Democratic and Popular Republic of Algeria
Ministry of Higher Education and Scientific Research
University of Kasdi Merbah, Ouargla

Faculty of new information and communication technologies
Department of computer science and Information technology

Thesis Submitted to the Department of Computer Science and Information Technology in Candidacy for the Degree of "Doctor" 3rd Cycle LMD in Computer Science.

Option: Networks and Telecommunication

Supply and consumption of services in vehicular networks

Presented by:

Mohamed Ben bezziane

Board of Examiners

Dr. Bachir Said	Kasdi Merbah university of Ouargla	President
Pr. Ahmed Korichi	Kasdi Merbah university of Ouargla	Supervisor
Dr. Akram zine eddine Boukhamla	Kasdi Merbah university of Ouargla	Examiner
Dr. Khaldi Amine	Kasdi Merbah university of Ouargla	Examiner
Dr. Mounir Beggas	Echahid Hamma Lakhdar university of Eloued	Examiner
Pr. Brahim Lejdel	Echahid Hamma Lakhdar university of Eloued	Examiner

Academic Year: 2021 – 2022

Acknowledgments

First, praise and thanks to ALLAH the sacred and the mighty for giving me the ability and helping me in accomplishing this thesis.

Then, I would like to express my sincerest gratitude to my supervisor Pr. Korichi Ahmed for his continuous support, help, guidance, motivation and patience throughout this thesis. I deeply appreciate his immense knowledge, constructive feedback and assistance that have considerably improved my experience.

I also greatly appreciate the collaboration and the cooperation of my friend Dr. Fekair Mohammed Elamine. I deeply acknowledge him for their assistance, discussions and efforts.

Avery special thank goes to the honourable board of examiners who accepted to examine and to evaluate this thesis namely, Dr. Bachir Said, Khaldi Amine and Akram Boukhamla (Ouargla University); Dr. Mounir Beggas and Pr. Brahim Lejdel (University of Eloued).

I owe an earnest thank and respect to the teachers in the Department of Computer Science and Information Technology, especially those whom I was a student for years, and the colleagues with whom I have been working such as Dr. Kerrache Chaker Abdelaziz.

.

Abstract

In recent years, there has been growing interest in accessibility to information and, in particular, in innovative approaches using remote services accessible from mobile devices across the world. At the same time, vehicle communication, using on-board sensors and wireless communication devices, has been introduced to improve road safety and the driving experience through what is commonly referred to as vehicle networks (VANETs).

The service offering or the vehicle cloud provides drivers with access to several services such as the Internet as a service, information as a service, storage and compute as a service, etc. However, given the high mobility of vehicles, consumer vehicles must first discover the supplier vehicles and their services before they can consume the required services.

In this thesis, we are interested in data offering and consumption of services in vehicular networks. In a first contribution, we suggested a novel approach (FR-VC) that allows a successful registration of services on the vehicular cloud in real time. The experimental results prove the performance of our proposal in terms of waiting time to register services in the VC.

In a second contribution, we proposed a novel protocol named RSU-aided Cluster-based Vehicular Clouds protocol (RCVC) which constructs the VC using the Road Side Unit (RSU) directory and Cluster Head (CH) directory to make the resources of supplier vehicles more visible. We carried out an experimental comparison, revealing that the proposed method outperformed several states of the art protocols.

Keywords: Vehicular cloud; VANET; cloud computing; vehicle clustering.

Résumé

Ces dernières années, l'accessibilité à l'information a suscité un intérêt croissant et, en particulier, des approches innovantes utilisant des services à distance accessibles depuis des appareils mobiles à travers le monde. Entre temps, la communication des véhicules, utilisant des capteurs embarqués et des dispositifs de communication sans fil, a été introduite pour améliorer la sécurité routière et l'expérience de conduite grâce à ce que l'on appelle communément les réseaux de véhicules (VANET).

L'offre de service où le Cloud véhicule permet aux conducteurs d'accéder à plusieurs services tels que l'Internet en tant que service, l'information en tant que service, le stockage et le calcul en tant que service, etc. Cependant, compte tenu de la grande mobilité des véhicules, les véhicules clients doivent d'abord découvrir les véhicules fournisseurs et leurs services avant qu'ils puissent consommer les services requis.

Dans cette thèse, nous nous intéressons à l'offre et à la consommation de services dans les réseaux véhiculaires. Dans une première contribution, nous avons proposé une nouvelle approche (FR-VC) qui permet d’enregistrer avec succès et en temps réel les services sur le Cloud véhiculaire. Les résultats expérimentaux prouvent la performance de notre proposition en termes de temps d'attente pour enregistrer des services dans le Cloud véhiculaire.

Dans une deuxième contribution, nous avons proposé un nouveau protocole nommé RSU-aided Cluster-based Vehicular Clouds protocol (RCVC) qui construit le VC en utilisant le répertoire Road Side Unit (RSU) et le répertoire Cluster Head (CH) pour rendre les ressources des véhicules des fournisseurs. plus visible. Nous avons effectué une comparaison expérimentale, révélant que la méthode proposée surpassait plusieurs protocoles de l'état de l'art.

Mots clés : Réseaux Ad hoc véhiculaire (VANet), Cloud véhiculaire, Sélection de services (fournisseurs), RCVC, FR-VC.

ملخص

إمكانية الوصول إلى المعلومات في عالم التكنولوجيات الحديثة كان محل إهتمام مجموعة من باحثي الحاسوب و الرقمنة في السنوات الأخيرة، ولا سيما الأساليب المبتكرة باستخدام الخدمات عن بُعد التي يمكن الوصول إليها من الأجهزة المحمولة في جميع أنحاء العالم. بعدها، تم إدخال إتصالات المركبات السيارة، بغية تحسين السلامة المرورية وتسهيل قيادة المركبات وباستخدام أجهزة الاستشعار وأجهزة الاتصال اللاسلكي، إذ قام عدد معتبر من مراكز البحث العلمي على مستوى العالم بمباشرة أبحاث كثيفة في مجال شبكات المركبات السيارة (VANET).

يوفر عرض الخدمة أو سحابة المركبات للسائقين إمكانية الوصول إلى العديد من الخدمات مثل الإنترنت، المعلومات والتخزين والحساب كخدمة ، وما إلى ذلك. ومع ذلك ، نظرًا للحركة العالية للمركبات السيارة، يجب أن تكتشف المركبات الاستهلاكية أولاً مركبات المورّد وخدماتها قبل أن تتمكن من استهلاك الخدمات المطلوبة.

في هذه الأطروحة، نحن مهتمون بتوريد واستهلاك الخدمات في شبكات السيارات. في مساهمة أولى، إقترحنا نهجًا جديدًا (FR-VC) يسمح بالتسجيل بنجاح وفي الوقت الفعلي للخدمات على سحابة المركبات. تثبت النتائج التجريبية أداء اقتراحنا من حيث وقت الانتظار لتسجيل الخدمات في سحابة المركبات.

في مساهمة ثانية، اقترحنا بروتوكولًا جديدًا يسمى (RCVC) بروتوكول سحابة المركبات المدعوم من قبل وحدة جانب الطريق (RSU) والذي ينشأ السحابة المركبية (VC) باستخدام دليل وحدة جانب الطريق ودليل المركبة الرائدة (Cluster Head) لإضفاء أكثر رؤية لخدمات المركبات الموردة. أكثر وضوحا. أجرينا مقارنة تجريبية، وكشفنا أن البروتوكول المقترح تجاوز العديد من البروتوكولات الحديثة.

الكلمات المفتاحية: شبكة المركبات المخصصة؛ السحابة الإلكترونية؛ السيارات الموردة للخدمات.

Table of Contents

Abstract **I**
Table of Contents **IV**
List of Figures **VI**
List of Tables **VIII**
List of acronyms **X**

1. General introduction

1.1. Introduction **13**
1.2. Technological trend **14**
1.3. Thesis context **14**
1.4. VANET networks **15**
1.4.1. VANET characteristics **16**
1.4.2. VANET applications **17**
1.4.3. VANET architectures **17**
1.4.4. VANET mobility models **18**
1.4.4.1 Random-based mobility models **18**
1.4.4.2 Geographic map-based mobility models **19**
1.4.4.3 Group-based mobility models **21**
1.4.4.4 Prediction-based mobility models **23**
1.4.5. Normalizations and standardization works **23**
1.5. Problematic **25**
1.6. Objectives and contributions **26**
1.7. Thesis structure **27**

2. Literature review on the supply and consumption of services in vehicular networks

2.1. Introduction **28**
2.2. Routing for VANET **29**
2.2.1 Overview **29**
2.2.2 Routing classification for VANET **30**
2.3. Cloud computing in general **35**
2.3.1. Smart city **35**
2.3.2. Cloud computing in smart cities **36**
2.4 Vehicular cloud **38**
2.4.1. Definition **38**
2.4.2. Vehicular cloud types **39**
2.4.3. The different mobile vehicular cloud services **39**
2.5. Works-related to the vehicular cloud **40**
2.6. VANETs simulation environments **42**
2.6.1. Software tools for mobility models **42**
2.6.2. Network simulators **42**
2.6.2.1 Integrated Simulators: Veins **43**
2.7. Conclusion **44**

3. Finding Resources in Vehicular Cloud (FR-VC)

3.1. Introduction **45**
3.2. FR-VC approach architecture **45**
3.2.1. VC creation at the cluster head level **45**
3.2.1.1. Cluster creation **48**
3.2.2 VC creation at the RSU level **50**
3.3. Performance and validation **51**
3.3.1. Simulation setup and parameters **51**
3.3.2. Result and discussions **53**
3.4. Conclusion et perspectives **54**

4. RSU-aided Cluster-based Vehicular Clouds architecture for urban Areas (RCVC)

4.1. Introduction **54**
4.2. RCVC protocol architecture **54**
4.2.1. Mobile cluster-based vehicular cloud **57**
4.2.2 VC creation at the RSU level **63**
4.3. A mathematical model for service selection **67**
4.3.1. Normalization **68**
4.3.2. Performance score **68**
4.4 Experimental analysis **69**
4.4.1. Simulation setup **69**
4.4.2. Results and discussions **71**
4.4. Conclusion **75**

5. Conclusion and perspectives

5.1. General conclusion and perspectives **76**

Publications list **78**
Bibliography **79**

List of Figures

Figure 1 : VANET transmission types 15
Figure 2 : VANET architectures 18
Figure 3 : Random waypoint mobility model 19
Figure 4 : Random walk mobility model 19
Figure 5 : Manhattan grid mobility model 20
Figure 6 : City section mobility model 20
Figure 7 : Freeway mobility model 21
Figure 8 : Reference point group mobility model 22
Figure 9 Virtual track mobility model 22
Figure 10 : WAVE 1609 model 23
Figure 11 : DSRC frequency band 24
Figure 12 : Single-hop beaconing 30
Figure 13 : Multi-hop beaconing 30
Figure 14 : Topology-based routing protocols 31
Figure 15 : Geography-based-routing 33
Figure 16 : Cluster-based-routing 34
Figure 17 : Veins Architecture 43
Figure 18 : Cluster creation 49
Figure 19 : . Performance evaluation comparisons of FR-VC with CROWN and TOPVISOR in terms of W_time 53
Figure 20 : Performance evaluation comparisons of FR-VC with CROWN, DCCS-VC and TOPVISOR in terms of HIT 53
Figure 21 : Performance evaluation comparisons of FR-VC with DCCS-VC in terms of SUC 54
Figure 22 : Vehicular cloud types in RCVC 57
Figure 23 : RCVC Registration/Request packets format 57
Figure 24 : Registration and data requests process at the RSU level to enable vehicular clouds 58
Figure 25 : Cluster section 61
Figure 26 : RCVC protocol packet format 63
Figure 27 : Sequence diagram of packets exchanged by RCVC 65
Figure 28 : State diagram of the supplier vehicle object 66
Figure 29 : State diagram of the consumer vehicle object 66
Figure 30 : Performance evaluation of RCVC while varying the SV density: (a) Discovery Delay. (b) Consuming Delay. (c) Vehicle Traffic. (d) End-to-End Delay. 72

Figure 31 : Performance evaluation comparisons of RCVC with: (a) CROWN, DCCS-VC, and TOPVISOR in terms of Discovery Delay. (b) CROWN and DCCS-VC in terms of Consuming Delay. (c) CROWN, DCCS-VC, FDCCS-VC, and TOPVISOR in terms of Vehicle Traffic. (d) CROWN, DCCS-VC in terms of End-to-End Delay **73**

LIST OF TABLES

Table 1 : Simulation parameters **52**
Table 2 : Quality criteria of SV **67**
Table 3 : Simulation parameters **69**
Table 4 : Distribution values of elapsed times to arrive to RSU **70**
Table 5 : Sufficient number of hops by varying Number of vehicles in a cluster **70**
Table 6 : A comparative ranking of RCVC with the existing state-of-the-art **74**
Table 7 : Normalized decision matrix **74**
Table 8 : Weighted normalized decision matrix **75**

List of acronyms

VANET	Vehicular Ad Hoc Network
VANETs	Vehicular ad-hoc Networks
ITS	Intelligent Transportation Systems
MANET	Mobile Ad hoc Network
RSU	Road Side Unit
ITA	Intermediate Trusted Authority
IVC	Inter-Vehicle Communication
VRC	Vehicle-to-Roadside Communication
QoS	Quality of Service
IOT	Internet of things
5G	Fifth generation technology standard for broadband cellular networks
MCC	Mobile Cloud Computing
VC	Vehicular Cloud
WLAN	Wireless Local Area Network
RP	Reference point
GM	Group motion vector
DSRC	Dedicated Short-Range Communication
WAVE	Wireless Access in Vehicular Environments
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
WSMP	Wave Short Message Protocol
PHY	Physical
LLC	Logical link control
SCH	Service Channel
CCH	Control Channel
OBU	On Board Unit
LTE	Long Term Evolution
WiMAX	Worldwide Interoperability for Microwave Access
AC	Access Categories
AODV	Ad hoc On-demand Distance Vector
AP	Access Point
BER	Bit Error Rate
BS	Base Station
E2E	End to End
E2ED	End to End Delay
EIFS	Extended Inter Frame Space
ETSI	European Telecommunications Standards Institute
FCC	Federal Communications Commission
GPS	Global Positioning System
IEEE	Institute of Electrical and Electronics Engineers

ISO	International Organization for Standardization
MAC	Medium Access Control
Ns-2	Network Simulator-2
Ns-3	Network Simulator-3
OLSR	Optimized Link State Routing Protocol
OMNet++	Objective Modular Network Testbed in C++
OSI	Open Systems Interconnection
PDR	Packet Delivery Network
PER	Packet Error Rate
TraCI	Traffic Control Interface
V2I	Vehicle-to-Infrastructure
V2V	Vehicle to Vehicle
Veins	Vehicles in Network Simulation
WSN	Wireless Sensor Networks.
WWW	World Wide Web
VC	Vehicular Cloud
SV	Supplier vehicle
CV	Consumer vehicle
CH	Cluster Head
FR-VC	Finding resources in the vehicular cloud
RCVC	RSU-Aided Cluster-Based Vehicular Clouds Architecture for Urban Areas
DSDV	Destination-Sequenced Distance Vector
DSR	Dynamic source routing
ZRP	Zone Routing Protocol
GPSR	Greedy perimeter stateless routing
ITU-T	International Telecommunication Union Telecommunication
NAS	Network Attached Storage
ICT	Information and Communication Technologies
SnaaS	Sensing as a Service
SEaaS	Sensor Event as a Service
IT	Information Technology
NoSQL	No Structured Query Language
XaaS	Anythings as a Service
CoT	Cloud of Things
PS	Partcipory Sensing
TQMS	Trip Quality Measurement System
SVC	Stationary vehicular cloud
MVC	Mobile vehicular cloud (MVC)
CAV	Cloud Ad hoc Vehicular
VuC	Vehicles use Cloud
NaaS	Network as a Service
SaaS	Storage as a Service
DaaS	Data as a Service
CROWN	disCoveRing and cOnsuming services WithiN vehicular clouds
DCCS-VC	Discovering and Consuming Cloud Services in Vehicular Clouds

GTC	Grid-based Tracking Cell
TBP	Tracking Bus Path
TBPI	Tracking Bus Path Identification
FDCCS-VC	Fast Discovering and Consuming Cloud Services in Vehicular Cloud
FcVcA	Fuzzy Clustering-based Vehicular Cloud Architecture
Q-learning	Quality learning
DHCV	Distributed D-Hop cluster of Vehicles formation algorithm
VWaaS	Vehicle Witnesses as a Service
GaaS	Gateway as a Service
TOPVISOR	Two-level cOntroller-based aPproach for serVice advertISement and discOveRy
SUMO	Simulation of Urban MObility
API	Application Programming Interface
ID	Identification
E-GPSR	Enhancement of Greedy Perimeter Stateless Routing
ListV	List of vehilces
Tth	Threshold
dly	Delay
AV	Average
Dist	Distance
RD	RSU directory
VD	Vehicle Density
W_time	Wait-time
HIT	Hit Percentage
SUC	Successful registration rate
RGP	registration packet
REQP	REQuest Packet
WT	Waiting Time
DPH	Discovery PHase
CP	Cluster Packet
Numhops	Number of Hops
RP	Response Packet
NACK	Negative Acknowledgement
SP	Service Packet
SR	Response Packet
SAW	Simple Additive Weighting
$Bandwidth_{Naas}$	Access bandwidth
$Duration\_bandwidth_{NaaS}$	The offered access bandwidth duration
$Cost_{NaaS}$	The offered bandwidth unit price
$Storage_{SaaS}$	Offered storage
$Duration\_Storage_{SaaS}$	The offered storage duration
$Days_{SaaS}$	Maximum overall storage time
$Cost_{SaaS}$	The offered storage unit price
$Data_{DaaS}$	Data capacity
$Cost_{DaaS}$	The offered data unit price
CondidSV	List of Supplier vehicles

MATDIC	Decision matrix
PERF	Performance score
MATDICnom	weighted normalized decision matrix
DD	Discovery Delay
CD	Consuming Delay
VT	Vehicle Traffic
UAV	Unmanned Aerial Vehicles

Chapter

1

GENERAL INTRODUCTION

Content

1.1. Introduction
1.2. Technological trend.
1.3. Thesis context
1.4. VANET networks
  1.4.1. VANET characteristics
  1.4.2. VANET applications
  1.4.3. VANET architectures
  1.4.4. VANET mobility models
  1.4.5. Normalizations and standardization works.
1.5. Problematic
1.6. Objectives and contributions
1.7. Thesis structure

1.1. Introduction

In recent times, remarkable advances have been achieved in the field of ad-hoc vehicular network (VANET) [1], which help enhancing driving by increasing safety and providing commercial services [2]. All exertions aim to increase the use of Intelligent Transportation Systems (ITS) [3]. The increase in the number of vehicles in circulation has increased the problems of road traffic congestion and the safety of road users. Indeed, the problem of traffic congestion in large cities results in traffic jams and lengthening travel times.

Due to their restricted range and motion (direction and speed), VANETs are considered as a specific type of Mobile Ad hoc Network (MANET) with particular mobility. In VANETs, vehicles move according to an organized pattern that is called a mobility model which is based on predefined roads, buildings, and junctions, etc. [4]. Two types of VANET nodes can be distinguished; these types are mobile nodes such as vehicles, and fixed nodes known as Road Side Units (RSUs) deployed at critical locations on roadsides to provide various services including internet access, and can play the role on an Intermediate Trusted Authority (ITA) [5]. Vehicles and RSUs transmit messages between the source and destination nodes using two types of wireless multi-hop transmission, which are Inter-Vehicle Communication (IVC) and Vehicle-to-Roadside Communication (VRC).

In this context, the main objective of our work is to study and model the techniques of supply and consumption of services in vehicular networks, as well as to assess the qualities of service (QoS) of our proposals in protocol and approach terms. In this vision, we study the impact of clustering by integrating a mathematical model for the service selection of the best services. After analyzing the existing protocols, we proposed two new techniques for both the supply and consumption of services in vehicular networks. Furthermore, we carried out an experimental comparison, revealing that their evaluating performance methods outperformed several states of the art protocols.

1.2. Technological trend

Nowadays, the research trend is directed towards the development of intelligent technological infrastructures such as edge computing, IOT and 5G, to provide useful and rapid information with low latency [6]. These infrastructures allow communications between their users. Undeniably, Mobile Cloud Computing (MCC) is one of these infrastructures that can comparatively resolve obvious problems that arise with the growing number of connected devices in traditional cloud computing [7]. Subsequently, the MCC implies the emergence of vehicular cloud VC [8]. To exchange and share information relevant to different types of applications such as those related to road safety and the management of the freeways, road weather, and drivers helping, it is necessary to give more importance to the development of the architectures that facilitate the communication between the vehicles. This trend aims to direct scientific research, within the wireless networking field, towards this new type of infrastructure such as vehicular networks.

Regarding the development of the next generation, the intelligent transportation system is intended to play an essential role of VANET to see its potential. Yet, many challenges need to be addressed including the collection of real-time vehicles information. In addition to the different specificities and high mobility of vehicles that characterize vehicular networks, and before being able to deploy such a network, many experimental phases and several issues, such as the supply and consumption of services in the vehicular network in real-time, must be taken into consideration. Among these issues, there are those which present constraints that require better quality of service.

1.3. Thesis context

In this section, we will present an overview of this thesis on this kind of network (VANET). We provide a definition and indicate its characteristics and the deployment environments of these networks. Hence, its useful applications can be envisaged and exploited on the intelligent transportation system (ITS) that concerns the integration of technology of information, as well as some normalizations and standardization work which is launched to respond to the specificities of vehicular networks. Regarding what constitutes the purpose of our work, we will present all details of our new proposal protocol of offering and consuming services in VANET networks and its architecture.

1.4. VANET networks

VANTs have emerged as an exciting research and application domain aiming to enhance driving by improving road safety and providing commercial services. Due to their restricted range and motion (direction and speed), MANET network nodes represent vehicles in VANET networks. These nodes circulate with some specific mobility feature, including a set of things that are based on predefined roads, buildings, and junctions, etc. [9]. VANET stands for Vehicular Ad-hoc Network. As the name implies, it is an ad-hoc network with two types of nodes: vehicles (On-Board Units (OBU)) and RSUs. Vehicles communicate among them and share traffic, comfort, and entertainment information. RSU generally has multiple network interfaces to connect to the vehicles or other RSUs. Vehicles and RSUs transmit messages between nodes using intermediate nodes with two types of wireless multi-hop transmission, which are Inter-Vehicle Communication (IVC) and Vehicle-to-Roadside Communication (VRC) [10]. Figure 1 shows all kinds of transmission in VANET.

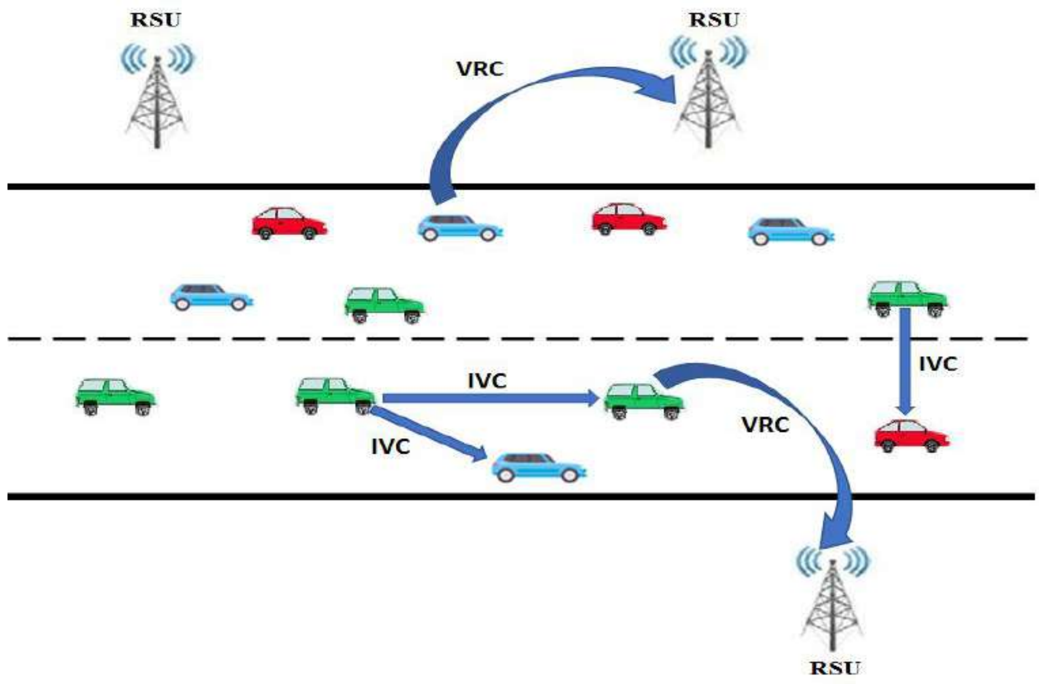


Figure 1. VANET transmission types.

In a VANET network, the communication among vehicles (nodes) can be via a Vehicle-to-Vehicle (V2V) wireless communication and with the infrastructures that are installed at the Road-Side Units (RSUs) via a communication Wireless Vehicle to Infrastructure (V2I). The main objectives of this type of network consist in offering two main classes of applications, namely the applications that make it possible to build an intelligent transport system ITS (Intelligent Transport System) and those linked to the safety and comfort of the driver and passengers.

1.4.1. VANET characteristics

VANET has particular properties, which influence vehicular applications [11]. The most important among them could be cited:

- **High mobility:** Easily the speed node is located between 0 k/h when the vehicle is stuck in traffic jams and 200 k/h on highways so it changes according to the environment. Consequently, there is a high dynamic change in vehicular networking.

- **Vehicle resources:** vehicles are equipped with large storage capacity, computing power and unlimited energy, not like MANET, which is limited in terms of energy.

- **Mobility model:** Contrary to other networks which contain notes moving around arbitrarily, VANETs are characterized by their predicted mobility because of the infrastructure (roads, highways, signs and limited speeds) and diver behaviours and their reactions against critical situations such as accidents and traffic jams, etc.

- **Radio channel:** The communication in VANET has been done on the extern environment; this last contains many obstacles such as urban area, which cause decreasing of quality of services.

- **Density:** The deployment of VANET nodes can cause a high density in the transmission range of one node that can exceed 250 nodes due to various reasons as well as traffic jams and accidents. However, a low density can occur if the area contains a low number of vehicles, particularly in rural areas or at night.

1.4.2. VANET applications

Different applications are found in VANETs, and are cited in the following:

- **Road safety applications:** it has a significant impact on saving people live by sharing and gathering information by safety messages.

- **Vehicles authority services:** this kind of application helps the transport authorities such as police and emergency to contribute to road safety by sending warning and emergency messages from authority vehicles to ordinary vehicles which also can send the information to the authority centres using surveillance applications like stolen vehicles tracking and electronic license plate verification.

- **Enhanced driving:** the information is disseminated in the local area to propose some beneficial actions in the case of awful weather and supply informative services as well as the nearest fuel station, etc.

- **Business and entertainment services:** they provide drivers and passengers with a set of services through the internet or other private networks such as parking payment and road usage payment. Moreover, entertainment and interactive multimedia by downloading movies, music and playing online games, etc.

1.4.3. VANET architectures

Three types of architectures as shown in Figure 2 [12] can be cited:

a. **Vehicular WLAN/cellular:** in this type of network, the antennas of the cellular networks ensure the coverage of the VANET network.

b. **Pure Ad-hoc:** In this type of network, communication is done purely between vehicles.

c. **Hybrid:** This type of network is the mixture of the two networks mentioned above.

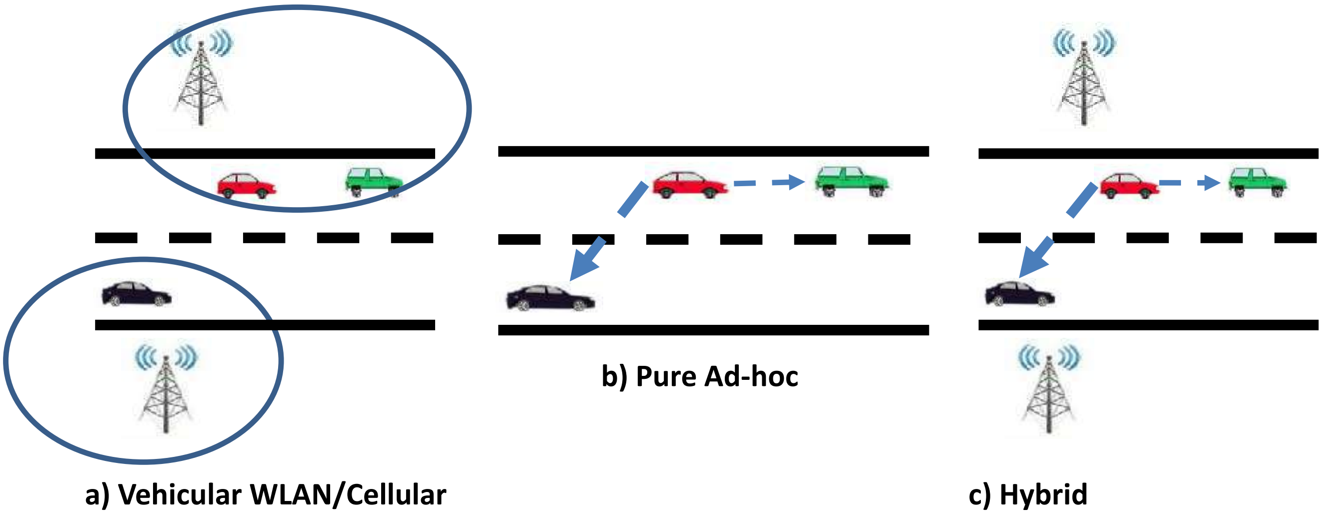


Figure 2. VANET architectures.

1.4.4. VANET Mobility models

Seizing the chance to exhibit four (04) mobility models which are:

1.4.4.1. Random-based mobility models:

In this model, nodes move randomly over time. There are two types of this model, which are:

- **Random waypoint mobility model:** Nodes move periodically and wait in their locations during a certain period. After they select randomly their directions from [0,π] and their speeds, this model remains theoretical because vehicles can not detect the obstacles in the simulation area. As shown in Figure 03 [13].

- **Random walk mobility model:** It is inspired by movements of particles in nature, nodes move through four (4) directions (north, south, east, or west) on a grid of cells. As shown in Figure 04 [14].

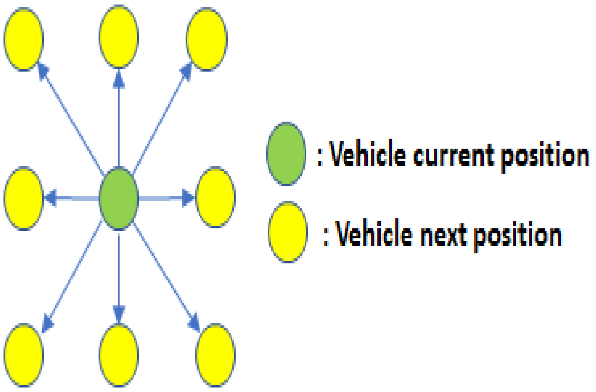


Figure 3. Random waypoint mobility model.

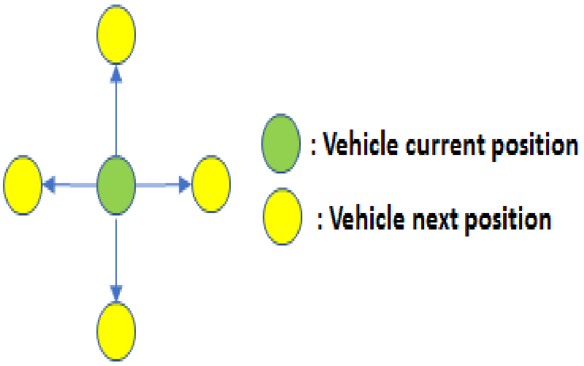


Figure 4. Random walk mobility model.

Limitation of Random-based mobility models:

- ✓ Random changing speeds do not exist in reality.
- ✓ Certain vehicle scenarios cannot be applied due to non-capture geographic restrictions.

1.4.4.2. Geographic map-based mobility models:

To get over the limitation of the last mobility model. Especially, non-capture geographic restrictions, it has been proposed vehicular maps within streets, buildings, and curves, etc. Three (03) mobility models are exhibited. They are:

- **Manhattan grid mobility model**: This model concerns urban areas where nodes move on the square which is divided into a grid of roads that contain two lanes for this opposite direction. Once a node reaches the intersection, it must keep direction with probability equal to ½ or turn left or right with probability equal to¼ [15]. Moreover, speed relatively changes with time and according to preceding vehicles in the same lanes. As shown in Figure 05.

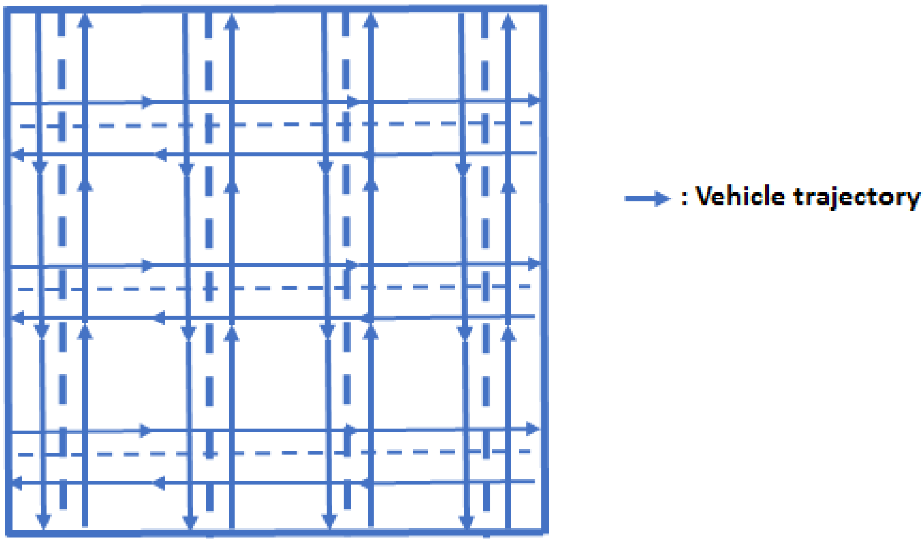


Figure 5. Manhattan grid mobility model.

- **City section mobility model:** This model concerns the downtown area where nodes move on roads that contain one lane. Once a node reaches the intersection, it chooses with one horizontal and one vertical movement at most. Moreover, speed relatively depends on the selected road such as slow way, expressway or others; when it arrives it pauses for a period and restarts travelling randomly [16]. As shown in Figure 6.

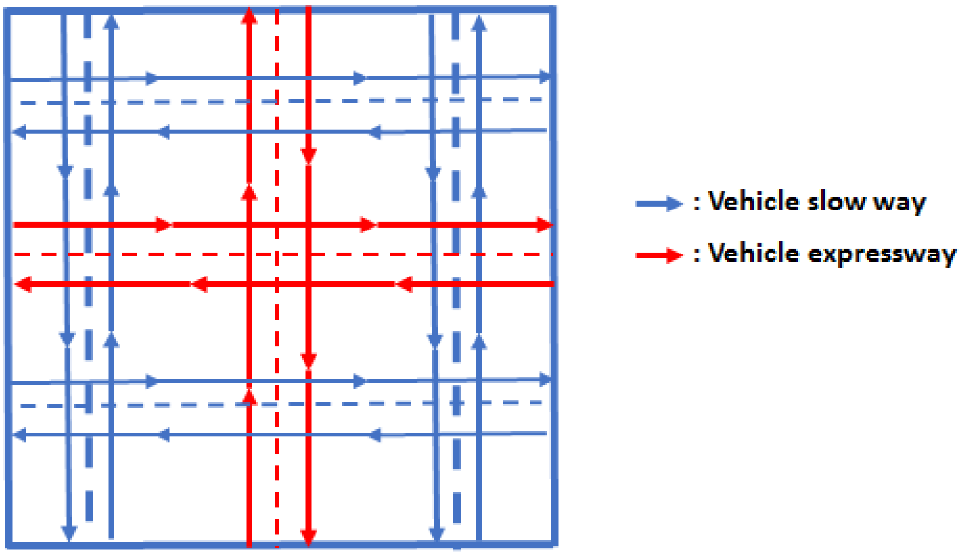


Figure 6. City section mobility model.

- **Freeway mobility model:** In this model, the nodes move on bidirectional roads (freeways) that contain multi-lanes. Moreover, speed relatively depends on the previous speed and the velocity of the successive vehicle [17]. It is worth noting that the security distance between vehicles is required. As shown in Figure 7.

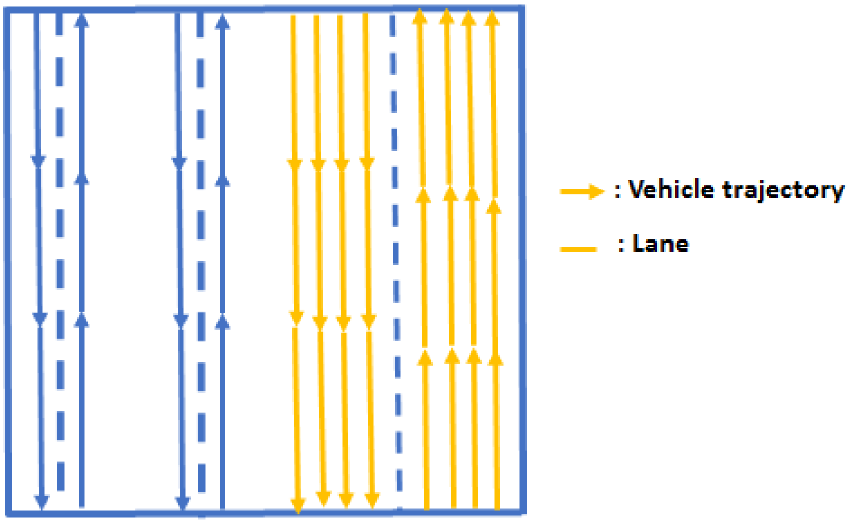


Figure 7. Freeway mobility model.

The limitation of Random-based mobility models is the non-prediction vehicles' motions that can cause high density.

1.4.4.3. Group-based mobility models:

In this model, the network is divided into groups of nodes according to their same travelling destination and motion behaviours. Two (02) mobility models would be cited:

- **Reference point group mobility model:** This model divides the network into several groups of vehicles that have the same target. Each group has a logic centre point that leads the group’s motion behaviour (speed, location, direction, and acceleration). Each node moves randomly in the neighborhood of its centre point Which moves at the moment t from the reference point RP(t) to the next RP(t+1) with group motion vector $\overrightarrow{GM}$ [18]. As shown in Figure 8.

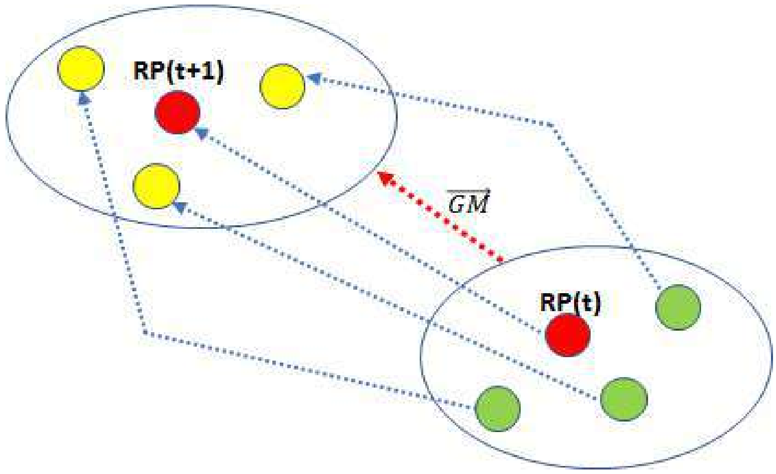


Figure 8. Reference point group mobility model.

- **Virtual track mobility model:** This model describes dynamics of group mobility in virtual tracks which are connected with some switch stations deployed randomly in the area of the network. Once the group reaches to switch station, the group can be scattered into multiple smaller groups or emerged into bigger groups. Moreover, some nodes do not belong to any group [19]. As shown in Figure 9.

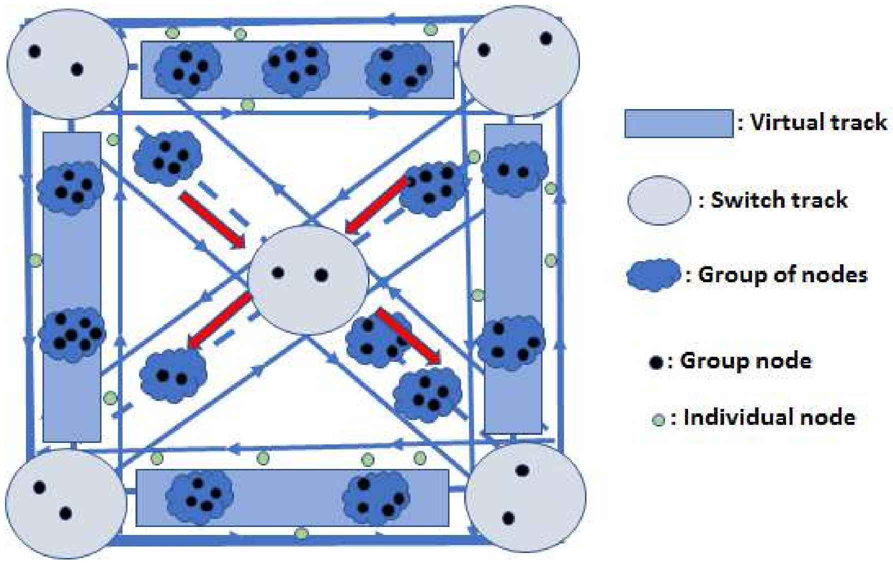


Figure 9. Virtual track mobility model.

Limitation of Group-based mobility models:

- ✓ It does not take into account the problem of density.
- ✓ In reality, the same direction of moving groups cannot be found.

1.4.4.4. Prediction-based mobility models:

This model is based on preempted prediction, for this case, each vehicle memorizes its previous speeds and locations targeting to expect the next future motions. The most known used model is Gauss-Markov based mobility model [20].

The limitation of Prediction-based mobility models is that this model is more realistic than others, but it hides some details such as obstacles and traffic lights which cannot be omitted.

1.4.5. Normalizations and standardization work.

It is worth mentioning the efforts of the automotive industry that developed the Dedicated Short-Range Communication (DSRC) technology, as shown in Figure 11 which enables high-speed direct communication between vehicles and the surrounding infrastructure. DSRC is an IEEE 802.11p-based wireless communication amendment for Wireless Access in Vehicular Environments (WAVE) [3, 4]. As shown in Figure 10. The IEEE 1609.2, 1609.3 and 1609.4 standards are structured for security, network services and multi-channel operation, respectively [21]. These results are advantageous to drivers, passengers and transport authorities such as freeway management, crash prevention and safety, road weather, collision avoidance, and driver assistance [22].

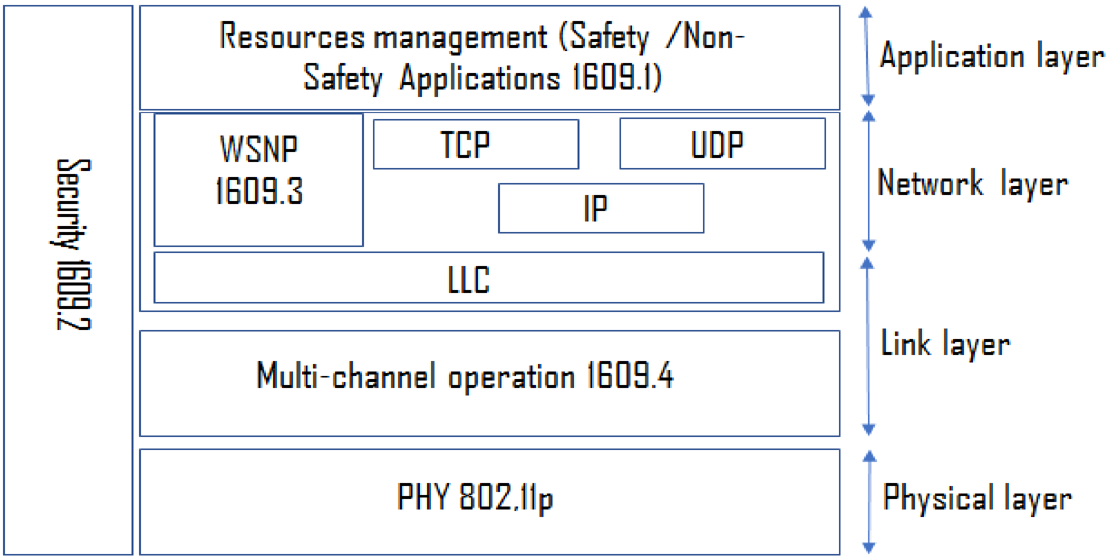


Figure 10. WAVE 1609 model.

IEEE 1609.1: is the standard specifying the services and interfaces of the WAVE Resource Manager application. It describes the data and management services offered within the WAVE architecture. It defines command message formats and the appropriate responses to those messages, data storage formats that must be used by applications to communicate between architecture components, and status and request message formats.

IEEE1609.2: is the standard specifying WAVE security services and describes secure message formats and their processing.

IEEE 1609.3: is the standard specifying WAVE networking service and offers to route and addressing services within a WAVE system.

IEEE 1609.4: is the standard specifying multi-channel wireless radio operations, physical layers and WAVE mode MAC. IEEE 1609.4 defines SCH and CCH interval timers, channel switching, routing parameters, and management services.

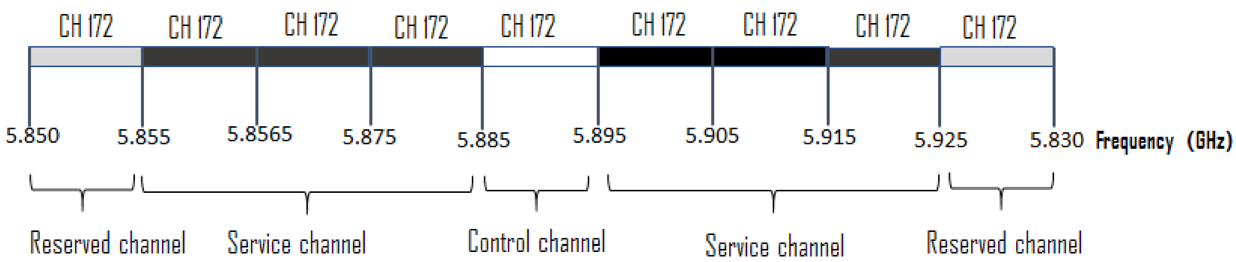


Figure 11. DSRC frequency band.

some DSRC Characteristics could be cited:

1. **Fast Network Acquisition**: Active safety applications require the immediate establishment of communication,
2. **Low Latency**: Active safety applications must transmit messages to each other in milliseconds without delay,
3. **High Reliability when Required**: Active safety applications require a high level of link reliability. DSRC works in high vehicle **speed mobility conditions** and delivers performance despite the **extreme weather conditions** (e.g., rain, fog, snow),

4. **Priority for Safety Applications**: Safety applications on DSRC are given priority over non-safety applications,

5. **Interoperability**: DSRC ensures interoperability, which is the key to the successful deployment of active safety applications, using widely accepted standards. It supports both V2V and V2I communications,

6. **Security and Privacy**: DSRC provides safety message authentication and privacy.

1.5. Problematic

Currently, intelligent vehicles in smart cities are equipped with some devices, such as wireless device On-Board Unit (OBU), radar, camera, localization systems, storage unit, and unlimited energy source. Similarly, RSUs offer high-speed internet access by wireless transmission technologies (LTE, WiMAX, and 5G) [23]. Yet, the systems have been suffering from delay, intermittent connections and packet loss due to the high-speed mobility of vehicles, and the overlapping of radiofrequency in crowded traffic.

Cloud computing is a paradigm for hosting and delivering services [24]; its providers sell computing resources in the form of data centres, servers, data storage, networks and applications through the Internet. In fact. Cloud computing is flexible, scalable and measurable to customers; they pay what they use. At the advent of edge computing, mobile cloud computing (MCC) is a promising technology that can comparatively resolve the inherent problems that appeared with the increasing number of connected devices in classical cloud computing [25]. Subsequently, MCC involves the emergence of VC [26] in which certain vehicles play the role of suppliers, while the others act as consumers. However, Due to the high-speed mobility of vehicles, users in consumer vehicles need a mechanism to discover services in their vicinity. Besides, the quality of service varies from one supplier vehicle to another.

In this study, some techniques are proposed to construct a vehicular cloud; hence, to achieve supply and computation of services in the vehicular networks, to facilitate our task, the following aspects would be taken into consideration:

- In this type of network, the vehicles, which offer their resources as services, represent supplier vehicles.
- The vehicles, which need to consume some services, request them for consumption, represent consumer vehicles.
- RSUs and cluster heads (CH) act as a repository of all offered services for consumption.

1.6. Objectives and contributions

The objectives of the present study are to assess and model the techniques of supply and consumption of services in vehicular networks (also called the vehicular cloud). For different types of applications, These techniques allow QoS to be guaranteed in this type of network.

The objectives of designing service supply and consumption protocols are set. These protocols allow consumer vehicles to discover supplier vehicles services. knowing suppliers services conduct consumer vehicles to send them queries on the required services to consume them. In this context, the contributions can be summarized as follows:

- Identify and study the different approaches and protocols within vehicular networks before establishing a comparative study according to several classification criteria. A state of the art research related to our goal in the area of VANET is established by focusing on the challenges of QoS.

- Propose a novel approach which is called Finding Resources in Vehicular Cloud (FR-VC) that allows a successful registration of services on the vehicular cloud in real-time. The experimental results prove the performance of our proposal in terms of waiting time to register services in the VC.

- Propose a novel protocol named RSU-aided Cluster-based Vehicular Clouds protocol (RCVC) which is the extension of FR-VC that constructs the VC using the RSU and CH directories to make the resources of supplier vehicles more visible. While clusters of vehicles that move on the same road form a mobile cloud, the remaining vehicles form a different cloud on the roadside unit. Furthermore, the consumption operation is achieved via the service selection method which is managed by the CHs and RSUs based on a mathematical model to select the best services. The experimental results prove the performance of our protocol in terms of service discovery and end-to-end delay.

1.7. Thesis structure

This thesis is divided into five chapters including a general introduction and a general conclusion.

Chapter two is entitled Literature review on the supply and consumption of services in vehicular networks. It is mainly devoted to some works that have been done in this direction. These works concern all protocols and approaches that allow some vehicles to offer their resources as services to others that request these services for consumption.

Chapter three, which is entitled Finding Resources in Vehicular Cloud, is dedicated to the presentation of our first contribution, the approach called FR-VC proposed in the context of this thesis. We start with the description of this approach. Then, we analyze its performance by simulations.

Chapter four is entitled RCVC: RSU-aided Cluster-based Vehicular Clouds Architecture for Urban Areas. It deals with the presentation of our second contribution, the protocol called RCVC proposed in the context of this thesis. We start with the description of this protocol. Then, we analyze its performance by simulations compared to other protocols.

The conclusion summarizes the overall proposed approaches and the obtained results, as it unveils the suggestions for further researches and future trends. It provides a synthesis of our contributions and shows the extent to which the objective, aimed at, has been achieved.

Chapter

2 Literature review on the supply and consumption of services in vehicular networks

Content

2.1. Introduction
2.2. Routing for VANET
2.2.1 Overview
2.2.2 Routing classification for VANET
2.3. Cloud computing in general
2.3.1. Smart city
2.3.2. Cloud computing in smart cities
2.4 Vehicular cloud
2.4.1. Definition
2.4.2. Vehicular cloud types
2.4.3. The different mobile vehicular cloud services
2.5. Works-related to the vehicular cloud
2.6. VANETs simulation environments
2.6.1. Software tools for mobility models
2.6.2. Network simulators
2.6.2.1 Integrated Simulators: Veins
2.7. Conclusion

2.1 Introduction

Recently, intelligent vehicles are equipped with wireless communication capabilities, detection devices: sensors, radars, cameras, a positioning system (GPS, Global Positioning System), large storage, processing and unlimited energy capacities. However, these vehicles spend several hours, either in traffic or in parking lots. During this time, their resources are unused. Recently, the idea of offering these vehicle resources as a service to other vehicles has given birth to the concept of the Vehicle Cloud [27].

The vehicular cloud can be considered as a cluster of vehicles that make available their underutilized resources and collaborate with each other to provide services to authorized users [28], [29]. In VC, resources need to analyzed and coordinated dynamically. With VC, resources can be used to provide services cooperatively. Conversely, VC can leverage roadside units (RSUs) networking capabilities to be connected to conventional clouds.

Several research studies have been carried out in the context of VCs. While some researchers proposed a set of protocols, the others had introduced new vehicular cloud architectures. We seize the opportunity to cite some works that have been done in this direction. These works concern all protocols and approaches that allow some vehicles to offer their resources as services to others that request these services for consumption.

The aim of this chapter is to present the review of the work carried out in the field of vehicular Cloud, in which the problematic of the present work is located, then trying to introduce all the concepts related to the vehicular cloud as well as its challenges.

2.2 Routing for VANET

2.2.1 Overview

Moving data over a network can be as simple as sending an email from one computer to another in the same building, or as complex as moving large volumes of data from one part of the world to the other. Routers make decisions about how to convey information using tables showing where other routers are located, and on which various algorithms process for accomplishing a task and making the best decision about which routers and paths make use.

Static network administrators can create route tables manually. Yet, this task is tedious. The only advantage is that the administrator knows the exact path that data takes to get to a destination. This makes static routing tables predictable and manageable. Static routing that works best in small networks [30].

Dynamic Routing is a less tedious way to create a dynamic routing table. Dynamic routing requires each device in a network to broadcast information about its location that other devices are using to update their routing tables. Frequent diffusion keeps the tables up to date. Dynamic routing protocols use different algorithms to help routers refine path selection [31].

The transmission of data packets in VANET is synchronized with beacons. The latter means control message that contains the identification such geographical position and velocity [32]. Two types of beaconing can distinguished:

- **Single-hop beaconing:** As shown in Figure 12**,** single-hop beaconing uses one hop and high transmit power to reach vehicles. Hence, it leads to less congested network and reduce delay, but the distance can limit this process to be efficient [33].

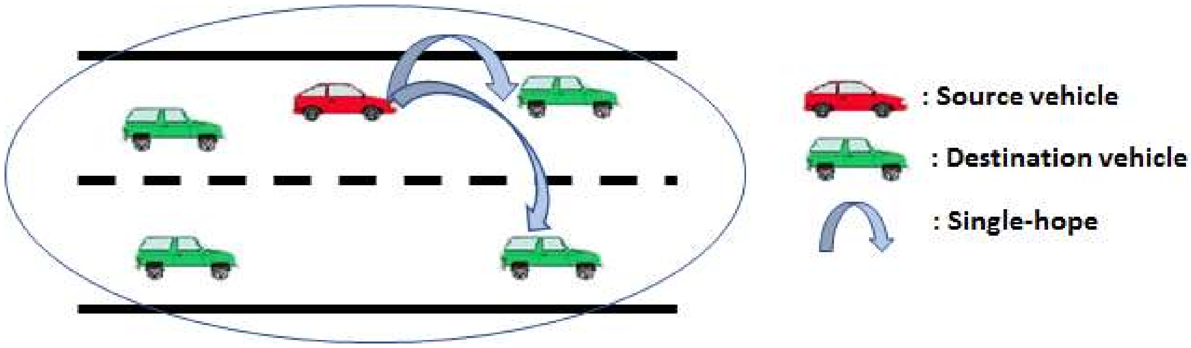


Figure 12. Single-hop beaconing.

- **Multi-hop beaconing:** As shown in Figure 13**,** multi-hop beaconing uses multi relayed vehicles and small transmit power for dissemination. Hence, it leads to high quality transmission, but the network can be very congested [34].

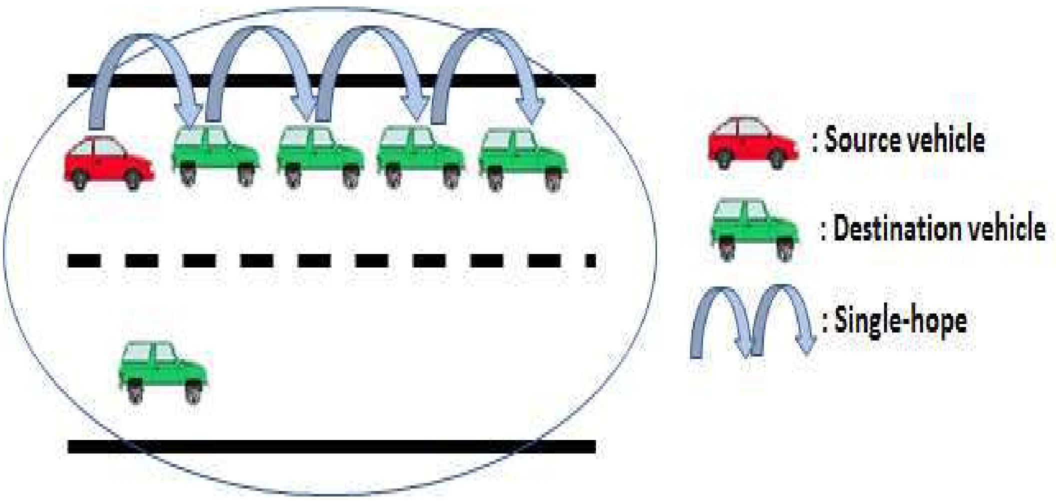
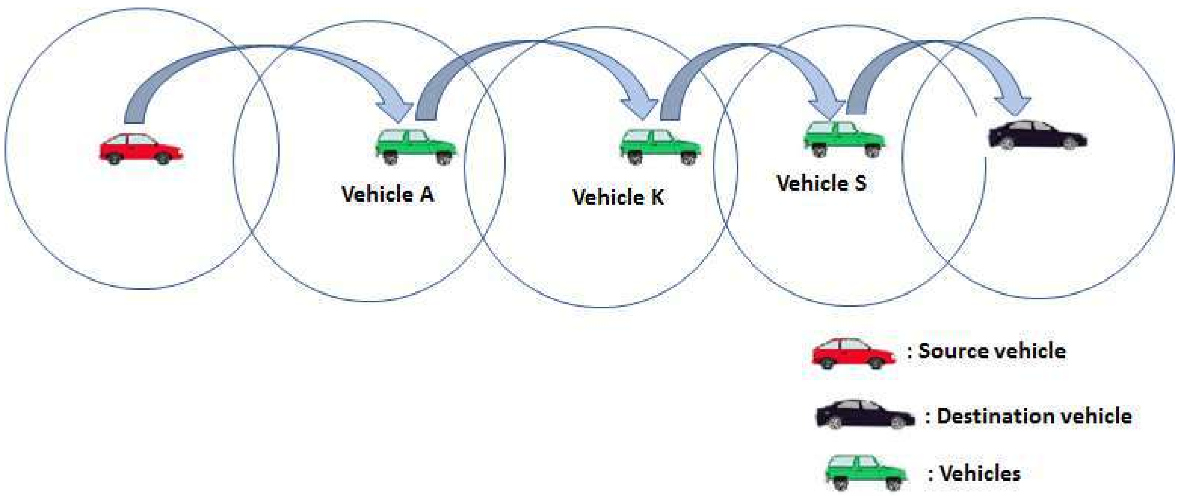


Figure 13. Multi-hop beaconing.

2.2.2 Routing classification for VANET

Three VANET routing categories could be mentioned:

- **Topology-based routing:** it takes into account the topological links between vehicles to discover the existing optimal source-destination route; to achieve this, an address to each vehicle is assigned and a set of control packets is sent through the existing vehicles, as shown in Figure 14. Three types of topology-based routing protocols are found in the literature [35].



Destination	Next hop
Vehicle K	@ Vehicle A
…………	…………
Vehicle S	@ Vehicle A
Routing table of the vehicle	

Destination	Next hop
Vehicle S	@ Vehicle K
…………	…………
Vehicle K	@ Vehicle K
Routing table of the vehicle A	

Figure 14. Topology-based routing protocols.

1. **Proactive routing:** some points of its characteristics are summarized in the following:
   - Each vehicle has paths to other vehicles in routing table.
   - Routing table is updated periodically and it can contain paths to vehicles even if they are needless.
   - Updating of routing table copes with frequent changes of network topology.
   - High mobility of vehicles influences on network topology; this propriety causes rapid topology changes. Consequently, proactive routing can generate and delivery many control packets.
   - Generation and delivery of many control packets consume a significant amount of network bandwidth.
   - Dynamic destination-sequenced distance-victor routing (**DSDV**) and optimized link state routing (**OLSR**) protocols can be cited.

2. **Reactive routing:** some points of its characteristics are summarized in the following:
   - Each vehicle has paths to other vehicles in routing table.
   - Source vehicle discovers paths only when it is to transmit data packet: it is on-demand process.

-Route discovery usually starts by flooding a route request packet through the network.

- When a vehicle is reached, route reply is sent back to the source vehicle using:
  - Link reversal if the route request has traveled on bidirectional links.
  - piggybacking the route-by-route reply packet via flooding.

- Two classes of reactive routing protocols could be distinguished. They are:

✓ **Next-hop routing (hop-by-hop routing):**

- Data packets can reach destination using two addresses in its header: the destination and next-hop addresses.
- Periodic beaconing process is mandatory to ensure neighbour connectivity.
- The next hop vehicle conducts the packet to the following hop and so on until destination.
- The intermediate vehicles keep in their routing tables the next hop to the requested destination.
- Each vehicle update its routing table when they receive the fresher topology information.
- Ad hoc on demand distance vector routing (**AODV**) protocol is one of this class.

✓ **Source routing:**

- The complete-route using all vehicle addresses, which is kept in the data packet header, helps this data packet to reach destination.
- Periodic **beaconing process** neighbour connectivity can be avoided because the transmission is not based on vehicle routing information.
- The generated control packets are reduced (routing overhead).
- The complete route cannot guarantee a right transmission due to the changes of network topology.

- Dynamic source routing (**DSR**) protocol is one of this class.

3. **Hybrid routing:** is to combine proactive and reactive routing [36], some points of its characteristics are summarized in the following:

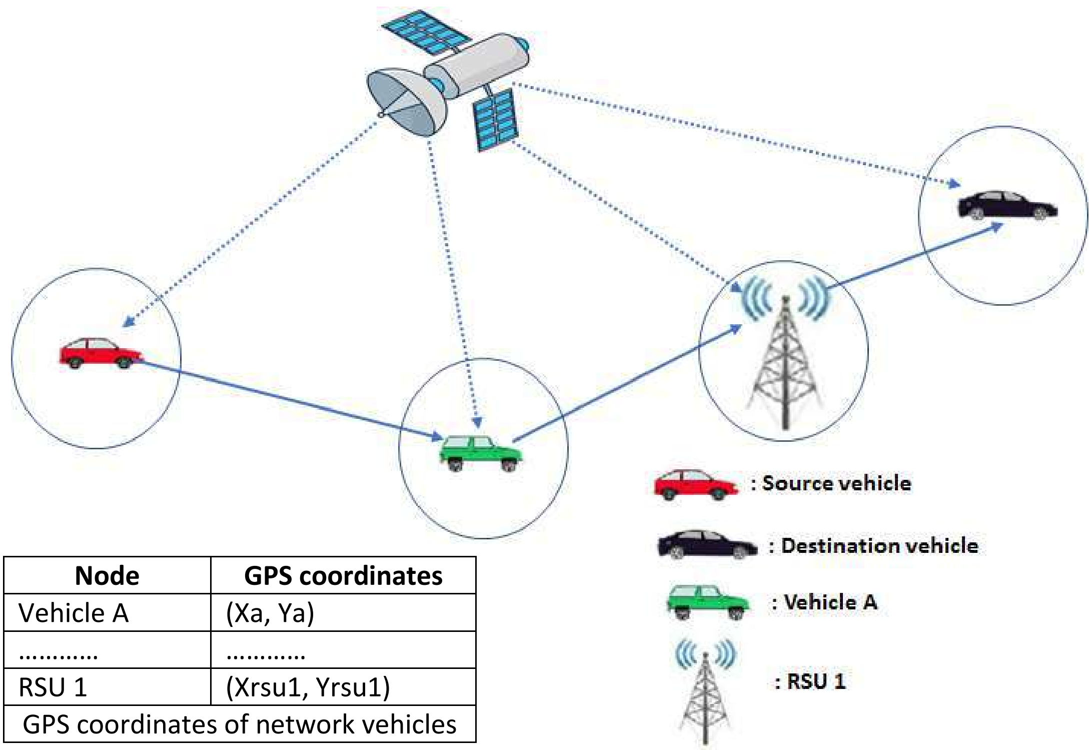
- Proactive routing in a local region is limited by some hops (usually three hops).

- Reactive routing between regions is used.
- The generated control packets are reduced (routing overhead).
- Zone routing (**ZRP**) protocol is one of this class.

- ## Geography-based-routing:

Each vehicle has geographic coordinates in the area [37]. As shown in Figure 15.

- This approach uses geographic coordinates to forward data packets to destination using geographic positioning system (**GPS**).
- The geographic coordinates provide to find the best path to send data packets.
- It does not need to know a prior route discovery.
- Greedy perimeter stateless routing (**GPSR**) protocol is one of this class.



Node	GPS coordinates
Vehicle A	(Xa, Ya)
…………	…………
RSU 1	(Xrsu1, Yrsu1)
GPS coordinates of network vehicles	

Figure 15. Geography-based-routing

- **Cluster-based-routing**

This approach creates the virtual partial network infrastructure (cluster = set of VANET nodes). Each cluster has cluster head which is responsible to forward data packets intra-cluster communication within cluster, and inter-cluster between clusters by direct links using cluster heads [38]. This kind of routing needs a policy to establish clusters by electing the cluster heads. As shown in Figure 15.

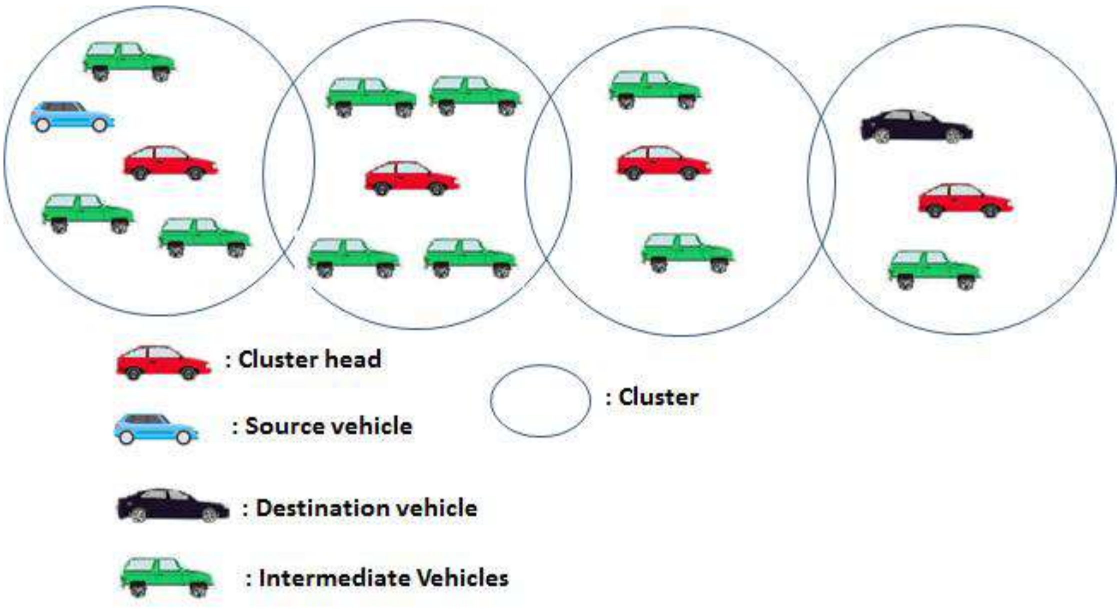


Figure 15. Cluster-based-routing.

According to the recommendation of the International Telecommunication Union Telecommunication Standardization Sector (ITU-T), In VANET routing, QoS is the ability to satisfy certain human and technical factors when disseminating data packets from source to destination [39]. These factors are described as follows:

- The human factors include stability of transmission, availability and correctness of service (packet delivery), and reduced delays.
- Technical factors consist of routing reliability and effectiveness (increased bandwidth and decreased routing overhead).

As a result, QoS in VANET routing ensures timely reception of safety messages and efficient dissemination of non-safety messages like multimedia informative services [40]. In addition, QoS also implies ensuring the successful delivery of the maximum number of transmitted messages (i.e., with a minimum number of dropped packets) where the network bandwidth is optimally used.

- **Quality-of-service criteria:**

There are several criteria or metrics used to evaluate the QoS in VANET routing [41]. For example:

- Average end-to-end delay (measured in milliseconds).
- Average jitter (measured in milliseconds)
- Average available bandwidth (measured in KB/s).
- Packet delivery ratio (it is number).
- Normalized overhead load (it is number).

2.3 Cloud computing in general

Cloud Computing is a general term used to refer to the delivery of resources and services on demand over the Internet. It refers to the storage of and access to data over the Internet rather than through the hard disc of a computer. Thus, it opposes the notion of local storage, consisting of storing data or launching programs from the hard disc [42]. The notion of Cloud should not be confused with that of Network Attached Storage (NAS), used by many companies via a server in residence. These local networks do not fall under the definition of the cloud. However, some NAS allow data to be accessed remotely from the Internet.

In general, speaking of Cloud Computing when it is possible to access data or programs from the Internet, or at least when this data is synchronised with other information on the Internet. To access it, all you need to do is to have an internet connection.

2.3.1 Smart city

A smart city is an urban area that uses various electronic sensors to collect data to provide information to efficiently manage resources and assets [43]. This includes data collected from citizens, mechanical devices and assets that are processed and analysed by monitors. It is important to mention that manage traffic and transportation systems, power plants, water supply networks, waste management, information systems, schools, libraries and finally hospitals are considered as data collected in smart cities.

The smart city concept integrates ICT (information and communication technologies) and various physical devices connected to the network, constituting the Internet of Things, to optimise the efficiency of urban operations and services, and to connect to citizens [44].

2.3.2 Cloud computing in smart cities

The term designates a city equipped with connected objects. Smart cities are rather connected cities, able to perceive their environment thanks to the many sensors that are deployed there. The terms Internet of Things and Big Data are terms that often emerge in research on smart cities [45]. Cloud Computing is often the answer to manage them. The following section summaries and reviews different articles on smart cities, trying to briefly provide new leads.

In the research [46], the authors present how Cloud Computing, the Internet of Things and sensor networks complement one another. Cloud computing enables the optimal use of resources for on-demand operations. Sensor networks and connected objects are a considerable source of data that must be processed and could be helped by Cloud Computing. The latter needs resources that connected objects are able to share when they are not using them. Then, the authors introduce and present the concept of sensing as a Service (SnaaS) and Sensor Event as a Service (SEaaS), also addressed in [47], which are the processes allowing to pass data generated by networks of sensors to an event for the user.

The study [45] shows the necessary relationship between the Internet of Things and Cloud Computing. The authors also recall the heterogeneous nature and the multitude of players involved in the Internet of Things and then present their contribution, which is a framework for a Cloud of sensors.

Then, the authors present their work which is already realised, and show how the two technologies (Cloud and Internet of Things) converge. They point out, however, that this convergence requires that adaptations should be made and that nothing has been proposed in this direction for that moment. They then discuss one of the major challenges in the field, as in many areas of IT: security. Several aspects are explored: virtualisation, resource control or privacy control.

The concept that the authors propose is a platform which is based on the virtualisation of infrastructures. It combines several services: Network as a Service, Platform as a Service and Sensing as a Service.

In article [48], the authors introduce their work by explaining that the IoT is booming. Between 2010 and 2013, the amount of information stored had increased; this growth is exponential.

The authors then discuss the interests of Big Data, particularly economic, and then discuss the technologies that make it possible to process these large amounts of data (NoSQL for example). The challenges surrounding the management of BigData are then set out, such as optimising requests or reducing energy consumption. The article looks like a big data mini tutorial. In the end, the authors evoke and discuss the association of BigData with Cloud Computing, transforming the Internet of Things into Sensing as a Service or rather into XaaS: Whatever it is as a service.

The authors recall in their study [49] that the world's population is increasingly focusing on urban spaces, contributing to the increase in digital data traffic. To manage these (collection and use), two paradigms are associated: Cloud Computing and the Internet of Things, which gives rise to the Cloud of Things (CoT). At the beginning of the article, different platforms for manipulating objects are presented. The trend is to develop middleware specialising in one area; to bridge the gap between a network of one type of object and a more global platform.

The study [50] presents a form of network of sensors called "partcipory sensing" (PS), which is translated as "participatory perception". This form of network involves the individual and smartphone sensors to record environmental parameters.

The authors present a platform called TQMS (Trip Quality Measurement System), which aims to measure the quality of an individual's journey. They list a set of areas of application to PS: transport, digital environmental survey, public health, urban safety, economics, education/research. Immediately afterward, the authors have introduced the challenges around the PS: security (in the IT sense of the term), motivation (motivation to participate), data quality.

In article [51], the authors explain on the one hand that smart cities and the VANET can improve the quality and safety of transport. They also assume that if one node in the network is not individually reliable, the entire network is reliable through node redundancy. Finally, they announce the data propagation method they propose: combining smartphone and vehicle, using the smartphone for long-distance communication.

They introduce, in their model, the notion of the small world. These are linked together through smartphones and are able to build routes for inter-vehicle communication over long distances.

In their proposal, they assume that the phone allows long-distance communication; actually, it is the connection to the cellular network that allows it. In European projects, the boxes in vehicles are able to communicate on this channel, and the smartphone is no longer necessary. In practice, the phone in a vehicle only serves as a graphical interface.

To conclude, if a definition of the smart city is to be provided, then it would be the organization of a city's sensors to offer services to users. The nature of the services offered is multiple, even limitless, the reason why, sometimes, the term XaaS is found. This means that anything can be a service.

2.4 Vehicular cloud

The Vehicular Cloud (VC) is the ultimate convergence between the concept of cloud computing and vehicular networks for service provisioning and management [52]. With this approach, vehicles can be connected to the cloud, where a multitude of services is available, or they can also be service providers. This is possible due to the variety of resources available in vehicles: computing, bandwidth, storage and sensors.

2.4.1 Definition

Vehicle Cloud Computing is a type of Mobile Cloud Computing. It can be seen in two ways. The first way is the provision of services dedicated to vehicles, on a remote Cloud. The second way of approaching Cloud Computing is to transform vehicles into suppliers and consumers of services. Thus, the vehicle shares its resources or associates with other vehicles to produce data; this represents our future work of Chapter 3 and 4.

2.4.2 Vehicular cloud types

There are two classes of vehicular cloud:

- **Stationary vehicular cloud (SVC):** this is the case where a set of vehicles parked in the parking of a company or any place, supplier vehicles can form a stationary vehicular cloud and consumer vehicles parked in the same parking, can consume these services.
- **Mobile vehicular cloud (MVC)**: where the supply and consumption of services takes place in a purely mobile environment. As the focus of the present research is on this type of vehicular cloud, it would be detailed in the following sections.

The authors in [53] have proposed another classification of vehicular cloud, namely the nature of consumption of services. For example, if the supply and consumption of services is done through V2V communication, then it is a CAV (Cloud Ad hoc Vehicular). If they are done through the classic cloud, it is VuC (Vehicles use Cloud), where the RSUs act as gateways to the Cloud. The hybrid architecture is the combination of VAC and VuC. In this case, vehicles can consume services either from other vehicles or from classic Cloud Computing through RSUs.

2.4.3 The different mobile vehicular cloud services

A supplier vehicle may offer its resources to neighbouring vehicles according to three fundamental types of services [54] with their attributes such as:

- Network as a Service (NaaS): it means to offer the extra bandwidth in terms of the Internet to others for a certain fee.
- Storage as a Service (SaaS): it means to offer storage for a certain fee.
- Data as a Service (DaaS): it means offering the data like books, city maps, and video files for a certain fee too.

2.5 Works-related to the vehicular cloud

Several research studies have been carried out in the context of VCs. While some researchers proposed a set of protocols, the others had introduced new vehicular cloud architectures. We seize the opportunity to cite some works that have been done in this direction. These works concern all protocols and approaches that allow some vehicles to offer their resources as services to others that request these services for consumption.

Mershad et al. [55] implemented a disCoveRing and cOnsuming services WithiN vehicular clouds protocol (CROWN), where they considered an urban scenario and suggested a set of vehicles as providers, known as STARs, and the remaining ones as consumers. The STARs register their resources in the nearest RSUs. Then, once the consumer vehicles seek some services, they request RSUs for their needs; the latter search in their directories and respond to consumer vehicles by informing them of the most suitable STARs.

Brik et al. [56] made an important push-forward toward finding a public bus to rent out services in vehicular clouds. The authors designed a Discovering and Consuming Cloud Services in Vehicular Clouds protocol (DCCS-VC) that finds adequate public buses and uses them as cloud directories to facilitate the discovery of supplier vehicles' services. The proposed protocol is based on the Grid-based Tracking Cell technique (GTC) that allows partitioning the route of each bus group into several cells. The cells are grouped in tracking bus path (TBP), and each TBP has a unique identification (TBPI). Then, once a supplier vehicle registers a service on a public bus, it will be informed by the corresponding TBPI. The authors in [57], [58] improved the DCCS-VC, where they added new research by proposing a protocol named Fast Discovering and Consuming Cloud Services in Vehicular Cloud (FDCCS-VC). Some enhancements have been made allowing vehicles to share all public bus routes and schedules.

Arkian, et al. [59], [60] proposed a Fuzzy Clustering-based Vehicular Cloud Architecture (FcVcA), where they introduced the concept of clustering in the vehicular cloud. The election of the cluster head (CH) is done using a fuzzy logic algorithm. The CH manages the cloud in real-time in terms of creation and maintenance. When a consumer vehicle wants to consume services, it sends a request to the CH which uses Q-learning to provide it with the adequate service provider vehicle. It is worth noting that the proposed protocol was implemented in a highway scenario.

Azizian, et al. [61] introduced an optimization vehicular cloud model. They used a distributed D-hop cluster formation algorithm (DHCV) to organize vehicles into clusters without overlapping. All clusters are considered as clouds and brokers (cloud coordinator), and the vehicle chooses its broker according to mobility calculations proportional to its neighbors D-hop. This model uses a mathematical optimization-scheduling algorithm to maximize bandwidth and minimize delays.

Hussain, et al. [62] implemented a Vehicle Witnesses as a Service (VWaaS) protocol that employs some vehicles which assist as witnesses using their cameras to build a VANET-cloud service. This protocol offers the use of the installed cameras on vehicles as well as roadside cameras to provide pictorial services to other entities (e.g., VANET users and/or law obligation). The event's occurrence allows some vehicles to serve as witnesses, and provides legal confirmation to law obligation for ex-amination.

Lin, et al. [63] proposed Gateway as a Service (GaaS) protocol, which includes three levels. The first is a gateway, which can be RSU or vehicle, connects vehicles to the Internet. The second is the consumer vehicle that requests internet access, while the third is the vehicular cloud which provides two sub-repositories which are:

- GaaS register: it is to save all data on the gateways.
- GaaS dispatcher: it is to send the associated gateways for consumer vehicles.

Jafari et al. [64] introduced a Two-level cOntroller-based aPproach for serVice advertISement and discOveRy in vehicular cloud network (TOPVISOR). They implemented the proposed scheme in a vehicular cloud network. This approach is based on two central controllers which are connected to the distributed directories of roadside units; the information discovery with its registration is achieved in a hierarchical proactive manner at the controller levels.

However, all previous studies used to focus on improving few performance metrics, whereas other metrics were neglected. Therefore, we proposed a protocol that finds compromises by negotiating all metrics, especially those that have a strong relationship with decreasing delays in urban environments.

2.6 VANETs simulation environments

2.6.1 Software tools for mobility models

Mobility models are generated by using software tools. There are several tools; the most important between them namely Simulation of Urban MObility (SUMO), VanetMobiSim, and MOVE could be cited.

- **SUMO framework:** SUMO is a free and open source traffic simulation. It allows modelling traffic systems, including road vehicles, public transport and pedestrians. The execution and evaluation of traffic simulations, such as network import, route calculations and visualisation. SUMO can be enhanced with custom models and provides various APIs to remotely control the simulation [65].
- **VanetMobiSim:** A free generator of realistic vehicular motion traces for network simulators. The traces generated by VanetMobiSim are validated first by illustrating how the interaction between featured motion constraints and traffic generator models is able to reproduce typical phenomena of vehicular traffic. This makes VanetMobiSim one of the few vehicular mobility simulators fully validated and freely available to the vehicular networks research community [66].
- **MOVE framework:** MOVE allows users to generate rapidly realistic mobility models for VANET simulations. This software provides, as a result, a trace file of a realistic mobility model which can be immediately used by frequent network simulators, such as ns-2 and qualnet. Two main entities that composed MOVE are the map editor and the vehicle movement editor. The Map Editor allows the road topology design which can be manually created by or by Google Earth [67].

2.6.2 Network simulators

The most famous simulator are:

- **OMNeT++ (Objective Modular Network)** : is an open-source discrete event simulation tool written in C ++. OMNeT ++ is a modular simulation space based on components, with hierarchical modules. Its main field of application is that of communication networks. OMNeT ++ based on the object-oriented concept which allows developing simulation scenarios using the set of tools and modules that it offers [68].

- **Ns-2:** Ns-2 is a discrete event simulator for network research. Ns-2 provides substantial support for simulating TCP, routing and multicast protocols over wired and wireless networks (local and satellite) [69].
- **ns-3**: ns-3 is a discrete event network simulator for Internet systems, primarily for research and education. ns-3 is free software, released under the GNU GPLv2 license, and is publicly available for research, development, and use [70].

2.6.2.1 Integrated Simulators: Veins

VEINS [71] is an open source framework which allows to run simulations of VANET vehicular ad-hoc networks. It is based on two simulators which are OMNeT ++ and SUMO. Its architecture is illustrated in Figure 17. The two simulators are combined to provide a complete platform of models for the simulation of VANETs. Road traffic simulation is carried out by SUMO, which is designated for the field of traffic engineering. The simulation of network communications is carried out by OMNeT ++ with the modeling of the physical layer thanks to the MIXIM models, which makes it possible to use precise models for radio interference, as well as observation by fixed obstacles and mobile. The two simulators are interconnected and simulations are performed. In this way, the influence of vehicle networks on road traffic can be modelled and complex interactions between the two simulators are possible through a connection via a TCP socket.

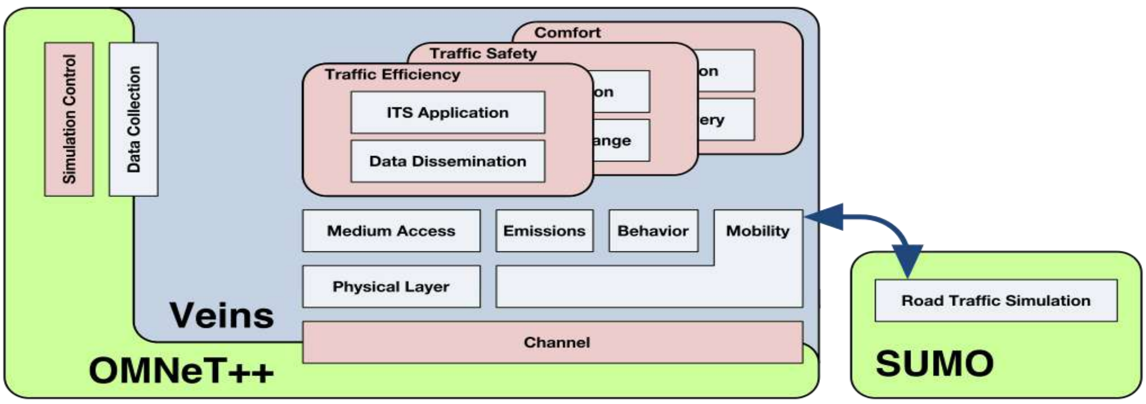


Figure 17. Veins Architecture[1].

[1] http://veins.car2x.org/documentation/

The framework Veins contains the IEEE 802.11p and IEEE 1609.4 protocols which are fully supported. It is based on fully detailed models of IEEE 802.11p and IEEE 1609.4 DSRC / WAVE network layers, with support for multichannel, noise and interference effects.

2.7 Conclusion

As a promising topic of research, Vehicular Cloud incorporates cloud computing and ad-hoc vehicular network. In VC, supplier vehicles provide their services to consumer vehicles in real-time. These services have a significant impact on the applications of internet access, storage and data. Due to the high-speed mobility of vehicles, users in consumer vehicles need a mechanism to discover services in their vicinity. Besides, quality of service varies from one supplier vehicle to another, thus, consumer vehicles attempt to pick out the most appropriate services. In this study, we have conducted extensive and detailed research into the vehicle cloud. This concept has become a very attractive field of research to exploit the resources of vehicles, which we presented in the previous section,

In this context, our research objective is twofold. First, design and propose a service supply and consumption protocol in the CAV by optimizing service discovery and consumption times. Second, to provide a service selection technique inside the mobile vehicular cloud that is based on a mathematical model which is managed by the CHs and RSUs based on a mathematical model to select the best services.

Chapter

3 Finding Resources in Vehicular Cloud (FR-VC)

Content

3.1. Introduction
3.2. FR-VC approach architecture
  3.2.1. VC creation at the cluster head level
    3.2.1.1. Cluster creation
  3.2.2 VC creation at the RSU level
3.3. Performance and validation
  3.3.1. Simulation setup and parameters
  3.3.2. Result and discussions
3.4. Conclusion et perspectives

3.1 Introduction

In recent years, sharing resources through the cloud-computing model has experienced different interactions with several fields of scientific research. After the ad hoc mobile network emerging. Particularly, ad hoc vehicular networks, cloud computing has been presented as one of the technologies that help information exchange in a mobile environment. Mixing the cloud-computing concept with an ad hoc vehicle network produces a vehicle cloud (VC). It also provides real-time services to consumer vehicles via supplier vehicles. These services have a considerable impact on road safety applications, improving driving, and serve advertisements. Due to the vehicle density in urban areas and the overlapping of transmission ranges, vehicles need a mechanism on their sensors providing an efficient collaboration.

In this chapter, we suggest a novel approach (FR-VC) that allows a successful registration of services on the vehicular cloud in real time [72]. The experimental results prove the performance of our proposal in terms of waiting time to register services in the VC.

3.2 FR-VC approach architecture

We designed the FR-VC approach termed Finding Resources in Vehicular Cloud considering the MANHATTAN mobility model [15] as a map, where all vehicles in that area move randomly. RSUs are chosen near to the junctions.

3.2.1 VC creation at the cluster head level

The vehicles communicate directly with each other via their OBU antennas or with the nearest RSUs. The RSUs and the routing protocol used by vehicles make it possible to build a set of clusters if they receive the services offered or the requests according to the vehicle needs of consumers who are on the same trajectory. These RSUs should only establish a cluster when they receive a set of registration or request packets in its neighborhood or via a routing protocol. Furthermore, to establish a cluster, it must elect a cluster head that will be capable to control the cluster and respond to the consumer vehicles' demands, which are referring to this cluster. The remaining of the vehicles that do not meet the circumstance mentioned above must register their services in the most adjacent RSU if they are supplier vehicles, or send their requests if they are general public vehicles. Nevertheless, with a random delay value (dly) between [0.05, 0.1] seconds, all RSUs are linked between them by a wired transmission, which are delayed by this value.

When a supplier vehicle desires to offer its resources to other vehicles, it sends a REGISTRATION PACKET to the closest RSU, which contains:

- The ID of the vehicle.
- The average speed of the vehicle.
- The direction of the vehicle.
- The geographical coordinates of supplier vehicle.
- The time to leave the VANET Tl.
- The offered resources.

Sending a REGISTRATION PACKET to the RSU from the supplier vehicle (its location is R1) uses the routing protocol. We used the geography-based routing such as Enhancement of Greedy Perimeter Stateless Routing Protocol (E-GPSR) [73], this kind of routing uses the positions to forward packets from the destination node to the source node and this is our case (we use the coordinates of vehicles).

A consumer vehicle that wants to consume certain services can use the services of one or more nearby supplier vehicles by sending a REQUEST PACKET to the closest RSU, which contains:

- The ID of the vehicle.
- The average speed of the vehicle.
- The direction of the vehicle.
- The geographical coordinates of the vehicle.
- The requested resources.

Accordingly, RSU's collection of registration and request packets launches certain processes to establish a cluster and pick a cluster head or build a directory at its level. Meanwhile, the holding of the first packet, being it a registration packet or a request packet. The RSU starts a clock, Then the sending process of packets is consecutive until the clock ends to a certain threshold Tth for example thirty (90) seconds (which we haven't chosen it randomly, but it comes down to several simulation attempts), some actions are launched.

- **RSU level action**

This task is called a discovery phase which enables all resources to be saved on the RSU directories or to create cluster head directories; we outline these activities as follows:

➢ Reset clock to zero.

➢ Put all packets on a list (ListV).

➢ If ListV contains only supplier vehicles, the RSU creates its proper directory (because we do not need to create a cluster that will only contain similar vehicles, and it will have no offer or consumption operations in the cluster), this is the second case of RSU-based vehicular cloud.

➢ If ListV contains only consumer vehicles, the RSU creates its proper directory (same reason mentioned above), this is the second case of RSU-based vehicular cloud.

➢ If ListV contains only one supplier vehicle or one consumer vehicle, the RSU creates its proper directory, and this is the second case of RSU-based vehicular cloud.

➢ If ListV contains at least one supplier vehicle or one consumer vehicle, the RSU creates a cluster and selects its cluster head (CH) (we will see it later).

Delete ListV's items in DR after creating the cluster.

Algorithm 1 shows the process of receiving the registration and request packets at RSU level to create vehicular clouds.

Algorithm 1 FR-VC algorithm to create the vehicular clouds

```
begin
Packets_reception();
if Simulation time = Tth
    begin
        if (ListV is suitable)
          begin
              Create_mobile_cluster_vehicular_cloud();
              Reset(ListV);
          end if
        else
              Create_RSU_vehicular cloud();
        end else
end if
else
 Packets_reception();
end else
end.
```

3.2.1.1 Cluster creation

FR-VC establishes clusters taking into account two manner of creation; we explain them in the following:

- ✓ **Autonomous creation**

In this case, RSUs do not react to create clusters but by exchanging beacons between vehicles each every 2 seconds allows some of vehicles recognize which others that go on the same direction and can form clusters [74]. The beacon contain this information:

- The ID of the vehicle.
- The average speed of the vehicle.
- The direction of the vehicle.

Once all vehicles that belong to the same cluster achieve the recognition, each vehicle uses the Euclidian distance (it will be explained after) to select the cluster head (CH) that will be able to manage the consumption between vehicles.

✓ **Cluster creation at the RSU level**

The second case, we defined what is called a cluster section, by dividing all routes on the map into sections allowing building clusters. With this technic, RSU can recognize the vehicles that belong to the same cluster section, and proceed to create the cluster. Each cluster section has a length Ls and a width Ws (as shown in Fig.1). Vehicles that circulate on the roads have the coordinates in this plan. So, to select a CHj, we used the Euclidean distance [75], taking as parameters the coordinates of the vehicles Vi (*Xi*, *Yi*) and their average speeds *AVi*,

Now we have a point P (*Xj*, *Yj*). To obtain the coordinates of the vehicle that acts as a cluster head in the section Sj taking the average speed as a factor, we calculated the distance *Dist* between all vehicles and P. The vehicle that has a shorter distance will be a CHj. As shown in Figure 18.

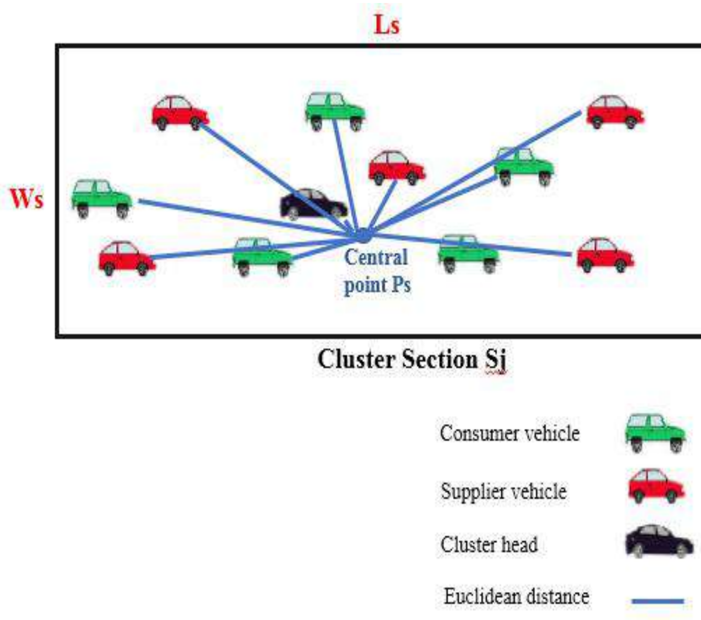


Figure 18. Cluster creation

The RSUs transmit to all vehicles, which will be on the same cluster for this creation by broadcasting the CLUSTER PACKET. The cluster packet contains the ID of the cluster head and all information about vehicles that are belonging to the same cluster.

The role of the CH is to control the operation of consumption between vehicles. Therefore, to optimally scheduling the queues of consumers. Now, the CH knows all consumer vehicles, and then it searches in its list that is received from RSU during the creation, it returns by a RESPONSE PACKET that holds the ID of supplier vehicles and their resources.

The consumer vehicle designates the needed resources via transmitting a SERVICE PACKET to supplier vehicles using the routing protocol. The service packet includes the identification of the vehicle and its requests, upon receiving of the vehicle service packet, the supplier vehicle answers either with a NEGATIVE ACKNOWLEDGMENT if it cannot meet the user's request or its waiting line is busy or with a SERVICE RESPONSE packet, inviting the consumer vehicle for the method of payment. The user from consumer vehicles and the supplier vehicles can negotiate data packets comparing to the resources.

3.2.2 VC creation at the RSU level

The packets collection on RSU directory RD of all information, which contain the ID, resources with their attributes, and Tl determines the influence area of each supplier vehicle Ai [55]. The supplier vehicle can offer certainly its resources to consumers and transfers a registration packet to all RSUs that are in Ai to add registration packet to their RD enabling them to identify the supplier vehicle and their resources, which belong on Ai. If the supplier vehicle goes to another RSU and is nearest than others, Ai will change its value and send R2 (for example) another registration packet. Additionally, this latest adds (updates) this packet to its RD and forwards it to all RSU within on Ai, for appending their RD.

The supplier vehicle, which does not belong to a cluster, it periodically transmits a beacon to the closest RSU (every 2 seconds). These beacons empower RSUs to follow the supplier vehicle's travel. While the beacons are delayed to arrive at RSU, it seeks to identify the position of the supplier vehicle's area, which is called estimated area Ae. With this formula, we can measure the radius Re of $Ae = AV * (Tc - T1)$, where T1 the time of sending the last beacon, Tc current time and AV is the average speed of the supplier vehicle. The RSU sends to all consumers the area Ae. It is meriting to note that the RSU will remove the ID of the supplier vehicle in its RD if does not receive beacons from it for a time X = 10 * (interval beacon).

When a vehicle that does not belong to a cluster needs to consume some services. It formulates a REQUEST PACKET and sends it to its nearest RSU. Besides, to the desired resources and their attributes, the request packet contains the vehicle's geographic coordinates. Wherever, if the RSU receives a request packet from the consumer vehicle, it searches its DR for one or more supplier vehicles that can face the vehicle's requirements.

The RSU that responds to a user's request will choose the corresponding supplier vehicle (s) from the candidate list Lc of supplier vehicles. Then, the RSU will formulate a RESPONSE PACKET, which contains the following elements for each supplier vehicle chosen by the RSU:

- The ID of the supplier vehicle.
- Latest supplier vehicle's locations from the latest beacon.
- Resources and their attributes.

The consumer vehicle defines the resources and their attributes that it requires from each supplier vehicle. Then it sends a SERVICE PACKET for each supplier vehicle picked using a routing protocol. The service packet includes the consumer vehicle's ID and their requests. Upon acquisition of the consumer vehicle's service packet, the supplier vehicle responds either with a NEGATIVE ACKNOWLEDGMENT if it cannot provide the user's request or its waiting line is busy, or with a SERVICE RESPONSE packet otherwise, by requesting the consumer vehicle for the payment method. The consumer and supplier vehicles can interchange data packets regarding the resources.

3.3 Performance and validation

3.3.1 Simulation setup and parameters

We used OMNeT++ 5.3 [68] for the behavioural aspect and Sumo-0.32.0 [65] as a mobility simulator to simulate FR-VC. FR-VC is implemented in MANHATTEN grid 9 x 9, where we put 72 RSUs near to the junctions. The vehicle density (VD) is varied between 100 and 500 vehicles; each scenario of simulation is jointed with the value of VD (100, 300, 400, and 500). The details of the simulation parameters are shown in Table 1.

Table 1. Simulation parameters.

Parameter	Value
Simulation framework	Veins (OMNet++ and Sumo)
Mobility model	Manhattan
Simulation time	1000 s
Simulation area	9x9 km^2
Transmission range	500 m
Transfer rate	18 Mb/s
Vehicle density	[100-500] vehicles
Vehicles speed	Up to 70 km/h
Supplier vehicle density	1/4, 1/3 and 1/2 of vehicle density
The size of registration and request packet	128 Kbytes
Data Packet Size	[1-5] Kbytes
Supplier vehicle's queue	5 consumer vehicles

FR-VC is evaluated by taking these metrics:

1- Wait Time to Register (W_time): measures the time from sending the registration packet by supplier vehicle to be stored in VC (second).

2- Hit Percentage (HIT): percent of requested services, which is acknowledged by the RSU (%).

3- Successful SVs registration rate (SUC): percent of successful registration packets at the RSU (%)

3.3.2 Result and discussions

To prove the performance of FR-VC, we performed an extensive comparison in terms of W_time, HIT and SUC taking the same simulation parameters against three protocols, which are: CROWN (2013), DCCS-VC (2018), and TOPVISOR (2020).

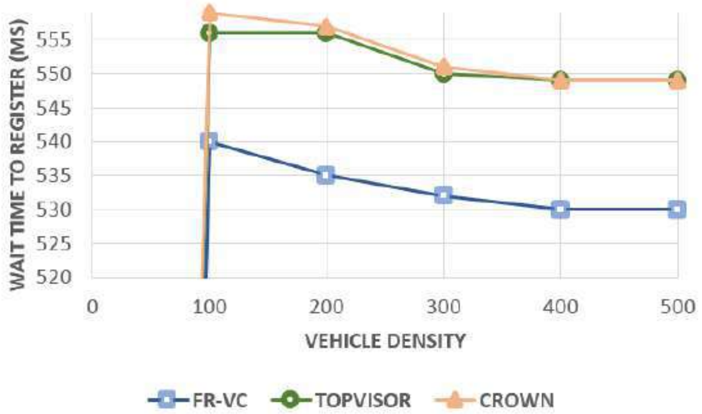


Figure 19. Performance evaluation comparisons of FR-VC with CROWN and TOPVISOR in terms of W_time.

As shown in Figure 19, FR-VC yielded better results than CROWN and TOPVISOR in terms of W_time. Compared to CROWN protocol, the outcome is justified by the use of the routing protocol E-GPSR in FR-VC that proves its efficiency, E-GPSR can deliver packets to reach RSUs better than CAN DELIVER [76] which is used in CROWN. Regarding the TOPVISOR it's clear that it uses two levels to store packets at the VC, which are the RSUs and the central controller, not like FR-VC that uses only RSUs to store packets.

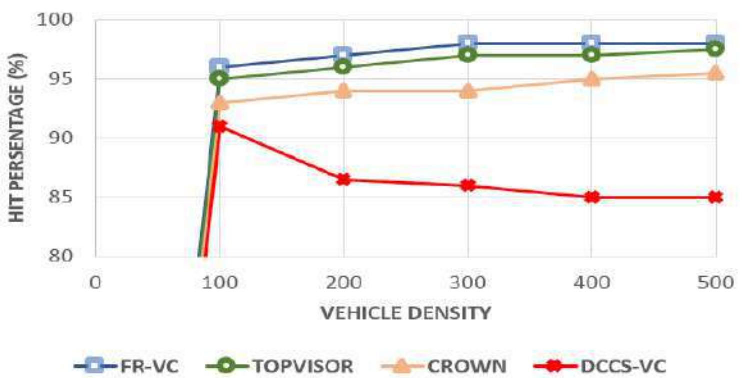


Figure 20. Performance evaluation comparisons of FR-VC with CROWN, DCCS-VC and TOPVISOR in terms of HIT.

As shown in Figure 20, CROWN and TOPVISOR protocols have ascending slopes except for the DCCS-VC protocol, which has a descending slope. By varying the number of vehicles (100, 200, 300 and over 400), FR-VC performed better than CROWN, DCCS-VC and TOPVISOR in terms of HIT, the reason behind this best outcome can be substantiated by the non-correlation between the VD and this metric. It has a strong relationship with the critical location of RSUs in the map, where FR-VC chooses to deploy them near to junctions.

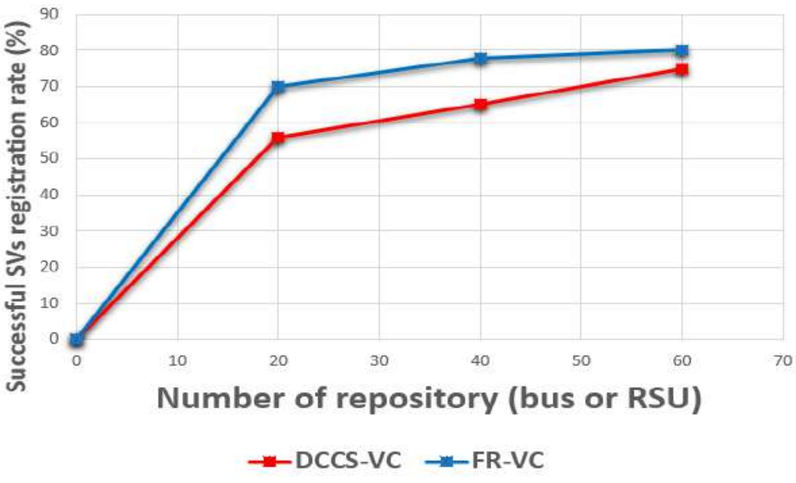


Figure 21. Performance evaluation comparisons of FR-VC with DCCS-VC in terms of SUC.

The figure 21 demonstrates that when we increase the number of deployment of RSUs the SUC increases, this outcome can be justified by the existence of more RSUs on the map the SVs have more the chance to register their offers.

Comparing the slope of FR-VC with DCCS-VC according to the successful SVs registration rate. FR-VC performed better then DCCS-VC, this result can be explained by the mobility of the repositories which are the buses that might influence on the successful registration of packets, considering the locations of RSUs that are closest to the junctions, this fact may help VCs to register their offers easily.

3.4 Conclusion

In this chapter, we proposed a new approach to register supplier vehicles services reliably. The clustering-assisted technique has proved an efficient performance for the successful registration of services in the vehicular cloud. The experiment results of our approach indicated some enhancements in terms of wait time to register, hit percentage and successful registration ratio metrics. In future work, we will try to extend FR-VC by adding a mathematical method inside the VC to select the best services, hence investigating other clustering techniques for effective consumption of services.

Chapter 4

RSU-aided Cluster-based Vehicular Clouds architecture for urban Areas (RCVC)

Content

4.1. Introduction
4.2. RCVC protocol architecture
  4.2.1. Mobile cluster-based vehicular cloud
  4.2.2 VC creation at the RSU level
4.3. A mathematical model for service selection
  4.3.1. Normalization
  4.3.2. Performance score
4.4 Experimental analysis
  4.4.1. Simulation setup
  4.4.2. Results and discussions
4.4. Conclusion

4.1. Introduction

In this chapter, we propose a novel protocol named RSU-aided Cluster-based Vehicular Clouds protocol (RCVC) which constructs the VC using the Road Side Unit (RSU) directory and Cluster Head (CH) directory to make the resources of supplier vehicles more visible [77]. While clusters of vehicles that move on the same road form a mobile cloud, the remaining vehicles form a different cloud on the road side unit. Furthermore, the consumption operation is achieved via the service selection method which is managed by the CHs and RSUs based on a mathematical model to select the best services. Simulation results prove the effectiveness of our protocol in terms of service discovery and end-to-end delay, where we achieved service discovery and end-to-end delay of 3x10-3s and 13x10-2s, respectively. Moreover, we carried out an experimental comparison, revealing that the proposed method outperformed several states of the art protocols.

4.2. RCVC protocol architecture

We designed RCVC taking into consideration the MANHATTAN mobility model [15] as an environment where all vehicles are moving randomly in this area. RSUs are chosen near junctions and are also installed uniformly on the board.

All vehicles can exchange messages between them or with the nearest RSUs via their OBU antennas. RSU range helps vehicles to construct a set of clusters if they receive offers and consumptions services according to their needs. Note that such RSU can create a cluster only when it receives a set of registration or request packets. Some vehicles going in the same direction, where there are SVs and CVs that have stable links with the CH can form a cluster, as it is shown in Figure 22. Therefore, based on this concept, to create a cluster, it is mandatory to select a CH that will be able to manage the cluster and responds to all requests, which are belonging to the cluster. The rest of the vehicles, which do not fulfill the condition mentioned above, must register their services in the closest RSU if they are SVs but if they are CVs, they must send their requests. However, a wired transmission connects all RSUs with delay between [0.05, 0.1] seconds.

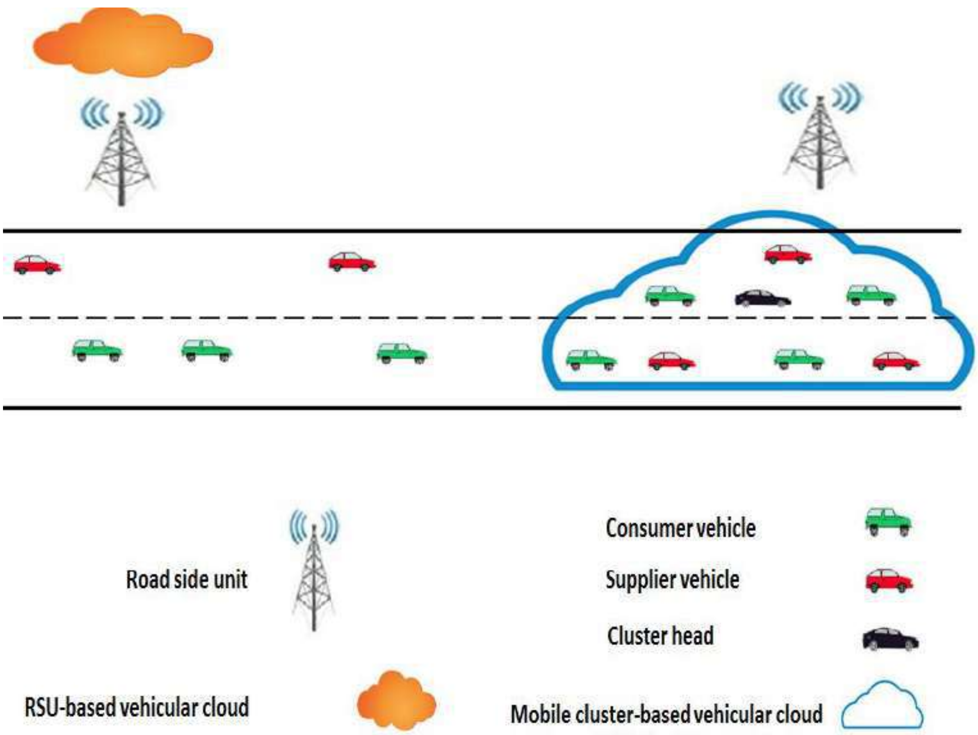


Figure 22. Vehicular cloud types in RCVC.

4.2.1. Mobile cluster-based vehicular cloud

Once a smart vehicle wants to offer its resources to other vehicles, it sends a registration packet (RGP) to the nearest RSU. If a user in an intelligent vehicle needs an internet connection, requires additional resources that his vehicle does not have or he is interested in certain data, he uses the services of one or more nearby SVs. The user's vehicle formulates a request packet (REQP) and sends it to the nearest RSU. The format of the registration and request packets are shown in Figure 23.

0 … 15	16	18 … 31
Vehicle ID	Dir.	Average speed
Geographical latitude		
Geographical longitude		
$Bandwidth_{Naas}$		
$Duration\_bandwidth_{NaaS}$		
$Cost_{NaaS}$		
$Storage_{SaaS}$		
$Duration\_Storage_{SaaS}$		
$Days_{SaaS}$		
$Cost_{SaaS}$		
$Data_{DaaS}$		
$Cost_{DaaS}$		

Figure 23. RCVC Registration/Request packets format.

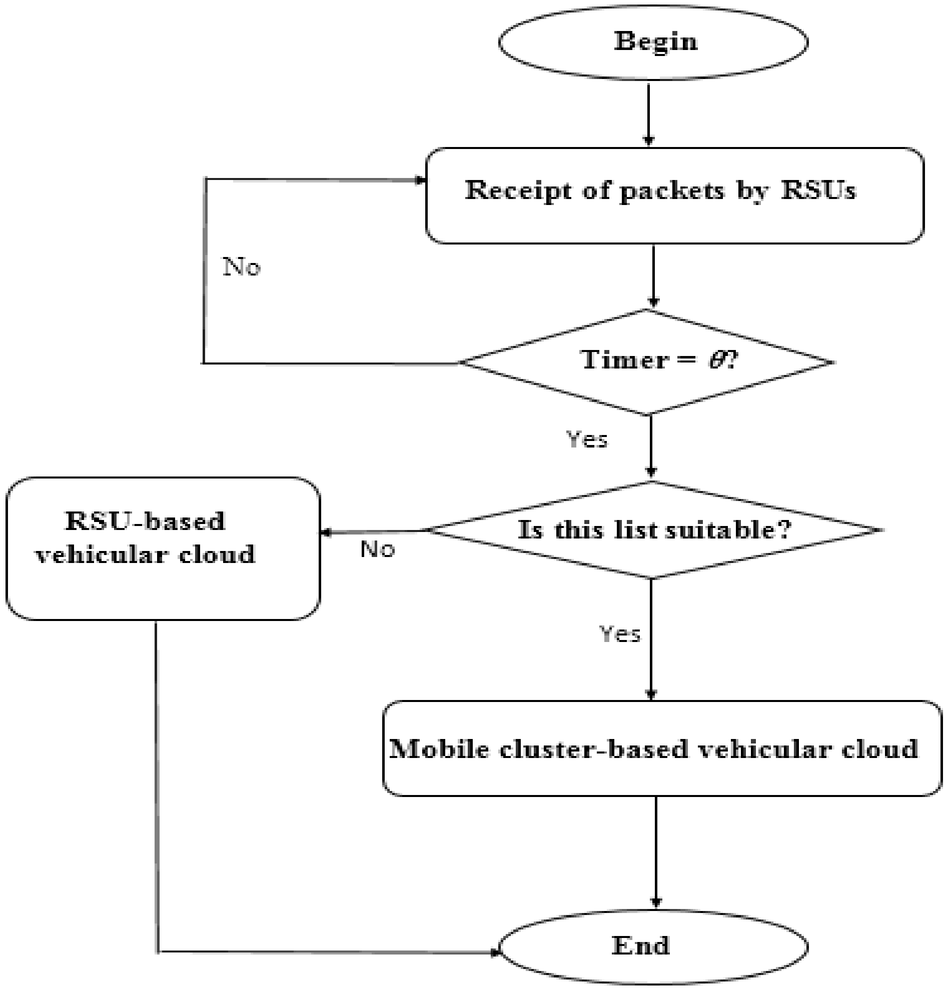


Figure 24. Registration and data requests process at the RSU level to enable vehicular clouds.

If CVs and SVs are not receiving the response from RSUs, they all keep constantly sending their packets after waiting time (WT) of ten (10) seconds (the ten seconds choice explanation is in the experimental section).

Consequently, the reception of the registration and request packets by RSUs triggers some operations to create clusters and select cluster heads or construct directories at RSUs level. Figure 24 depicts this process. When such an RSU receives the first packet, either a registration packet or request packet, it starts a timer. Then, the transmission process of packets continues until the timer reaches a certain threshold θ (e.g., 1.5 minutes), some operations are launched.

- **RSU level operation after θ**

This step is called Discovery Phase (DPH) that allows saving all resources on RSU's directory or creating clusters. We summarize these operations as follows:

- Reset the time θ
- Put all packets on the list.
- If the list contains the same type of vehicles, either CVs or SVs, the RSU creates its proper directory
- If the list contains at least one SV or one CV and the sum of vehicles is higher than one (1), the RSU creates a cluster and selects its CH (described in section 3.1.2)
- Delete the list's items in the directory after creating the cluster.

The following algorithm highlights how RSUs try to create VCs.

Algorithm 2 RCVC process Pseudo-code, which checks the ability either to create a cluster or RSU's cloud.

```
input: stack temporary_cloud // temporary_cloud is a stack that contains the
                              // temporary cloud built using the received packets.
stack RSU_cloud;
boolean find_supplier ← false;
boolean find_client ← false;
j ← θ; // It is a counter.
while j <= simulation time do
begin
for each V in temporary_cloud
 begin
  if  V = "client"
    begin
      find_client ← true;
    end if
  if  V = "supplier"
    begin
      find_supplier ← true;
    end if
 end for.
    if  (temporary_cloud.size() >= 2) and (find_client) and (find_supplier)
      begin
        create_cluster(temporary_cloud); // This function informs vehicles that
                              // belong in the same cluster for cluster creation.
        temporary_cloud.clear(); // This function deletes the temporary cloud.
       end if
    else
      begin
       for each item i in temporary_cloud
        begin
          RSU_cloud.push_back(i); // This function creates a cloud at RSU's level.
        end for
          temporary_cloud.clear();
    end else
 j ← j+ θ;
 end while
```

Algorithm 2 verifies the existence of at least one SV and one CV in the temporary cloud which is created at the beginning, and that will be deleted after a threshold θ if it does not fulfill the conditions of creating a cluster. Note that in our proposed protocol, not all vehicles can join clusters. Thus, in the case of a vehicle that cannot join any cluster, it registers the offers and requests on the VC at the RSU level.

- **Clustering**

The RSUs proceed with creating vehicle clusters using vehicles' coordinates in a plan of Euclidean space by defining what is called a section cluster. RCVC considers all routes on the map as sections from 1 to j, as shown in Figure 25.

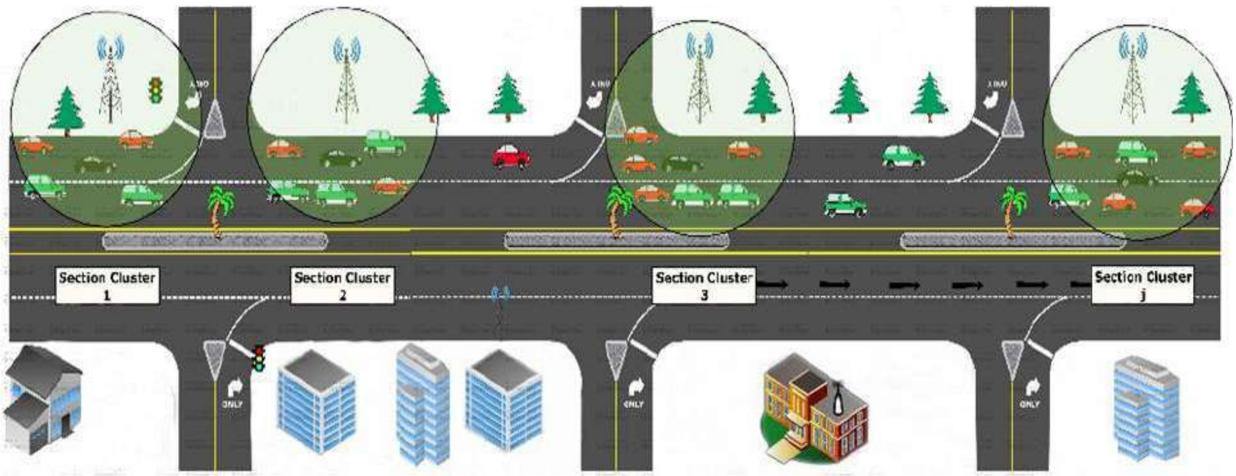


Figure 25. Cluster section.

Thus, to select CH_j in section j in real-time, we utilize the Euclidean distance [75], taking as parameters all vehicles' coordinates V_i (X_i, Y_i) and their average speeds AV_i at the time t. The formulas used to calculate the central point P_j (X_j, Y_j), are expressed mathematically as:

$$X_j = \frac{\sum_{i=1}^{n} X_i}{n} \quad (1)$$

$$Y_j = \frac{\sum_{i=1}^{n} Y_i}{n} \quad (2)$$

$$AV_j = \frac{\sum_{i=1}^{n} AV_i}{n} \quad (3)$$

where n is the number of vehicles in cluster section S_j.

To get the coordinates of the vehicle which is a cluster head in the cluster section S_j, we calculate the distance (*Dist*) between all vehicles and P_j. The vehicle that has a smaller distance will be a cluster head CH_j in S_j. The average speed AV is included because the accelerations and decelerations are very frequent within the urban area. That is why it is considered as a representative and stable value and used to maintain the cluster formation for as long as possible.

This formula calculates the Euclidean distance between vehicle V_x and P_j:

$$Dist(Vx, P_j) = \sqrt{(X_{V_x} - X_{P_j})^2 + (Y_{V_x} - Y_{p_j})^2 + (AV_{V_x} - AV_{p_j})^2} \quad (4)$$

Besides the minimum required number of vehicles to form a cluster, all vehicles having similar mobility patterns can also join the formed cluster. The similarity of the mobility patterns is mainly measured by the link stability $LS_{Pj,Vi}$ between the joining vehicle and the cluster head P_j [78].

The $LS_{Pj,Vi}$ is calculated by the Equation (5).

$$LS_{Pj,Vi} = \alpha * LS_{Pj,Vi} + (1+\alpha) * \left[\frac{1}{\frac{\Delta V_{Pj,Vi}(t+p)}{\Delta V_{Pj,Vi,}(t)} * \frac{\Delta D_{Pj,Vi}(t+p)}{\Delta D_{Pj,Vi}(t)}}\right]$$

where α: a constant that is used to avoid the influence of peak cases, such as unexpected braking;

V_{Vi} (t) the velocity of vehicle V_i at time t;

$\Delta V_{Pj,Vi}$ = V_{Pj} (t) – V_{Vi} (t); V_{Pj} and V_{Vi} speed variation at instant t;

$\Delta D_{Pj,Vi}(t)$: Distance between P_j and V_i at the time t.

Each RSU informs all vehicles that are included on the same cluster for this creation and selection by broadcasting the cluster packet (CP). The CP does not contain only the ID of the CH, but all the information about offered/requested services and *Numhops* (a variable that defines the number of authorized hops between the CV and the SV) that promote the clusters to the autonomous management as well. As the CH has all the pertinent information on its cluster vehicles, it starts to manage the consumption between vehicles by applying the service selection operation (it will be detailed hereafter) to optimally schedule the CVs queue. RCVC searches in its SV's candidate list (L_C) that is obtained from the RSU during the clustering to respond to all CVs via response packet (RP) which contains the ID of SV and their resources if:

1. There are some SVs whose queues contain only less than the value of the $Queue_{CV}$ variable. The latter is scalable according the global number of vehicles and SVs in a VC (it will be explained in the experimental).
2. There are some SVs that can satisfy CVs requirements. On the contrary, the CH responds by negative acknowledgement (NACK).

The user in CV specifies its needed resources and attributes by sending a service packet (SP) to the SV using *Numhops* hopes (explained in the experimental section). SP contains the ID of the vehicle and its requests. Upon receipt of the user service packet, the SV responds with a service response packet (SR) packet, asking the CV for the payment method. The CV and the SV exchange data packets corresponding to the resources. Figure 26 shows the structure of the RCVC protocol packet, where C, R, N, P, S are flags for the CP, RP, NACK, SP and SR packets, respectively.

0 15	16 31
CH ID	Numhops
Vehicle ID	C R N P S
Resource	

Figure 26. RCVC protocol packet format.

4.2.2.VC creation at the RSU level

The storage of all SV's and CVs information on RSU directory (DR) in the case of no cluster creation triggers the calculation of the SV's influence area A_i in which the SVs can offer their resources reliably to consumers [55]. Moreover, A_i allows tracking the SVs itineraries. The fact that all RSUs are wired enables them to share their DRs and SVs Ai to form a global cloud of RSUs. When the SV moves to other RSU and becomes nearest, A_i changes its value on all RSUs DR by sending another registration packet to other RSU (R2, for example). Consequently, the RSU adds (updates) the pack-et to its DR and forwards it to all RSUs, which they update their DRs.

The SV, which does not belong to any cluster, always sends a beacon to the nearest RSU (every 2 seconds). These beacons allow RSUs to follow the supplier vehicle's movements. When the beacons are delayed to arrive at RSU, the latter tries to know voluntarily the location of the SV's area which is called estimated area A_e. With this formula, we can estimate the A_e, $A_e = 1.3*V*(T_c-T_1)$, where T_1 is the time of sending the last beacon, T_c is the current time, and V is the average speed of supplier vehicle). The RSU sends the value of A_e to all RSUs. It is worth noting that the RSU will delete the supplier vehicle from its DR if it does not receive beacons from it, for a time $X = 10$ * (interval beacon) [76].

When a CV needs to consume some services, it formulates a REQP and sends it to its nearest RSU. In addition to the needed resources and their attributes, the REQP packet contains the vehicle's geographical coordinates. Wherever, if the RSU receives a REQP from the CV, it searches in its DR for one or more supplier vehicles that can meet the vehicle's requirements.

The RSU responds to a CV's request by choosing the corresponding SV from its list L_c, which has undergone a service selection operation (it will be detailed hereafter). Then, the RSU formulates an RP and sends it to each CV, this kind of packet contains the following elements:

- The ID of supplier vehicle.
- Latest SV's location from the latest beacon.
- Resources and their attributes.

If L_c does not contain any SVs which can satisfy the CV's request, or if SVs' queues of this list are busy, then the RSU responds with a negative acknowledgement (NACK).

The user specifies the resources and their attributes of each resource that he needs. His vehicle prepares an SP for the supplier vehicle selected and sends it, using an RSU's backbone. The service packet contains the CV's ID and their requests, upon receipt of the CV's service packet, the SV responds via SR to the user for payment method. The CV and the SV exchange data packets corresponding to the resources.

Figure 27 illustrates how all objects (vehicles and RSUs), orderly, interact with each other in RCVC by using the sequence of events according to packets sending and receiving. This diagram provides a better visualization of all RCVC operations.

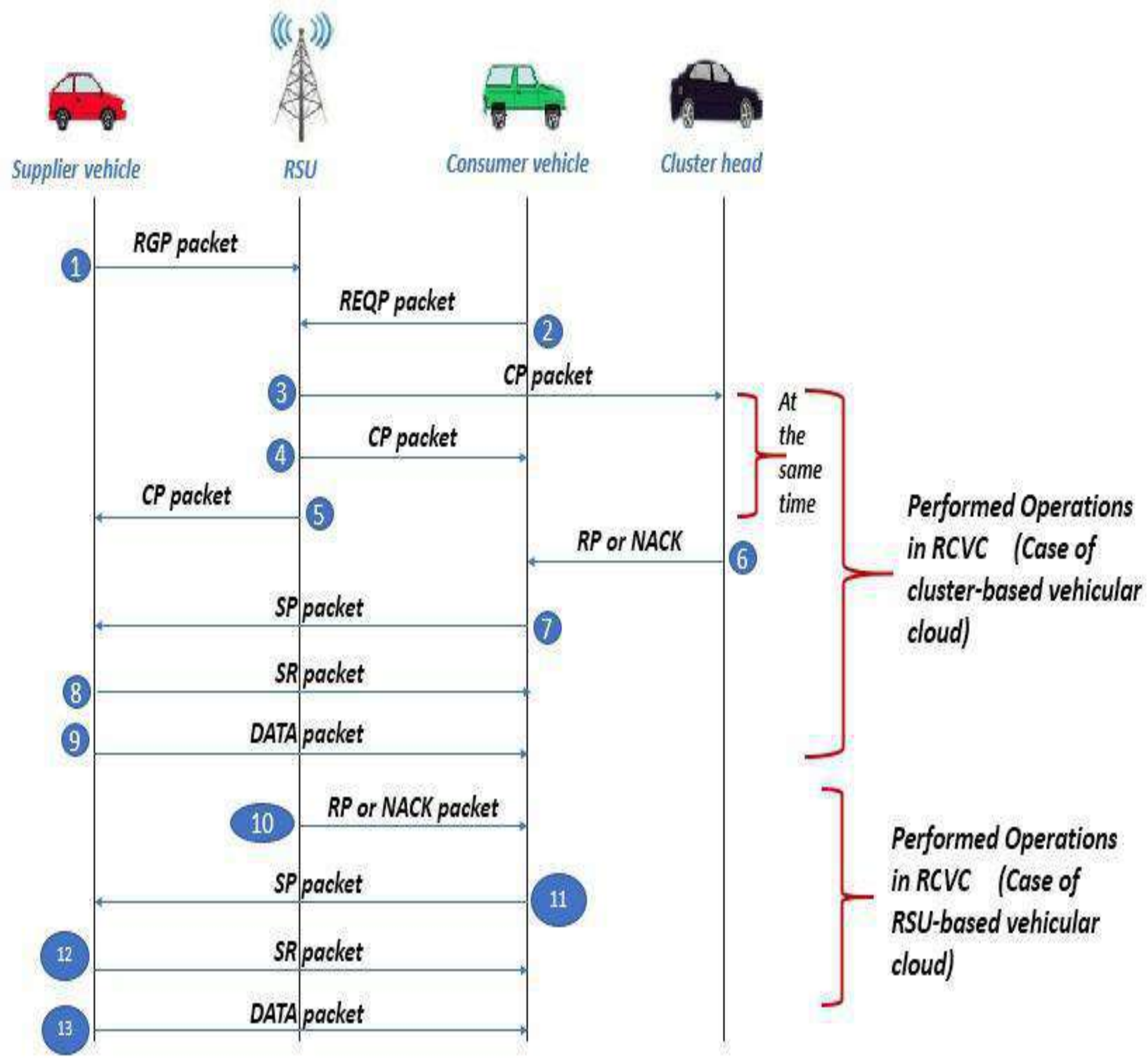


Figure 27. Sequence diagram of packets exchanged by RCVC.

The supply and consume operation begins by registering all packets at the RSUs level, whether it is an RGP registration packet or a REQP request packet, as it is shown in Figure 27. The RSUs create the clusters by informing the vehicles belonging to the cluster by this creation through sending CP packets, or else by creating VCs at their levels.

In the case of a cluster-based vehicular cloud, the CH has the role of managing the operation of consumption by sending the RP or NACK packets. In the second case of RSU-based vehicular cloud, the RSUs also continue to achieve the consumption operation by the RP or NACK packets. The rest of the operations is accomplished between vehicles using the SP, SR, and data packets.

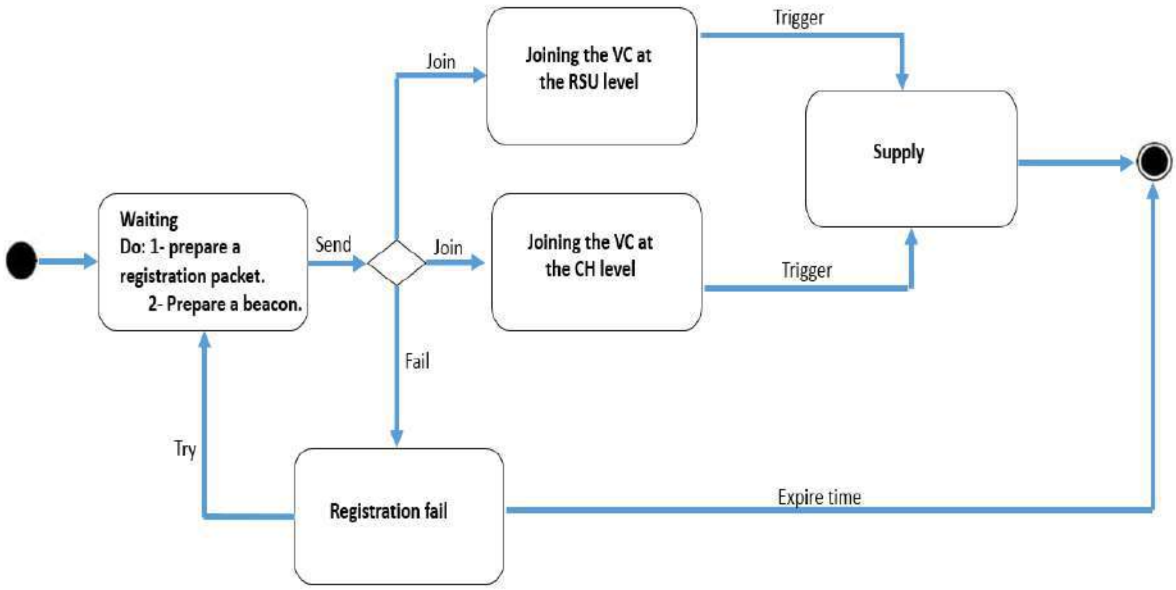


Figure 28. State diagram of the supplier vehicle object.

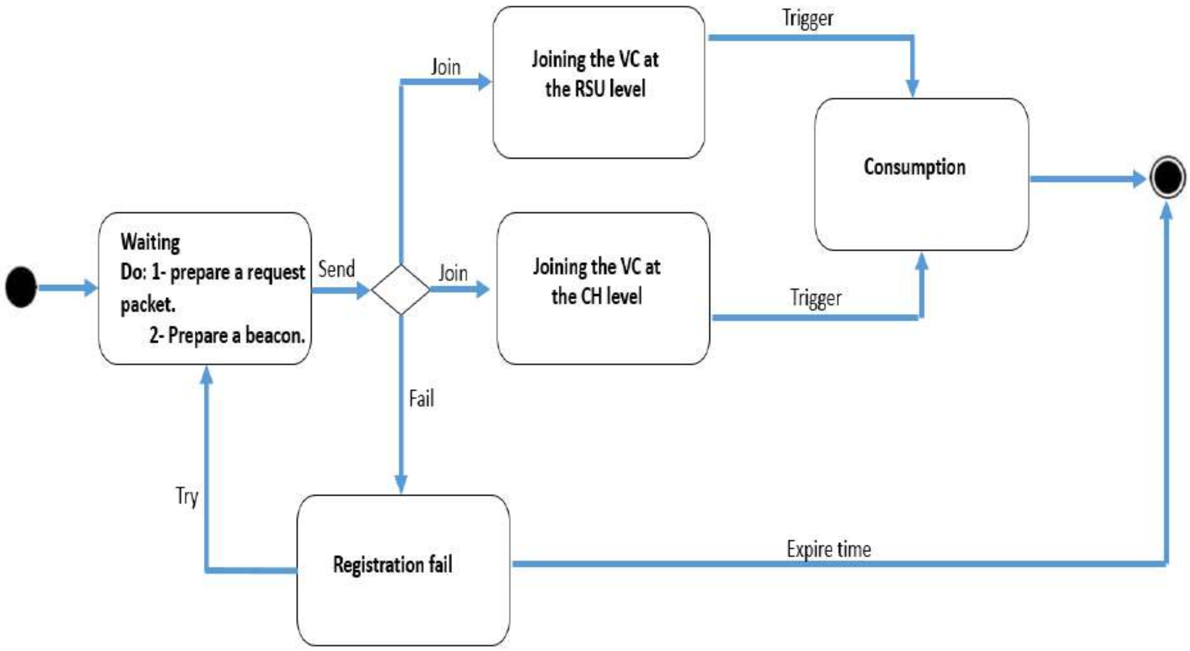


Figure 29. State diagram of the consumer vehicle object.

In order to prove that no deadlocks can occur, we used the state diagram to describe all possible states and to better understand the behavior of the objects. The state diagram, as shown in Figures 28 and 29, is a set of a finite number of states which captures an abstract description of the behavior of SV and CV objects.

4.3. A mathematical model for service selection

We extended RCVC by adding a service selection method in the VC either at the RSUs level or the CHs'. Based on the collected data in real-time, the RSUs and CHs select the best services in VC by using the Simple Additive Weighting method (SAW) [79] which exploits the multi-criteria making. Table 2 illustrates the offered services' attributes of any SV, such as quality criteria.

Table 2. Quality criteria of SV.

Criteria	Definition	Type
$Bandwidth_{Naas}$	Access bandwidth.	Double (bit/s)
$Duration\_bandwidth_{NaaS}$	The offered access bandwidth duration.	Double (h)
$Cost_{NaaS}$	The offered bandwidth unit price.	Double ($)
$Storage_{SaaS}$	Offered storage.	Double (Mo)
$Duration\_Storage_{SaaS}$	The offered storage duration.	Double (h)
$Days_{SaaS}$	Maximum overall storage time.	Double (Days)
$Cost_{SaaS}$	The offered storage unit price.	Double ($)
$Data_{DaaS}$	Data capacity.	Double (Mo)
$Cost_{DaaS}$	The offered data unit price.	Double ($)

Note that the criteria are as performance values, and they are numbered from 1 to 9 with:

$Bandwidth_{NaaS}$ =1; $Duration\_bandwidth_{NaaS}$ =2; $Cost_{NaaS}$ =3; $Storage_{SaaS}$ =4; $Duration\_Storage_{SaaS}$ =5; $Days_{SaaS}$ =6; $Cost_{SaaS}$ =7; $Data_{DaaS}$ =8; $Cost_{DaaS}$ =9.

In each VC, we consider a set of candidate vehicles $CondidSV = \{SV1, SV2, SV3 \ldots\ldots SVn\}$ that leads to get a decision matrix $MATDIC = (MATDIC_{ij};\ 1 \leq i \leq n;\ 1 \leq j \leq 9)$. The *MATDIC* will experience a normalization operation. Therefore, the final score is calculated for each SV to be able to classify them. The browsing of the decreasing list, which is sorted by a score from highest to lowest, allows proposing to each CV such an SV that its queue contains less than five CVs, and it can satisfy the CV's requirements.

4.3.1. Normalization

Before combining the performance values, a normalization operation is carried out to obtain a normalized decision matrix that allows all values to be compared. We applied the Equation (6) to maximize the beneficial criteria of $Bandwidth_{NaaS}$, $Duration\_bandwidth_{NaaS}$, $Storage_{SaaS}$, $Duration\_Storage_{SaaS}$ and $Data_{DaaS}$, the Equation (7) is applied to minimize the non-beneficial criteria for $Cost_{NaaS}$, $Cost_{SaaS}$ and $Cost_{DaaS}$, and to maximize the beneficial criterion $Days_{SaaS}$ we used the formula in Equation (8).

$$MATDICnorm_{ij} = \begin{cases} \frac{MATDIC_{ij} - MATDIC_j^{min}}{MATDIC_j^{max} - MATDIC_j^{min}} & if\ (MATDIC_j^{max} - MATDIC_j^{min} \neq 0) \\ 1 & else\ (MATDIC_j^{max} - MATDIC_j^{min} = 0) \end{cases} \quad (6)$$

$$MATDICnorm_{ij} = \begin{cases} \frac{MATDIC_j^{max} - MATDIC_{ij}}{MATDIC_j^{max} - MATDIC_j^{min}} & if\ (MATDIC_j^{max} - MATDIC_j^{min} \neq 0) \\ 1 & else\ (MATDIC_j^{max} - MATDIC_j^{min} = 0) \end{cases} \quad (7)$$

$$MATDICnorm_{ij} = \begin{cases} \frac{lenght(MATDIC_{ij}) - MATDIC_j^{\mathrm{min.length}}}{MATDIC_j^{max.length} - MATDIC_j^{\mathrm{min.length}}} & if\ \left(MATDIC_j^{max.length} - MATDIC_j^{\mathrm{min.length}} \neq 0\right) \\ 1 & else\ \left(MATDIC_j^{max.length} - MATDIC_j^{\mathrm{min.length}} = 0\right) \end{cases} \quad (8)$$

$Days_{SaaS}$ can be seen as a character string that represents days. For example, 1453 means Sunday, Wednesday, Thursday, and Tuesday. Where $MATDIC_j^{max.length}$ and $MATDIC_j^{min.length}$ are the max and the min length of the character string of days, respectively.

4.3.2. Performance score

We assigned a weightage to all criteria to get a weighted normalized decision matrix. In our case, we allocated an equal weightage (W_j) to each criterion, where the sum of all weightage is equal to one. Then, we multiplied the weight of each criterion by its normalized performance values [80]. Finally, we added them for each alternative to get a performance score ($PERF$), as demonstrated by Equation (9).

$$PERF(SV_i) = \sum_{j=1}^{9} MATDICnorm_{ij} * W_j \quad \text{where} \quad \sum_{j=1}^{9} W_j = 1 \quad (9)$$

Rankings can be assigned to all SVs' services in the cloud, based on the performance score to classify all services from best to lowest.

4.4. Experimental analysis

4.4.1. Simulation setup

To implement and evaluate our proposed protocol RCVC, we used OMNeT++ 5.3 [68] for the behavioral aspect and Sumo-0.32.0 [65] as a mobility simulator. RCVC is deployed in MANHATTEN grid 4 x 4 km^2, which contains 16 junctions, where the distance between two junctions is 1 km. We covered this map by thirty (16) RSUs. The vehicle density (VD) is varied between 100 and 500 vehicles taking three ratios of supplier vehicle density: one-fourth, one-third and one-half of VD, in each time we vary the value of VD (100, 300, 400, and 500). The details of the simulation parameters are shown in Table 3.

Table 3. Simulation parameters.

Parameter	**Value**
Simulation framework	Veins (OMNeT++ and Sumo)
Mobility model	Manhattan
Simulation time	1000 s
Simulation area	4x4 km2
Transmission range	500 m
Transfer rate	18 Mb/s
Vehicle density	[100-500] vehicles
Vehicles speed	Up to 70 km/h
Supplier vehicle density	1/4, 1/3 and 1/2 of vehicle density
The size of registration and request packet	128 Kbytes
Data Packet Size	[1-5] Kbytes
Maximum number of offered services per supplier	Three (3) services
Maximum number of requested services per consumer	Three (3) services

Based on proved facts, the vehicle's average speed in Manhattan city is almost 24 k/h [81], this allows determining the value of θ by taking into consideration the distance between the RSUs (equals 1 kilometer). One vehicle that crosses all the distance between two RSUs, it must spend 2.5 minutes. Therefore, the observation of the following distribution, in Table 4, leads to calculating the median [82]. This allows having a realistic value of θ.

Table 4. Distribution values of elapsed times to arrive to RSU.

The distance between the vehicle and the next RSU (meters).	200	400	600	800	1000
The elapsed time to arrive near to the next RSU (minutes).	0.5	1	1.5	2	2.5

The threshold θ is the median of the distribution that equals 1.5 minute.

For the scalability reasons, we have either the VC at the RSU level or the CH's. The management of queues is accomplished by assigning a value $\mathrm{Queue_{CV}}$ to each CV depending on two factors that are the number of vehicles and the number of SVs in the VC at the RSU level or the CH as shown in the formula in the Equation (10).

$$\mathrm{Queue_{CV}} = \begin{cases} \frac{\mathrm{N_V}}{\mathrm{N_{SV}}} & \mathit{If}\ \mathrm{N_{SV}} \leq \frac{1}{3} \\ \frac{\mathrm{N_V}}{2\mathrm{N_{SV}}} & \mathit{If}\ \frac{1}{3} < \mathrm{N_{SV}} \leq \frac{1}{2} \\ 0 & \mathit{Otherwise} \end{cases} \quad (10)$$

where NV is the number of vehicles and $\mathrm{N_{SV}}$ is the number of SVs in the VC.

The value of *Numhops* is scalable according to the number of vehicles in the cluster. Equation (11) defines its value:

$$Numhops = \frac{\mathrm{Nv}}{5} \quad (11)$$

Table 5. Sufficient number of hops by varying Number of vehicles in a cluster.

Number of vehicles in a cluster	2 to 5	5 to 10	10 to 15	15 to 20	20 to 25	25 to 30
Sufficient number of hops	1	2	3	4	5	6

As shown in Table 5, the choice of number five (5) comes from several simulation experiments. If we have less than five (5) vehicles in the cluster, then *Numhops* = 1, If we have less than ten (10) vehicles and more than five (5) in the cluster, then *Numhops* = 2 and it makes sense compared to the average speed of vehicles in the MANHATTAN city (24 k/h). Taking the case where the number of vehicles thirty (30) vehicles, the *Numhops* = 6, and so on.

RCVC's performance has been evaluated taking these metrics:

1) Discovery Delay (DD): the time delay between sending a request packet and receiving a response packet from RSU.

2) Consuming Delay (CD): the time delay between sending a request for resources to the SV and receiving three data packets.

3) Vehicle Traffic (VT): the average generated, received, and forwarded amount of traffic by a vehicle.

4) End-to-End Delay (E2ED): the average time delay that a data packet takes to reach the CV from the SV through RSUs and CHs.

To prove the effectiveness of RCVC, we varied the number of supplier vehicles and performed a comparison in terms of DD, CD, VT and E2ED. Then, we compared it with four state-of-the-art protocols, which are the CROWN [55], DCCS-VC [57], FDCCS-VC [58] and TOPVISOR [64] taking the same performance metrics and simulation parameters.

4.4.2. Results and discussions

To evaluate the performance of RCVC, we compared three experimental scenarios (One-fourth, One-third and one-half VD), and at the level of each scenario, we take different VDs while using four performance measures.

The DD, CD, VT, and E2ED performance metrics were almost stable. Figure 30 (a) shows that varying the number of SVs did not affect the DD because all vehicles continued to send their request packets to the nearest RSUs every 10 seconds until the acknowledgement is achieved. Note that all request packets did not pass through the neighboring vehicles. We defined *WT*=10 seconds because using a *WT* greater than that results in bypassing the nearest RSUs as vehicles move quickly. While using *WT* less than 10 seconds leads to generating more packets and flooding the network. This *WT* came for many simulation attempts.

In Figure 30 (b), the CD decreases when the number of SVs increases; this is attributed to the fact that when the number of SVs increases, there is more chance of finding resources faster. The VT has an ascending slope in Figure 30 (c), even during varying SV's number. This fact can be substantiated by the non-relationship between this metric and the number of SVs; it depends on the density of all vehicles. As soon as we increased the VD, the VT increased because more packets were generated and transmitted. Figure 30 (d) shows that the E2ED is strongly related to the number of SV.

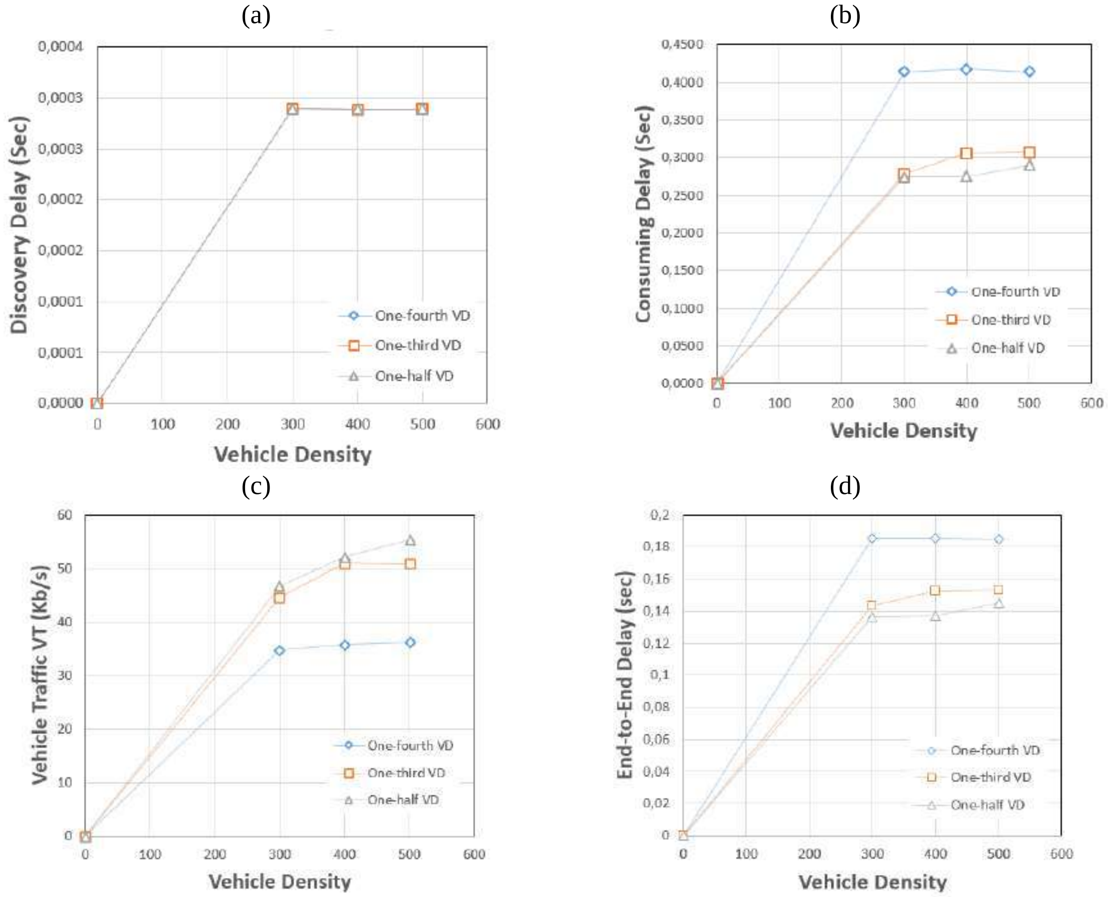


Figure 30. Performance evaluation of RCVC while varying the SV density: (a) Discovery Delay. (b) Consuming Delay. (c) Vehicle Traffic. (d) End-to-End Delay.

RCVC yielded a better result than CROWN, DCCS-VC, and TOPVISOR in terms of DD. This outcome can be justified by the use of the routing protocols to route packets between RSUs in the case of CROWN, which is called CAN DELIVER [76], and among buses in case of DCCS-VC that is presented in [83]. Moreover, RCVC has superior performance than TOPVISOR because this latter uses two control levels on top of the RSUs to advertise the VC, as shown in Figure 31 (a). The same reason for the CD in Figure 31 (b), CROWN and DCCS-VC use routing protocols to route all packets from discovery to data consumption. Nevertheless, DCCS-VC has a good result that almost converges into the same RCVC's outcome, which may be due to the efficiency of its routing protocol between buses.

RCVC provided some improvements to DD because it did not use any routing protocol. When an SV attempts to register their resources or a CV sends requests, they try each time (every 10s) to find the nearest RSU in their vicinity. Once found, they send their packets which will be acknowledged immediately. Even for the CD that yields good results, the reason is that the consumption is achieved either in the cluster that uses only one hop in minimum till six hopes in maximum or between RSUs.

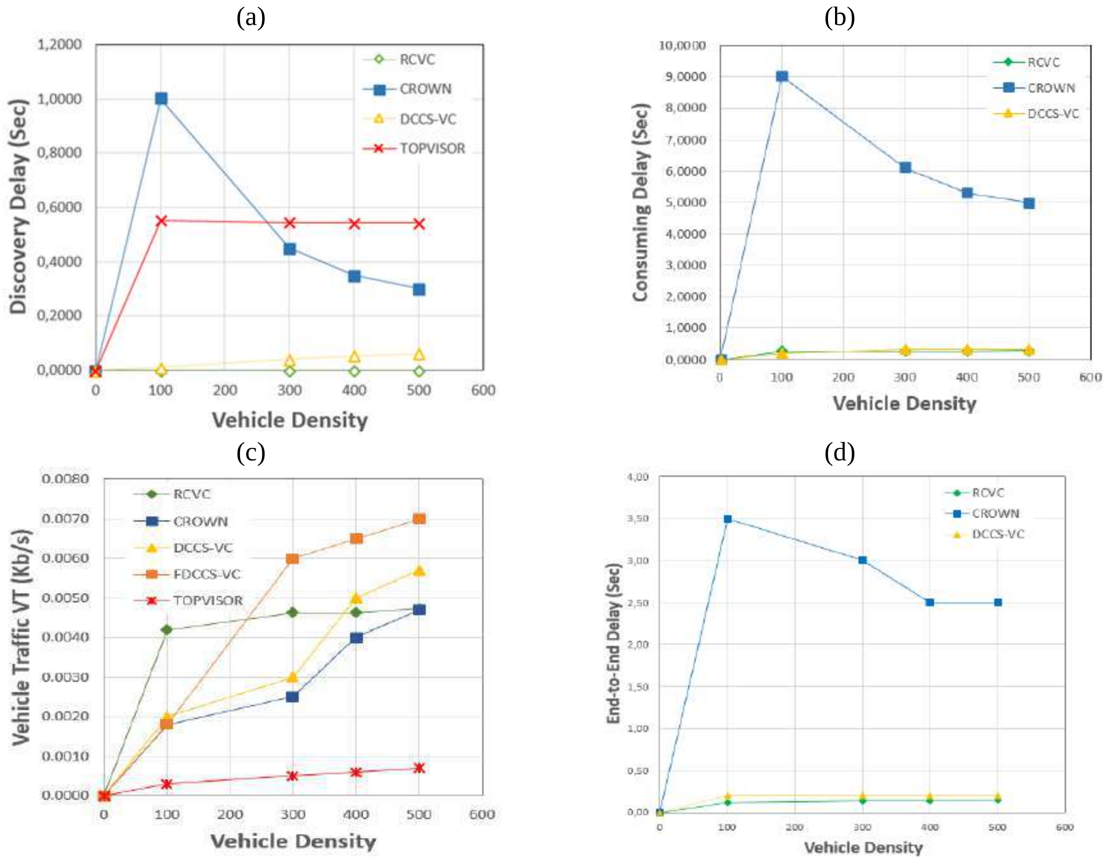


Figure 31. Performance evaluation comparisons of RCVC with: (a) CROWN, DCCS-VC, and TOPVISOR in terms of Discovery Delay. (b) CROWN and DCCS-VC in terms of Consuming Delay. (c) CROWN, DCCS-VC, FDCCS-VC, and TOPVISOR in terms of Vehicle Traffic. (d) CROWN, DCCS-VC in terms of End-to-End Delay.

In figure 31 (c), we measured the VT generated by vehicles through increasing vehicle densities. The objective is to evaluate the protocols by monitoring the stability of the VT, where the best performance is obtained by the protocol which maintains the least traffic generation. Up to 100 vehicles, all protocols generated traffic increasingly. The proposed RCVC protocol provided stable performance in case of density greater than 100 vehicles, unlike the rest of the protocols, they generated more packets. Over 400 vehicles, RCVC performed better than DCCS-VC and FDCCS-VC in terms of VT, and almost the same as CROWN. The reason behind this result is that the exchange of more packets between buses and vehicles by DCCS-VC and FDCCS-CV compared to RCVC which only transmits packets inside clusters or across the RSUs backbone. Note that CROWN shares the same feature of using RSUs to route packets in case of discovery or consumption. However, TOPVISOR performed better than all protocols in terms of VT because it is dedicated to building the VC, and hence discovering the services it contains; it is not devoted to the consumption operation.

The obtained results, shown in Figure 31 (d), illustrate that RCVC is always better than CROWN and DCCS-VC in terms of E2ED, which can be justified by the same reason of CD; it is clear that E2ED is a part of the CD.

We can consider Table 6 as a decision matrix which allows applying SAW to select the best protocol among all available alternatives based on various criteria like DD, CD, VT and E2ED.

Table 6. A comparative ranking of RCVC with the existing state-of-the-art.

	DD	**CD**	**VT**	**E2ED**
RCVC	1	1	3	1
CROWN	3	3	2	3
DCCS-VC	2	2	4	2
FDCCS-VC	-	-	5	-
TOPVISOR	4	-	1	-

The normalized decision matrix is shown in Table 7.

Table 7. Normalized decision matrix.

	DD	**CD**	**VT**	**E2ED**
RCVC	1	1	0,5	1
CROWN	0,33	0	0,75	0
DCCS-VC	0,67	0,33	0,25	0,33
FDCCS-VC	-	-	0	-
TOPVISOR	0	-	0,8	-

The next step is to assign the weightage to each criterion. Here we assign an equal weightage to all criteria, which is 0.25 %, and then to multiply each weight with its normalized performance values on solving, we get a weighted normalized decision matrix as shown in Table 8.

Table 8. Weighted normalized decision matrix.

Weightage	0,25 %	0,25 %	0,25 %	0,25 %	100 %
	DD	**CD**	**VT**	**E2ED**	**Performance score**
RCVC	0,25	0,25	0,125	0,25	0,87 %
CROWN	0,08	0	0,19	0	0,27 %
DCCS-VC	0,17	0,08	0,06	0,08	0,40 %
FDCCS-VC	-	-	0	-	0 %
TOPVISOR	0	-	0,2	-	0,2 %

As shown in Table 7, RCVC compared to other protocols has a highest score, which proves better performances considering all metrics.

4.5. Conclusion

In this chapter, we proposed a new protocol that allows building vehicular clouds in an urban area, focusing on Mobile cluster-based vehicular cloud and RSU-based vehicular cloud. This protocol makes the possibility to offer services by supplier vehicles to consumers for effective consumption in real-time. We extend our protocol with a mathematical model for service selection. The clustering mechanism that cooperates with the RSUs yielded a set of improvements in terms of performance evaluation metrics. RCVC's outcomes were calculated, discussed, and compared to four other protocols which are CROWN, DCCS-VC, FDCCS-VC and TOPVISOR. Finally, the simulation results proved the performance of RCH-VC in terms of discovery delay, consuming delay, vehicle traffic, and end-to-end delay. As future work, we will try to extend this protocol, by involving blockchain, UAVs, some security enhancements.

Chapter 5 Conclusion and perspectives

Content

5.1. General conclusion and perspectives

5.1. General conclusion and perspectives

In the first part of this thesis, we began to give a general view of the vehicular networks as well as the standardisations that have been completed until now. After that, we have presented a thematic and chronological state of the art on the subject treated. Since we used a routing protocol in our first contribution, we have taken into consideration to present the routing in the VANETs as a first exhibit. Subsequently, knowing that vehicles are considered as smart objects in the future smart cities, we started the second chapter with the basic concepts of Cloud Computing, touching on their importance in smart cities. Thus, we were able to reveal what defines each of these areas and the functionalities that characterise them. The objectives of the work presented in this thesis is the study and the modelling of the techniques of supply and consumption of services in vehicular networks.

In the second part, this is where we have identified the lack of attention to intelligent vehicle communication technologies and have suggested a suitable techniques for the supply and consumption of services in the vehicular cloud. We have proposed two solutions for offering services in the vehicular Cloud. The first is an approach that is based on the creation of vehicular clouds at the level of cluster heads and roadside units using the technic of clustering, allowing each consumer vehicle to join a cluster hoping to find services to consume. The vehicular Clouds created at the RSU and CH levels have the objective of giving more visibility to the supplier vehicles services. While the second protocol is based on a mathematical model for service selection.

As a future work, we will try to enhance our work by involving blockchain, UAVs, some security enhancements. Many research perspectives can be envisaged. In what follows, we present those perspectives that seem to be the most interesting:

- Secure our protocols in order to protect the privacy of drivers and their data offered.
- In the context of the vehicular Cloud, several challenges remain to be overcome, such as:

- ✓ The formation of a flexible and robust vehicular cloud architecture in order to face the mobility of nodes.
- ✓ The selection of services in the event that consumers require more than a single service.
- ✓ Study implementation of recovery mechanisms in the event that there are breakdowns in supplier vehicles during a consumption operation.
- ✓ Secure V2V and V2I communications in order to guarantee the integrity of the data exchanged.
- ✓ Help the vehicles through the UAVs to achieve the consumption operation.

Publications list

International journal

1. Ben Bezziane, M., Korichi, A., Kerrache, C. A., & Fekair, M. E. A. (2021). RCVC: RSU-Aided Cluster-Based Vehicular Clouds Architecture for Urban Areas. *ELECTRONICS*, *10*(2).

International conference

1. Bezziane, M. B., Korichi, A., el Amine Fekair, M., & Azzaoui, N. (2021). FR-VC: A novel approach to finding resources in the vehicular cloud. In Innovative and Intelligent Technology-Based Services for Smart Environments–Smart Sensing and Artificial Intelligence (pp. 149-155). CRC Press.

BIBLIOGRAPHY

[1] Hartenstein, H., & Laberteaux, L. P. (2008). A tutorial survey on vehicular ad hoc networks. IEEE Communications magazine, 46(6), 164-171.

[2] Li, L., Liu, J., Cheng, L., Qiu, S., Wang, W., Zhang, X., & Zhang, Z. (2018). Creditcoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles. *IEEE Transactions on Intelligent Transportation Systems*, *19*(7), 2204-2220.

[3] ITS. https://www.its.dot.gov/ consulted on 19/06/2020.

[4] Martinez, F. J., Fogue, M., Coll, M., Cano, J. C., Calafate, C. T., & Manzoni, P. (2010, July). Assessing the impact of a realistic radio propagation model on VANET scenarios using real maps. In 2010 Ninth IEEE International Symposium on Network Computing and Applications (pp. 132-139). IEEE.

[5] Shakeel, S. M., Ould-Khaoua, M., Rehman, O. M. H., Al Maashri, A., & Bourdoucen, H. (2015). Experimental evaluation of safety beacons dissemination in VANETs. Procedia Computer Science, 56, 618-623.

[6] Ren, J., Yu, G., He, Y., & Li, G. Y. (2019). Collaborative cloud and edge computing for latency minimization. IEEE Transactions on Vehicular Technology, 68(5), 5031-5044.

[7] Lo'ai, A. T., Bakheder, W., & Song, H. (2016, June). A mobile cloud computing model using the cloudlet scheme for big data applications. In *2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)* (pp. 73-77). IEEE.

[8] Hagenauer, F., Higuchi, T., Altintas, O., & Dressler, F. (2019). Efficient data handling in vehicular micro clouds. Ad Hoc Networks, 91, 101871.

[9] Hasrouny, H., Samhat, A. E., Bassil, C., & Laouiti, A. (2017). VANet security challenges and solutions: A survey. *Vehicular Communications*, *7*, 7-20.

[10] Khosroshahi, A. H., Keshavarzi, P., KoozehKanani, Z. D., & Sobhi, J. (2011, August). Acquiring real time traffic information using VANET and dynamic route guidance. In *2011 IEEE 2nd International Conference on Computing, Control and Industrial Engineering* (Vol. 1, pp. 9-13). IEEE.

[11] Akhtar, N., Ozkasap, O., & Ergen, S. C. (2013, April). VANET topology characteristics under realistic mobility and channel models. In *2013 IEEE Wireless Communications and Networking Conference (WCNC)* (pp. 1774-1779). IEEE.

[12] Ku, I., Lu, Y., Gerla, M., Gomes, R. L., Ongaro, F., & Cerqueira, E. (2014, June). Towards software-defined VANET: Architecture and services. In *2014 13th annual Mediterranean ad hoc networking workshop (MED-HOC-NET)* (pp. 103-110). IEEE.

[13] Saha, A. K., & Johnson, D. B. (2004, October). Modeling mobility for vehicular ad-hoc networks. In *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks* (pp. 91-92).

[14] Saleet, H., Basir, O., Langar, R., & Boutaba, R. (2009). Region-based location-service-management protocol for VANETs. *IEEE Transactions on Vehicular Technology*, *59*(2), 917-931.

[15] Nagel, R., & Eichler, S. (2008, March). Efficient and realistic mobility and channel modeling for VANET scenarios using OMNeT++ and INET-framework. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops* (pp. 1-8).

[16] Hossain, M. S., & Atiquzzaman, M. (2009). Stochastic properties and application of city section mobility model. In *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference* (pp. 1-6). IEEE.

[17] Bai, F., Sadagopan, N., & Helmy, A. (2003). The IMPORTANT framework for analyzing the Impact of Mobility on Performance Of RouTing protocols for Adhoc NeTworks. *Ad hoc networks*, *1*(4), 383-403.

[18] Hong, X., Gerla, M., Pei, G., & Chiang, C. C. (1999, August). A group mobility model for ad hoc wireless networks. In *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems* (pp. 53-60).

[19] Zhou, B., Xu, K., & Gerla, M. (2004, October). Group and swarm mobility models for ad hoc network scenarios using virtual tracks. In *IEEE MILCOM 2004. Military Communications Conference, 2004.* (Vol. 1, pp. 289-294). IEEE.

[20] Liang, B., & Haas, Z. J. (1999, March). Predictive distance-based mobility management for PCS networks. In *IEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320)* (Vol. 3, pp. 1377-1384). IEEE.

[21] Cheek, E., Alghodhaifi, H., Adam, C., Andres, R., & Lakshmanan, S. (2020, April). Dedicated short range communications used as fail-safe in autonomous navigation. In *Unmanned Systems Technology XXII* (Vol. 11425, p. 114250P). International Society for Optics and Photonics.

[22] Yogarayan, S.; Razak, S.F.; Azman, A.; Abdullah, MF; Ibrahim, SZ; Raman, KJ. A Review of Routing Protocols for Vehicular Ad-Hoc Networks (VANETs). In2020 8th International Conference on Information and Communication Technology (ICoICT) 2020 Jun 24, 1-7.

[23] Qureshi, K. N., Bashir, F., & Iqbal, S. (2018, October). Cloud computing model for vehicular ad hoc networks. In *2018 IEEE 7th International Conference on Cloud Networking (CloudNet)* (pp. 1-3). IEEE.

[24] Skourletopoulos, G., Mavromoustakis, C. X., Mastorakis, G., Batalla, J. M., Dobre, C., Panagiotakis, S., & Pallis, E. (2017). Big data and cloud computing: a survey of the state-of-the-art and research challenges. In *Advances in mobile cloud computing and big data in the 5G Era* (pp. 23-41). Springer, Cham..

[25] Fernando, N., Loke, S. W., & Rahayu, W. (2013). Mobile cloud computing: A survey. *Future generation computer systems*, *29*(1), 84-106.

[26] Hagenauer, F., Higuchi, T., Altintas, O., & Dressler, F. (2019). Efficient data handling in vehicular micro clouds. *Ad Hoc Networks*, *91*, 101871.

[27] Gerla, M. (2012, June). Vehicular cloud computing. In *2012 The 11th annual mediterranean ad hoc networking workshop (Med-Hoc-Net)* (pp. 152-155). IEEE.

[28] Eltoweissy, M., Olariu, S., & Younis, M. (2010, August). Towards autonomous vehicular clouds. In *International Conference on Ad hoc networks* (pp. 1-16). Springer, Berlin, Heidelberg.

[29] Whaiduzzaman, M., Sookhak, M., Gani, A., & Buyya, R. (2014). A survey on vehicular cloud computing. *Journal of Network and Computer applications*, *40*, 325-344.

[30] Melaouene, N., & Romadi, R. (2019). An enhanced routing algorithm using ant colony optimization and VANET infrastructure. In *MATEC Web of Conferences* (Vol. 259, p. 02009). EDP Sciences.

[31] Poonia, R. C., Bhargava, D., & Kumar, B. S. (2015, January). CDRA: Cluster-based dynamic routing approach as a development of the AODV in vehicular ad-hoc networks. In *2015 International Conference on Signal Processing and Communication Engineering Systems* (pp. 397-401). IEEE.

[32] J. Bitam, S., & Mellouk, A. (2011, June). QoS swarm bee routing protocol for vehicular ad hoc networks. In *2011 IEEE International Conference on Communications (ICC)* (pp. 1-5). IEEE.

[33] Hussain, R., Kim, S., & Oh, H. (2016). Traffic information dissemination system: Extending cooperative awareness among smart vehicles with only single-hop beacons in VANET. *Wireless Personal Communications*, *88*(2), 151-172.

[34] Mittag, J., Thomas, F., Härri, J., & Hartenstein, H. (2009, September). A comparison of single-and multi-hop beaconing in VANETs. In *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking* (pp. 69-78).

[35] Qureshi, K. N., & Abdullah, H. (2013). Topology based Routing Protocols for VANET and their comparison with MANET. *Journal of Theoretical and Applied Information Technology*, *58*(3), 707-715.

[36] Setiabudi, A., Pratiwi, A. A., Perdana, D., & Sari, R. F. (2016, May). Performance comparison of GPSR and ZRP routing protocols in VANET environment. In *2016 IEEE region 10 symposium (TENSYMP)* (pp. 42-47). IEEE.

[37] Yan, S. H. I., JIN, X. Y., & CHEN, S. Z. (2011). AGP: an anchor-geography based routing protocol with mobility prediction for VANET in city scenarios. *The Journal of China Universities of Posts and Telecommunications*, *18*, 112-117.

[38] Luo, Y., Zhang, W., & Hu, Y. (2010, April). 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). IEEE.

[39] Rahim, A., Qiu, T., Ning, Z., Wang, J., Ullah, N., Tolba, A., & Xia, F. (2019). Social acquaintance based routing in vehicular social networks. *Future Generation Computer Systems*, *93*, 751-760.

[40] Gillani, M., Ullah, A., & Niaz, H. A. (2018, November). Trust management schemes for secure routing in VANETs—a survey. In *2018 12th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)* (pp. 1-6). IEEE.

[41] Fekair, M. E. A., Lakas, A., & Korichi, A. (2016, April). CBQoS-Vanet: Cluster-based artificial bee colony algorithm for QoS routing protocol in VANET. In *2016 International conference on selected topics in mobile & wireless networking (MoWNeT)* (pp. 1-8). IEEE.

[42] Malathi, M. (2011, April). Cloud computing concepts. In *2011 3rd International Conference on Electronics Computer Technology* (Vol. 6, pp. 236-239). IEEE.

[43] Eremia, M., Toma, L., & Sanduleac, M. (2017). The smart city concept in the 21st century. *Procedia Engineering*, *181*, 12-19.

[44] Ejaz, W., Naeem, M., Shahid, A., Anpalagan, A., & Jo, M. (2017). Efficient energy management for the internet of things in smart cities. *IEEE Communications magazine*, *55*(1), 84-91.

[45] Suciu, G., Vulpe, A., Halunga, S., Fratu, O., Todoran, G., & Suciu, V. (2013, May). Smart cities built on resilient cloud computing and secure internet of things. In *2013 19th international conference on control systems and computer science* (pp. 513-518). IEEE.

[46] Rao, B. P., Saluia, P., Sharma, N., Mittal, A., & Sharma, S. V. (2012, December). Cloud computing for Internet of Things & sensing based applications. In *2012 Sixth International Conference on Sensing Technology (ICST)* (pp. 374-380). IEEE.

[47] Morello, R., Mukhopadhyay, S. C., Liu, Z., Slomovitz, D., & Samantaray, S. R. (2017). Advances on sensing technologies for smart cities and power grids: A review. *IEEE Sensors Journal*, *17*(23), 7596-7610.

[48] Zaslavsky, A., Perera, C., & Georgakopoulos, D. (2013). Sensing as a service and big data. *arXiv preprint arXiv:1301.0159*.

[49] Petrolo, R., Loscri, V., & Mitton, N. (2017). Towards a smart city based on cloud of things, a survey on the smart city vision and paradigms. *Transactions on Emerging Telecommunications Technologies*, *28*(1), e2931.

[50] Xiao, Z., Lim, H. B., & Ponnambalam, L. (2017). Participatory sensing for smart cities: A case study on transport trip quality measurement. *IEEE Transactions on Industrial Informatics*, *13*(2), 759-770.

[51] Qiu, T., Liu, X., Li, K., Hu, Q., Sangaiah, A. K., & Chen, N. (2018). Community-aware data propagation with small world feature for internet of vehicles. *IEEE Communications Magazine*, *56*(1), 86-91.

[52] Bezziane, M. B., Korichi, A., el Amine Fekair, M., & Azzaoui, N. (2021). FR-VC: A novel approach to finding resources in the vehicular cloud. In *Innovative and Intelligent Technology-Based Services for Smart Environments–Smart Sensing and Artificial Intelligence* (pp. 149-155). CRC Press.

[53] Hussain, R., Son, J., Eun, H., Kim, S., & Oh, H. (2012, December). Rethinking vehicular communications: Merging VANET with cloud computing. In *4th IEEE International Conference on Cloud Computing Technology and Science Proceedings* (pp. 606-609). IEEE.

[54] Ben bezziane, M., Korichi, A., Kerrache, C. A., & Fekair, M. E. A. (2021). RCVC: RSU-Aided Cluster-Based Vehicular Clouds Architecture for Urban Areas. *Electronics*, *10*(2), 193.

[55] Mershad, K., & Artail, H. (2013). Finding a STAR in a Vehicular Cloud. *IEEE Intelligent transportation systems magazine*, *5*(2), 55-68.

[56] Brik, B., Lagraa, N., Lakas, A., & Ghamri-Doudane, Y. (2015, September). Finding a public bus to rent out services in vehicular clouds. In *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)* (pp. 1-7). IEEE.

[57] Brik, B., Lagraa, N., Ghamri-Doudane, Y., & Lakas, A. (2016, October). Finding the most adequate public bus in vehicular clouds. In *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)* (pp. 67-74). IEEE.

[58] Brik, B., Lagraa, N., Tamani, N., Lakas, A., & Ghamri-Doudane, Y. (2018). Renting out cloud services in mobile vehicular cloud. *IEEE Transactions on Vehicular Technology*, *67*(10), 9882-9895.

[59] Arkian, H. R., Atani, R. E., & Kamali, S. (2014, October). FcVcA: A fuzzy clustering-based vehicular cloud architecture. In *2014 7th International Workshop on Communication Technologies for Vehicles (Nets4Cars-Fall)* (pp. 24-28). IEEE.

[60] Arkian, H. R., Atani, R. E., Diyanat, A., & Pourkhalili, A. (2015). A cluster-based vehicular cloud architecture with learning-based resource management. *The Journal of Supercomputing*, *71*(4), 1401-1426.

[61] Azizian, M., Cherkaoui, S., & Hafid, A. (2016). An optimized flow allocation in vehicular cloud. *IEEE Access*, *4*, 6766-6779.

[62] Hussain, R., Abbas, F., Son, J., Kim, D., Kim, S., & Oh, H. (2013, December). Vehicle witnesses as a service: Leveraging vehicles as witnesses on the road in vanet clouds. In *2013 IEEE 5th International Conference on Cloud Computing Technology and Science* (Vol. 1, pp. 439-444). IEEE.

[63] Lin, Y. W., Shen, J. M., & Weng, H. C. (2013). Cloud-supported seamless internet access in intelligent transportation systems. *Wireless personal communications*, *72*(4), 2081-2106.

[64] Jafari Kaleibar, F., & Abbaspour, M. (2020). TOPVISOR: Two-level controller-based approach for service advertisement and discovery in vehicular cloud network. *International Journal of Communication Systems*, *33*(3), e4197.

[65] Sumo, simulation for urban mobility. http://sourceforge.net/apps/mediawiki/sumo (accessed on 04/12/2019).

[66] VanetMobiSim, simulation for urban mobility. http://vanet.eurecom.fr/ (accessed on 06/03/2020).

[67] Karnadi, F. K., Mo, Z. H., & Lan, K. C. (2007, March). Rapid generation of realistic mobility models for VANET. In *2007 IEEE wireless communications and networking conference* (pp. 2506-2511). IEEE.

[68] Omnet++ network simulation framework. http://www.omnetpp.org (accessed on 12/05/2019).

[69] ns-2 network simulation framework. https://www.isi.edu/nsnam/ns/ (accessed on 01/02/2019).

[70] ns-3 network simulation framework. https://www.nsnam.org/ (accessed on 03/03/2019).

[71] Veins, VANET simulator https://veins.car2x.org/ (accessed on 12/01/2019).

[72] Bezziane, M. B., Korichi, A., el Amine Fekair, M., & Azzaoui, N. (2021). FR-VC: A novel approach to finding resources in the vehicular cloud. In *Innovative and Intelligent Technology-Based Services for Smart Environments–Smart Sensing and Artificial Intelligence* (pp. 149-155). CRC Press.

[73] Houmer, M., & Hasnaoui, M. L. (2019). An Enhancement of Greedy Perimeter Stateless Routing Protocol in VANET. *Procedia Computer Science*, *160*, 101-108.

[74] Teshima, S., Ohta, T., Kohno, E., & Kakuda, Y. (2011, March). A data transfer scheme using autonomous clustering in VANETs environment. In *2011 Tenth International Symposium on Autonomous Decentralized Systems* (pp. 477-482). IEEE.

[75] Mesquita, D. P., Gomes, J. P., Junior, A. H. S., & Nobre, J. S. (2017). Euclidean distance estimation in incomplete datasets. *Neurocomputing*, *248*, 11-18.

[76] Mershad, K., Artail, H., & Gerla, M. (2012). We can deliver messages to far vehicles. *IEEE Transactions on Intelligent Transportation Systems*, *13*(3), 1099-1115.

[77] Bezziane, M. B., Korichi, A., Kerrache, C. A., & Fekair, M. E. A. (2021). RCVC: RSU-Aided Cluster-Based Vehicular Clouds Architecture for Urban Areas. *Electronics*, *10*(2), 193.

[78] Kerrache, C. A., Lagraa, N., Calafate, C. T., & Lakas, A. (2017). TFDD: A trust-based framework for reliable data delivery and DoS defense in VANETs. *Vehicular Communications*, *9*, 254-267.

[79] Yoon, K., & Kwang, C. L. (1981). Multiple criteria decision making. In *Lecture Notes in Economics and Mathematical Systems*. Springer-Verlag.

[80] Afshari, A., Mojahed, M., & Yusuff, R. M. (2010). Simple additive weighting approach to personnel selection problem. *International Journal of Innovation, Management and Technology*, *1*(5), 511.

[81] Mavromatis, I., Tassi, A., Piechocki, R. J., & Sooriyabandara, M. (2020, May). On Urban Traffic Flow Benefits of Connected and Automated Vehicles. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)* (pp. 1-7). IEEE.

[82] Qi, L., & Gonzalez, C. (2019). Math matters: mathematical knowledge plays an essential role in Chinese undergraduates' stock-and-flow task performance. *System Dynamics Review*, *35*(3), 208-231.

[83] Li, T., Hazra, S. K., & Seah, W. (2005, May). A position-based routing protocol for metropolitan bus networks. In *2005 IEEE 61st Vehicular Technology Conference* (Vol. 4, pp. 2315-2319). IEEE.