other aware of their existence. Status and attribute information of a vehicle is contained in the originated CAM. Status information includes the time, position and motion state. Attribute information includes the dimensions, vehicle type and its role in the road traffic. Each CAM is transmitted in a single-hop to the vehicles located in the direct communication range of the originating vehicle. And received CAMs should not be forwarded to other vehicles.

Table 3.3 shows which protocols employ CAMs. Most of the geo-routing protocols require only the information of one hop, i.e., direct neighbor vehicles. They are more compatible with the standard CAMs. On the other hand, some of the geo-routing protocols require the information of multi-hop neighbors. Some of these geo-routing protocols include this information in the CAMs, e.g., GpsrJ+ [60] and TO-GO [67], while others employ separated control messages, which yields in a higher network overhead. Therefore, having the information of multi-hop neighbors, geo-routing protocols can make more precise decisions at the cost of a higher network overhead. Table 3.3 shows which protocols require the information of single-hop or multi-hop.

Schnaufer et al. [64] proposed Greedy Routing with Abstract Neighbor Table (GRANT), in which vehicles add an Abstract Neighbor Table (ANT) to the beacons. These tables work as the core of GRANT and separates the VANET into different areas and represents one neighbor per area.

4

GEOUNICAST

As mentioned before, my methodology is the simulation-based analysis and evaluation of geo-routing protocols. For this purpose, I have studied and implemented the most outstanding VANET geo-routing protocols. In order to validate my implementations and to evaluate these VANET geo-routing protocols, the network simulator OMNeT++ and the road traffic simulator SUMO are coupled employing TRaCI.[1]

4.1 SIMULATION ENVIRONMENT

As shown in Figure 4.1, a Manhattan grid of 1800×1800 m^2 including buildings is generated, in which all the streets have two lanes. Manhattan grid is reproducable and it complies with the state of the art work which makes it more suitable for simulations comparing my proposed geo-routing protocols with the state of the art routing protocols. Moreover, the considered Non-Line-of-Sight (NLoS) measurement-based model is not applicable to the non-Manhattan grids. This model is explained more in Section 4.1.3. All intersections are equipped with traffic lights and road segments have the speed limit of 15 m/s. The distance between each two neighboring intersections is 300 m.

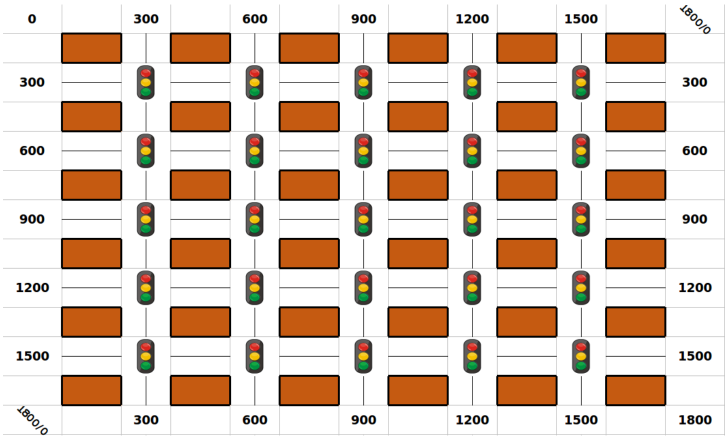


Figure 4.1: Grid street plan (1800 m by 1800 m)

4.1.1 *Network simulator*

OMNeT++ is an open-source, modular, component-based C++ simulation library and framework, mainly for building network simulators. The simulation model of OMNeT++ consists of different modules that communicate

---

[1] Parts of this chapter have been published in Garrosi et al. [97] and Garrosi et al. [98].

with each other using gates. The structure of the simulation model is defined using the NED (Network Description) language. The modules in the lowest layer of the modular hierarchy of OMNeT++ are called single modules. These simple modules define the behavior of the simulation model and are written in C++ programming language. Different modules within the simulation model can communicate with each other exchanging the OMNeT++ messages.

The advantage of the modular structure of OMNeT++ is that one can develop libraries for different applications. INET framework is one of the several model libraries that are available for simulation of the wired, wireless and mobile communication networks. INET framework includes the implementation of different protocols, e.g., IPv4, IPv6, TCP and UDP, and it is especially useful for the purpose of designing and validating new protocols, and in order to explore new scenarios. I run the OMNeT++ version 4.5 and employ the INETMANET as the foundation of my network simulations, that is a fork of INET regarding mobile and ad-hoc networks. In my simulations, OMNeT++ works based on the documentations of ETSI regarding ITS.

4.1.2 *Traffic simulator*

SUMO is an open-source traffic simulator that has different tools regarding the navigation of the vehicles, visualization of the traffic simulations, controlling traffic lights and importing networks of streets. I run the SUMO version 0.21.0. Also, I employ *randomTrips* tool of SUMO to generates a set of random trips for my Manhattan grid. It chooses the starting and ending road segments randomly. The resulting trips are used as input for another tool called *DUAROUTER* to compute the vehicle routes using shortest path computation. These vehicle routes will be afterwards used by SUMO. In my simulations, several random networks of vehicles with random trips and mobility routes have been generated employing SUMO.

4.1.3 *Simulation Scenarios*

After validating my implementations and for the evaluation purposes, and in order to follow the ETSI standards and the state of the art studies, the simulation scenarios and parameters were configured accordingly.

In order to have the simulation results independent from the traffic density, mobility routes have been generated employing SUMO for 100 up to 300 vehicles with an increment step of 25 vehicles. Since DCC is out of the scope of this work, an upper limit for the number of the vehicles in the VANET should have been chosen. In order to be in accordance with the state of the art scenarios and simulations, the upper limit of 300 vehicles is considered.

Moreover, in order to have the simulation results independent from the traffic scenarios, several networks of vehicles with random trips have been generated employing SUMO. Considering each traffic density, there were 49 simulation runs, in which every second, five random pairs of source-destination try to communicate with each other sending DENMs. This way, the simulation results are independent from the positions of source and destination, and their distance from each other.

To consider the attenuation, shadowing and fading effects in a more realistic way, the urban intersection propagation model proposed by Tchouankem et al. [45] is considered. For Line-of-Sight (LoS), a log-distance path-loss model is considered as shown in Equation 4.1.

$$PL = PL_0 + 10\gamma \log_{10} \frac{d}{d_0} + \chi_g \tag{4.1}$$

Here, PL is the total path-loss in dB. PL_0 is the total path-loss at the reference distance of d_0. γ is the path-loss exponent. d is the length of the path. And, χ_g is a normal random variable with zero mean to count for fading attenuation in dB.

For NLoS, based on the distances of the sender and receiver vehicles from the center of their common intersection, i.e., respectively d_t and d_r, the corresponding path-loss can be read from a look-up table that is built based on measurements. This look-up table is depicted in Figure 4.2. Finally, considering transmit power and the resulting total LoS and NLoS path-loss, it is possible to calculate the receiving power on the side of the receiver. Therefore, it is possible to compare the receiving power with the sensitivity, to find out if the receiver has received a packet from the sender.

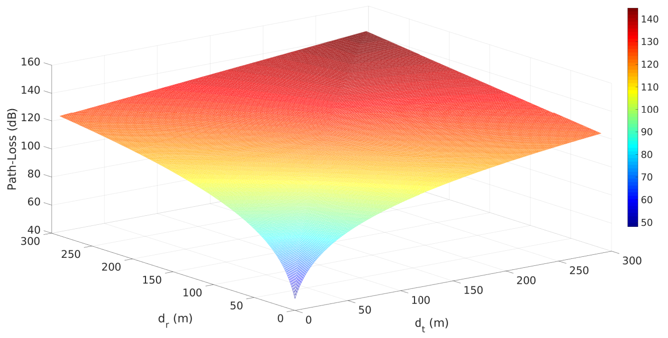


Figure 4.2: NLoS path-loss

In case of the access layer of C-ITS, ITS-G5 with bit-rate of 27 Mbps, transmission power of 23 dBm, thermal noise of -110 dBm and sensitivity of -85 dBm have been considered. Table 4.1 shows the simulation parameters.

According to ETSI EN 302 637-2 [20], the CAM generation interval shall not be inferior to 100 ms (corresponding to the CAM generation rate of 10 Hz) and shall not be superior to 1000 ms (corresponding to the CAM generation rate of 1 Hz). As mentioned before, DCC is out of the scope of this work. Moreover, in order be in accordance with the state of the art scenarios and simulations, the fixed CAM generation rate of 10 Hz is considered to have the most updated information of vehicles.

Table 4.1: Simulation parameters

Network simulator's parameters	
Network simulator	OMNeT++ 4.5
Transmission power	23 dBm
Frequency band	5.9 GHz
Thermal noise	-110 dBm
Sensitivity	-85 dBm
Data rate	27 Mbps
Propagation model	Log-distance path-loss
CAM interval	100 ms
MAC and PHY	ITS-G5
Traffic simulator's parameters	
Traffic simulator	SUMO 0.21.0
Map size	1800×1800 m^2
Speed limit	15 m/s
Number of lanes	2 lanes per direction
Intersection type	Traffic light equipped
Number of vehicles	100:25:300

4.2 OUTSTANDING GEOUNICAST ROUTING PROTOCOLS

Geo-routing protocols only employ the geographic position information of vehicles to route the data packets from the source vehicle to the destination vehicle in VANETs. This simplicity and efficiency of geo-routing protocols, make them suitable for VANETs. Also, GPS devices cost less and vehicles equipped with GPS devices are getting more common. On the other hand, GLSs have been addressed by a considerable number of research work. Thus, geo-routing protocols became a reasonable solution for VANETs.

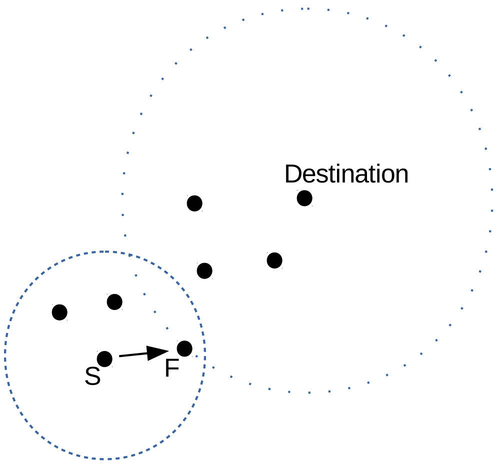


Figure 4.3: Greedy forwarding

Karp et al. [99] has introduced GPSR, in which each node starts the routing process in the so called *greedy* mode. It means that each node tries to find a neighbor node which is geographically closer to the destination and forwards the packet to it. As shown in Figure 4.3, when node S wants to send a packet to the destination, node F is the closest direct neighbor of node S towards destination.

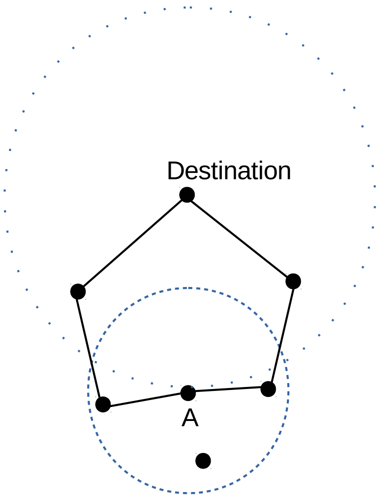


Figure 4.4: Local optimum

But because greedy forwarding only employs the local information, it is possible that a packet reaches a *local optimum*, in which a node can not find any neighbor node that is closer to the destination. As shown in Figure 4.4, node A can not find any neighbor closer than itself to the destination. It can happen as a result of the non-uniform distribution of vehicles, limited communication range, and having buildings as obstacles in urban areas.

At this point, GPSR employs a repair strategy to forward the packet to a node which is closer to the destination. After the packet arrives at a node which is closer to the destination than the point at which it hit the local optimum, it switches back to the greedy mode.

Many recovery algorithms have been suggested to solve this issue. GPSR employs the right-hand rule in the so called *perimeter* mode, as shown in Figure 4.5. Considering node D as destination, when a data packet enters the perimeter mode at node A, GPSR considers the virtual line $\overline{AD}$. Afterwards, it considers the faces of the planar graph, that are crossed by the line $\overline{AD}$, and forwards the data packet on the progressively closer faces. The data packet employs the right-hand rule to find an edge of the current face that crosses the line $\overline{AD}$. Then, the data packet goes to the adjacent face crossed by the line $\overline{AD}$.

This method requires a planar graph, i.e., a graph that can be drawn on the plane having its edges intersect only at its vertices, as shown in Figure 4.6. But this is mostly not the case in urban areas. Therefore, GPSR employs planarization algorithms in a decentralized manner to first have a planar

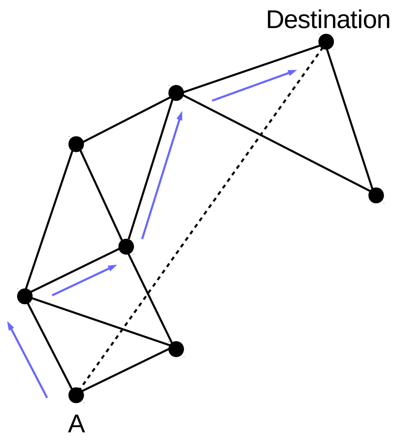


Figure 4.5: Perimeter forwarding

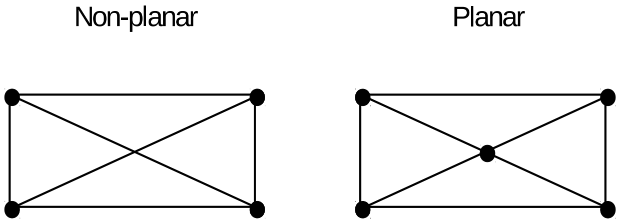


Figure 4.6: Planar graph

graph and then apply the perimeter mode. But this approach suffers from the fast movement of vehicles and also high computational costs.

Figure 4.7 shows the PDR of GPSR comparing to the *network connectivity*, corresponding to different number of vehicles in urban VANETs. In order to have a bench-mark to be able to evaluate the PDR of any geo-routing protocol, I have calculated the network connectivity employing the Depth-First Search (DFS) algorithm. In which I assume that a centralized routing entity with sufficient resources is available to find any possible route between any source-destination pairs. As shown in Figure 4.7, the more vehicles on the streets, the more connected is the VANET, i.e., it is more probable that a pair of source and destination find a route to communicate. Simulations show a big gap between the PDR of GPSR and the network connectivity.

Lochert et al. [56] have realized that the streets and intersections form a natural planar graph, therefore there is no need for planarization of the graph of urban VANETs. Based on this, they have proposed GPCR, which has two parts: a restricted greedy forwarding procedure and a repair strategy based on the fact that the graph of a city is by nature planar.

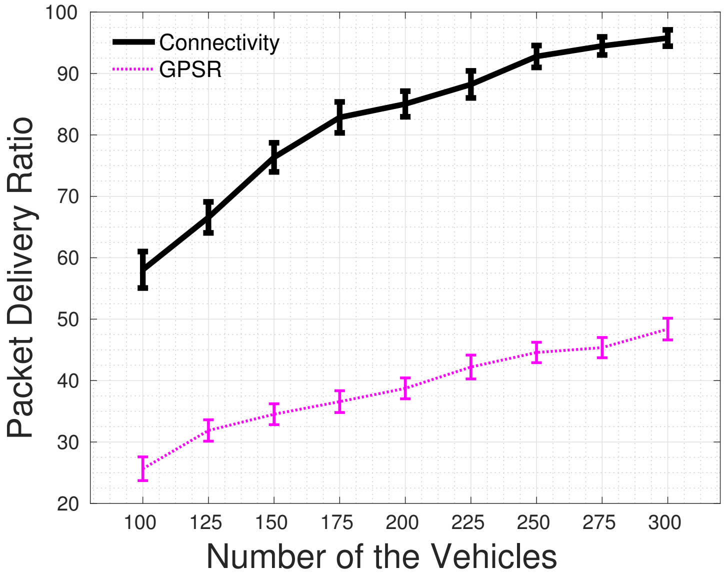


Figure 4.7: PDR of GPSR

GPCR calls the vehicles located at intersections *coordinator*, and the rest of vehicles *non-coordinator*. In GPCR, as long as no local optimum is encountered, non-coordinator vehicles forward the packet along the street towards the next intersection, which is called restricted greedy forwarding. Packets should always be forwarded to coordinator vehicles and should not be forwarded across the intersections to other non-coordinator vehicles. Here, coordinators are the vehicles that make the main routing decisions. This requires that all the coordinators inform their neighbors, with the aid of beacon messages, that they are located at intersections.

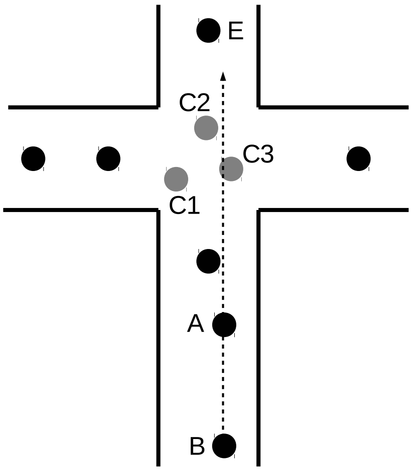


Figure 4.8: Restricted greedy forwarding

As shown in Figure 4.8, in order to perform the restricted greedy forwarding, each non-coordinator vehicle considers a virtual line from the previous vehicle to itself, e.g., the virtual line from B to A, and lists the neighbor vehicles whose positions approximate an extension of this line as potential forwarders. If there is any coordinator in potential forwarders, e.g., C1, C2 and C3, the vehicle chooses one of them randomly and forwards the packet to it. Otherwise, the non-coordinator vehicle with the largest distance to the forwarding vehicle is chosen. When a coordinator vehicle receives a packet, it should decide about the next street that the packet should be forwarded along, i.e., the neighbor vehicle with the maximum progress towards destination is chosen.

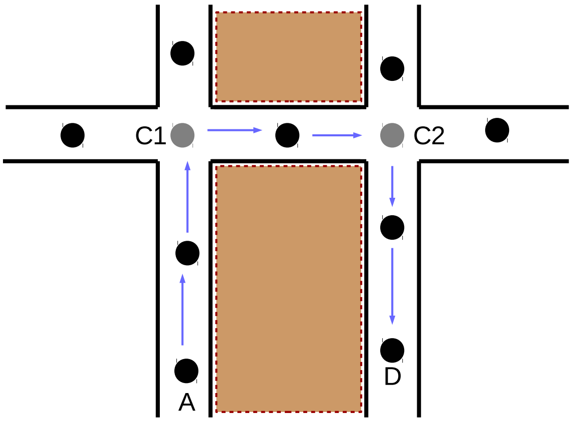


Figure 4.9: Perimeter forwarding

It is still possible in GPCR that a packet encounters the local optimum problem. For such a case as shown in Figure 4.9, GPCR uses a repair strategy which is consisting of two parts. Non-coordinator vehicles, e.g., A, again employ the restricted greedy forwarding to forward the packet along the street towards the next intersection. Coordinator vehicles, e.g., C1 and C2, use the right-hand rule to choose the street which is the next one counter-clockwise from the street from which the packet has arrived.

To find out if a vehicle is a coordinator or not, GPCR has proposed two different approaches. In one of them, a vehicle should also include the position of all its neighbors in its beacon messages, which causes overhead for the network. Having this information, each vehicle can check, if it has two neighbor vehicles, which are within the transmission range of each other but do not list each other as neighbor. If this is the case, this vehicle finds itself as coordinator. This approach works fine but it also needs to know the transmission range of neighbor vehicles, which considering the effects of fading and also TPC mechanisms, is not practical. In the second approach, each vehicle calculates a correlation coefficient with respect to the position of its neighbors and tries to find out, if this correlation coefficient is less than 0.9. This approach is not able to differentiate all the coordinators from non-coordinators and it seems that the threshold changes with respect to the city layout and needs to be adapted manually beforehand.

Figure 4.10 shows the PDR of GPCR comparing to the GPSR and the *network connectivity*, corresponding to different number of vehicles in urban VANETs. As shown, the more vehicles on the streets, the more connected is the VANET, i.e., it is more probable that a pair of source and destination find a route to communicate. Simulations show a big gap between the PDR of GPCR and the network connectivity.

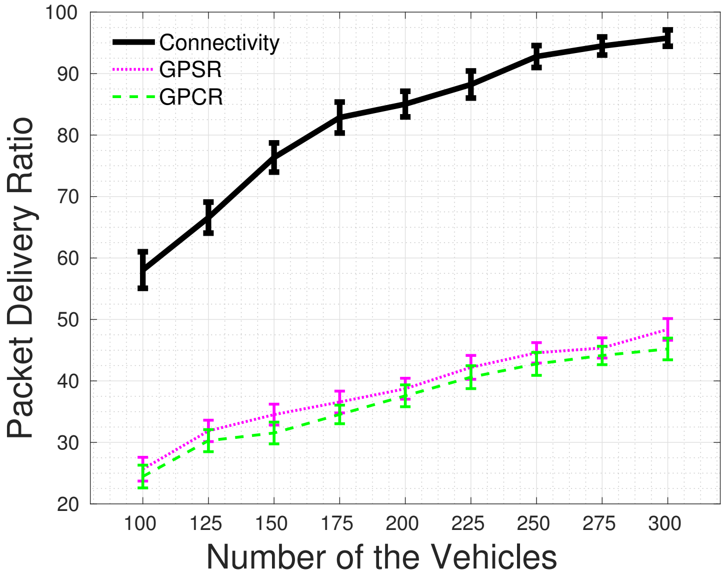


Figure 4.10: PDR of GPCR

In my simulations GPCR performs poorer than GPSR, because it employs correlation coefficients to determine the coordinator vehicles. And as it is mentioned before, the performance of GPCR is highly depending on the performance of the coordinator finding approach, which in this case works poorly.

Lee et al. [60] have realized that in the repair strategy of GPCR, packets are greedily backtracked along the perimeter of roads to come back as fast as possible to an intersection. They have proposed GpsrJ+, in which it is not requisite to always forward the packets to intersections. It also includes in beacon messages, the IDs of the road segments, on which neighbor vehicles are located, to be able to perform a prediction, taking advantage of on-board topological maps. As shown in Figure 4.11, GpsrJ+ lets a non-coordinator vehicle, e.g., A, that has a coordinator neighbor vehicle, e.g., C1, predict, to which road segment its coordinator neighbor vehicle will forward the packet. Therefore, overpass it, if the next neighbor vehicle, e.g., B, has the same x or y coordinate as the coordinates of the predicted neighbor vehicle. Otherwise, simply forward the packet to the coordinator vehicle. This way, they try to save one hop where applicable, but they cause overhead, having extra information in the beacon messages. This prediction might also worsen the problem of cross-links, which is addressed by Lee et al. [1].

Figure 4.12 shows the PDR of GpsrJ+ comparing to the GPSR, GPCR and the *network connectivity*, corresponding to different number of vehicles in urban

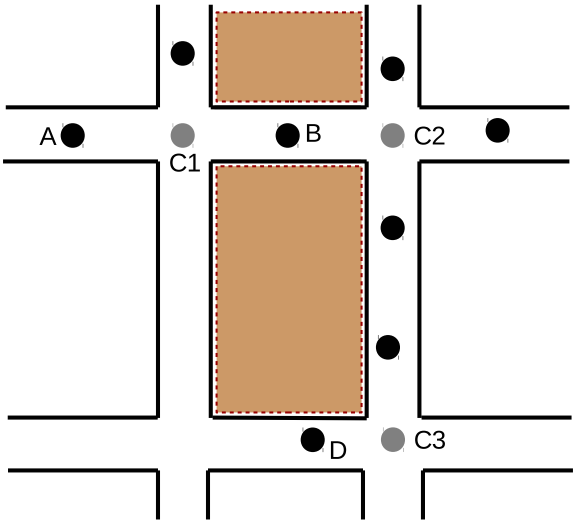


Figure 4.11: Prediction in GpsrJ+

VANETs. As shown, the more vehicles on the streets, the more connected is the VANET, i.e., it is more probable that a pair of source and destination find a route to communicate. Simulations show a big gap between the PDR of GpsrJ+ and the network connectivity.

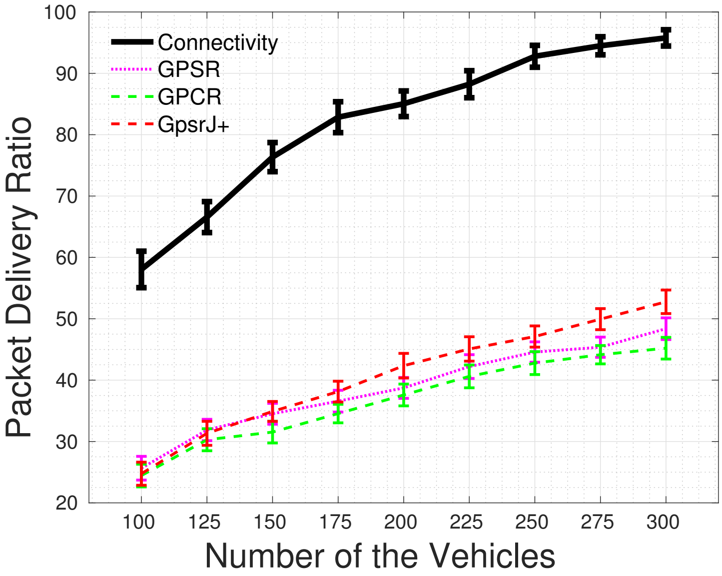


Figure 4.12: PDR of GpsrJ+

Kim et al. [100, 101] observed anomalies in GPSR as a result of the planarization algorithms, i.e., network partitions, asymmetric links, and cross-links. Cross-link problem occurs whenever there are intersections without coordinator vehicles in urban VANETs, as depicted in Figure 4.13. Lets assume the planar urban grid without any obstruction on the road segments and without any tunnels and bridges. If at least one ITS station exists at each junction, there exist no cross-links.

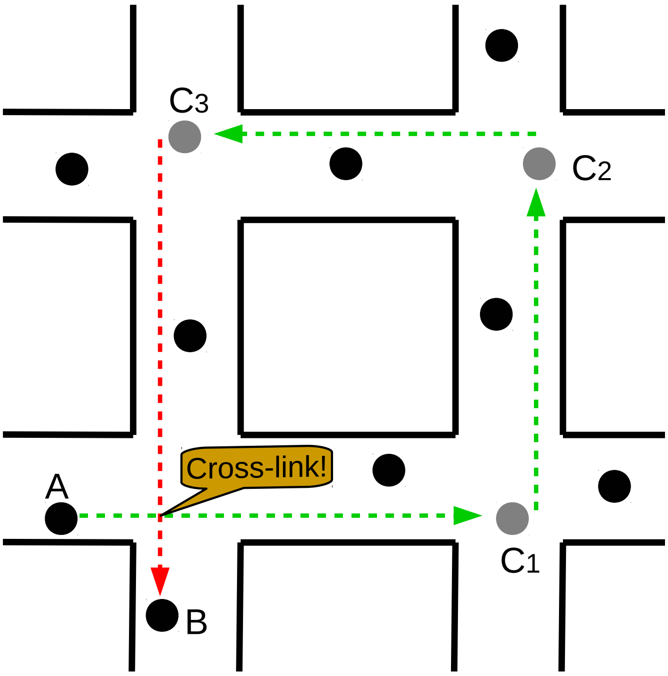


Figure 4.13: Cross-link

Kim et al. [102, 101] also observed that the graph of an urban VANET violates the unit graph assumption, i.e., the length of the edges is not the same. Therefore, two vehicle ITS stations can be connected even if their distance is more than their communication range. Also, two vehicle ITS stations might not be connected even if their distance is less than their communication range. This will worsen the three aforementioned pathologies, i.e., network partitions, asymmetric links, and cross-links.

Lee et al. [1] calculated the frequency of cross-links as the number of junctions without coordinator vehicle ITS stations whose road segments are filled with vehicle ITS stations, divided by the total number of junctions whose road segments are filled with vehicle ITS stations. To do so, they considered the vehicle ITS stations that are in the 15 m radius of the junction as coordinator.

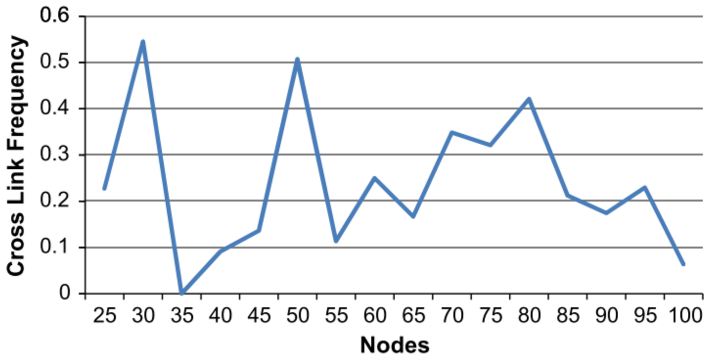


Figure 4.14: Cross-link frequency [1]

As seen in Figure 4.14, as the number of vehicle ITS stations increases in the network, the cross-link frequency decreases. Despite the descending trend, cross-links exists always. Also, Figure 4.14 shows a high standard deviation from the average cross-link frequency. This happens because the cross-link frequency is greatly affected by the fast movement of vehicle ITS stations.

Lee et al. [1] proposed GeoCross, that also consists of greedy forwarding and perimeter forwarding modes like GPSR. GeoCross also takes advantage of the natural planar graph of the urban streets and intersections. GeoCross defines a *missing-junction* as "a link such that packets travel directly from one road segment to another because they are connected by an empty junction [1]". An empty junction is a junction at which no vehicle is located.

Cross-link is considered as a communication link that has been crossed by another communication link. Only cross-links that happen at missing-junctions are problematic and should be detected. As depicted in Figure 4.13, a packet takes the route $A \rightarrow C_1 \rightarrow C_2 \rightarrow C_3 \rightarrow B$, where link $\overline{AC_1}$ crosses with link $\overline{C_3B}$ at a missing-junction. Therefore, this cross-link should be detected and either $\overline{AC_1}$ or link $\overline{C_3B}$ should be removed.

Crossing links that happen at road segments are not problematic. As depicted in Figure 4.15, although $\overline{AB}$ crosses with link $\overline{CD}$, there is no need to remove any of them.

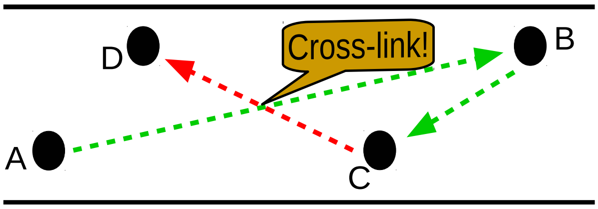


Figure 4.15: Cross-link at a road segment

As mentioned before, the CA basic service of the receiving ITS station makes the content of the CAMs accessible for its ITS applications and other facilities, e.g., LDM. Also, the ITS stations that receive the CAM become informed about the type, presence, and the status of the originating ITS station. The receiving ITS stations can employ these information to support several ITS applications. Employing GPS devices and LDM, each vehicle knows not only its position information but also the road ID of the road segment at which the vehicle is located. Thus, GeoCross takes advantage of the road IDs and the junction IDs in order to detect the cross-links.

GeoCross applies an on-demand cross-link removing approach, that does not remove any cross-links unless a cross-link is detected while routing a packet in perimeter forwarding mode. To do so, it defines two main parameters as follows:

- *Probe* records the road IDs, the junction IDs and the missing-junction IDs that a packet travels.

- *Unroutable Road (UR)* records the road IDs, that packets are not permitted to travel in future.

When a vehicle ITS station receives a packet in perimeter forwarding mode, GeoCross calculates its distance from the destination ITS station, i.e., d_2 and compares it to the distance from the destination ITS station at the point the packets first entered the perimeter forwarding mode, i.e., d_1. If d_2 is less than d_1, then GeoCross switches back to the greedy forwarding mode. If it decides to stay in perimeter forwarding mode, then it checks if the forwarding vehicle ITS station is a coordinator or not. If it is a non-coordinator, then GeoCross looks for the furthest potential vehicle ITS station forwarder.

If it is a coordinator, GeoCross starts the on-demand cross-link detection and removing approach. Afterwards, GeoCross employs the right-hand rule to find the road segment that is counterclockwise the first one from the current road segment and forwards the packet to a vehicle ITS station located on that road segment. But before forwarding the packet, GeoCross determines whether the links between the current vehicle ITS station and the next vehicle ITS station are saved in the UR. If the links are saved in UR, GeoCross will not forward the packet to the corresponding vehicle ITS station and will find a new vehicle ITS station, again employing right-hand-rule. If the links are not saved in UR, GeoCross forwards the packet to the next vehicle ITS station.

It is important to check whether the current road ID or junction ID is already saved in the probe. If the road ID or junction ID is already saved in the probe, it means that the current road or junction has been traveled before. Therefore, GeoCross needs to check whether a cross-link happens or no.

GeoCross searches for cross-link and symmetric-links at each missing-junction link. If GeoCross finds a cross-link and the vehicle ITS station is adjacent to it, it decides whether its adjacent link or its crossing link should be placed in UR. GeoCross tries to avoid partitioning the network while determining whether a cross-link is removable. In particular, GeoCross removes a link only if its symmetric link does not exist within the probe. Removing a symmetric link can partition the network because the link is traveled in both directions. In case both cross-links do not have symmetric links, GeoCross randomly removes one of them. The removed link will be recorded in UR. Moreover, the recorded elements of probe should be flushed. Next time that vehicle ITS stations make a routing decision, they will ignore all the links listed in UR.

A detailed example of cross-link detection within perimeter forwarding is depicted in Figure 4.16. Vehicle S attempts to forward a packet in perimeter forwarding toward the destination vehicle D. GeoCross employs the right-hand rule and forwards the packet to vehicle A, and afterwards to vehicle B which is a coordinator at junction J_1. So far, the road ID R_1, junction ID J_1 and the missing-junction ID R_1R_2 are recorded in the probe. Afterwards, the packet gets routed through $B \rightarrow E \rightarrow F \rightarrow H \rightarrow C \rightarrow B$ and comes back to vehicle B. So far, the probe looks like $[R_1, R_1R_2, J_1, J_2, J_3, R_5R_6, J_4, J_5, J_1]$.

At this point, GeoCross realizes that the junction ID J_1 appears twice in the probe. Therefore, GeoCross considers the part of the probe enclosed by J_1, i.e., [J_1, J_2, J_3, R_5R_6, J_4, J_5, J_1], and checks whether a cross-link happened, which is not the case. Therefore, GeoCross continues to route the packet in perimeter forwarding mode and forwards the packet to vehicle A. So far, the probe looks like [R_1, R_1R_2, J_1, J_2, J_3, R_5R_6, J_4, J_5, J_1, R_2R_1, R_1]. At this point, GeoCross realizes that the road ID R_1 appears twice in the probe. Therefore, GeoCross considers the part of the probe enclosed by R_1, i.e., [R_1, R_1R_2, J_1, J_2, J_3, R_5R_6, J_4, J_5, J_1, R_2R_1, R_1], and checks whether a cross-link happened. This time, GeoCross finds out that there is a cross-link, i.e., [R_1R_2, R_5R_6].

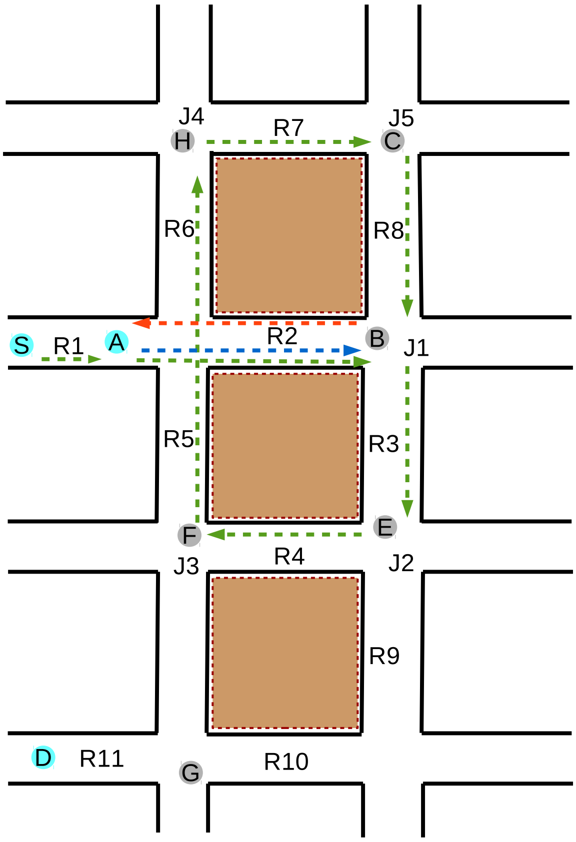


Figure 4.16: An example of perimeter forwarding in GeoCross

At this point, GeoCross decides which link should to be added to UR. This decision should be made in a way not to disconnect the network. R_1R_2 and its symmetric link R_2R_1 are both included in the probe. Therefore, removing R_1R_2 yields in network partitioning. On the other hand, R_5R_6 is an asymmetric link. Therefore, removing R_5R_6 does not yield in network partitioning. As a result, GeoCross adds R_5R_6 to the UR and flushes the probe.

After removing the cross-link, GeoCross sends the packet to the furthest node on the next road segment or next intersection recorded in the loop.

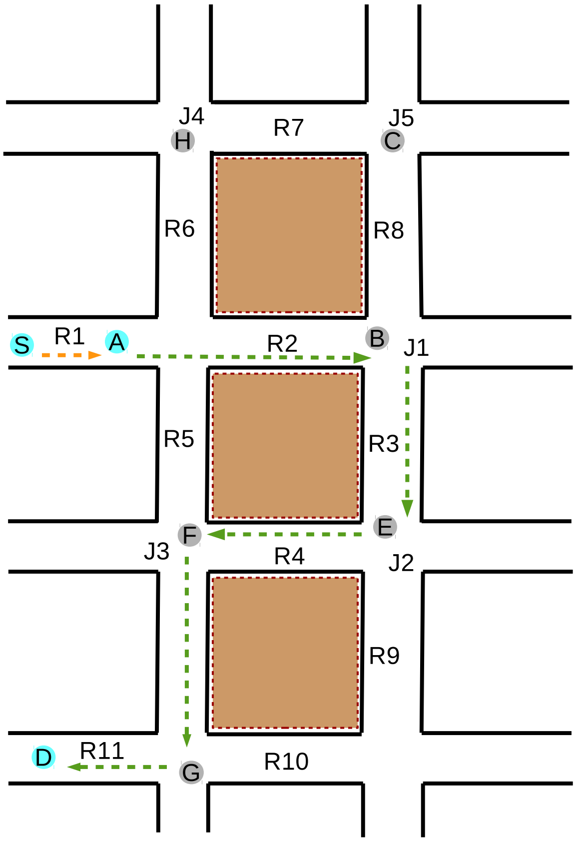


Figure 4.17: New path after removing the cross-link

Therefore, vehicle A forwards the packet in perimeter forwarding towards the vehicle B which is a coordinator at junction J_1. Subsequently, GeoCross employs the right-hand rule and forwards the packet to vehicle E and vehicle E forwards the packet to vehicle F. Vehicle F employs the right-hand rule and finds the vehicle H as the next possible forwarder. But, it checks the UR and finds the R_5R_6 recorded in it. Therefore, Vehicle F employs the right-hand rule to find the next possible forwarder, i.e., vehicle G and forwards the packet to it. The distance between vehicle G and the destination vehicle D is less than the distance between source vehicle S and the destination vehicle D. Thus, GeoCross switches back to greedy routing mode and sends the packet successfully to the destination vehicle D. The new route, i.e., $S \rightarrow A \rightarrow B \rightarrow E \rightarrow F \rightarrow G \rightarrow D$, is shown in Figure 4.17.

Figure 4.18 shows the PDR of GeoCross comparing to the GPSR, GPCR, GpsrJ+ and the *network connectivity*, corresponding to different number of vehicles in urban VANETs. As shown, the more vehicles on the streets, the more connected is the VANET, i.e., it is more probable that a pair of source and destination find a route to communicate. Simulations show a big gap between the PDR of GeoCross and the network connectivity.

Figure 4.19 shows the cross-link frequency of GeoCross, corresponding to different number of vehicles in urban VANETs. As expected, when the

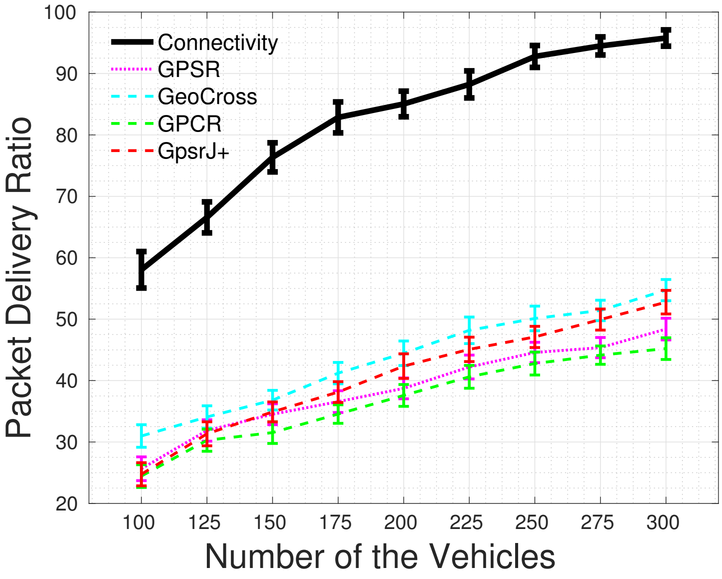


Figure 4.18: PDR of GeoCross

number of vehicle ITS stations increases in the network, the cross-link frequency decreases. Also, despite the descending trend, cross-links exist always. Moreover, Figure 4.19 shows a high standard deviation from the average cross-link frequency. This happens because the cross-link frequency is greatly affected by the fast movement of the vehicle ITS stations.

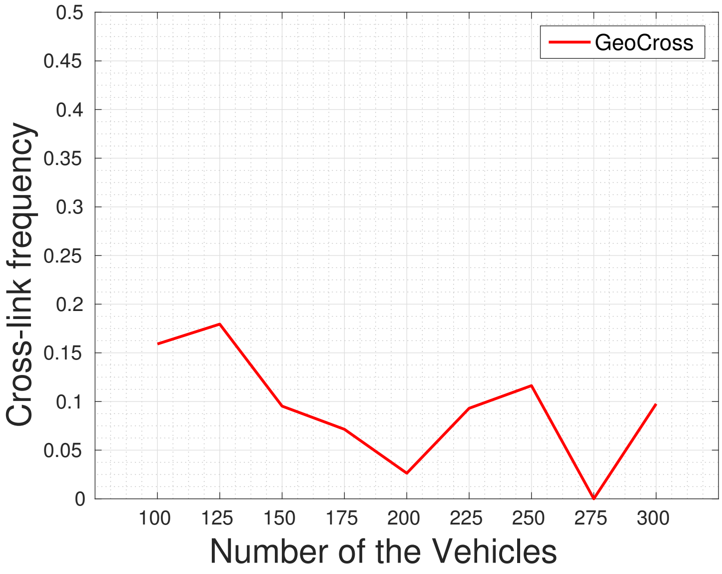


Figure 4.19: Cross-link frequency of GeoCross

Jerbi et al. [103] proposed GyTAR. It dynamically selects the intersections one after each other, to forward the packets through them. GyTAR calculates the number of vehicles located between each two intersections, based on the pre-loaded digital street-level maps. Considering the traffic density and the curvemetric distance of the intersection to the destination, it assigns a score to each intersection. Each vehicle predicts the new position of the neighbors and forwards the data packet to one of them, based on greedy forwarding. GyTAR employs CaF as a recovery strategy, in which a vehicle carries the data packet until it finds a suitable vehicle or it arrives to the intersection itself.

GyTAR consists of three mechanisms:

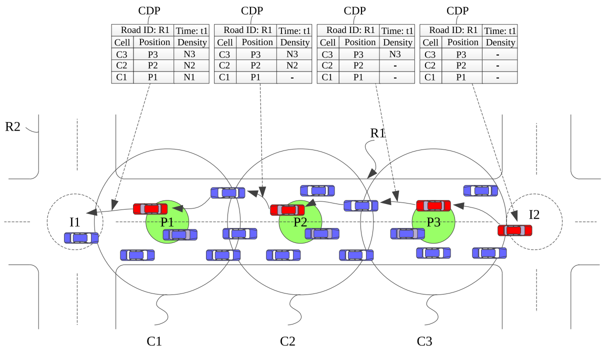
- *Traffic Density Estimation*, a completely decentralized mechanism to estimate the traffic density on the streets,
- *Intersection Selection*, a mechanism to select the intersections from the source to the destination in a dynamic way, and
- *Forwarding Data between Two Intersections*, an improved greedy forwarding process to forward the packets between intersections.

Employing the aforementioned three mechanisms, GyTAR forwards the packets in the direction of the destination, wherever there is enough vehicle density.

In order to estimate the traffic density on the streets decentrally, the traffic information of the streets should be exchanged between groups of vehicle ITS stations. Vehicle ITS stations are organized into different groups based on their positions. To do so, each street in divided into small fixed-area cells. The size of the cells depends of the communication range of the vehicle ITS stations and each cell has a unique ID based on the position of its center. These cells overlap in a way that, when a vehicle ITS station drives from one cell to another cell, it always belongs to at least one cell. Between the vehicle ITS stations that are located in one common cell, the vehicle ITS station that is the closest to the center of the cell will be the group leader for a specific duration of time. The cells overlap is small enough, so that it is not possible for a vehicle ITS station to be the group leader for two neighboring cells.

The local traffic density information are calculated by each group leader and are relayed between these groups employing the Cell-Density Packets (CDPs). The vehicle ITS stations that have been already a group leader, generate a CDP, as soon as they reach an intersection, i.e., only the vehicle ITS stations that already have generated a CDP will send a new CDP before leaving that road segment. This way, the CDP generation is limited in order to avoid the overhead issues. Moreover, the CDP generation is adapted to the dynamic of the traffic density on that street.

As depicted in Figure 4.20, when generating a CDP, the group leader vehicle ITS station records the generation time, road ID and the sequence of the cells that the CDP should go through to reach the intersection on the other side of the road segment. As a result, the CDP will be sent to the intersection on the opposite side of the road segment in an improved greedy forwarding strategy. As soon as a group leader receives a CDP, it updates the CDP and sends it towards the next cell. The group leaders repeat this process until the CDP reaches the intersection on the other side of the road segment. Therefore, all the vehicle ITS stations that are located around the intersection and are going to cross the intersection will receive the traffic density information of this road segment, and in a similar way, they will receive the traffic density information of all the other road segments of the intersection.



CDP

Road ID: R1		Time: t1
Cell	Position	Density
C3	P3	N3
C2	P2	N2
C1	P1	N1

CDP

Road ID: R1		Time: t1
Cell	Position	Density
C3	P3	N3
C2	P2	N2
C1	P1	-

CDP

Road ID: R1		Time: t1
Cell	Position	Density
C3	P3	N3
C2	P2	-
C1	P1	-

CDP

Road ID: R1		Time: t1
Cell	Position	Density
C3	P3	-
C2	P2	-
C1	P1	-

Figure 4.20: Relaying CDP between intersections

These receiving vehicle ITS stations, calculate the traffic density of each road segment, based on the average number of member vehicle ITS stations of a cell and its standard deviation. N_i is the number of the vehicles in the cell i and N_c is the total number of the cells on a road segment.

$$N_{avg} = \frac{1}{N_c} \cdot \sum_{i=1}^{N_c} N_i \tag{4.2}$$

$$\sigma = \sqrt{\frac{1}{N_c} \cdot \left(\sum_{i=1}^{N_c} (N_i - N_{avg})^2 \right)} \tag{4.3}$$

In order to choose the next intersection, GyTAR takes advantage of the traffic density information of the streets and consider the distance to the destination vehicle ITS station and chooses the intersections sequentially and dynamically so that the packets can get routed following the topology of the street map. This approach has the advantage that the sequence of the intersections towards the destination is calculated using the updated information of traffic density, comparing it to the other approaches like in GSR and A-STAR.

With the help of the LDM, GyTAR looks for the position of the neighboring intersections. Each neighboring intersection is assigned an score, based on the traffic density information and its curvemetric[2] distance to the destination vehicle ITS station. The best intersection, i.e., the intersection with the highest score, is the closest one to the destination with the highest traffic density.

As depicted in figure 4.21, once vehicle S receives a packet, it calculates the score of each of the neighboring intersections, i.e., J_1, J_2 and J_3. Here, J_2 has the highest score and will be selected as the next intersection.

This way the selected route to the destination vehicle ITS station will have

---

[2] The distance measured following the geometric topology of the roads

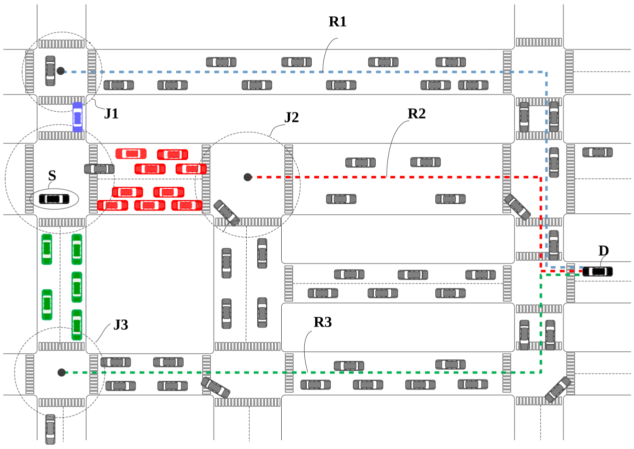


Figure 4.21: Selection the next intersection

the maximum connectivity. Therefore, GyTAR has two weighting factors; for traffic density β and for distance α, and it assumes that $\beta + \alpha = 1$. $f(D_j)$ calculates the distance score and $g(T_j)$ calculates the density score.

$$Score(J) = \alpha \cdot f(D_j) + \beta \cdot g(T_j) \tag{4.4}$$

$$f(D_j) = 1 - \frac{D_j}{D_i} \tag{4.5}$$

$$g(T_j) = \min\left(\frac{1}{\sigma + 1} \cdot \frac{N_{avg}}{N_{con}}, 1\right) \tag{4.6}$$

J is the next candidate intersection, I is the current intersection, D_j is the curvemetric distance of the candidate intersection from the destination vehicle ITS station, D_i is the curvemetric distance of the current intersection from the destination vehicle ITS station, and N_{con} is a constant number that gives the ideal connectivity in a cell. The multiplication of $(\frac{1}{\sigma+1} \cdot \frac{N_{avg}}{N_{con}})$ in $g(T_j)$, addresses the issue of having gaps between groups of vehicles on a road segment.

GyTAR applies an improved greedy strategy in order to forward a packet between two intersections. The requirements are the position and velocity vector of the neighbor vehicle ITS stations, that are obtained from the CAMs. Having this information, GyTAR can predict the future position of the neighbor vehicle ITS stations and choose the closest one to the next intersection.

Figure 4.22(a) shows the position of the vehicle ITS stations at time t_1, i.e., last time that the neighbor table was updated. Considering $t_1 < t_2$, Figure 4.22(b) shows the predicted position of the vehicle ITS stations at time t_2. Assuming that vehicle V_1 wants to forward a packet towards intersection

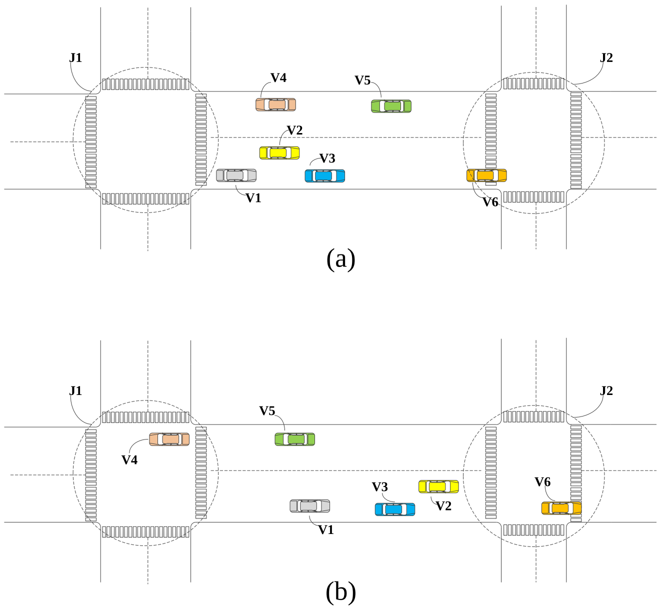


Figure 4.22: Forwarding a packet between two intersections

J_2. Vehicle V_2 is driving in that direction and has a greater velocity than vehicle V_3. Thus vehicle V_2 is selected as the next forwarder vehicle. Without predicting the future position of the vehicle ITS stations, vehicle V_1 would have forwarded the packet to vehicle V_5, which at time t_2 is closer to the intersection J_1.

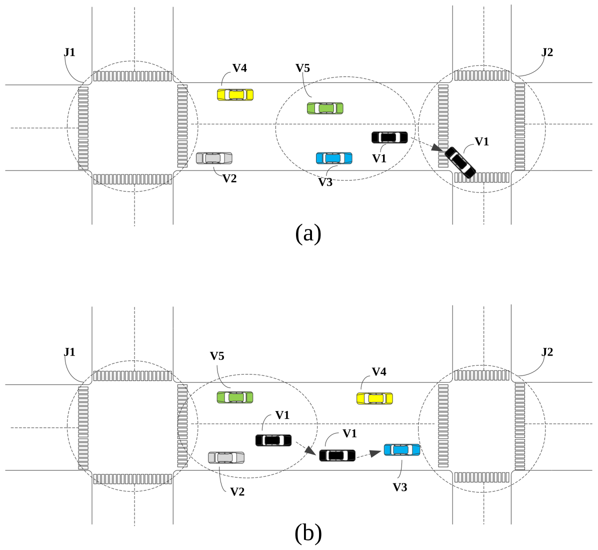


Figure 4.23: Recovery strategy of GyTAR

Considering the aforementioned basic version of GyTAR, it is still possible for a packet to get stuck in a so called local optimum, i.e., the forwarding vehicle ITS station does not have any neighbor vehicle ITS station that is closer to the destination vehicle ITS station than itself. Therefore, GyTAR adopts a recovery strategy that works based on the CaF [104] concept. In such a case, the the forwarding vehicle ITS station, i.e., V_1, will carry the packet until it reaches the next intersection, as shown in Figure 4.23(a), or until it finds another vehicle closer to the next intersection than itself, i.e., V_3, as shown in Figure 4.23(b).

Figure 4.24 shows the PDR of basic GyTAR considering different values for α and β, corresponding to different number of vehicles in urban VANETs. It is interesting to note that, for the low traffic densities, the PDR of basic GyTAR for the three different configurations show a small difference of about 1%. In case of high traffic densities, this difference increases up to 8%. In other words, putting more weight on the distance score leads us to a better performance in case of high traffic densities.

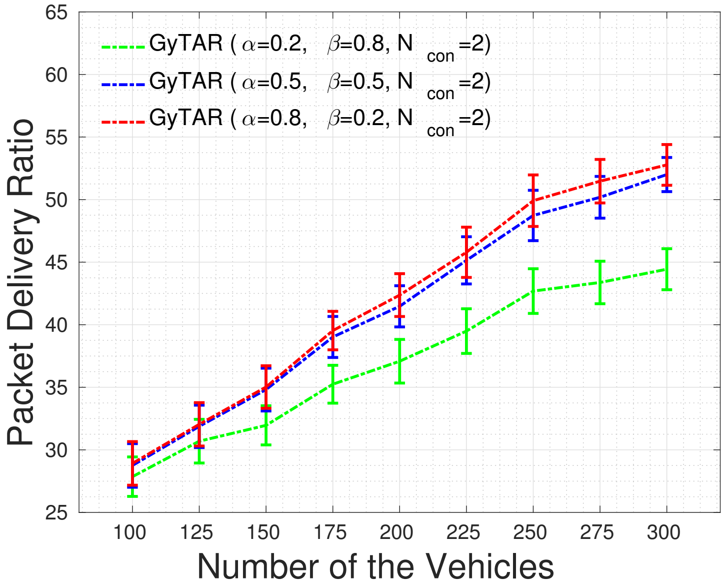


Figure 4.24: PDR of basic GyTAR for different α and β

Figure 4.25 shows the PDR of basic GyTAR comparing to the GPSR, GPCR, GpsrJ+, GeoCross and the *network connectivity*, corresponding to different number of vehicles in urban VANETs. It is assumed that $N_{con} = 2$ and $\alpha = \beta = 0.5$. As shown, the more vehicles on the streets, the more connected is the VANET, i.e., it is more probable that a pair of source and destination find a route to communicate. Simulations show a big gap between the PDR of GyTAR and the network connectivity.

4.3 EIPG

In this section, I propose EIPG, that addresses the problem of local optimum. It inherits the restricted greedy forwarding of GpsrJ+ but employs a new

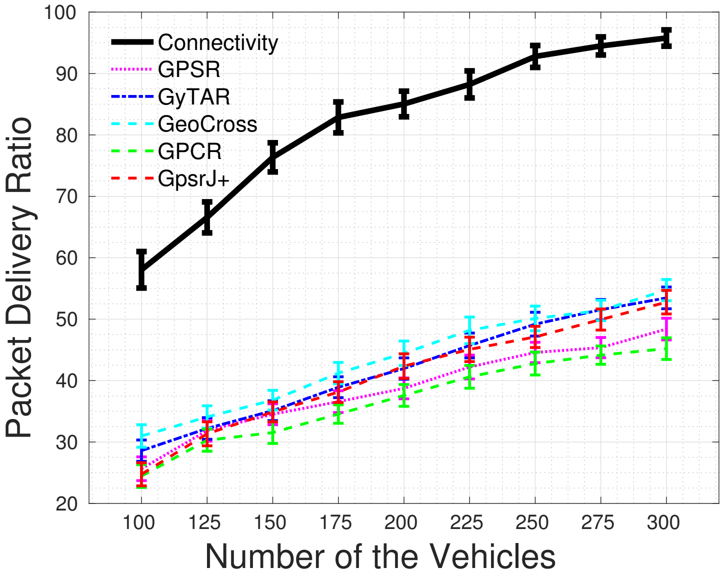


Figure 4.25: PDR of basic GyTAR

intersection-based perimeter forwarding in order to avoid the problem of WSE.

4.3.1 *Wrong Street Estimation*

As explained in the previous section, in perimeter mode and at intersections, GPCR and GpsrJ+ try to find the next road segment to forward the packet, employing the right-hand rule. First they calculate the center points of all the road segments around an intersection, assuming that it is possible to extract the coordinates of any road segment's two ends from the available LDM. Then, having the position information of the coordinator vehicle, they calculate the θ_i for each road segment, which is the angle that the edge $\overline{e_i}$, from the coordinator vehicle to the road segment i's center point, forms with the X-axis[3]. Afterwards, they calculate α, which is the angle that the edge from the coordinator vehicle to the previous sender vehicle, forms with the X-axis. Finally, they choose the road segment with the smallest θ_i which is bigger than α.

As depicted in Figure 4.26, the aforementioned procedure does not always lead to the correct decision. Assume that the coordinator vehicle V_3 has received a packet in perimeter mode from vehicle V_2. Vehicle V_3 first calculates all the θ_i angles and α, which is the angle of the edge $\overline{V_3V_2}$ counterclockwise from the X-axis. After that, vehicle V_3 tries to find out the smallest θ_i which is bigger than α. In this case, it is θ_2, which is the angle of the edge $\overline{e_2}$ counterclockwise from the X-axis. Finally, vehicle V_3 will choose the road segment towards South as the right-hand road segment, which is the correct decision to make. Now lets assume, that vehicle V_3 has received a packet in perimeter mode from vehicle V_1. First vehicle V_3

---

[3] The West-East horizontal direction

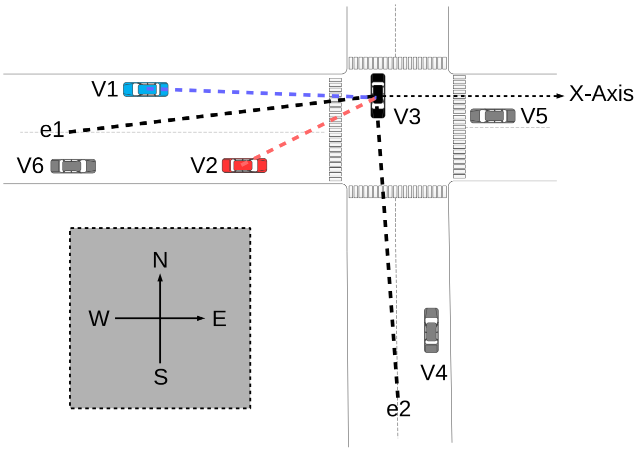


Figure 4.26: Illustration of the WSE problem

tries to find out the smallest θ_i which is bigger than α. In this case, it is θ_1, which is the angle of the edge $\overline{e_1}$ counterclockwise from the X-axis. Finally, vehicle V_3 will choose the road segment towards West as the right-hand road segment. But the forwarder vehicle V_1 is also located on the road segment towards West. Therefore, it is obviously a wrong decision to make. I termed this problem as WSE.

In order to avoid the problem of WSE, the procedure of choosing the right-hand street should be as proposed in the following. Having the digital map available, the coordinator vehicle V_3 can acquire the position of the intersection's node point O as it is shown in Figure 4.27. Then, it calculates θ_i for each road segment, which is the angle that the edge $\overline{e_i}$, from the point O to the road segment i's center points, forms with the X-axis.

Figure 4.27: Intersection's node point (from OpenStreetMap & Bing)

Considering Figure 4.28, and knowing θ_i for each road segment, it is possible to calculate the angle between each angle bisector, i.e., $\overline{oa}$, $\overline{ob}$, $\overline{oc}$ and $\overline{od}$, and the $\overline{e_i}$ edges located on their sides. For example, the angle between $\overline{oa}$ and the edges $\overline{e_1}$ or $\overline{e_2}$, is equal to $0.5 \times (\theta_2 - \theta_1)$. Having these angles, it is possible to divide the area around the intersection of Figure 4.28 into four slices, i.e., the area of $\angle aob$, $\angle boc$, $\angle cod$, and $\angle doa$. As a result, based on the angle of each neighbor vehicle's virtual edge from the point O, it is possible to calculate on which road segment is any neighbor vehicle located. Therefore, the coordinator vehicle can correctly calculate the road segment the packet comes from and the road segment it should take based on the right-hand rule.

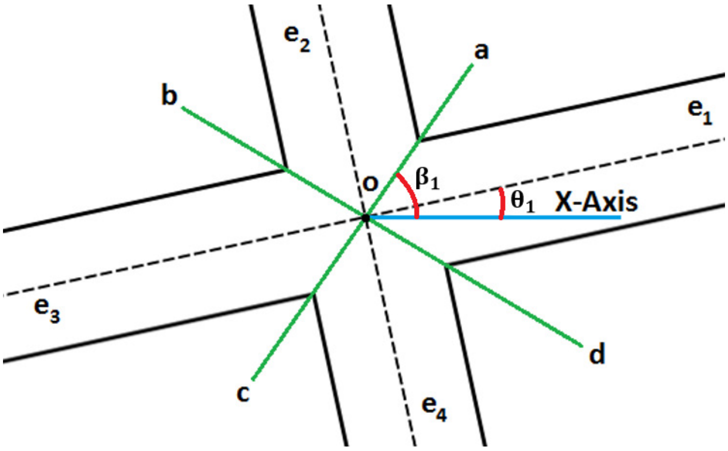


Figure 4.28: Perimeter forwarding with the help of angle bisectors

In other words, in perimeter mode, GpsrJ+ tries to find the road segment with the smallest θ_i which is bigger than α. But EIPG tries to find the angle bisector with the smallest angle β_i from X-axis, which is bigger than α. And then starts rotating counterclockwise from this angle bisector around the point O to find the potential forwarder vehicles. This way, the problem of WSE is avoided.

Another difference between GpsrJ+ and EIPG is that, EIPG requires only the information of one hop, i.e., direct neighbor vehicles to avoid the unnecessary overhead caused by the extra information included in the beacon messages of GPCR and GpsrJ+, and to be more compatible with the standard CAMs.

According to ETSI EN 302 637-2 [20], CAMs are exchanged in VANETs between vehicles to make each other aware of their existence. Status and attribute information of a vehicle is contained in the originated CAMs. Status information includes the time, position and motion state. Attribute information includes the dimensions, vehicle's type and role in the road traffic. The CAM is transmitted in a single hop to the vehicles located in the direct communication range of the originating vehicle. None of the vehicles should forward a received CAM to other vehicles.

4.3.2 *Simulation Evaluations*

In this section, the performance of EIPG is evaluated, comparing it with GPSR, GPCR, GpsrJ+, GeoCross and basic GyTAR. The performance of these geo-routing protocols has been evaluated concerning reachability, scalability and latency:

- *Reachability*: PDR is the number of packets received by the destination, divided by the number of packets originated by the source.
- *Latency*: Transmission latency is defined as the "time interval between the time when a V2X message is delivered from the facilities layer to the network and transport layer at the sending ITS station and the time when a V2X message is delivered from the network and transport layer to the facilities layer at the receiving ITS station [16]". In other words, end-to-end latency is the time taken for a packet to be transmitted from the source to the destination.
- *Scalability*: Routing overhead is the average number of routing packets received by all the nodes.

4.3.2.1 *Assessment of Reachability*

Figure 4.29 shows the PDR of GPSR, GPCR, GpsrJ+, GeoCross, basic GyTAR and EIPG, corresponding to different number of vehicles in urban VANETs. The more vehicles on the streets, the more connected is the VANET, i.e., it is more probable that a pair of source and destination find a route to communicate. That is why the graphs show an upward trend. EIPG shows an improvement of about 5% in average in comparison to the second best geo-routing protocol, i.e., GeoCross. This improvement tends to increase with the number of vehicles so that it reaches to more than 7% at the end. This upward trend can be explained, considering the fact that the more dense is the VANET, the more is the probability that the WSE problem occurs. Simulations still show a big gap between the PDR of the geo-routing protocols and the network connectivity.

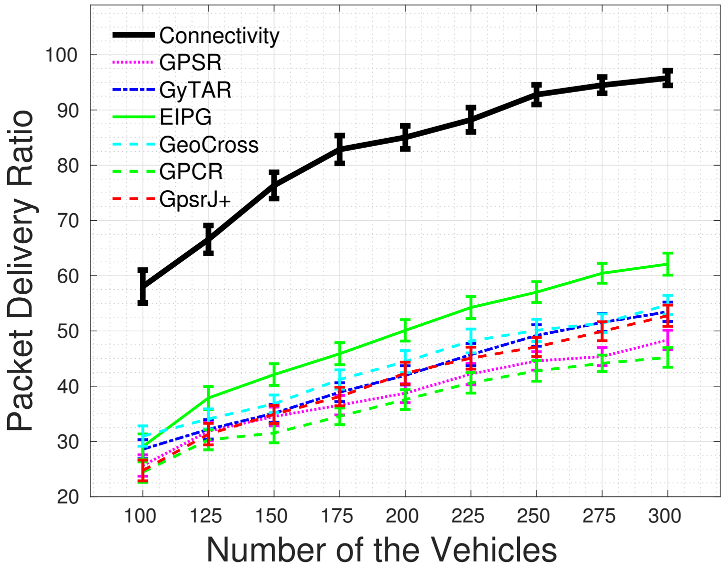


Figure 4.29: PDR for different number of vehicles

4.3.2.2 *Assessment of Latency*

Figure 4.30 shows the Cumulative Distribution Function (CDF) of end-to-end latency for GPSR, GPCR, GpsrJ+, GeoCross, basic GyTAR and EIPG, corresponding to an aggregation of all the different number of vehicles. This figure gives us the probability that a packet can be received by any destination in VANET within a specific time. In order to have more meaningful graphs, all the packets which are not delivered, are included in Figure 4.30 considering an end-to-end latency of infinity. As an example, the probability that a packet can be received by any destination in less than 40 ms, employing EIPG is about 49%. In case of GyTAR, it is about 40%, in case of GpsrJ+, it is about 39%. For GPCR and GeoCross it is about 36% and for GPSR about 37%. Therefore, EIPG shows up to 9% improvement in comparison to the second best performing geo-routing protocol, i.e., GyTAR, in terms of end-to-end latency.

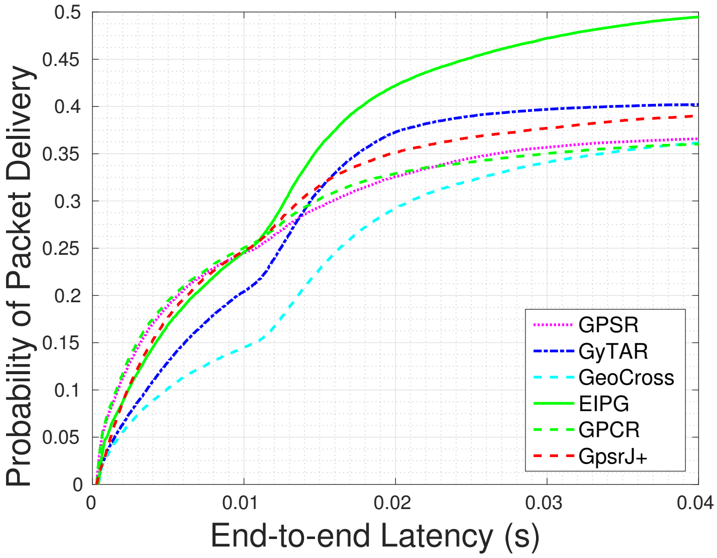


Figure 4.30: CDF of end-to-end latency, aggregating all packets

4.3.2.3 *Assessment of Scalability*

Figure 4.31 shows the routing overhead of GPSR, GPCR, GpsrJ+, GeoCross, basic GyTAR and EIPG, corresponding to different number of vehicles in urban VANETs. This figure illustrates the required average number of routing packets for successful delivery of a packet to any destination. For example, considering 300 vehicles in VANET, EIPG needs to send about 20 packets for routing purposes to deliver a packet from any source to any destination. GpsrJ+ accomplishes the same routing task sending about 31 packets for routing purposes to deliver a packet from any source to any destination. Therefore, in terms of routing overhead, EIPG improves GpsrJ+ by about 35% in average. GeoCross performs this routing task sending about 13 routing packets and GyTAR accomplishes the same task sending about 6 routing packets. Thus, in case of routing overhead, GyTAR shows the best performance, because it does not employ perimeter forwarding. Also, GeoCross performs better than EIPG, because of reacting to the cross-links.

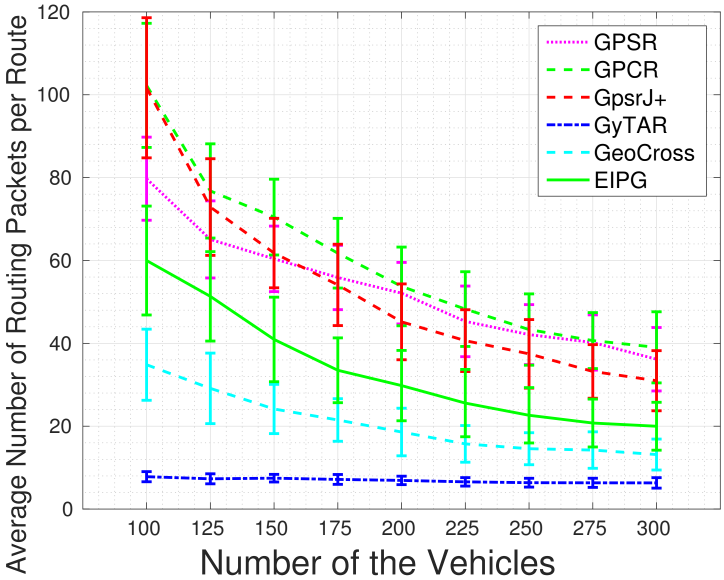


Figure 4.31: Routing overhead

As explained previously, GpsrJ+ requires the road ID information of neighbors of neighbors, i.e., information of two hops, in order to perform the required prediction in perimeter mode. GPCR also proposed an alternative mechanism to recognize coordinator vehicles which needs the position information of two-hop neighbors. According to [105], it is needed to have 8 extra Bytes to represent the latitude and longitude information of each two-hop neighbor. Based on my simulations, in average, each vehicle can have between 7 to 23 neighbor vehicles. This means that, to consider the information about two hops in GPCR, vehicles need to send between 56 to 184 extra Bytes for each generated CAM. Considering an average CAM size of 400 Bytes [106], this leads to an overhead in the range of 14% up to 46% for urban VANETs. Note that, in GPCR the time stamp of each two-hop neighbor is not included in the beacon messages and therefore not in the calculated overhead, which can increase it even more. As a result, EIPG employs only the information of one hop, therefore do not employ the prediction mechanism of GpsrJ+ to avoid this overload.

4.4 EIPG2

In this section, I propose EIPG2, that addresses the problem of partitioned networks and routing loops. It applies a new preferential intersection-based perimeter forwarding based on the intersection-based perimeter forwarding of EIPG [97] in order to avoid the problem of WSE. Also, instead of the restricted greedy forwarding of EIPG [97], it employs a new preferential unrestricted greedy forwarding. In order to avoid the cross-link problem, it adopts the CLD mechanism of GeoCross [1]. Also, EIPG2 takes advantage of new LD and CaF approaches to improve EIPG in terms of PDR, end-to-end delay, and network overhead. Unlike its predecessors, EIPG2 has three forwarding modes: *greedy*, *perimeter* and *carry* forwarding mode. The source vehicle always starts in *greedy* forwarding mode.

4.4.1 *Greedy Forwarding Mode*

In urban VANETs, geo-routing protocols often have difficulties to get the packets behind the building blocks and the coordinator vehicles are the only ones that can perform this task. Since coordinator vehicles have more neighbor vehicles comparing to the non-coordinator vehicles, they have more and normally better options to forward the data packet. Therefore, in the *greedy* forwarding mode, unlike EIPG [97] and GpsrJ+ [60], EIPG2 prefers to send data packets to the coordinator neighbor vehicles. If EIPG2 could not find a suitable coordinator neighbor vehicle, then it searches for a suitable non-coordinator neighbor vehicle. This new preferential greedy forwarding results in an increased PDR and fewer number of hops, i.e., less overhead and latency.

Algorithm 1 Greedy forwarding mode

```
procedure GREEDY FORWARDING
    if a loop is detected then
        switch to carry forwarding mode
    else
        find mpcnv
        if mpcnv is specified then
            forward the packet to mpcnv
        else
            find mpnnv
            if mpnnv is specified then
                forward the packet to mpnnv
            else
                if I am a coordinator then
                    switch to perimeter forwarding mode
                else
                    switch to carry forwarding mode
```

As a result of the new preferential greedy forwarding, it is no longer required to send the packets in a restricted greedy manner towards the next intersection. In VANETs, it is important to use the whole communication range of vehicles to forward the packet in order to decrease the number of the hops to reduce the latency. Thus, unlike EIPG [97] and GpsrJ+ [60], EIPG2 sends the packets as far as the communication range of the vehicle allows. Also, unlike in EIPG [97] and GpsrJ+ [60], in EIPG2 vehicles do not need to consider a virtual line from the previous vehicle to themselves, in order to keep forwarding the packet in the same direction. Taking advantage of the new LD approach, EIPG2 can send the packets to any qualified vehicle in any direction. This new unrestricted greedy forwarding again leads to an increased PDR and fewer number of hops, i.e., less overhead and latency.

In *greedy* forwarding mode of EIPG2, first the source/forwarder vehicle checks if a loop is detected, i.e., if it has already forwarded this packet before. The purpose of the LD is to avoid redundant forwarding. If yes, it switches to *carry* forwarding mode. If no, it searches within its coordinator neighbors to find the coordinator neighbor with the maximum progress towards the destination vehicle and forwards the packet to it. If the source/forwarder vehicle has no coordinator vehicle, that is closer than itself to the destination vehicle, then it searches within its non-coordinator neighbors to find the non-coordinator neighbor with the maximum progress towards the destination vehicle and forwards the packet to it. If the source/forwarder vehicle also has no non-coordinator neighbor, that is closer than itself to the destination vehicle, then it switches to *carry* forwarding mode if it is a non-coordinator vehicle, and it switches to *perimeter* forwarding mode if it is a coordinator vehicle.

Algorithm 1 outlines the *greedy* forwarding mode, in which *mpcnv* stands for the maximum progress coordinator neighbor vehicle and *mpnnv* stands for the maximum progress non-coordinator neighbor vehicle.

4.4.2 *Perimeter Forwarding Mode*

As mentioned before, coordinator vehicles have more neighbor vehicles comparing to the non-coordinator vehicles. As a result, they have more and normally better options to forward the data packet. EIPG [97] has implemented an intersection-based perimeter forwarding, in which coordinator vehicles decide about the next hand of the intersection to forward the data packet to it, i.e., the coordinator vehicles know which direction is the best to take. This provides the opportunity to go one step further and look for the best vehicle in that direction, instead of simply sending the packet to the next vehicle counterclockwise located on that direction. Thus, in the *perimeter* forwarding mode, unlike EIPG [97] and GpsrJ+ [60], EIPG2 prefers to send data packets to the coordinator neighbor vehicles. If EIPG2 could not find a suitable coordinator neighbor vehicle, then it searches for a suitable non-coordinator neighbor vehicle. This new preferential intersection-based perimeter forwarding yields in an increased PDR and fewer number of hops, i.e., less overhead and latency.

In *perimeter* forwarding mode of EIPG2, first the source/forwarder vehicle checks if it is closer to the destination than the point at which this packet hit the local optimum. If yes, it switches to *greedy* forwarding mode. If no, then it checks if a cross-link is detected. If yes, it marks the asymmetric link as unroutable and sends the packet back to its sender. If no, then it checks if a loop is detected. If yes, it switches to *carry* forwarding mode. If no, it tries to find the next vehicle based on the new preferential intersection-based perimeter forwarding, in which coordinator neighbors have preference over non-coordinator neighbors. If it does not find a suitable neighbor vehicle

Algorithm 2 Perimeter forwarding mode

```
procedure PERIMETER FORWARDING
    if closer to the destination than the local optimum then
        switch to greedy forwarding mode
    else
        if a cross-link is detected then
            - mark the asymmetric link as unroutable
            - forward the packet back to its sender
        else
            if a loop is detected then
                switch to carry forwarding mode
            else
                find pcnv
                if pcnv is specified then
                    forward the packet to pcnv
                else
                    find pnnv
                    if pnnv is specified then
                        forward the packet to pnnv
                    else
                        switch to carry forwarding mode
```

employing preferential intersection-based perimeter forwarding, then it switches to *carry* forwarding mode.

Algorithm 2 outlines the *perimeter* forwarding mode, in which *pcnv* stands for the perimeter coordinator neighbor vehicle and *pnnv* stands for the perimeter non-coordinator neighbor vehicle.

4.4.3 *Carry Forwarding Mode*

Although GyTAR [103] employs an improved greedy routing strategy, it is still possible that a packet hits a local optimum, i.e., the forwarding vehicle is the closest to the next intersection. In this case, it applies a recovery strategy, in which the source/forwarder vehicle carry the packet until the next intersection or until it senses another vehicle closer to the next intersection. In my simulations, I have realized that this carrying vehicle might be driving not towards the closer intersection to the destination. As a result, it is going to carry the packet for an unnecessarily longer time. Also, it is not always worthy to apply the CaF method, for example when the source/forwarder vehicle is too far from the next intersection. In such a case, applying the *perimeter* forwarding mode can improve the performance of the geo-routing protocol. Therefore, EIPG2 considers the driving direction of the source/forwarder vehicle and its distance to the closest intersection to the destination in order to have a higher PDR and a lower latency.

In *carry* forwarding mode of EIPG2, first the source/forwarder vehicle checks if there is any neighbor closer than itself to the destination vehicle. If

Algorithm 3 Carry forwarding mode

```
procedure CARRY FORWARDING
    find mpnv
    if mpnv is specified then
        forward the packet to mpnv
    else
        find cnvdi
        if cnvdi is specified then
            - forward the packet to cnvdi
            - cnvdi queues the packet
        else
            switch to perimeter forwarding mode
```

yes, then it forwards the packet to it. If no, it tries to find the closest vehicle around its intersection driving towards the intersection and forwards the packet to it. If there is no vehicle close to and driving to this intersection, it switches to *perimeter* forwarding mode. Otherwise, this vehicle queues the packet and waits until it becomes a coordinator and then continues forwarding the packet in *greedy* forwarding mode.

Algorithm 3 outlines the *carry* forwarding mode, in which *mpnv* stands for the maximum progress neighbor vehicle and *cnvdi* stands for the closest neighbor vehicle around the intersection driving towards it.

4.4.4 *Cross-link Detection*

As depicted in Figure 4.13, only two links that are at different road segments and cross each other at an intersection cause a problematic cross-link. Figure 4.32 shows the cross-link frequency in EIPG, corresponding to different number of vehicles in urban VANETs. The adapted cross-link detection and avoidance approach of EIPG2 defines a cross-link just like it is defined in GeoCross. It is also on-demand, which means it does not exclude any link, i.e., vehicle, until a cross-link is detected during the routing. Therefore, they have the same CLD mechanism up to the point that they detect a cross-link. At this point, when GeoCross detects a cross-link, it records the asymmetric link as unroutable and forwards the packet to the furthest vehicle on the next road segment or next intersection recorded in the loop. EIPG2 records the asymmetric link as unroutable and sends the packet back to its sender.

A detailed example of cross-link detection within perimeter forwarding is depicted in Figure 4.33. Vehicle S attempts to forward a packet in preferential intersection-based perimeter forwarding toward the destination vehicle D. EIPG2 employs the right-hand rule and forwards the packet to vehicle B which is a coordinator at junction J_1. So far, the road ID R_1, the missing-junction ID R_1R_2 and the junction ID J_1 are recorded in the probe. Afterwards, the packet get routed through $B \to E \to F \to H \to C \to B$ and comes back to vehicle B. So far, the probe looks like $[R_1, R_1R_2, J_1, J_2, J_3, R_5R_6, J_4, J_5, J_1]$. At this point, EIPG2 realizes that the junction ID J_1 appears

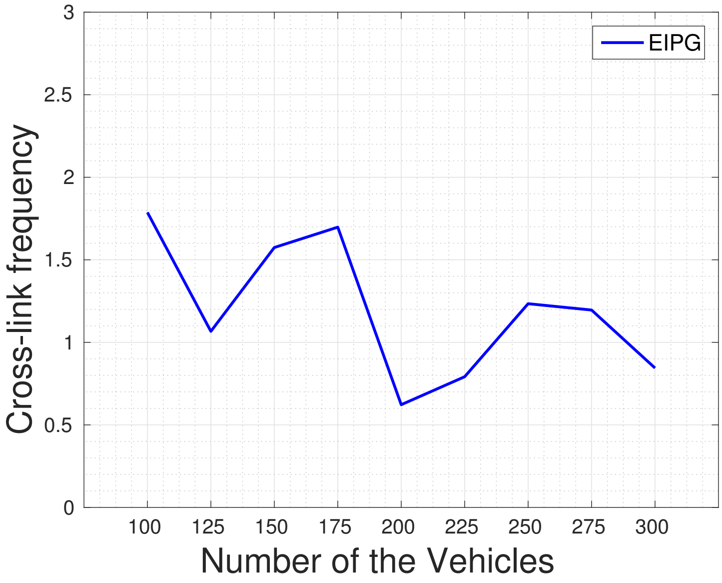


Figure 4.32: Cross-link frequency of EIPG

twice in the probe. Therefore, EIPG2 checks the part of the probe enclosed by J_1, i.e., [J_1, J_2, J_3, R_5R_6, J_4, J_5, J_1], and checks whether a cross-link happened, which is not the case. Therefore, EIPG2 continues to route the packet in its perimeter forwarding mode and forwards the packet to vehicle A. So far, the probe looks like [R_1, R_1R_2, J_1, J_2, J_3, R_5R_6, J_4, J_5, J_1, R_2R_1, R_1]. At this point, EIPG2 realizes that the road ID R_1 appears twice in the probe. Therefore, EIPG2 checks the part of the probe enclosed by R_1, i.e., [R_1, R_1R_2, J_1, J_2, J_3, R_5R_6, J_4, J_5, J_1, R_2R_1, R_1], and checks whether a cross-link happened. This time, EIPG2 finds out that there is a cross-link, i.e., [R_1R_2, R_5R_6]. Afterwards, EIPG2 records the asymmetric link R_5R_6 as unroutable and sends the packet back to its sender, i.e., vehicle B.

Subsequently, EIPG2 employs the right-hand rule and forwards the packet to vehicle E and vehicle E forwards the packet to vehicle F. Vehicle F employs the right-hand rule and finds the vehicle H as the next possible forwarder. But, it checks the UR and finds the R_5R_6 recorded in it. Therefore, Vehicle F employs the right-hand rule to find the next possible forwarder, i.e., vehicle G and forwards the packet to it. The distance between vehicle G and the destination vehicle D is less than the distance between source vehicle S and the destination vehicle D. Thus, EIPG2 switches back to its greedy forwarding mode and sends the packet successfully to the destination vehicle D. The new route, i.e., $S \rightarrow A \rightarrow B \rightarrow E \rightarrow F \rightarrow G \rightarrow D$, is shown in Figure 4.17.

4.4.5 *Loop Detection*

One of the main objectives of this work is to decrease the number of the hops to reduce the latency. Figure 4.34 depicts the loop frequency, i.e., total number of the routing loops induced divided by the total number of the packets sent by the source ITS stations. My excessive simulations show that

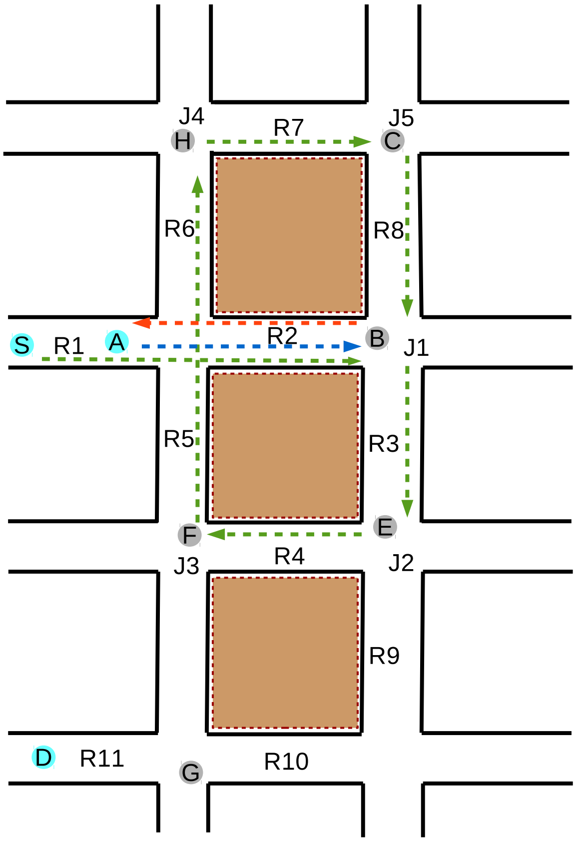


Figure 4.33: An example of perimeter forwarding mode in EIPG2

often packets get stuck in routing loops and go back and forth between a couple of vehicles. Therefore, it is required to recognize the possibility of loop construction and to react accordingly to increase the performance of geo-routing protocols in urban VANETs. In order to avoid redundant forwarding, LD approach of EIPG2 investigates if it has already forwarded the data packet before.

In *greedy* forwarding mode, it switches to *carry* forwarding mode as soon as it detects a loop. In *perimeter* forwarding mode, EIPG2 tries to find any possible route to get closer to the destination vehicle than the local optimum. During this process, a coordinator vehicle sends the data packet to different hands of the intersection one by one to find a possible route. Therefore, it might receive the same packet several times. Also, when the packet is forwarded back to the coordinator vehicle, it can be done by the same non-coordinator vehicle that has received this data packet from the coordinator vehicle, i.e., it has received this data packet for the second time. Thus, in *perimeter* forwarding mode, LD approach of EIPG2 does not interrupt the coordinator vehicles. Also, LD approach of EIPG2 forces the non-coordinator vehicle to switch to *carry* forwarding mode only if it has received a data packet for the third time. This way, EIPG2 does not disrupt the normal process of perimeter forwarding.

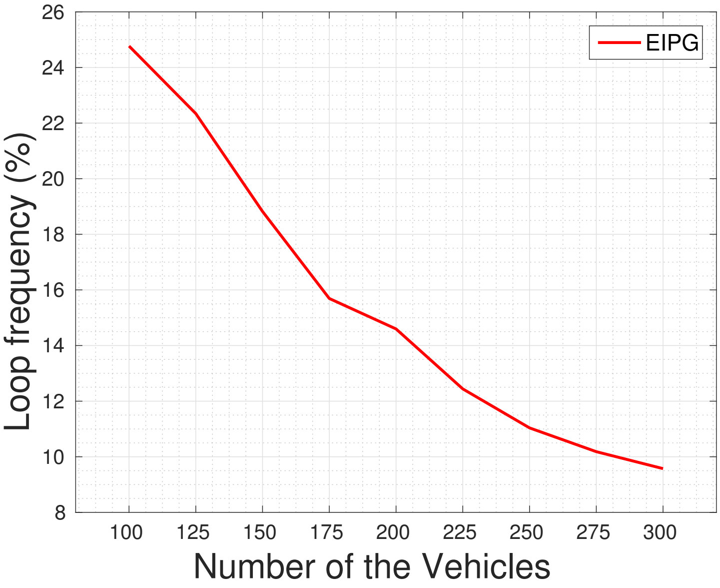


Figure 4.34: Loop frequency

4.4.6 *Simulation Evaluations*

In this section, the performance of EIPG2 is evaluated, comparing it with GeoCross, EIPG, basic GyTAR and GyTAR+CaF. The performance of these geo-routing protocols has been evaluated concerning reachability, scalability and latency:

- *Reachability*: PDR is the number of packets received by the destination divided by the number of packets originated by the source.
- *Latency*: Transmission latency is defined as the "time interval between the time when a V2X message is delivered from the facilities layer to the network and transport layer at the sending ITS station and the time when a V2X message is delivered from the network and transport layer to the facilities layer at the receiving ITS station [16]". In other words, end-to-end latency is the time taken for a packet to be transmitted from the source to the destination.
- *Scalability*: Routing overhead is the average number of routing packets received by all the nodes.

Also, the performance of the CLD and LD mechanisms of EIPG2 is evaluated in the following.

4.4.6.1 *Assessment of Cross-link Detection*

Lee et al. [1] has investigated the cross-link frequency in the range of 25 up to 100 vehicles in a Manhattan Grid of 1500×1500 m^2. They have shown that increasing the traffic density, the cross-link frequency has the overall trend of being downward but its presence does not disappear. I have investigated the cross-link frequency in the range of 100 up to 300 vehicles in my scenario and as expected, it does not happen that often in this range of traffic density. Figure 4.35 shows the cross-link frequency in EIPG. For instance, in case of 100 vehicles, 1.8% of packets in EIPG face a cross-link. Therefore, after adapting this cross-link detection and avoidance approach to EIPG2, major improvements are not expected, which is shown in Figure 4.36.

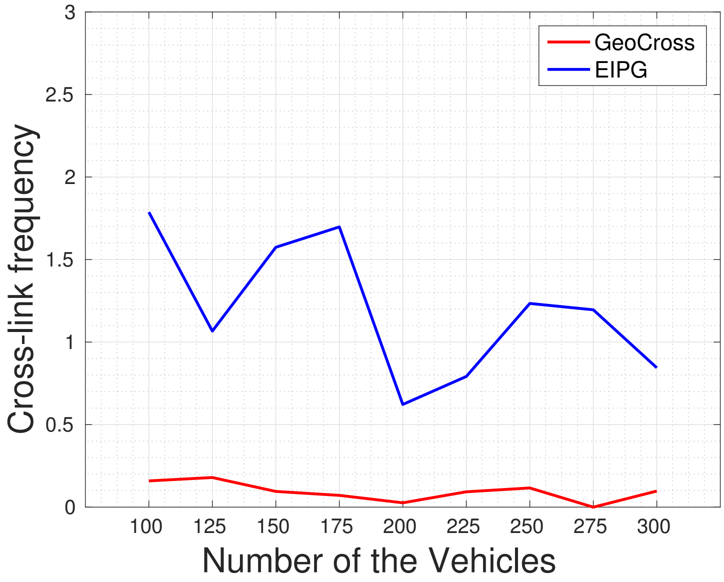


Figure 4.35: Cross-link frequency in EIPG and GeoCross

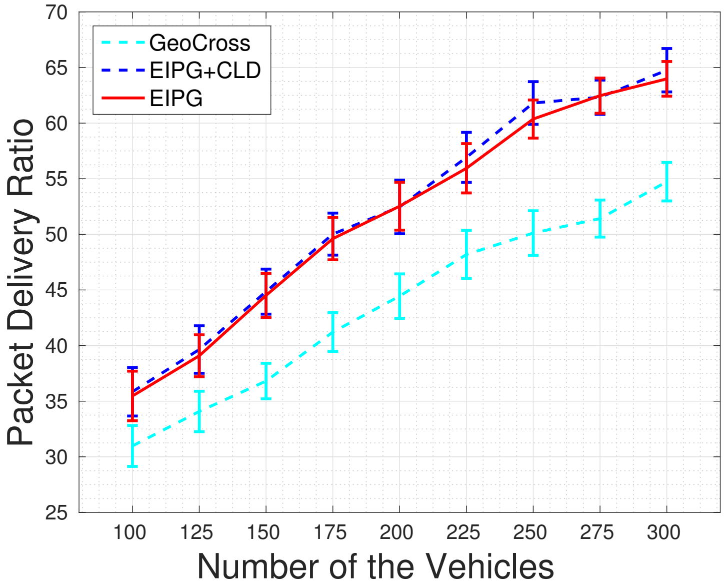


Figure 4.36: PDR for different number of vehicles

4.4.6.2 *Assessment of Loop Detection*

As depicted in Figure 4.37, my excessive simulations show that often packets get stuck in loops and go back and forth between a couple of vehicles.

Higher number of the loops in EIPG is because of not finding a coordinator vehicle during perimeter and greedy forwarding to change the path from a road segment to its crossing road segments at an intersection. EIPG2 applies the aforementioned LD mechanism in order to avoid redundant forwarding. Figure 4.37 shows the loop frequency in EIPG and EIPG2, corresponding to different number of vehicles in urban VANETs. With the help of this LD approach, loop frequency in EIPG2 is in average 10% decreased comparing to EIPG. It is good to mention that not all the loops are inevitable, e.g., in perimeter forwarding mode. And that explains the remaining average 5% of loops in this case in EIPG2.

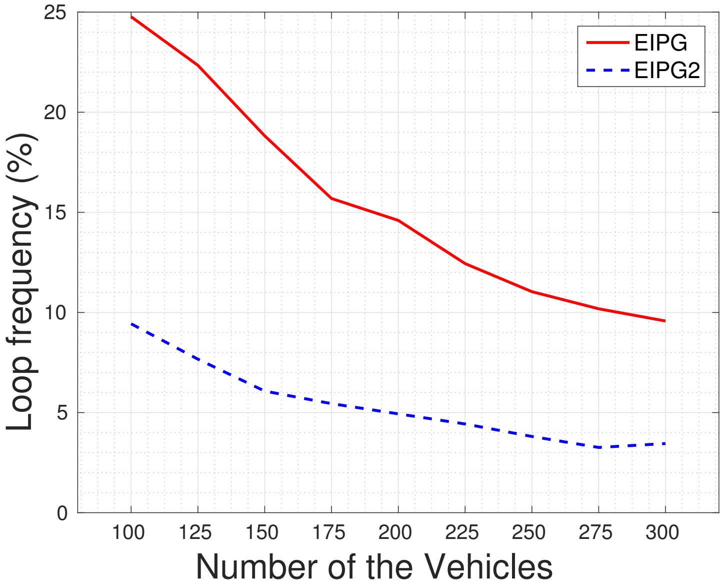


Figure 4.37: Loop frequency in EIPG and EIPG2

4.4.6.3 *Assessment of Reachability*

In order to assess the reachability of these protocols, PDR is evaluated, which is defined as the number of packets received by the destination divided by the number of packets originated by the source.

Figure 4.38 shows the packet delivery ratio of EIPG, EIPG2, basic version of GyTAR, i.e. without CaF, and GyTAR with CaF, corresponding to different number of vehicles in urban VANETs. In terms of PDR, EIPG2 performs in average about 1.5 times better than EIPG, which is a great improvement. GyTAR+CaF performs in average about 1.8 times better than basic GyTAR. Also, EIPG2 works in average 2% better than GyTAR+CaF.

4.4.6.4 *Assessment of Latency*

In order to assess the latency of these geo-routing protocols, end-to-end delay is evaluated, which is defined as the time taken for a packet to be transmitted from the source to the destination.

Figure 4.39 shows the CDF of end-to-end latency for EIPG, EIPG2, basic

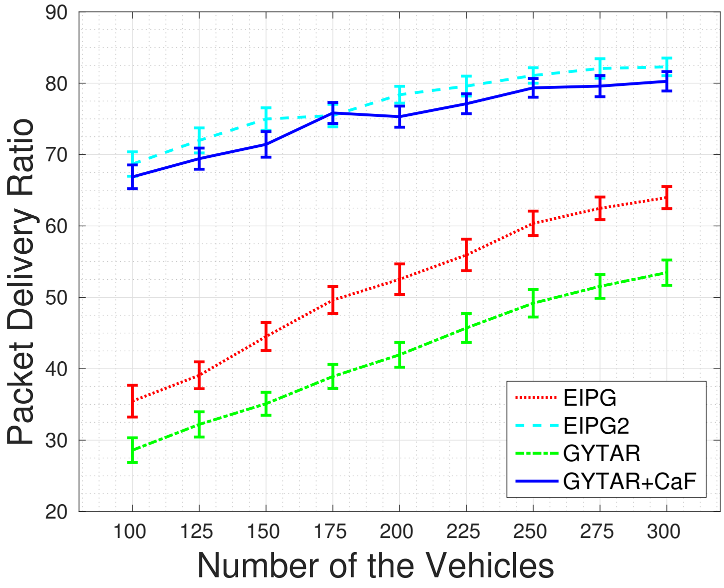


Figure 4.38: PDR for different number of vehicles

GyTAR and GyTAR+CaF, corresponding to an aggregation of all the different number of vehicles. This figure gives us the probability that a packet can be received by any destination in urban VANETs within a specific time. In order to have more meaningful graphs, all the packets which are not delivered, are included in Figure 4.39 considering an end-to-end latency of infinity. The probability that a packet can be received by any destination in less than 40 ms, employing EIPG is about 49% and employing EIPG2 is about 50%. In case of basic GyTAR, it is about 40% and for GyTAR+CaF it is about 36%. It is interesting to note that EIPG2 shows about 1% improvement comparing to EIPG, but GyTAR+CaF shows about 4% deterioration in comparison to the basic GyTAR. Also, EIPG2 shows up to 14% improvement in comparison to GyTAR+CaF, in terms of end-to-end latency.

4.4.6.5 *Assessment of Scalability*

In order to assess the scalability of these geo-routing protocols, routing overhead is evaluated, which is defined as the average number of routing packets received by all the nodes. In other words, how many extra packets for routing purposes should be sent in an urban VANET so that a data packet can be routed from the source vehicle to the destination vehicle.

Figure 4.40 shows the routing overhead of basic GyTAR, GyTAR+CaF, EIPG and EIPG2, corresponding to different number of vehicles in urban VANETs. Figure 4.40 illustrates the required average number of the routing packets for the successful delivery of a packet to any destination. In average, EIPG needs to send about 34 packets for routing purposes to deliver a packet from any source to any destination. EIPG2 accomplishes the same routing task sending about 11 packets for routing purposes to deliver a packet from any source to any destination. Therefore, in terms of routing overhead, EIPG2 improves EIPG by about 67% in average, which is a great improvement.

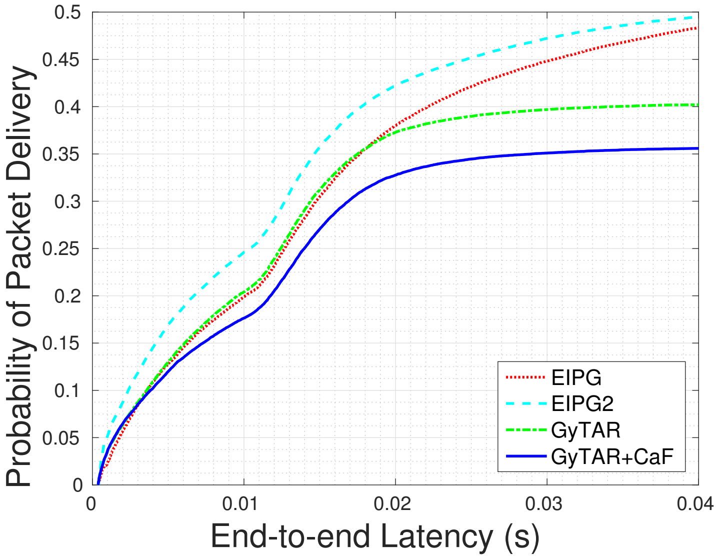


Figure 4.39: CDF of end-to-end latency, aggregating all packets

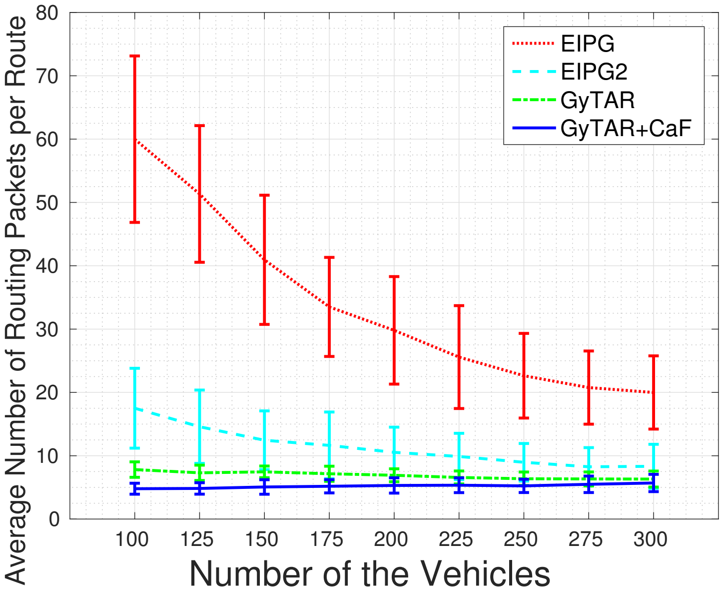


Figure 4.40: Routing overhead

Basic GyTAR needs to send about 7 packets for routing purposes to deliver a packet from any source to any destination. GyTAR+CaF accomplishes the same routing task sending about 5 packets for routing purposes to deliver a packet from any source to any destination. Therefore, in terms of routing overhead, GyTAR+CaF improves GyTAR by about 28% in average. GyTAR+CaF depends too much on CaF, in comparison with EIPG2. That's why it needs in average 6 routing packets less than EIPG2 to deliver a packet from any source to any destination. And for the same reason, as seen in Figure 4.39, in case of end-to-end latency, it works worse than basic GyTAR, EIPG and EIPG2.

My extensive simulations show that, EIPG2 performs in average about 1.5 times better than EIPG in terms of reachability. In terms of scalability, EIPG2 improves EIPG by about 67% in average. Also, in terms of latency, EIPG2 improves EIPG by about 2% in average. Moreover, EIPG2 outperforms GyTAR+CaF in terms of reachability and latency.

5

GEOBROADCAST

According to ETSI TS 102 636-2 [7], communication between different communication endpoints may be realized by geoBroadcast, i.e., communication from a single ITS station to all the ITS stations within a geographical target area. Based on ETSI TS 102 636-2 [7], geographical areas shall be specified by geometric shapes, e.g., circular area, rectangular area, and ellipsoidal area. ETSI TC ITS defines the specifications of DEN basic service in ETSI EN 302 637-3 [26], that supports the RHW applications by constructing, managing and processing the DENMs. Based on Akamatsu et al. [48], performing geoBroadcast, the broadcast storm problem arises that needs the broadcast suppression methods to be tackled. Therefore, designing an efficient multi-hop geoBroadcast protocol is very important to avoid the broadcast storm problem.[1]

The existing geoBroadcast routing protocols have some drawbacks that can be critical. Some of them suffer from the routing issues mentioned in Chapter 3, e.g., local optimum and partitioned networks. Others are optimized for a specific scenario, e.g., highway, and perform worse in urban VANETs. The contributions and research problems of the existing geoBroadcast routing protocols are discussed in more details in Section 3.1.

5.1 GEOBROADCAST DESTINATION AREA

ETSI EN 302 931 [2] specifies a method for location referencing. It defines geographical areas with the aid of geometric shapes, i.e., circle, rectangle and ellipse. Moreover, it introduces a function to determine the geospatial relation of a point to the geographical area, i.e., to determine if the point is located at the center, inside, at the border, or outside of a geographical area.

According to ETSI EN 302 931 [2], circular, rectangular, and elliptical areas are defined. Figure 5.1 gives an overview of the geometric shapes defining the geographical areas. Here, a represents the distance between the center point and the short side of a rectangle or the length of the short semi-axis of an ellipse. b represents the distance between the center point and the long side of a rectangle or the length of the long semi-axis of an ellipse. r is the radius of a circle. x represents the abscissa of a Cartesian coordination system with the origin in the center of the geographical area and parallel to the long side of a geometric shape. y represents the ordinate of a Cartesian coordination system with the origin in the center of the geographical area and parallel to the short side of a geometric shape. θ is the azimuth angle of the long side of a rectangle or the long semi-axis of an ellipse. And φ is

---

[1] Parts of this chapter have been published in Garrosi et al. [43].

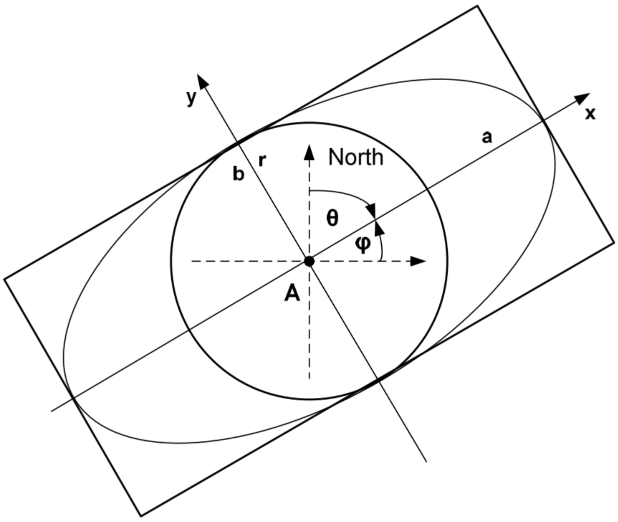


Figure 5.1: Geometric shapes defining geographical areas [2]

the zenith angle of the long side of a rectangle or the long semi-axis of an ellipse.

5.2 OUTSTANDING GEOBROADCAST ROUTING PROTOCOLS

In order to deliver a message to all the vehicles located in a geo-region, the most trivial way is to simply flood the message in the whole VANET. In simple flooding, every new arriving packet is rebroadcasted through the VANET. As mentioned in Section 2.1.2.5, the DENM reception management is a sub-function of the DEN basic service, that updates the receiving ITS station's message table and discards the received invalid DENMs. This way, the packet will be eventually received by all the reachable vehicle ITS stations of the VANET.

This type of algorithms cause unnecessary overhead in the network because of the high number of rebroadcasts. Therefore, more controlled geoBroadcast protocols are required to deal with the broadcast storm problem. The high number of rebroadcasts can be controlled in terms of time duration, geographical area and/or by allowing only a subset of vehicle ITS stations to rebroadcast. As mentioned in Section 2.1.2.5, the DENM KAF is an optional sub-function of the DEN basic service, that implements the DENM protocol operation of the forwarding ITS station, e.g., storing a received DENM as long as it is still valid, i.e., validity duration is not expired, and forwarding the DENM when applicable. There are more complex geoBroadcast routing protocols available for urban VANETs that tried to limit the number of rebroadcasts in terms of the geographical area and/or by allowing only a subset of vehicle ITS stations to rebroadcast.

Akamatsu et al. [48] proposed UGAD to suppress the unnecessary retransmissions by controlling the packet forwarding. It defines the forwarding zone as a region that is closer to the geo-region than the sender vehicle. Each vehicle ITS station compares its geographical position with geographical position of the sender vehicle and the destination area that is included in the DENM. This way, vehicle ITS stations check if they are located within the forwarding zone. If the vehicle ITS station finds itself within the forwarding zone, it calculates its own back-off time. The way vehicle ITS stations calculate their waiting time, depends on the forwarding mode that they employ. After calculating the back-off time, the vehicle ITS station delays its rebroadcast for the duration of the back-off time. The receiver vehicles are not required to rebroadcast the packet, if they receive duplicated packets from other vehicles, before the back-off timer expires.

UGAD considers two different forwarding modes, i.e., the *greedy forwarding* mode and the *intersection-based* forwarding mode. Vehicles calculate their own back-off time based on the transmission range, distance to the sender, and the forwarding mode, when they are located within the forwarding zone or the geo-region. The greedy forwarding mode should be employed to reduce the number of broadcasts, whenever rebroadcasting at intersections is not needed. On the other hand, whenever the reachability of the packets has more priority over reducing the number of broadcasts, the intersection-based forwarding mode should be employed.

When the vehicle ITS station i receives a packet from vehicle j, if it finds itself in the greedy forwarding mode, then it calculates the back-off time T_{GF_i} as follows:

$$T_{GF\_i} = T_{max\_R} \cdot \left(\frac{R - d_{ij}}{R}\right) \tag{5.1}$$

Here, d_{ij} is the distance between vehicle i and vehicle j. T_{max_R} is the maximum waiting time for the vehicle ITS stations. And, R is the transmission range of the vehicle ITS stations that is considered to be equal for all the vehicles. Therefore, the performance of UGAD depends on a very accurate estimate of the actual transmission range in order to calculate the back-off time.

When the vehicle ITS station i receives a packet from vehicle j, if it finds itself in the intersection-based forwarding mode, then it calculates the back-off time T_{IF_i} as follows:

$$T_{IF\_i} = \begin{cases} T_{max\_I} \cdot \left(\frac{R-d_{ij}}{R}\right) & \text{intersection} \\ T_{max\_I} + T_{max\_R} \cdot \left(\frac{R-d_{ij}}{R}\right) & \text{otherwise} \end{cases} \tag{5.2}$$

Here, T_{max_I} is the maximum back-off time for the vehicle ITS stations locating at intersections. And, T_{max_R} is the maximum back-off time for the vehicle ITS stations locating on the road segments.

Vehicles at intersections rebroadcast faster considering preferential delay values over in-road vehicles. An example of packet forwarding in the intersection-based mode is illustrated in Figure 5.2. The source vehicle S broadcasts a packet. All the other vehicles receive the packet and calculate their back-off time considering their distance to the sender vehicle S. Moreover, vehicles B, C and F are at intersections, thus they calculate a smaller back-off time. As a result, vehicle B with the smallest back-off time

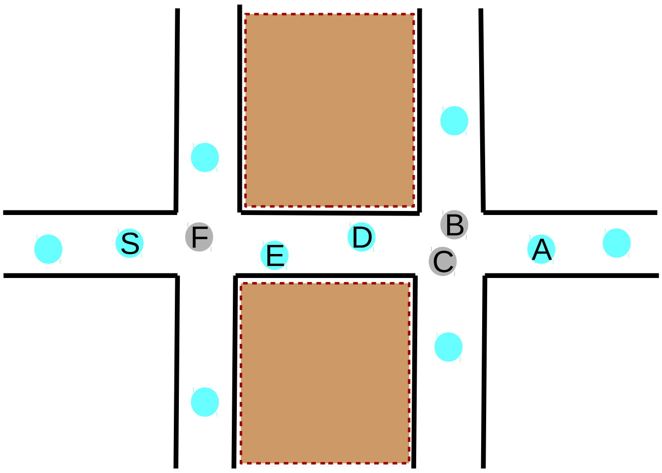


Figure 5.2: Intersection-based packet forwarding in UGAD

rebroadcasts the packet first. Vehicles A, C, D, E and F receive the packet for the second time from B and terminate their rebroadcasting processes.

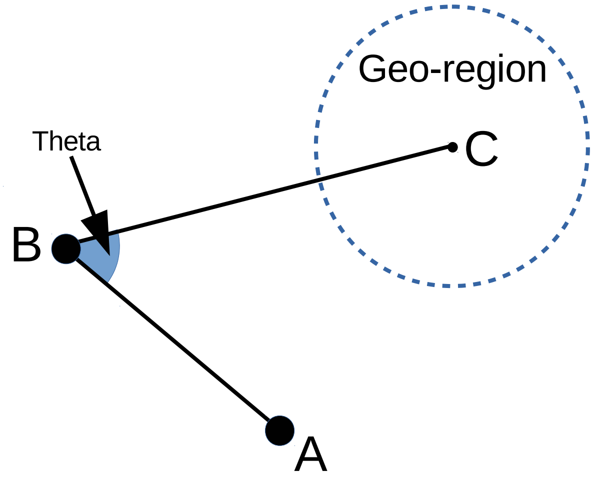


Figure 5.3: Calculation of the angle θ

The decision to perform the intersection-based forwarding mode or the greedy forwarding mode is made, based on an angle calculated from the positions of receiver, sender and the geo-region. The vehicle ITS stations should employ the greedy forwarding mode, if they have a one-dimensional path to the geo-region. Otherwise, the vehicle ITS stations should employ the intersection-based forwarding mode. To decide if vehicle ITS station has a one-dimensional path to the geo-region or not, UGAD calculates the angle θ from the positions of receiver, sender and the geo-region. This is shown in Figure 5.3. Vehicle B receives a packet from Vehicle A and calculates the angle θ between $\overrightarrow{BA}$ and $\overrightarrow{BC}$ as follows:

$$\theta = \arccos\left(\frac{\overrightarrow{BA} \cdot \overrightarrow{BC}}{|\overrightarrow{BA}| \cdot |\overrightarrow{BC}|}\right) \tag{5.3}$$

UGAD defines a threshold α in order to select the forwarding mode, comparing $\cos(\theta)$ with α. Whenever the vehicle ITS station is located within the geo-region, it employ the intersection-based mode. Otherwise, the vehicle ITS station calculates the angle θ and compares $\cos(\theta)$ with the threshold α. The vehicle ITS station employs the intersection-based forwarding mode, if the absolute value of $\cos(\theta)$ is less than the threshold α. Otherwise, the vehicle ITS station employs the greedy forwarding mode.

5.3 UNICAST-ASSISTED GEOBROADCAST

In order to deliver a message to all the vehicles located in a geo-region, the most trivial way is to simply flood the message in the whole network. But simple flooding causes the worst case of the broadcast storm problem. To tackle this problem, some protocols, e.g., GRUV [51] and UGAD [48] consider a forwarding zone to flood the message not in the whole network but to flood it in a smaller part of the network. In a similar way, other protocols, e.g., GRUV [51] and GeoMob [54] consider a path to the geo-region to forward the packet through a geographically smaller area. On the other hand, some protocols, e.g., DRG [50] and UGAD [48] apply a back-off timer to avoid the unnecessary retransmission of the message. Upon receiving a duplicated packet during the back-off, the vehicle breaks the back-off and cancels the scheduled rebroadcast to suppress the redundant rebroadcasts. Similarly, other protocols, e.g., T-TSG [52], GeoMob [54] and CAG [53] select the forwarding vehicle considering the microscopic mobility, traffic light situation and the position of vehicles. But of course, limiting the forwarding zone and retransmitters yield in an decrease of reachability, i.e., fewer vehicles located in the geo-region will receive the message. Some protocols, e.g., DRG [50] employ retransmission to compensate for the decreased reachability.

The idea of this work is to deliver the data packet from the source vehicle to the vehicles inside the geo-region employing UAG in order to tackle the problem of broadcast storm. UAG can be divided into two main parts: first the message is unicasted to the target positions within the geo-region with the aid of EIPG2 proposed in Chapter 4. As soon as a vehicle within the geo-region receives a unicast packet, it broadcasts the packet. In other words, all the vehicles located in the geo-region that forward the unicast packet towards any target position will broadcast the packet. Afterwards, some of the vehicles located in the geo-region are selected to rebroadcast the packet. This way we can tackle the problem of broadcast storm.

In order to compensate for the decreased reachability, I propose to unicast copies of the message to different target positions within the geo-region. Care should be taken that these copies should take different paths towards the destination in order to increase the probability of reaching the geo-region. To make sure that different packets take different paths, I make a slight change in EIPG2 which will be explained in Section 5.3.4. Moreover, we can define a forwarding zone in which vehicles are allowed to rebroadcast the message. I propose to define a forwarding zone similar to the geo-region but with an increased geographic size applying the Forwarding-zone Breathing Coefficient (FBC). Forwarding-zone Breathing will be explained in Section 5.3.2.

We can summarize the proposed UAG into the following steps:

- Select the intersection-based or road-based approach, choose the target positions within the geo-region, and calculate the FBC,
- Make copies of the message and unicast each copy to a selected target position employing EIPG2 considering both right-hande and left-hand rules,
- Any vehicle within the geo-region that receives a unicast message will broadcast the message,
- Any vehicle within the geo-region that receives a broadcast message will only rebroadcast the message, if it is qualified to be a rebroadcaster.

5.3.1 *Target Position Selection*

Choosing the number of the targets and their positions, depend on the road topology in the geo-region and also on the unicast geo-routing protocol. Considering EIPG2, it is preferred to select a target position close to the center of the geo-region and moreover to select a couple of target positions close to the boundary of the geo-region but in different cardinal directions.

I have proposed two approaches: intersection-based and road-based. In the intersection-based approach, EIPG2 chooses the closest intersection to the center of the geo-region as the central target position. Moreover, in the intersection-based approach, EIPG2 chooses some of the closest intersections to the boundary of the geo-region, considering them as the gates to enter the geo-region. Figure 5.4 shows the target positions selected within a circular geo-region employing the intersection-based approach. In the road-based approach, EIPG2 chooses the middle points of some of the roads intersecting with the boundary of the geo-region, considering them as the gates to enter the geo-region. Figure 5.5 shows the target positions selected within a circular geo-region employing the road-based approach.

5.3.2 *Forwarding-zone Breathing*

As mentioned before, some of the geoBroadcast routing protocols, e.g. UGAD, define a forwarding zone in which nodes are allowed to forward the message. Obviously, increasing the size of the forwarding zone can increase the reachability of the routing protocols. But on the other side, it causes scalability issues as well. Defining a forwarding zone can be static, e.g. a

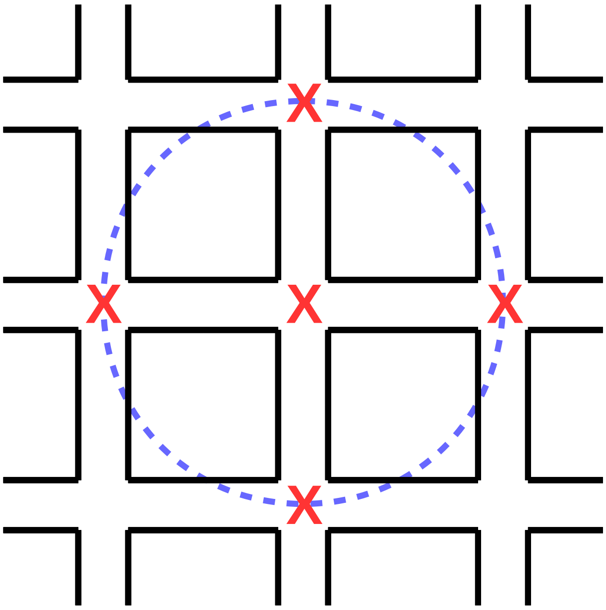


Figure 5.4: Intersection-based approach

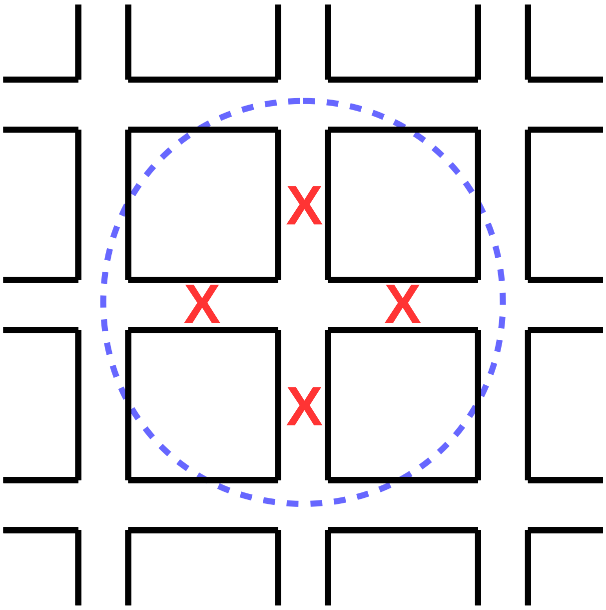


Figure 5.5: Road-based approach

region that is closer to the geo-region than the sender vehicle as in UGAD. It can also be defined dynamically, considering different input information, e.g., the road topology in the geo-region and the mobility pattern in the geo-region.

In this work, I propose to define a forwarding zone similar to the geo-region but with an increased geographic size. The size of the circular, rectangular, or elliptical geo-region is increased, multiplying the semi-axes or the radius of the geo-region with a coefficient, i.e., FBC. The idea is to calculate this coefficient based on the road topology in the geo-region and its neighborhood in order to include the intersections and streets that are not located inside the geo-region but connect different parts of the geo-region, e.g., intersections I_1 to I_4 and road segments R_1 to R_8 in Figure 5.6. This way we can find a trade-off between increasing the reachability and increasing the overhead in VANETs. Figure 5.6 depicts an example of forwarding-zone breathing for a circular geo-region. In this case, the FBC is equal to the distance between the center of the geo-region and the closest intersection outside of the geo-region, e.g., I_1, divided by the radius of the circular geo-region.

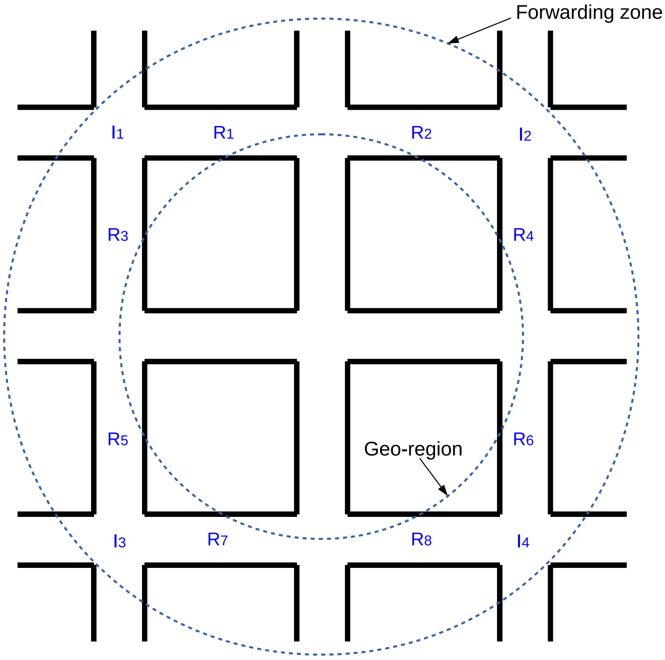


Figure 5.6: Forwarding-zone breathing

5.3.3 *EIPG2*

As mentioned before, in this work, a modified version of EIPG2 that is proposed in Chapter 4 is considered as the unicast geo-routing protocol. It has three forwarding modes: *greedy*, *perimeter* and *carry* forwarding mode. Having the *perimeter* forwarding mode, EIPG2 can tackle the local optimum problem. Also, employing the *carry* forwarding mode, EIPG2 can tackle the patitioned network problem. In order to avoid the cross-link problem, it adopts the cross-link detection mechanism of GeoCross [1]. Also, EIPG2 has its own LD and CaF approaches.

In *greedy* forwarding mode, the sender vehicle searches within its coordinator neighbors to find the coordinator neighbor with the maximum progress towards the destination vehicle and forwards the packet to it. If the source/forwarder vehicle has no such coordinator vehicle, then it searches within its non-coordinator neighbors to find the non-coordinator neighbor with the maximum progress towards the destination vehicle and forwards the packet to it. If the source/forwarder vehicle has also no such non-coordinator neighbor, then it switches to *carry* forwarding mode if it is a non-coordinator vehicle, and it switches to *perimeter* forwarding mode if it is a coordinator vehicle.

In *perimeter* forwarding mode, first the source/forwarder vehicle checks if it is closer to the destination than the point at which this packet hit the local optimum. If yes, then it switches back to *greedy* forwarding mode. If no, then it tries to find the next vehicle based on the preferential intersection-based perimeter forwarding, in which coordinator neighbors have preference over non-coordinator neighbors. If it can not find a suitable neighbor vehicle employing preferential intersection-based perimeter forwarding, then it switches to *carry* forwarding mode.

In *carry* forwarding mode, first the source/forwarder vehicle checks if there is any neighbor vehicle closer than itself to the destination vehicle. If yes, then it forwards the packet to this vehicle. If no, then it tries to find the closest vehicle around its intersection driving towards the intersection and forwards the packet to this vehicle. This vehicle queues the packet and waits until it becomes a coordinator and then continues forwarding the packet in *greedy* forwarding mode. If there is no vehicle close to and driving to this intersection, it switches to *perimeter* forwarding mode.

5.3.4 *Perimeter Forwarding with Left-hand Rule*

Karp et al. [99] has introduced GPSR, in which each vehicle starts the routing process in the so called greedy mode. It means that each vehicle tries to find a neighbor vehicle which is geographically closer to the destination and forwards the packet to it. But, because the greedy forwarding only employs the local information, it is possible that a packet reaches a local optimum, in which a vehicle can not find any neighbor vehicle that is closer to the destination. It can happen as a result of the non-uniform distribution of vehicles, limited communication range, and having buildings as obstacles in urban areas. At this point, GPSR employs a repair strategy to forward the packet to a vehicle which is closer to the destination. After the packet arrives at a vehicle which is closer to the destination than the point at which it hit the local optimum, it switches back to the greedy mode.

Many recovery algorithms have been suggested to solve this issue. GPSR employs the right-hand rule in the so called perimeter mode. Lochert et al. [56] have realized that the streets and intersections form a natural planar graph. GPCR calls the vehicles located at intersections *coordinator*, and the rest of vehicles *non-coordinator*. GPCR uses a repair strategy, in which coordinator vehicles use the right-hand rule to choose the street that is the next one counterclockwise from the street from which the packet has arrived. In

Chapter 4 I proposed EIPG, that employs a new intersection-based perimeter forwarding in order to avoid the problem of WSE. Also in EIPG, coordinator vehicles use the right-hand rule to choose the street that is the next one counterclockwise from the street from which the packet has arrived. I also proposed EIPG2 in Chapter 4, that employs a new preferential intersection-based perimeter forwarding in order to avoid the problem of WSE. Also in EIPG2, coordinator vehicles use the right-hand rule to choose the street that is the next one counterclockwise from the street from which the packet has arrived.

Simulations show that, it is possible that the coordinator vehicles either forward the packet to the next street counterclockwise from the street from which the packet has arrived, or clockwise from the street from which the packet has arrived. Thus, the coordinator vehicles can use the right-hand rule or the left-hand rule to choose the next street to forward the packet. In fact, for any local optimum, the vehicles should choose left or right-hand rule and should not switch between them until the packet arrives to a point that is closer to the destination than the local optimum. For the next local optimum, again the vehicles can choose between left or right-hand rule. Therefore, for each local optimum, vehicles can decide if employing the right-hand rule is more probable to get the packet closer to the destination vehicle or employing the left-hand rule. This can be done considering different parameters, e.g., position of destination, traffic density information and etc.

Considering unicasting several copies of the data packet to the geo-region, it is preferred that these copies take different routes towards the geo-region. To do so, some vehicles should employ the right-hand rule and others should employ the left-hand rule to take different paths towards the destination. Therefore, I propose to check whether the next street counterclockwise or clockwise is closer to the destination target position, and based on that, the coordinator vehicles can decide to employ the right-hand rule or the left-hand rule to perform the perimeter forwarding. Having the LDM available, the coordinator vehicles can acquire the intersection's position. And as a result, they can find out on which side of the intersection are the sender vehicle and destination target position located.

If the sender vehicle is located on the West of the intersection, then the coordinator vehicle employs the left-hand rule if the destination is located on the North of the intersection. If the destination is located on the South of the intersection, then the coordinator vehicle employs the right-hand rule. If the sender vehicle is located on the East of the intersection, then the coordinator vehicle employs the right-hand rule if the destination is located on the North of the intersection. If the destination is located on the South of the intersection, then the coordinator vehicle employs the left-hand rule.

As shown in Figure 5.7, for instance imagine that vehicle V_1 forwards a data packet to the coordinator vehicle V_3 in perimeter forwarding mode and vehicle V_3 needs to forward it also in perimeter forwarding mode. If the destination vehicle is located on the South of the intersection, then the coordinator vehicle V_3 employs the right-hand rule and forwards the data

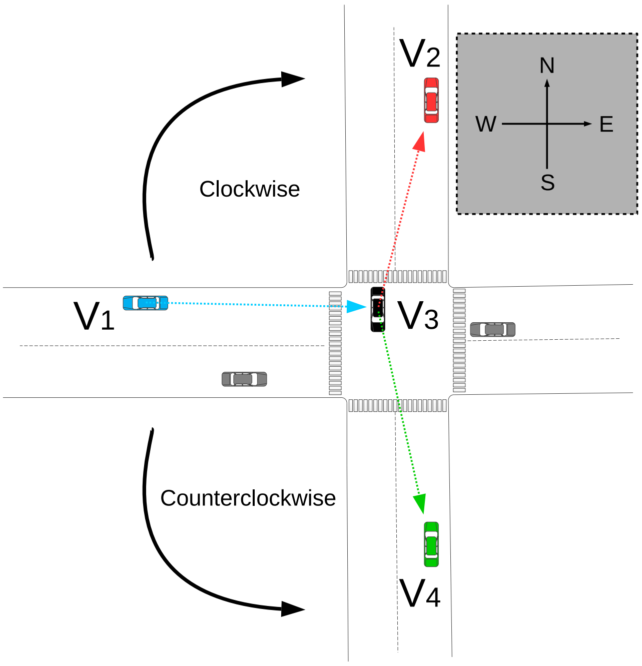


Figure 5.7: Left-hand rule or right-hand rule

packet to vehicle V_4. But, if the destination vehicle is located on the North of the intersection, then the coordinator vehicle V_3 employs the left-hand rule and forwards the data packet to vehicle V_2.

5.3.5 *Broadcaster and Rebroadcaster Selection*

Choosing the broadcasters depends on the selection of the target positions and the road topology inside the geo-region. In order to avoid the problem of broadcast storm, I propose that only the vehicles inside the geo-region that receive the unicast packet and forwards it towards any of the target positions, broadcast the packet. Also, only the coordinator vehicles within the geo-region or forwarding-zone, rebroadcast the packet after receiving a new broadcast packet. Coordinator vehicles are the vehicles located on the intersections.

5.4 SIMULATION EVALUATIONS

I have implemented the simple flooding and UGAD to evaluate them. After validating my implementations, the performance of the UAG with EIPG2 in comparison with UGAD has been evaluated concerning reachability and scalability. To assess reachability, PDR is evaluated, which is defined as the number of the vehicles that received the data packet and are located within the geo-region, divided by the total number of the vehicles locating within the geo-region. To assess scalability, routing overhead is evaluated, which is defined as the average number of the broadcasted packets per each packet delivered, i.e., how many times in average we need to rebroadcast a packet to deliver it to a vehicle located within the geo-region.

In order to evaluate the geoBroadcast protocols, the simulation environment and parameters as explained in Chapter 4, are considered. To be able to evaluate these geoBroadcast protocols in different traffic densities, I have considered different number of vehicles starting from 100 vehicles and increasing it with an increment step of 25 vehicles up to 300 vehicles. All my simulation results are based on 49 runs per each traffic density. For each simulation run, a random network of vehicles with random trips and mobility routes have been generated employing SUMO to simulate the road traffic as close as possible to the reality. Again, in each simulation run, every second, 1 random source chooses a random circular geo-region and sends a DENM packet to all the vehicles located within this geo-region. Finally, the results have been averaged over 441 runs of simulations with random seeds.

According to ETSI EN 302 637-2 [20], the CAM generation frequency should be between 1 Hz and 10 Hz. In order to have the maximum freshness of vehicles' information, CAMs are sent every 100 ms containing the position, time-stamp, speed, and road-ID of the originating vehicle.

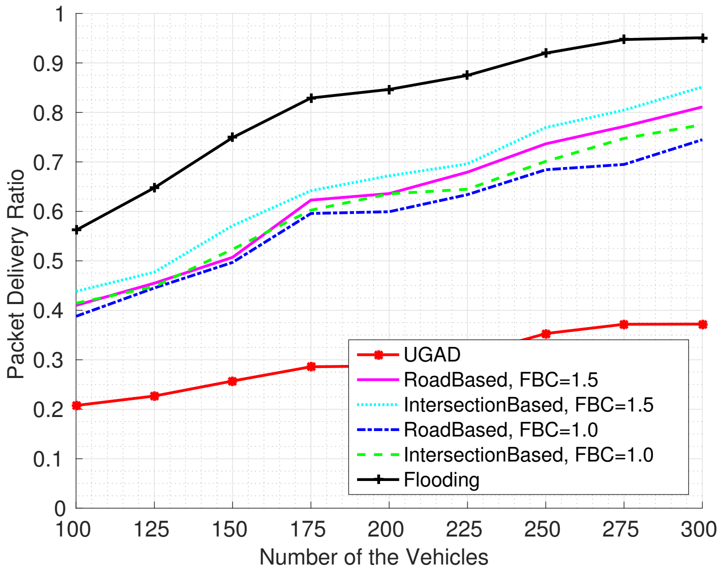


Figure 5.8: Reachability

Figure 5.8 shows the results of the simulations in term of reachability. Simple flooding, UGAD, intersection-based UAG and road-based UAG, with FBC=1.0 and FBC=1.5 are compared with each other. As expected, simple flooding shows the best performance in terms of reachability at a cost of lower scalability, i.e., high overhead in the network (see Figure 5.9). After simple flooding, UAG shows a good performance that is better than UGAD's performance. UAG with FBC=1.5 performs better than UAG with FBC=1.0, as a result of the forwarding-zone breathing and at a cost of a slightly increased overhead (see Figure 5.9). Also, intersection-based UAG shows a slightly better performance in comparison with road-based UAG. Because in this scenario, EIPG2 chose five target positions in case of intersection-based approach but four target positions in case of road-based approach. Also, the distance between two target positions in the intersection-based approach is bigger, which makes it more probable that the two copies that are sent to two different target positions take different routes towards the geo-region. And it increases the probability of reaching the geo-region.

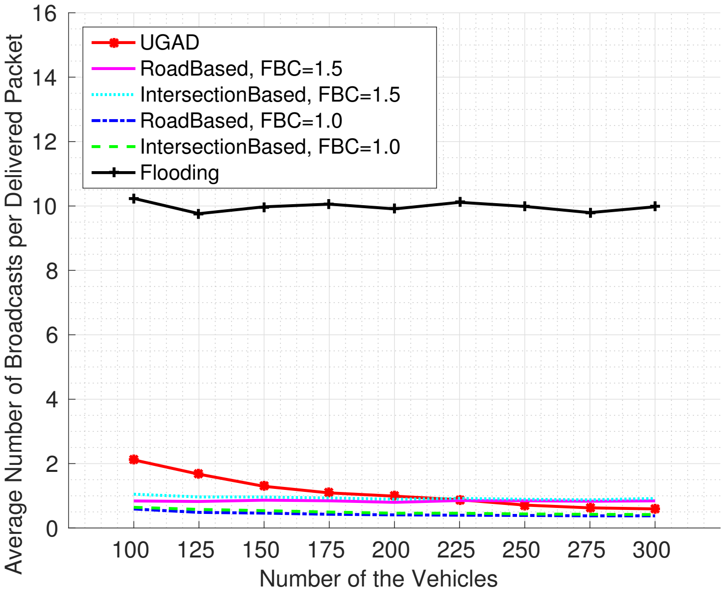


Figure 5.9: Scalability

Figure 5.9 shows the results of the simulations in term of scalability. Simple flooding, UGAD, intersection-based UAG and road-based UAG, with FBC=1.0 and FBC=1.5 are compared with each other. As expected, simple flooding shows the worst performance in terms of scalability. After simple flooding, UGAD shows an acceptable performance that is better than simple flooding's performance. Intersection-based UAG and road-based UAG show the best performance in case of scalability. The performance of UAG with FBC=1.5 is slightly worse than UAG with FBC=1.0, as a result of the forwarding-zone breathing. Also, intersection-based UAG shows a slightly better performance in comparison with road-based UAG. Because in this scenario, EIPG2 chose four target positions in case of road-based approach but five target positions in case of intersection-based approach.

My extensive simulations show that, UAG outperforms UGAD by about 30% in average, in terms of reachability. Also, in terms of scalability, UAG outperforms UGAD by about 1% in average.

6

CONCLUSION AND OUTLOOK

As reported by the ETSI, road traffic is constantly increasing, that causes serious problems, e.g., congested roads, road-safety and environmental effects. Stand-alone driver assistance systems have several benefits, e.g., maintaining a safe speed and distance. These advantages can be boosted by means of cooperation between vehicles. The development of ITS aims to enable the V2X communications to reduce the number of the accidents and to provide the traffic management, road-safety and comfort applications.

A series of standards for ITS have been established in Europe, US and Japan. The ITS Info-communications Forum of ARIB promotes the R&D and standardization of communication technologies in order to the successful introduction of ITS in Japan. ASTM has developed DSRC standards for ITS in the United States. ETSI has developed C-ITS standards for ITS in Europe.

In VANETs, vehicles dynamically set up an ad-hoc network without any aid of infrastructure. Moreover, vehicles move fast and are constrained within the layout of the roads, which leads to frequent reception failure and network disconnections. Also, in urban scenarios, vehicles are facing the shadowing effects of buildings and are suffering from the fading phenomena. When, the source and destination of a data packet are located outside of each other's communication range, other vehicles in between should work as router, so that they can receive the packet from the source and relay it through the network towards the destination. Therefore, successful establishment of VANETs, depends on the routing protocols, which help vehicles to find short, robust and reliable routing paths to deliver the data packets.

VANET routing protocols inherit the problems of traditional routing protocols of MANETs. Moreover, because of the aforementioned unique characteristics of urban VANETs, they are facing new problems. Therefore, the research issues of VANET routing protocols should have been identified and appropriate solutions should have been introduced.

While topology-based and cluster-based routing approaches are not suitable for the networks with high mobility and frequent topology changes, geo-routing protocols rely on the geographic position information of vehicles. As a result of this simplicity, geo-routing protocols scale better in large networks.

ETSI has introduced geoNetworking as a family of network protocols that employ the geographical positions for addressing and transport of data packets in VANETs. GeoNetworking is appropriate for the networks with high mobility and frequent changes in the network topology. Moreover, it is flexible in supporting different applications and their heterogeneous requirements. ETSI has defined two main types of connection in VANETs; as geoUnicast, i.e., one ITS station sends a packet to one ITS station, and geoBroadcast, i.e., one ITS station sends a packet to all the ITS stations located in a geographical target area. ETSI has also introduced geoUnicast and geoBroadcast forwarding algorithms for geoNetworking. But these algorithms does not address the research issues of urban VANETs, e.g., local optimum, and are only applicable in highway scenarios. Therefore, the focus of this work was on the geo-routing protocols for urban VANETs.

In this work, I have studied the relevant ETSI standards regarding ITS and also studied the state of the art VANET routing protocols, especially the VANET geo-routing protocols. I have identified the research issues of geo-routing protocols and classified them based on their specifications, approaches and the research issues that they have addressed. Also, I have implemented the most outstanding VANET geo-routing protocols in order to evaluate them.

Afterwards, I have introduced the EIPG geoUnicast routing protocol and subsequently the EIPG2 geoUnicast routing protocol. EIPG addresses the problem of WSE and EIPG2 addresses the routing loop and partitioned networks problems. They show a significant improvement in comparison to their predecessors, in terms of PDR, end-to-end delay, and network overhead. Finally, I have introduced the UAG geoBroadcast routing protocol to address the problem of broadcast storming. UAG shows a better performance in terms of reachability and scalability, comparing to simple flooding and UGAD.

My methodology was to analyze and evaluate the VANET routing protocols based on simulations. For this purpose, the network simulator OMNeT++ and the road traffic simulator SUMO were coupled employing TRaCI. In my simulations, OMNeT++ worked based on ETSI documentations of ITS, and SUMO generated vehicles with random trips and random mobility routes.

In conclusion, I have developed and validated scalable and robust geoUnicast and geoBroadcast routing protocols for urban VANETs that address the most troublesome research issues in the field of urban VANET geo-routing protocols and show a better performance in terms of reachability, latency and scalability, comparing to other outstanding state of the art geo-routing protocols.

In future, it is interesting to analyze and evaluate the VANET geo-routing protocols having bigger areas of different cities and their maps. It is also compelling to consider the traffic density information and to predict the future position of vehicles in order to improve the performance of EIPG2 even more. It is likewise engaging to work on developing an algorithm to calculate the target positions and the FBC based on the LDM and the traffic density information.

Part II

APPENDICES

A

TABLES OF VANET ROUTING PROTOCOLS

Table A.1: Cluster-based VANET routing protocols

Protocol	**Year**	**Network**	**Type**	**Activity**	**Application**	**Position**	**Direction**	**Velocity**	**Map**	**Traffic**	**Prediction**	**Channel**	**# of hops**
DPP [107]	2005	VANET	Cluster-based	Reactive	DT	Yes	Yes	Yes	No	Yes	No	No	Single-hop
GVGrid [108]	2006	VANET	Cluster-based	Reactive	RSU	Yes	Yes	No	Yes	No	Yes	No	Multi-hop
OPERA [109]	2008	VANET	Cluster-based	Reactive	DT	Yes	Yes	Yes	No	No	No	No	Multi-hop
CBR [110]	2010	VANET	Cluster-based	Reactive	RSU	Yes	No	No	No	No	No	No	Single-hop
SRD [111]	2011	VANET	Cluster-based	Reactive	DT	Yes	Yes	No	No	No	No	No	Single-hop
VWCA [112]	2011	VANET	Cluster-based	Reactive	RSU	Yes	Yes	Yes	No	Yes	No	No	Multi-hop
MDDC [113]	2012	VANET	Cluster-based	Reactive	DI	Yes	Yes	Yes	Yes	No	Yes	No	Multi-hop
PassCAR [114]	2013	VANET	Cluster-based	Reactive	DI	Yes	No	Yes	No	Yes	No	Yes	Multi-hop
PMTR [115]	2013	VANET	Cluster-based	Reactive	RSU	Yes	No	No	Yes	Yes	No	No	Single-hop
AODV-PNT [116]	2014	VANET	Cluster-based	Reactive	RSU	Yes	Yes	Yes	No	No	Yes	No	Multi-hop

Table A.2: Topology-based VANET routing protocols

Protocol	**Year**	**Network**	**Type**	**Activity**	**Application**	**Position**	**Direction**	**Velocity**	**Map**	**Traffic**	**Prediction**	**Channel**	**# of hops**
PBR [117]	2007	VANET	Topology-based	Proactive	DI	No	Yes	Yes	No	No	Yes	No	Multi-hop
NDMR [118]	2008	VANET	Topology-based	Reactive	DI	No	No	No	No	No	No	No	Multi-hop
LEAPER [119]	2011	VANET	Topology-based	Reactive	DI	No	No	No	No	No	No	No	Multi-hop
AID [120]	2011	VANET	Topology-based	Reactive	DI	No	No	No	No	No	No	No	Multi-hop
OLSR [121]	2012	VANET	Topology-based	Proactive	DI	No	No	No	No	No	No	No	Multi-hop
EEDAHRP [122]	2012	VANET	Topology-based	Hybrid	DI	No	No	No	No	No	No	No	Multi-hop

Table A.3: Position-based VANET routing protocols

Protocol	Year	Network	Type	Activity	Application	Position	Direction	Velocity	Map	Traffic	Prediction	Channel	# of hops
GSR [84]	2003	VANET	Position-based	Reactive	DI	Yes	No	No	Yes	No	No	No	Multi-hop
A-STAR [55]	2004	VANET	Position-based	Reactive	DI	Yes	No	No	Yes	Yes	No	No	Multi-hop
MDDV [96]	2004	VANET	Position-based	Reactive	DT	Yes	Yes	No	Yes	Yes	No	No	Multi-hop
PRAODV [94]	2004	VANET	Position-based	Reactive	DI	Yes	No	Yes	No	No	Yes	No	Multi-hop
MoVe [57]	2005	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	No	No	No	No	Single-hop
GPCR [56]	2005	VANET	Position-based	Reactive	DI	Yes	No	No	Yes	No	No	No	Single-hop
STAR [58]	2005	VANET	Position-based	Reactive	DI	Yes	Yes	No	Yes	Yes	No	No	Single-hop
MORA [59]	2006	VANET	Position-based	Reactive	DI	Yes	Yes	No	No	No	No	No	Multi-hop
VADD [87]	2006	VANET	Position-based	Reactive	DT	Yes	Yes	Yes	Yes	Yes	Yes	No	Multi-hop
CAR [62]	2007	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	No	Yes	No	No	Multi-hop
GyTAR [61]	2007	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	Yes	Yes	Yes	No	Single-hop
GpsrJ+ [60]	2007	VANET	Position-based	Reactive	DI	Yes	Yes	No	Yes	No	Yes	No	Multi-hop
MOPR [63]	2007	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	Yes	No	Yes	No	Multi-hop
GeOpps [92]	2007	VANET	Position-based	Reactive	DT	Yes	No	Yes	Yes	No	No	No	Single-hop
SADV [89]	2007	VANET	Position-based	Reactive	DT	Yes	Yes	Yes	Yes	Yes	No	No	Single-hop
GRANT [64]	2008	VANET	Position-based	Reactive	DI	Yes	No	No	No	No	No	No	Multi-hop
REAR [85]	2008	VANET	Position-based	Reactive	DI	Yes	Yes	No	No	No	No	Yes	Multi-hop
ACAR [65]	2008	VANET	Position-based	Reactive	DT	Yes	No	No	Yes	Yes	No	No	Multi-hop
TO-GO [67]	2009	VANET	Position-based	Reactive	DI	Yes	Yes	No	Yes	No	Yes	No	Multi-hop
Fcar [68]	2009	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	No	No	No	No	Multi-hop
RBVT-R [66]	2009	VANET	Position-based	Hybrid	DI	Yes	Yes	No	Yes	Yes	No	No	Multi-hop
LAGAD [123]	2009	VANET	Position-based	Hybrid	RSU	Yes	No	Yes	No	No	No	Yes	Multi-hop
GeoCross [1]	2010	VANET	Position-based	Reactive	DI	Yes	No	No	Yes	No	No	No	Single-hop
GeoDTN+Nav [90]	2010	VANET	Position-based	Reactive	DT	Yes	Yes	No	Yes	No	No	No	Single-hop
JARR [69]	2010	VANET	Position-based	Reactive	DT	Yes	Yes	Yes	Yes	Yes	No	No	Single-hop
TrafRoute [124]	2010	VANET	Position-based	Reactive	RSU	Yes	No	No	Yes	Yes	No	No	Multi-hop
DIR [70]	2010	VANET	Position-based	Reactive	DI	Yes	No	Yes	Yes	Yes	No	No	Multi-hop
ARM [86]	2011	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	No	No	No	No	Single-hop
IDVR [71]	2011	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	Yes	Yes	No	No	Single-hop
SRR [91]	2011	VANET	Position-based	Reactive	DT	Yes	Yes	No	No	No	No	No	Single-hop
HLAR [72]	2012	VANET	Position-based	Reactive	DI	Yes	No	No	No	No	No	No	Multi-hop
RIVER [73]	2012	VANET	Position-based	Reactive	DI	Yes	No	No	Yes	Yes	No	No	Single-hop
ROAMER [125]	2012	VANET	Position-based	Reactive	RSU	Yes	Yes	Yes	Yes	Yes	No	No	Single-hop
GeoSVR [74]	2013	VANET	Position-based	Reactive	DI	Yes	No	No	Yes	Yes	No	Yes	Single-hop
GSPR-MV [75]	2014	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	No	No	No	No	Single-hop
GeoSpray [93]	2014	VANET	Position-based	Reactive	DT	Yes	No	No	Yes	No	Yes	No	Multi-hop
TROUVE [76]	2015	VANET	Position-based	Reactive	DI	Yes	No	Yes	Yes	Yes	No	No	Single-hop
LPRV [77]	2015	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	Yes	No	Yes	No	Single-hop
EHTAR [78]	2015	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Multi-hop
MBMPR [79]	2015	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Single-hop
SCRP [82, 83]	2015	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	Yes	No	No	No	Multi-hop
ETAR [95]	2015	VANET	Position-based	Reactive	DT	Yes	No	Yes	Yes	Yes	No	No	Multi-hop
EIPG [97]	2016	VANET	Position-based	Reactive	DI	Yes	No	No	Yes	No	No	No	Single-hop
RPGR [80]	2016	VANET	Position-based	Reactive	DI	Yes	Yes	No	Yes	Yes	No	No	Single-hop
WNPRP [81]	2016	VANET	Position-based	Reactive	DI	Yes	Yes	Yes	Yes	No	No	No	Single-hop

B

PUBLICATIONS

[1] M. T. Garrosi, Tongxing Lu, Lei Zhong and Yusheng Ji, "Unicast Assisted GeoBroadcast in Urban Vehicular Ad-hoc Networks," 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall), Chicago, 2018.

[2] M. T. Garrosi, X. Xiang and M. Noroozi, "Loop-Free Enhanced Intersection-Based Perimeter Geo-Routing with Carry-and-Forward for Urban VANETs," 2017 IEEE 86th Vehicular Technology Conference (VTC-Fall), Toronto, ON, 2017, pp. 1-7.

[3] M. T. Garrosi, M. Kalac and T. Lorenzen, "Geo-routing in urban Vehicular Ad-hoc Networks: A literature review," 2017 International Conference on Computing, Networking and Communications (ICNC), Santa Clara, CA, 2017, pp. 865-871.

[4] Torsten Lorenzen and Mehdi Tavakoli Garrosi. 2016. Achieving global fairness at urban intersections using cooperative DSRC congestion control. In Proceedings of the First ACM International Workshop on Smart, Autonomous, and Connected Vehicular Systems and Services (CarSys ′16).

[5] M. T. Garrosi, "Enhanced Intersection-Based Perimeter Geo-Routing in Urban Vehicular Ad-Hoc Networks," 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall), Montreal, QC, 2016, pp. 1-5.

[6] M. T. Garrosi, B. Zafar and M. Haardt, "Prolonged network life-time in self-organizing peer-to-peer networks with E-RSSI clustering," 2012 IEEE International Conference on Communications (ICC), Ottawa, ON, 2012, pp. 5558-5562.

[7] B. Zafar, S. Gherekhloo, A. Asgharzadeh, M. T. Garrosi and M. Haardt, "Self-Organizing Network with Intelligent Relaying (SONIR)," The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS 2010), San Francisco, CA, 2010, pp. 765-767.

C

SCIENTIFIC CAREER

Name	Mehdi Tavakoli Garrosi
Date of Birth	04.06.1983
Place of Birth	Gorgan, Iran

Educational Background

Since 07/2014	Research assistant and Ph.D. student
	Institute of Communications Technology
	Leibniz Universität Hannover, Germany
10/2009 - 08/2013	M.Sc. in Electrical Engineering
	Ilmenau University of Technology, Germany
10/2002 - 02/2008	B.Sc. in Electrical Engineering
	K.N.Toosi University of Technology, Iran

Teaching Expirience

04/2015 - 09/2015	Lab exercise on "Networks and Protocols"
10/2014 - 03/2018	Lab exercise on "Computer Networks"
08/2016 - 04/2017	Supervision of Master thesis
	Study and Implementation of Network Architecture of Vehicular Communications
	for Intelligent Transport Systems
08/2016- 04/2017	Supervision of Master thesis
	Study, Classification and Implementation of Diffrerent Approaches of Employing CAM Containers
	in Geo-routing Protocols in Urban VANETs
07/2016 - 03/2017	Supervision of Master thesis
	Implementierung und Analyse von kreuzungsbasierten Geo-Routing Protokollen
	für innerstädtische Fahrzeug-Ad-hoc-Netzwerke
07/2016- 01/2017	Supervision of Master thesis
	Implementation and Analysis of Geo-routing Protocols for Vehicular Ad-hoc Networks in Urban Areas
09/2015 - 02/2016	Supervision of Bachelor thesis
	Ein Überblick über die verschiedenen Routing-Protokolle in der Car-to-X Kommunikation

BIBLIOGRAPHY

[1] K. C. Lee, P. C. Cheng, and M. Gerla, "Geocross: A geographic routing protocol in the presence of loops in urban scenarios," *Ad Hoc Networks*, vol. 8, no. 5, pp. 474–488, 2010.

[2] *Intelligent Transport Systems (ITS), Vehicular Communications, Geographical Area Definition*. ETSI EN 302 931 V1.1.1, Jul 2011.

[3] "Statistisches Bundesamt, Anzahl der Unfälle im deutschen Strassenverkehr in den Jahren 2012 bis 2017."

[4] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 1: Requirements*. ETSI TS 102 636-1 V1.1.1, Mar 2010.

[5] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 1: Requirements*. ETSI EN 302 636-1 V1.2.1, Apr 2014.

[6] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications, Sub-part 1: Media-Independent Functionality*. Draft ETSI EN 302 636-4-1, Oct 2013.

[7] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 2: Scenarios*, vol. 1. ETSI TS 102 636-2 V1.1.1, 2010.

[8] *Intelligent Transport Systems (ITS), Vehicular Communications, Basic Set of Applications, Definitions*. ETSI TR 102 638 V1.1.1, Jun 2009.

[9] A. Festag, "Standards for vehicular communication—from ieee 802.11p to 5g," *e & i Elektrotechnik und Informationstechnik*, vol. 132, pp. 409–416, Nov 2015.

[10] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 3: Network architecture*. ETSI TS 102 636-3 V1.1.1, Mar 2010.

[11] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 3: Network architecture*. Final draft ETSI EN 302 636-3 V1.2.0, Oct 2014.

[12] *Intelligent Transport Systems (ITS), Communications Architecture*. ETSI EN 302 665 V1.1.1, Sep 2010.

[13] *International Mobile Telecommunications 2000 (IMT-2000)*. ITU-R Recommendation M.687-2.

[14] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 2: Scenarios*. ETSI EN 302 636-2 V1.2.1, Nov 2013.

[15] *Co-operative systems for Intelligent Transport in the field of information and communication technologies to support interoperability of cooperative systems for intelligent transport in the European Community.* Mandate M/453.

[16] *Intelligent Transport Systems (ITS), Vehicular Communications, Basic Set of Applications, Part 1: Functional Requirements.* ETSI TS 102 637-1 V1.1.1, Sep 2010.

[17] *Intelligent Transport Systems (ITS), V2X Applications, Part 1: Road Hazard Signalling (RHS) application requirements specification.* ETSI TS 101 539-1 V1.1.1, Aug 2013.

[18] *Intelligent Transport Systems (ITS), V2X Applications, Part 2: Intersection Collision Risk Warning (ICRW) application requirements specification.* ETSI TS 101 539-2.

[19] *Intelligent Transport Systems (ITS), V2X Applications, Part 3: Longitudinal Collision Risk Warning (LCRW) application requirements specification.* ETSI TS 101 539-3 V1.1.1, Nov 2013.

[20] *Intelligent Transport Systems (ITS), Vehicular Communications, Basic Set of Applications, Part 2: Specification of Cooperative Awareness Basic Service.* ETSI EN 302 637-2 V1.3.2, Nov 2014.

[21] *Intelligent Transport Systems (ITS), Facilities layer function, Position and time facility specification.* ETSI TS 102 890-3.

[22] *Intelligent Transport Systems (ITS), Vehicular Communications, Basic Set of Applications, Local Dynamic Map (LDM), Rationale for and guidance on standardization.* ETSI TR 102 863 V1.1.1, Jun 2011.

[23] *Intelligent Transport Systems (ITS), Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band.* Draft ETSI EN 302 663 V1.2.0, Nov 2012.

[24] *Intelligent Transport Systems (ITS), Harmonized Channel Specifications for Intelligent Transport Systems operating in the 5 GHz frequency band.* ETSI TS 102 724 V1.1.1, Oct 2012.

[25] *Intelligent Transport Systems (ITS), Users and applications requirements, Part 2: Applications and facilities layer common data dictionary.* ETSI TS 102 894-2 V1.2.1, Sep 2014.

[26] *Intelligent Transport Systems (ITS), Vehicular Communications, Basic Set of Applications, Part 3: Specifications of Decentralized Environmental Notification Basic Service.* ETSI EN 302 637-3 V1.2.2, Nov 2014.

[27] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking.* ETSI TS 102 636 (all parts).

[28] *Intelligent Transport Systems (ITS), Communications access for land mobiles (CALM), IPv6 networking.* ISO/IEC 21210, Jun 2012.

[29] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 6: Internet Integration, Subpart 1: Transmission of IPv6 Packets over GeoNetworking Protocols.* ETSI TS 102 636-6-1 V1.1.1, Mar 2011.

[30] *Intelligent Transport Systems (ITS), Communications access for land mobiles (CALM), Non-IP networking.* ISO/IEC 29281, Apr 2011.

[31] *Intelligent Transport Systems (ITS), Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range, Access layer part.* ETSI TS 102 687 V1.1.1, Jul 2011.

[32] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 5: Transport Protocols, Sub-part 1: Basic Transport Protocol.* ETSI EN 302 636-5-1 V2.1.1, Aug 2017.

[33] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 5: Transport Protocols, Sub-part 1: Basic Transport Protocol.* ETSI TS 102 636-5-1 V1.1.1, Feb 2011.

[34] *User Datagram Protocol.* IETF RFC 768.

[35] *Intelligent Transport Systems (ITS), Vehicular communications, GeoNetworking, Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications, Sub-part 1: Media-Independent Functionality.* ETSI TS 102 636-4-1 V1.1.1, Jun 2011.

[36] *Intelligent Transport Systems (ITS), Vehicular Communications, GeoNetworking, Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications, Sub-part 2: Media-dependent functionalities for ITS-G5.* ETSI TS 102 636-4-2 V1.1.1, Oct 2013.

[37] *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 802.11, 2012.

[38] *IEEE Standard for Information technology, Telecommunications and information exchange between systems, Local and metropolitan area networks, Specific requirements, Part 2: Logical Link Control.* ANSI/IEEE Std 802.2, 1998.

[39] *Intelligent Transport Systems (ITS), European profile standard for the physical and medium access control layer of Intelligent Transport Systems operating in the 5 GHz frequency band.* Final draft ETSI ES 202 663 V1.1.0, Nov 2009.

[40] *Intelligent Transport Systems (ITS), Radiocommunications equipment operating in the 5 855 MHz to 5 925 MHz frequency band, Harmonized EN covering the essential requirements of article 3.2 of the RTTE Directive.* ETSI EN 302 571 V1.1.1, Sep 2008.

[41] *Broadband Radio Access Networks (BRAN), 5 GHz high performance RLAN, Harmonized EN covering the essential requirements of article 3.2 of the RTTE Directive*. ETSI EN 301 893 V1.5.1, Dec 2008.

[42] M. T. Garrosi, M. Kalac, and T. Lorenzen, "Geo-routing in urban vehicular ad-hoc networks: A literature review," in *2017 International Conference on Computing, Networking and Communications (ICNC)*, pp. 865–871, Jan 2017.

[43] M. T. Garrosi, T. Lu, L. Zhong, and Y. Ji, "Unicast assisted geobroadcast in urban vehicular ad-hoc networks," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, Aug 2018.

[44] F. D. da Cunha, A. Boukerche, L. Villas, A. C. Viana, and A. A. F. Loureiro, "Data communication in vanets: A survey, challenges and applications. [research report] rr-8498," 2014.

[45] H. Tchouankem, T. Zinchenko, and H. Schumacher, "Impact of buildings on vehicle-to-vehicle communication at urban intersections," in *CCNC, 12th Annual IEEE*, pp. 206–212, Jan 2015.

[46] M. T. Garrosi, B. Zafar, and M. Haardt, "Prolonged network life-time in self-organizing peer-to-peer networks with e-rssi clustering," in *2012 IEEE International Conference on Communications (ICC)*, pp. 5558–5562, June 2012.

[47] S. Rehman, M. A. Khan, T. Zia, and L. Zheng, "Vehicular ad-hoc networks (vanets), an overview and challenges," vol. 3, pp. 29–38, 01 2013.

[48] R. Akamatsu, M. Suzuki, T. Okamoto, K. Hara, and H. Shigeno, "Adaptive delay-based geocast protocol for data dissemination in urban vanet," in *2014 Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU)*, pp. 141–146, Jan 2014.

[49] Y. B. Ko and N. H. Vaidya, "Flooding-based geocasting protocols for mobile ad hoc networks," *Mob. Netw. Appl.*, vol. 7, pp. 471–480, Dec. 2002.

[50] H. Joshi, M. Sichitiu, and M. Kihl, "Distributed robust geocast multicast routing for inter-vehicle communication," 05 2007.

[51] G. Zhang, W. Chen, Z. Xu, H. Liang, D. Mu, and L. Gao, *Geocast Routing in Urban Vehicular Ad Hoc Networks*, pp. 23–31. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009.

[52] O. Kaiwartya, S. Kumar, and R. Kasana, "Traffic light based time stable geocast (t-tsg) routing for urban vanets," in *Sixth International Conference on Contemporary Computing (IC3)*, pp. 113–117, Aug 2013.

[53] O. Kaiwartya and S. Kumar, "Cache agent based geocasting (cag) in vanets," in *International Journal of Information and Communication Technology*, vol. 7, pp. 562–584, Sep 2015.

[54] L. Zhang, B. Yu, and J. Pan, “Geomob: A mobility-aware geocast scheme in metropolitans via taxicabs and buses,” in *IEEE INFOCOM 2014, IEEE Conference on Computer Communications*, pp. 1279–1787, April 2014.

[55] B. C. Seet, G. Liu, B. S. Lee, C. H. Foh, K. J. Wong, and K. K. Lee, *Networking 2004*, ch. A-STAR: A Mobile Ad Hoc Routing Strategy for Metropolis Vehicular Communications, pp. 989–999. Springer Berlin Heidelberg, 2004.

[56] C. Lochert, M. Mauve, H. Füßler, and H. Hartenstein, “Geographic routing in city scenarios,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, p. 69, Jan 2005.

[57] J. LeBrun, C. N. Chuah, D. Ghosal, and M. Zhang, “Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks,” in *IEEE 61st Vehicular Technology Conference*, vol. 4, May 2005.

[58] F. Giudici and E. Pagani, *Spatial and Traffic-Aware Routing (STAR) for Vehicular Systems*, pp. 77–86. Springer Berlin Heidelberg, 2005.

[59] F. Granelli, G. Boato, and D. Kliazovich, “Mora: A movement-based routing algorithm for vehicle ad hoc networks,” in *IEEE Workshop AutoNet*, Dec 2006.

[60] K. C. Lee, J. Häerri, U. Lee, and M. Gerla, “Enhanced perimeter routing for geographic forwarding protocols in urban vehicular scenarios,” *IEEE GLOBECOM*, 2007.

[61] M. Jerbi, S. M. Senouci, R. Meraihi, and Y. Ghamri-Doudane, “An improved vehicular ad hoc routing protocol for city environments,” in *IEEE International Conference on Communications*, June 2007.

[62] V. Naumov and T. R. Gross, “Connectivity-aware routing (car) in vehicular ad-hoc networks,” in *26th IEEE INFOCOM*, May 2007.

[63] H. Menouar, M. Lenardi, and F. Filali, “Movement prediction-based routing (mopr) concept for position-based routing in vehicular networks,” in *IEEE 66th Vehicular Technology Conference*, Sept 2007.

[64] S. Schnaufer and W. Effelsberg, “Position-based unicast routing for city scenarios,” in *International Symposium on WoWMoM*, June 2008.

[65] Q. Yang, A. Lim, S. Li, J. Fang, and P. Agrawal, “Acar: Adaptive connectivity aware routing protocol for vehicular ad hoc networks,” in *2008 Proceedings of 17th International Conference on Computer Communications and Networks*, pp. 1–6, Aug 2008.

[66] J. Nzouonta, N. Rajgure, and C. Borcea, “VANET Routing on City Roads Using Real-Time Vehicular Traffic Information,” in *IEEE Transactions on Vehicular Technology*, vol. 58, Sep 2009.

[67] K. C. Lee, U. Lee, and M. Gerla, “To-go: Topology-assist geo-opportunistic routing in urban vehicular grids,” in *6th International Conference on WONS*, pp. 11–18, Feb 2009.

[68] X. b. Wang, Y. l. Yang, and J. w. An, “Multi-metric routing decisions in vanet,” in *8th IEEE International Conference on DASC*, Dec 2009.

[69] C. A. T. H. Tee and A. Lee, “A novel routing protocol, junction based adaptive reactive routing (jarr) for vanet in city environments,” in *Wireless Conference (EW), 2010 European*, pp. 1–6, April 2010.

[70] Y. S. Chen, Y. W. Lin, and C. Y. Pan, “DIR: diagonal-intersection-based routing protocol for vehicular ad hoc networks,” *Telecommunication Systems*, vol. 46, no. 4, pp. 299–316, 2010.

[71] H. Shi and Z. Ding, “Idvr: An intersection dispatch-based vanet routing protocol,” in *7th International Conference on WiCOM*, Sept 2011.

[72] M. Al-Rabayah and R. Malaney, “A new scalable hybrid routing protocol for vanets,” *IEEE Trans. on Vehicular Technology*, July 2012.

[73] J. Bernsen and D. Manivannan, “River: A reliable inter-vehicular routing protocol for vehicular ad hoc networks,” *Computer Networks*, vol. 56, no. 17, pp. 3795, 3807, 2012.

[74] Y. Xiang, Z. Liu, R. Liu, W. Sun, and W. Wang, “Geosvr: A map-based stateless vanet routing,” *Ad Hoc Networks*, vol. 11, no. 7, 2013.

[75] H. Tu, L. Peng, H. Li, and F. Liu, “Gspr-mv: A routing protocol based on motion vector for vanet,” in *12th ICSP*, pp. 2354–2359, Oct 2014.

[76] C. A. Kerrache, N. Lagraa, C. T. Calafate, and A. Lakas, “Trouve: A trusted routing protocol for urban vehicular environments,” in *IEEE 11th International Conference on WiMob*, pp. 260–267, Oct 2015.

[77] L. N. Balico, H. A. B. F. Oliveira, R. S. Barreto, A. A. F. Loureiro, and R. W. Pazzi, “A prediction-based routing algorithm for vehicular ad hoc networks,” in *ISCC*, pp. 365–370, July 2015.

[78] C. C. Lo and Y. H. Kuo, “Enhanced hybrid traffic-aware routing protocol for vehicular ad hoc networks,” in *IEEE 82nd VTC Fall*, Sept 2015.

[79] Y. Lin, W. Huang, and Y. Tang, “Map-based multi-path routing protocol in vanets,” in *IEEE 9th International Conference on ASID*, Sept 2015.

[80] K. N. Qureshi, A. H. Abdullah, and J. Lloret, “Road perception based geographical routing protocol for vehicular ad hoc networks,” *International Journal of Distributed Sensor Networks*, p. 16, 2016.

[81] A. Chinnasamy, S. Prakash, and P. Selvakumari, "Wagon next point routing protocol (wnprp) in vanet," *Wireless Personal Comm.*, 2016.

[82] M. A. Togou, A. Hafid, and L. Khoukhi, "A novel cds-based routing protocol for vehicular ad hoc networks in urban environments," in *IEEE GLOBECOM*, pp. 1–6, Dec 2015.

[83] M. A. Togou, A. Hafid, and L. Khoukhi, "Scrp: Stable cds-based routing protocol for urban vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, pp. 1298–1307, May 2016.

[84] C. Lochert, H. Hartenstein, J. Tian, H. Fussler, D. Hermann, and M. Mauve, "A routing strategy for vehicular ad hoc networks in city environments," in *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pp. 156–161, June 2003.

[85] H. Jiang, H. Guo, and L. Chen, "Reliable and efficient alarm message routing in vanet," in *The 28th International Conference on Distributed Computing Systems Workshops*, pp. 186–191, June 2008.

[86] D. Borsetti, M. Fiore, C. Casetti, and C. F. Chiasserini, "An application-level framework for information dissemination and collection in vehicular networks," Elsevier Science Publishers B. V., Sept. 2011.

[87] J. Zhao and G. Cao, "Vadd: Vehicle-assisted data delivery in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 57, pp. 1910–1922, May 2008.

[88] J. Li, J. Jannotti, D. S. J. De Couto, D. R. Karger, and R. Morris, "A scalable location service for geographic ad hoc routing," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, MobiCom, pp. 120–130, ACM, 2000.

[89] Y. Ding, C. Wang, and L. Xiao, "A static-node assisted adaptive routing protocol in vehicular networks," in *Proceedings of the Fourth ACM International Workshop on Vehicular Ad Hoc Networks*, VANET, pp. 59–68, ACM, 2007.

[90] P. C. Cheng, K. C. Lee, M. Gerla, and J. Härri, "Geodtn+nav: Geographic dtn routing with navigator prediction for urban vehicular environments," *Mobile Networks and Applications*, vol. 15, no. 1, pp. 61–82, 2010.

[91] K. Z. Ghafoor, K. A. Bakar, S. Salleh, K. C. Lee, M. M. Mohammad, M. Kamat, and M. M. Arshad, "Fuzzy logic-assisted geographical routing over vehicular ad hoc networks," *International journal of innovative computing, information and control*, July 2012.

[92] I. Leontiadis and C. Mascolo, "Geopps: Geographical opportunistic routing for vehicular networks," in *2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pp. 1–6, June 2007.

[93] V. Soares, J. Rodrigues, and F.Farahmand, "Geospray: A geographic routing protocol for vehicular delay-tolerant networks," *Information Fusion*, vol. 15, pp. 102, 113, 2014.

[94] V. Namboodiri, M. Agarwal, and L. Gao, "A study on the feasibility of mobile gateways for vehicular ad-hoc networks," in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks*, VANET, pp. 66–75, ACM, 2004.

[95] O. S. Oubbati, A. Lakas, N. Lagraa, and M. B. Yagoubi, "Etar: Efficient traffic light aware routing protocol for vehicular networks," in *IWCMC*, pp. 297–301, Aug 2015.

[96] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter, "Mddv: A mobility-centric data dissemination algorithm for vehicular networks," in *Proceedings of the 1st ACM International Workshop on VANETs*, pp. 47–56, ACM, 2004.

[97] M. T. Garrosi, "Enhanced intersection-based perimeter geo-routing in urban vehicular ad-hoc networks," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, pp. 1–5, Sept 2016.

[98] M. T. Garrosi, X. Xiang, and M. Noroozi, "Loop-free enhanced intersection-based perimeter geo-routing with carry-and-forward for urban vanets," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, Sept 2017.

[99] B. Karp and H. T. Kung, "Gpsr: Greedy perimeter stateless routing for wireless networks," *MobiCom*, 2000.

[100] Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker, "On the pitfalls of geographic face routing," in *Proceedings of the 2005 Joint Workshop on Foundations of Mobile Computing*, DIALM-POMC '05, (New York, NY, USA), pp. 34–43, ACM, 2005.

[101] Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker, "Geographic routing made practical," in *Proceedings of the 2Nd Conference on Symposium on Networked Systems Design & Implementation, Volume 2*, NSDI'05, (Berkeley, CA, USA), pp. 217–230, USENIX Association, 2005.

[102] Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker, "Lazy cross-link removal for geographic routing," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, SenSys, pp. 112–124, ACM, 2006.

[103] M. Jerbi, S. M. Senouci, T. Rasheed, and Y. Ghamri-Doudane, "Towards efficient geographic routing in urban vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 5048–5059, Nov 2009.

[104] J. A. Davis, A. H. Fagg, and B. N. Levine, "Wearable computers as packet transport mechanisms in highly-partitioned ad-hoc networks,"

in *Proceedings Fifth International Symposium on Wearable Computers*, pp. 141–148, 2001.

[105] *Algorithms and data structures for the real-time processing of traffic data*. Miller, J. A., Jun 2007.

[106] T. Lorenzen and H. Tchouankem, "Evaluation of an awareness control algorithm for vanets based on etsi en 302 637-2 v1.3.2," in *Communication Workshop (ICCW), IEEE International Conference on*, pp. 2458–2464, June 2015.

[107] T. D. C. Little, S. Member, A. Agarwal, and S. Member, "An information propagation scheme for vanets," in *IEEE Conference on Intelligent Transportation Systems*, pp. 13–16, 2005.

[108] W. Sun, H. Yamaguchi, K. Yukimasa, and S. Kusumoto, "Gvgrid: A qos routing protocol for vehicular ad hoc networks," in *14th IEEE International Workshop on QoS*, 2006.

[109] M. Abuelela, S. Olariu, and I. Stojmenovic, "Opera: Opportunistic packet relaying in disconnected vehicular ad hoc networks," in *5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, pp. 285–294, Sept 2008.

[110] Y. Luo, W. Zhang, and Y. Hu, "A new cluster based routing protocol for vanet," in *2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing*, vol. 1, pp. 176–180, April 2010.

[111] R. S. Schwartz, H. Scholten, and P. Havinga, "A scalable data dissemination protocol for both highway and urban vehicular environments," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, p. 257, Nov 2013.

[112] A. Daeinabi, A. G. P. Rahbar, and A. Khademzadeh, "Vwca: An efficient clustering algorithm in vehicular ad hoc networks," *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 207, 222, 2011.

[113] M. S. Kakkasageri and S. S. Manvi, "Multiagent driven dynamic clustering of vehicles in vanets," *J. Netw. Comput. Appl.*, vol. 35, pp. 1771–1780, Nov. 2012.

[114] S.-S. Wang and Y.-S. Lin, "Passcar: A passive clustering aided routing protocol for vehicular ad hoc networks," *Computer Communications*, vol. 36, no. 2, pp. 170, 179, 2013.

[115] B. Aslam, F. Amjad, and C. C. Zou, "Pmtr: Privacy-enhancing multilayer trajectory-based routing protocol for vehicular ad hoc networks," in *MILCOM, IEEE Military Communications Conference*, pp. 882–887, Nov 2013.

[116] X. Shen, Y. Wu, Z. Xu, and X. Lind, "Aodv-pnt: An improved version of aodv routing protocol with predicting node trend in vanet," in *The 7th IEEE/International Conference on Advanced Infocomm Technology*, pp. 91–97, Nov 2014.

[117] V. Namboodiri and L. Gao, "Prediction-based routing for vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 2332–2345, July 2007.

[118] X. Huang and Y. Fang, "Performance study of node-disjoint multipath routing in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 1942–1950, May 2009.

[119] Z. Li and C. Chigan, "Leaper: A lightweight reliable and faithful data packet relaying framework for vanets," *Ad Hoc Networks*, vol. 9, no. 3, pp. 418, 429, 2011.

[120] M. Bakhouya, J. Gaber, and P. Lorenz, "An adaptive approach for information dissemination in vehicular ad hoc networks," *Journal of Network and Computer Applications*, vol. 34, no. 6, pp. 1971, 1978, 2011.

[121] J. Toutouh, J. Garcia-Nieto, and E. Alba, "Intelligent olsr routing protocol optimization for vanets," *IEEE Transactions on Vehicular Technology*, vol. 61, pp. 1884–1894, May 2012.

[122] L. V. Minh, Y. MingChuan, and G. Qing, "End-to-end delay assessment and hybrid routing protocol for vehicular ad hoc network," *IERI Procedia*, vol. 2, pp. 727, 733, 2012. International Conference on Future Computer Supported Education, August 22- 23, 2012.

[123] A. Boukerche, K. Abrougui, and R. W. N. Pazzi, "Location-aided gateway advertisement and discovery protocol for vanets: Proof of correctness," in *IEEE 34th Conference on Local Computer Networks*, pp. 778–785, Oct 2009.

[124] R. Frank, E. Giordano, P. Cataldi, and M. Gerla, "Trafroute: A different approach to routing in vehicular networks," in *IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications*, pp. 521–528, Oct 2010.

[125] K. Mershad, H. Artail, and M. Gerla, "Roamer, roadside units as message routers in vanets," *Ad Hoc Networks*, vol. 10, no. 3, pp. 479, 496, 2012.