Thèse
de doctorat
de l'UTT

Amal HBAIEB

Trust Based Management for V2X

Champ disciplinaire :
Sciences pour l'Ingénieur

2024TROY0002 **Année 2024**

Thèse en cotutelle avec l'Université de Sfax - Tunisie

THESE

pour l'obtention du grade de

DOCTEUR
de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

en SCIENCES POUR L'INGENIEUR

Spécialité : SYSTEMES SOCIOTECHNIQUES

présentée et soutenue par

Amal HBAIEB

le 4 mars 2024

Trust Based Management for V2X

JURY

M. Lotfi KAMOUN	PROFESSEUR DES UNIVERSITES (Tunisie)	Président
Mme Hanen IDOUDI	MAITRE DE CONFERENCES (Tunisie) - HDR	Rapporteure
M. Lyes KHOUKHI	PROFESSEUR DES UNIVERSITES	Rapporteur
Mme Hakima CHAOUCHI	PROFESSEUR IMT	Examinatrice
Mme Samiha AYED	MAITRE DE CONFERENCES	Directrice de thèse
Mme Lamia CHAARI FOURATI	PROFESSEUR DES UNIVERSITES (Tunisie)	Directrice de thèse

Acknowledgments

Ces années de thèse constituent pour moi une expérience inoubliable, tant humaine qu'intellectuelle. En écrivant ces lignes de nombreux souvenirs se sont bousculés dans ma tête. Cet instant marque pour moi la fin d'une expérience d'autant plus exceptionnelle pour moi durant laquelle de nombreuses personnes m'ont aidée, encouragée et soutenue. Ces quelques lignes leur sont dédiées.

De prime abord, je tiens à remercier Mme Lamia Chaari et Mme Samiha Ayed, mes directrices de thèse, pour la confiance qu'elles ont eue en moi en me proposant de travailler sur ce sujet de thèse, ainsi que pour leur disponibilité.

Mes vifs remerciements vont à M. Lyes Khoukhi et Mme Hanen Idoudi, d'avoir accepté de rapporter mon travail. Je les remercie pour la lecture approfondie et l'évaluation de cette thèse. Je remercie également Mr Lotfi Kamoun et Mme Hakima Chaouchi qui ont accepté d'évaluer ce travail de thèse en tant qu'examinateurs.

Un grand merci à l'école doctorale de l'Université de Technologie de Troyes. Tout particulièrement, je suis reconnaissante envers M. le directeur Antoine Grall, et les secrétaires Mme Pascale Denis et Mme Isabelle Leclercq. Je les remercie pour leur attention portée aux doctorants et leur bienveillance constamment témoignée lors de mon séjour en France. Qu'ils trouvent ici toute ma gratitude. J'associe à ces remerciements M. Hassen Mnif, directeur de l'École doctorale de l'ENET'COM et Mme Rim Ayedi, secrétaire à l'École doctorale de l'ÉNET'COM pour son efficacité.

Enfin, je ne trouverai sans doute pas les mots pour remercier mes parents et ma famille pour leur soutien sans faille et leur encouragements constants. Merci du fond du cœur de m'épauler au quotidien et de m'avoir rappelée que je suis assez forte pour faire aboutir cette thèse.

À vous tous, merci.

Abstract

The Internet of Vehicles (IoV) opens up new requirements regarding security, privacy, and trust provisioning. This raises the research question of how trust may be concurrently considered during the design of security solution for the IoV. This thesis addresses this question by designing an IoV security framework that maintains the trust between involved actors. We first propose a trust and Blockchain based framework that relies on reputation and location metrics to validate trustworthiness of the communication system. The Blockchain is leveraged to protect derived trust information from tamper. Next, we extend the proposed framework to better take into account the QoS while enforcing security. We use a clustering scheme to construct the extended framework. Besides, we propose to detect untrustworthy nodes through a lightweight federated learning-based Intrusion Detection System (IDS). We adopt the Software-Defined Networking (SDN)-IoV infrastructure to the network to build the collaborative IDS. The trust features-based detection is proposed along with the SDN to enable a QoS-aware IDS. After that, we proceed to improve the trust-based IDS. We apply multi cluster head concept to boost local detection and overall network performances. Finally, we suggest a reactive Unmanned Aerial Vehicle (UAV)-aided routing solution with security and QoS trade-off. We produce an IoV-UAV_Fog framework for the UAV-aided IoV routing. The proposed routing consists of selecting the optimal UAV relay that maximizes both trust and QoS.

Keywords

Vehicular ad hoc networks (Computer networks)
Trust (digital)
Intrusion detection systems (Computer security)
Routing (Computer network management)

Résumé

L'Internet des Véhicules (IoV) expose des exigences en matière de sécurité et de provisionnement de la confiance. Cette thèse se penche sur la conception d'une plateforme de sécurité pour l'IoV qui prend en considération l'aspect de la performance. Nous proposons tout d'abord une plateforme de sécurité basée sur la gestion de la confiance et la Blockchain et qui s'appuie sur des métriques de réputation et de localisation pour valider la fiabilité du système de communication. La Blockchain est utilisée pour protéger les informations de confiance contre la falsification. En outre, nous étendons la plateforme proposée afin de mieux considérer l'aspect de la performance tout en renforçant la sécurité. Nous utilisons le regroupement pour construire la plateforme de sécurité étendue. Ensuite, nous proposons un système de détection des intrusions (IDS) léger qui s'appuie sur l'apprentissage fédéré et les métriques de confiance. Nous adoptons une infrastructure IoV basée sur le Software-Defined Networking (SDN) pour construire l'IDS collaboratif. De plus, nous améliorons l'IDS proposé. Nous appliquons le regroupement pour renforcer la détection et les performances globales du réseau. Enfin, nous proposons une solution de routage réactif assisté par drone qui offre un compromis entre la sécurité et la performance. Nous produisons une plateforme IoV-UAV_Fog pour le routage assisté par drone. Le routage proposé consiste à sélectionner le drone optimal maximisant à la fois la confiance et la performance pour servir comme relais.

Mots clés

Réseaux ad hoc de véhicules
Confiance numérique
Systèmes de détection d'intrusion (informatique)
Routage (informatique)

Contents

List of Figures

List of Tables

List of Abbreviations

IoV	Internet of Vehicles
QoS	Quality of Service
IDS	Intrusion Detection System
SDN	Software-Defined Networking
UAV	Unmanned Aerial Vehicle
IoT	Internet of Things
VANET	Vehicular Ad hoc Network
ITS	Intelligent Transportation System
AI	Artificial Intelligence
V2X	Vehicle-to-Everything
OBU	On-Board Units
IT	Information Technology
MANET	Mobile Ad hoc Network
V2V	Vehicle-to-Vehicle
RSUs	Roadside Unit
V2I	Vehicle-to-Infrastructure
V2R	Vehicle-to-RSU
DRSC	Dedicated Short Range Communication
WAVE	Wireless Access in Vehicular Environment
ECU	Electronic Control Unit
ADAS	Advanced Driver Assistance System
LoS	Line of Sight
NFC	Near Field Communication
V2P	Vehicle-to-Pedestrian
V2C	Vehicle-to-Cellular
V2S	Vehicle-to-Sensor
V2N	Vehicle-to-Network
V2PD	Vehicle-to-Personal Devices
V2Edge	Vehicle-to-Edge
MEC	Mobile Edge Computing
IoD	Internet of Drones
HD	High Definition
ICN	Information-Centric Networking
CCN	Content-Centric Network
NDN	Named Data Networking
DST	Dempster–Shafer Theory
SVM	Support Vector Machine
IPFS	Interplanetary File System
PKI	Public Key Infrastructure
TA	Trusted Authority

TP	True Positive
FN	False Negative
FP	False Positive
CNN	Convolutional Neural Network
Fuzzy AHP	Fuzzy Analytic Hierarchy Process
SCF	Store-Carry-and-Forward
QoE	Quality-of-Experience
G2A	Ground-to-Aerial
A2G	Aerial-to-Ground
V2U	Vehicle-to-local UAV
R2F	RSU-to-central UAV_Fog
U2U	local UAV-to-local UAV
U2F	local UAV-to-central UAV_Fog
C2F	UAV_Fog Coordinator-to-central UAV_Fog
F2F	central UAV_Fog-to-central UAV_Fog
SINR	Signal-to-Interference-plus-Noise Ratio
NLoS	Non- Line of Sight
GPS	Global Positioning System
LT-REQ	Link Test Request
LT-REP	Link Test Response
R-REQ	Route Request
R-REP	Route Response
R-ER	Route Error
PDR	Packet delivery ratio
EED	End-to-End Delay

Chapter 1

General Introduction

1.1 Introduction and research problem statement

The pace at which technology has developed in recent years, frames bulks of connected smart apparatuses symbolizing the Internet of Things (IoT) [**1**]. The IoT has created the Vehicular Adhoc Network (VANET) for the transportation area . VANET is a vehicle-centric networking [**2**]. With the advent of IoT, VANET network evolves to Internet of Vehicles (IoV) [**3**]. The IoV is an advanced version of VANET. The evolution to the IoV is accelerated by the cross-sectional synergies of the IoT and other enablers such as Artificial Intelligence (AI), 5G/6G, technological infrastructures, big data, robotics, as well as Unmanned Aerial Vehicles (UAVs). The IoV consists of vehicles, infrastructures, people (e.g., drivers, passengers, roadside pedestrians), and other smart connected devices. The IoV synchronizes vehicle's networking and intelligence' high-tech dream. The IoV brings new usages reinforcing the transformation towards automation. The IoV exhibits self-driving [**4**], safety driving, social driving, entertainment services, intelligent mobile applications, and emerging technologies [**5**]. The IoV aims to enhance safety, intelligence, and efficiency of traffic systems. Various Vehicle-to-Everything communications (V2X) exist between IoV nodes. Vehicles in the IoV with advanced sensors, On-Board Units (OBU), local computing units, and gateways [**6**], enabling them to collect, generate, process, analyze, and exchange big environmental data. The IoV is in an evolution stage, and trial runs and research have being performed by Information Technology (IT) companies like Google and Tesla to project the IoV in real use.

However, IoV characteristics such as high dynamic network topology and exposed communication links between IoV nodes inevitably pose exceptional complexity in terms of security challenges. Security challenge is one of the main factors that affect the success of IoV. IoV security is a serious research topic since it impacts directly the lives of its users. Security failure in IoV can threaten human life, as well as causes damages to vehicles, road and city infrastructure. For example, malicious vehicles can alter messages or continuously circulates fake messages. The security concern for a connected vehicle is serious as there was incidents in past [**7**]. Therefore, fulfilling the security requirements is important to instill trust, acceptability, and success in the IoV. Besides, maintaining the QoS during the security evaluation is critical for real-time applications in the IoV.

Actually, a significant gap in the state-of-the-art of IoV security concerns security-QoS balance. Given high restrictive QoS requirements and heterogeneity in IoV, the conventional security methods are inadequate. Several works have adopted conventional security mechanisms for the IoV without taking into account the impact of the suggested solutions on the performance of the network during the security evaluation. How maintaining QoS during security design is a problem worth paying attention to. The IoV security mechanism need to maintain QoS requirements such as low time and low scheme complexity and resource utilization (e.g., memory and energy). As vehicles are becoming more autonomous, a poor QoS is a key factor that triggers an IoV service failure. This imposes restrictions on security mechanism to consider the QoS. Therefore, it is significant to use a trust management mechanism between the IoV nodes to guarantee a security that is aware of QoS. The trust issues in such an untrustworthy environment require an effective trust management approach to enhance overall security. A proper trust management-based security approach can distinguish malicious nodes and mitigate attacks. Trust-based approaches can provide reliability and stability of communication process, which improves IoV safety and boosts the IoV use. The "trust management" concept was introduced by Blaze et al in 1996 [**8**]. Trust management-based approaches can include identity verification, trust and reputation, attack or intrusion detection, and other dimensions. The trust of nodes can be evaluated so that the malicious nodes in the network can be avoided and interaction with trustworthy nodes is prioritized. However, it must be noted that most of existing trust management-based works for the IoV have shown some limitations with respect to QoS features. They have not exhibited the effect of trust management mechanism on the network QoS performance. Besides, the major of trust proposals for the IoV did not take into consideration the system communication architecture.

1.2 Contributions

This thesis focuses on the trust management and the network part of the IoV ecosystem. We address trust and network architecture using emerging technologies. Three aspects of work are proposed in this thesis: (1) trust management framework, (2) trust-based Intrusion Detection System (IDS),

and (3) trust-based routing protocol.

In the first, we propose a trust management and Blockchain based framework to ensure trustworthiness and security of the IoV. The framework architecture is two layered Blockchain that consists of local Blockchain to maintain trust of local nodes and global Blockchain that saves global trust information. The Blockchain technology is leveraged considering its key properties such as decentralization and immutability. We adopt a centralized trust management process. Afterwards, we extend the proposed framework to better consider the QoS metric while enforcing security. We highlight the energy metric as it directly effects the network continuity, and can facilitate the balance of security and QoS. We propose a decentralized trust management to allow for a greater scalability. For that, the trust approach applies a clustering process.

In second hand, we propose a trust evaluation through collaborative IDS in the IoV. The collaborative IDS uses the federated learning technique. We examine the collaborative trust learning under the Software-Defined Networking (SDN) architecture. We use a lightweight detection process to alleviate the complex trust learning process. In fact, collaborative IDS by its distributed learning is a promising solution for securing the IoV. Besides, the adoption of the SDN architecture enables a QoS-aware IDS. The SDN provides some solutions to network limitations allowing for good network performances such as flexible and granular management, efficient resource allocation to run the IDS, and localized decision-making. Besides, there exist few works integrating trust, federated learning, and SDN in an IDS for joint security and QoS in the IoV. Furthermore, we reconsider trust metrics, federated learning and SND, and proceed to enhance the IDS. We integrate the clustering with the dolphin swarm algorithm to improve network topology, intrusion detection, and overall QoS performances. We define nodes to act as dolphin that scans nearby nodes in order to select trusted cluster heads and identify malicious nodes.

In another hand, we apply the concept of trust for the routing aspect in the IoV. Our target is to conduct a routing solution with a trade-off between security and QoS. A routing protocol should be reliable and not vulnerable. As we pinpoint metrics and routing infrastructure limitations in existing routing solutions for the IoV, we suggest a reactive UAV-aided IoV routing. We introduce an IoV-UAV;_Fog framework to facilitate the UAV-aided IoV

routing. The UAVs as Fog nodes find optimal path to destination node. The optimal path is built through selection of optimal UAV relay maximizing both trust and QoS.

The contributions in this thesis are summarized as follows:

- A review of literature devoted to trust management mechanism in vehicular networks. We present existing trust-based approaches for vehicular networks and cover major trust' aspects such as trust proprieties, related metrics, trust modules, and trust challenges). This allows to position our work in the existing literature.

- A trust management and Blockchain based framework to obtain the reliability of the IoV network. We evaluate our proposal to validate its effectiveness. Next, we propose a more scalable solution based on clustering scheme in order to reach good QoS when coping with mobility. Clusters formation phase along with selection of clusters heads step and clusters maintenance phase relies on the trust management' process, the metric of safety distance and the factor of energy. Extensive performance analysis of the proposed approach is conducted.

- An IDS that is based on federated learning technique, SDN architecture and trust metrics to deal with the maliciousness of nodes in the IoV. The objective is to detect anomalies while maintaining the QoS. The IDS learns node behavior using trust-related metrics to decide whether the observed behavior is an anomaly. We apply a lightweight detection process in which SDN controllers are the local IDS nodes. Also, we present an extension of the IDS subjected to enhance the detection process by means of clustering. The performance evaluation for the IDS is provided.

- A trust and QoS based UAV-aided routing protocol for the IoV. The core element of the proposed protocol is the selection of optimal UAV relays forming the best path towards destination nodes when ground routing is unreliable. The performance evaluation is conducted to validate the routing solution.

1.3 Manuscript organization

Throughout the present chapter, we briefly relate our contribution to the context of our thesis. The rest of this manuscript is organized as follows.

Chapter 2 gives background on vehicular networks to define the thesis context and the motivations behind our work. Evolution from VANET to IoV, along with IoV characteristics, patterns, architecture, and applications is described. The chapter concludes with IoV security challenges.

Chapter 3 deals with the state-of-the art on trust management mechanism in vehicular networks. The chapter covers (1) introduction of trust concept, (2) major challenges of trust management in IoV, (3) taxonomy and comparison of works on trust management for the IoV, and (4) Overall discussion of the relevance of reviewed works.

Chapter 4 presents our first contribution, a trust management mechanism which is built on a two-layer Blockchain-IoV architecture. We structure the chapter into two parts. The first part is focused on applying a centralized trust scheme that uses metrics related to node reputation and message credibility. The second part describes a decentralized trust management scheme applied on nodes classification.

Chapter 5 introduces an IDS for the IoV that uses SDN, federated learning, and trust. We outline the collaborative detection of distrusted nodes under an SDN architecture. The chapter focuses on enhancing the proposed IDS in its second part.

Chapter 6 targets an UAV-aided routing solution to accomplish a secure and reliable data transmission condition in the IoV. The potential of combining UAVs and Fog computing to assist in routing for the IoV is discussed in this chapter.

Chapter 7 concludes this thesis and gives an outlook on perspectives for research on IoV security.

Chapter 2

Context of the work: Internet of IoV

2.1 Introduction

This chapter starts with basics on VANET and IoV, and goes through IoV characteristics, IoV patterns and architecture, and IoV applications. Besides, the chapter outlines security challenges to highlight the critical need to adhere requirement of security and trust building in the IoV.

2.2 From VANET to IoV

The IoV is an innovation activated by the VANET. In fact, VANET is a subclass of Mobile Ad hoc Network (MANET) which is an infrastructure-free network [**9**]. VANET operates in two typical communication environments: highway traffic scenario and urban area conditions. VANET applications focus on traffic management, congestion monitoring, and information dissemination among vehicles and infrastructure. VANET has Vehicle-to-Vehicle (denoted V2V) communication and Vehicle-to-Infrastructure (denoted (V2I) or (V2R)) communication. VANET mainly relies on Dedicated Short Range Communication (DSRC)/Wireless Access in Vehicular Environment(WAVE) [**10**], and cellular-based LTE (Long-Term Evolution) standards [**11**]. The typical architecture of VANET is depicted in figure 2.1. The evolution from

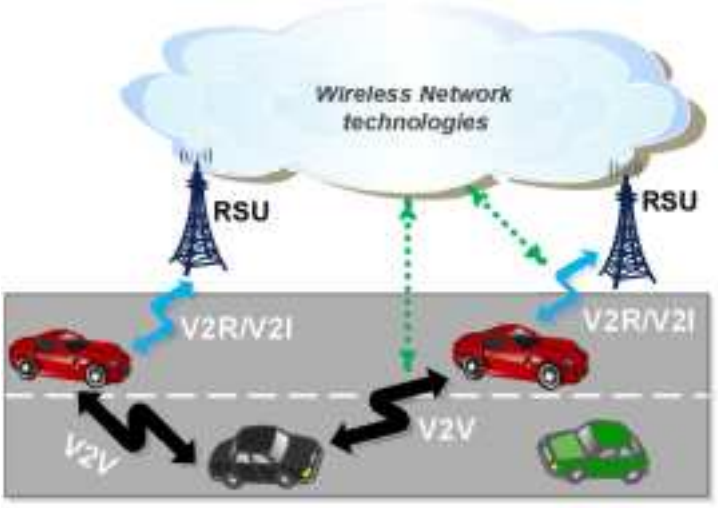


Figure 2.1: VANET architecture

VANET to IoV includes a gradual expansion and trans- formation of the vehicular communication concept. This evolution is driven the needs and expectation of more comprehensive and interconnected ecosystem that covers

more than V2V and V2I and vehicular communication. The IoV surpasses VANET in terms of scope, used technologies, characteristics, communication patterns, infrastructure and architecture, and applications.

Indeed, the IoV connects the vehicular environment with other domains, such as smart city, healthcare, entertainment, and logistic. The IoV integrates server-side computing, advanced in-vehicle computing and communication standards (e.g., 5G and 6G), emerging AI, digitization (e.g., digital twins), and multi-domain collaboration. The multidisciplinary nature of IoV falls under connectivity on a global scale and extensive communication infrastructure. The IoV can combine smart city infrastructure-based, Cloud, Fog and edge-computing-based communications, taking advantage of the strengths of different communication modes. In fact, the IoV can leverage Cloud-based services to enable a range of storage, processing, and networking capabilities to big data, and data analytics (e.g., for vehicle predictive maintenance). Likewise, the IoV applies edge computing and allows for data processing at the edge of network. Besides, the IoV benefits from Blockchain and SDN. The IoV with Blockchain and SDN makes provision for scalable and secure IoV service distribution. Furthermore, the IoV plans the path toward collaborative intelligence which aligns with Federated learning [**12**] (e.g,cooperative intelligent driver model, collaborative perception, collaborative computing, and cooperative decision). The evolution from VANET to IoV accommodates the increasing intelligence of vehicular networks, and embraces the integration of diverse technologies, which makes the IoV context unique and exciting to investigate.

2.3 IoV characteristics

The IoV characteristics can be both challenging and beneficial, depending on how they are addressed. Particularly, these characteristics can introduce additional considerations for managing communication and addressing security challenge [[**13**]-[**15**]]. The specific characteristics of IoV that go beyond VANET are as follows.

- Heterogeneous interactions and complexity of communication

The complexity of communication in IoV is higher than in VANET. While

VANET mainly focuses on V2V and V2R, the IoV goes beyond this by enabling heterogeneous connectivity. In IoV, vehicles can communicate with various entities such as pedestrians, cyclists, smart city infrastructure, and computing servers (Cloud, Fog, edge). This broadens the scope of IoV and enhances its potential for diverse applications. Yet, this diversity in communication entities introduces complex interactions and various communication environments requiring robust technologies to handle the heterogeneity of participants.

- Highly dynamic topology

Vehicles interact with many heterogeneous components in the IoV. Compared with common mobile nodes, vehicles are dynamic and may move with a quite high speed, leading to frequent topology changes and rapid fluctuations in network connectivity. This change in the network presents challenges in maintaining reliable and stable links between vehicles and infrastructure. The high dynamicity of the IoV requires adaptive communication protocols to ensure network reliability and data delivery.

- Network partitioning

Due to the dynamic nature of the topology, the IoV network can experience temporary partitioning where certain zones become disconnected from the rest of the network. This can hinder communication and data dissemination. Considering possibilities of integrating innovative technologies such UAVs, helps to enhance IoV users coverage and assist in exchange between nodes.

- Scalability

Vehicular network has a non-uniform density. The network density varies significantly depending on the traffic density in a particular area. In dense urban area such as city center, or the entrance of the big cities , the scale of the vehicular network is vast. The IoV is required to be highly scalable. Scalability allows to handle the growing traffic and demand for communication without compromising network performance. Nonetheless, a large scale network can lead to increase in communication load between nodes. Deploying an emerging technology-based architecture can permit for smoother communication and data exchange in a large scale IoV.

- Geographical communication

In VANET, geographical communication are mainly used, where packets are forwarded based on geographical proximity. In the case of IoV, communication can occur through various modes to cater to its dynamic nature. The IoV is more comprehensive, and the communication paradigm encompasses both geographical communication and non-geographical communication.

- Persistence of connections

As the IoV encompasses a diverse communication scenarios, the connectivity patterns varies based on communication scenarios. Some IoV scenarios offer stable and long-lasting communications, wherein connections between nodes can persist for longer periods compared to VANET. Such characteristic may extend particularly to privacy and security challenges.

- QoS and real-time constraints

VANET may have more flexible QoS requirements. Yet, the QoS is more challenging in the context of the IoV. A variety of high quality IoV applications imposes strict QoS like very hard delay, optimized energy consumption, and low overhead in scenarios with limited bandwidth.

- Energy, storage, and computing capabilities

The IoV involves nodes with varying resource constraints, which introduces additional complexities. Efficient energy management, data processing, and storage mechanisms are critical in IoV to accommodate the diverse set of nodes and ensure the smooth operation of the IoV ecosystem.

2.4 IoV patterns and architecture

The IoV architecture can be defined from different perspectives.

- From the perspective of communicating node

The IoV network model is depicted in figure 2.2, representing three building blocks of the network. These three blocks are: (1) vehicle zone, (2) cooperative zone, and (3) smart city zone. Following the VANET, the vehicle zone covers the inter and intra communications. The vehicles in the IoV carry innovations such as advanced sensor technology, intelligent and open vehicle terminal system platform, and speech recognition technology. The IoV realizes the cooperative zone through various communication types termed as V2X (Vehicle-to-Everything). Figure 2.3 illustrates IoV under-

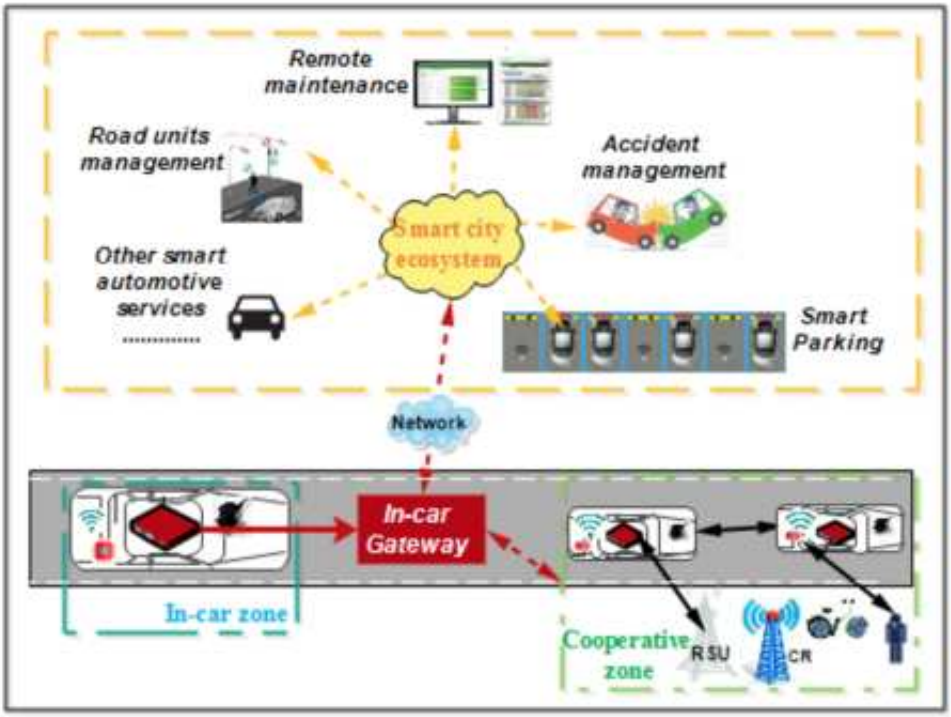


Figure 2.2: Overall IoV communication system

lying cooperative zone and contributing technologies towards V2X. The V2X extends the vehicle communications to all entities, it covers (i) V2V, (ii) V2I/V2R, (iii) Vehicle-to-Sensors (denoted V2S), (iv) Vehicle-to-Network (denoted V2N, such as Vehicle-to-Cloud or Vehicle-to-Fog), (v) Vehicle-to-Cellular (denoted (V2C), (vi) Vehicle-to-Pedestrian (denoted V2P), and (vii) Vehicle-to-Personal Devices (denoted V2PD). It is worth mentioning that the I2I (Infrastructure-to Infrastructure) communication complements the IoV by adding infrastructure-layer interaction within the V2X. The V2P involves the exchange of information between vehicles and pedestrians or their per-

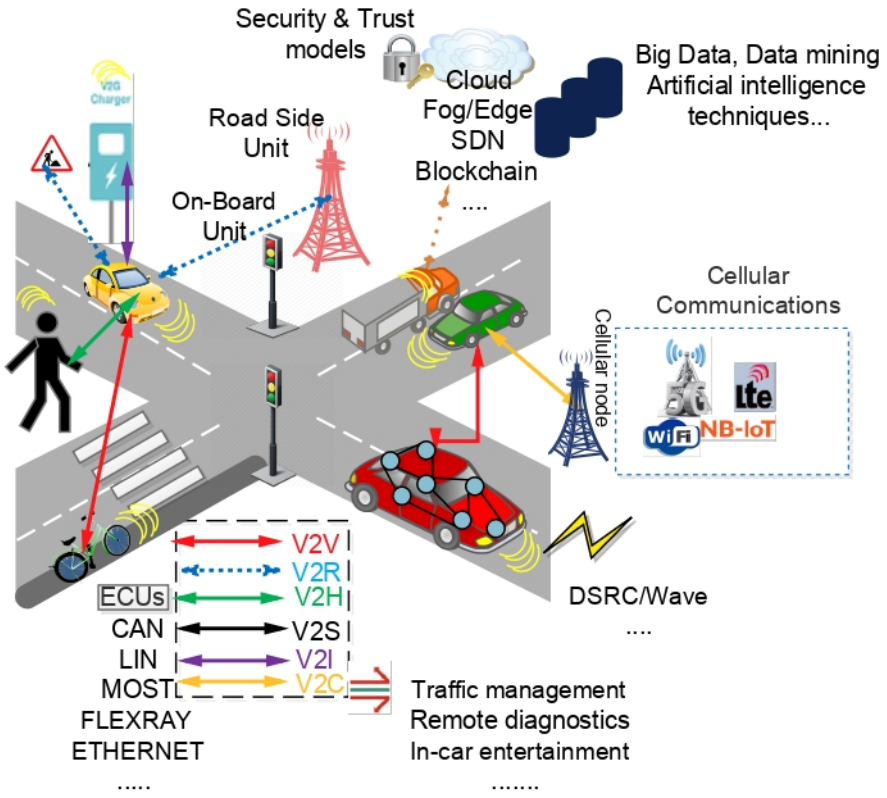


Figure 2.3: IoV cooperative zone

sonal devices. In situations where a pedestrian might be in a blind spot or not directly visible to the driver, V2P can generate alerts to both the pedestrian and the driver. Likewise, the V2P can provide insights into pedestrian intent. For instance, if a pedestrian device indicates that he is about to cross the road, vehicles can adjust their speed and initiate safety measures. The V2P can be integrated with smart city infrastructure, such as smart crosswalks or sensor-equipped pedestrian zones, contributing to pedestrian awareness, enhanced pedestrian safety, crosswalk safety and pedestrian-friendly traffic signals. The V2S involves the use of a vehicle's internal sensors to monitor its self-status and gather data about the surrounding environment. The connected smart vehicle carries sensor information integration, and enables numerous environmental data to be captured and processed into valuable information. The system border of the smart vehicle creates user-centered connected vehicle services. Also, the information that drivers or passengers would like to see can extend the system border of the vehicle to be on a second dashboard (e.g., smartphone, tablet) [**16**]. Thus, the V2S empowers vehicles to share valuable sensor-generated information, assisting

the health of vehicle components (e.g., predictive maintenance), and resulting in enhanced infrastructure interaction, environmental sensing, and cooperative situational awareness. The V2C enables the interaction between vehicles and cellular networks such as 4G LTE, 5G, and upcoming generations of cellular technology. The V2C is used for an extensive communication infrastructure, it provides vehicles broader connectivity and access to services. The V2C can be is particularly valuable for long-distance travel scenario, where continuous connectivity, mapping services, and route guidance are needed for reliable travel experience. Recent projects such as the 3rd Generation Partnership Project (3GPP project) has suggested specifications concerning 4G-LTE-enabled V2X within the IoV ecosystem. The V2N falls under the umbrella of V2-Internet data exchange. The V2N refers to the exchange of data and services between vehicles and network-based services or technological paradigms, such as server-based platform (e.g., Cloud), edge computing resources (e.g., Fog, edge), and software-based platform (e.g., SDN). The V2N can represent the brain of communication system, where services such as storage, computing, big data, analysis units, and network are provided to users. For example, Vehicle-to-Edge (V2Edge) can provide dynamic routing suggestions to vehicles. Finally, the V2PD extends the IoV ecosystem to personal items. V2PD enables vehicles to build connection with personal devices carried by occupants of the vehicle or pedestrians (e.g., smartphones, tablets, smartwatches, and other wearable). The V2PD enhances overall user experience for both drivers and passengers.

- From the perspective of architectural layers

The typical layers in an IoV architecture are often referred to the perception/sensing layer, coordination/communication layer, layer of AI and technology, application layer, business layer, and security management layer (see figure 2.4). The sensing layer involves sensing technologies and devices that gather data from the environment and the vehicle itself. It is laminated with a data filtering and preprocessing. The coordination layer ensures coordination and interoperability among the different components in the IoV. The AI and technology layer encompasses the emerging technologies shaping IoV intelligence and enabling the various aspects of IoV architecture. It provides intelligence, infrastructure for communication, storage, processing, analyzing, and management. This layer includes AI, data analytics, and big data to derive patterns and predictions from the perceived data, as well

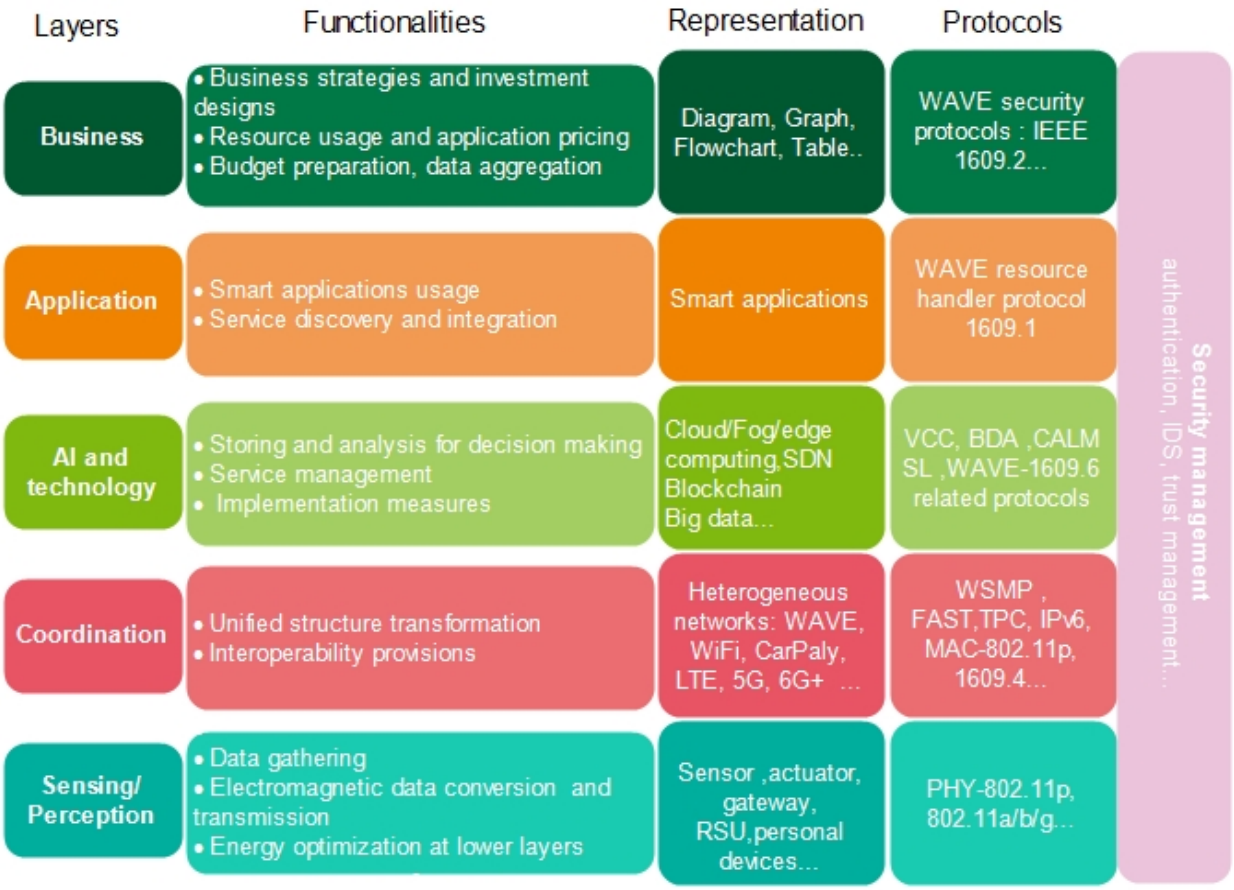


Figure 2.4: IoV layered architecture

as cellular technologies, Cloud and data centers, network infrastructure, and Blockchain. The application layer is where IoV materializes. It's at this layer that the patterns gained from data analysis and AI processing are put to practical use through smart applications. The business layer deals with the monetization, deployment, and management of IoV applications. It includes business strategies, investment designs, service provisioning, and overall system management. The security layer is a transversal layer that spans across other layers to provide security policies. The security layer implements authentication mechanisms, access control mechanisms, encryption mechanisms, intrusion detection, trust management, as well as secure software components' update.

- From the perspective of network model architecture

IoV architecture has largely been driven by different paradigms such as Cloud [[**17**]-[**19**]], Fog/edge [**20**][**21**], SDN [[**22**]-[**24**]], and Blockchain [[**25**]-[**28**]]. The IoV architectures are classified into three broad categories (1) centralized, (2) distributed, and (3) decentralized. Each architecture has advantages and limitations.

1. Centralized architecture: IoV centralized architecture depends mainly on Cloud computing. The Cloud in the IoV permits to alleviate the computation and storage load on vehicles. The use of Cloud involves remote servers as central data center to provide abundant computing resources and storage to process and store data that would be resource-intensive for vehicles. The vehicles in the IoV can connect to the Cloud, directly or through RSUs. Besides, IoV centralized architecture can be defined so that vehicles can form a shared Cloud under the concept of vehicular Cloud Computing. For example, when a driver searches for parking, the shared Cloud provides personalized parking recommendations. Pointing that, the vehicular Cloud computing reduces vehicle requests processing latency compared to vehicles using a traditional Cloud setup. Besides, the IoV-based cloud architecture can combine both vehicular Cloud computing and vehicles using Cloud. This hybrid architecture provides a third-party Cloud service. However, Cloud-based IoV architecture raises security and QoS limitations such as network access devices' energy consumption and lack of mobility support. For example attacks can be present due to long-distance transmission between Cloud and vehicles. Also, in vehicular Cloud computing, a malicious node can disrupt vehicular Cloud services. Moreover, factors like packet loss make the vehicular Cloud computing not less suitable for example for transmission of accurate road-traffic data.

2. Distributed architecture: IoV architecture can rely on Mobile Edge Computing (MEC)/Fog computing to bring the Cloud closer to users. Cloud resources are distributed to edge servers in a MEC/Fog-based IoV, forming a vehicular Edge computing. An MEC/Fog-based IoV architecture implements caching politics, and improves the QoS of the Cloud by minimizing resources accessing delay, reducing offloading costs, and enhancing the location awareness. The typical MEC/Fog-based IoV architecture consists of edge servers (e.g, RSUs), vehicular layer, and Cloud layer. A distributed IoV

architecture can be defined as also to be SDN-based. The SDN separates network control plane from the forwarding data plane. The SDN-based IoV architecture exploits the heterogeneous Edge computing nodes to optimize IoV-edge computing capacity. Such architecture provides comprehensive network control, and offers benefits in terms of QoS such as highly adaptive and scalable management, and efficient resource utilization. The SDN accompanies the IoV-edge computing to deal with complex and heterogeneous computations, infrastructure abstraction, resource utilization, large data flow, and frequent topology changes. Despite the advantages of an IoV distributed architecture, it comes along with the difficulty to secure and uniformly supervise the edge servers because of their high distribution. Edge nodes or SDN controllers are prone to attacks and errors. They might be malicious and thus compromise trust of the network and violate data integrity and transparency.

3. Decentralized architecture: A decentralized IoV architecture allows the network nodes to have a degree of autonomy to cooperate and perform independently decision-making, data processing, and control without relying on third party central entities. An IoV decentralized architecture often offer fault tolerance since there isn't a single point of failure. Lately, decentralized IoV architecture has put forward the Blockchain. Blockchain features like tamper resistance, self-healing and redundancy are promising to overcome security limitations in both centralized and distributed IoV architectures. A Blockchain-based IoV architecture affords resilience and error reduction in most IoV applications through verification and transparency. Such architecture prevents data altering and enables availability of applications. In a decentralized IoV architecture, vehicles or RSUs are often relied upon as Blockchain nodes. Also, the Blockchain-based IoV architecture can be coupled with an Edge Cloud or Fog computing where often the RSUs are supported by edge nodes to manage a Blockchain. Finally, we should state that it is important to look at coordination, consensus protocols, reliability of Blockchain nodes, and size of the Blockchain when adopting a Blockchain-based IoV architecture.

In this thesis, we aim to secure IoV applications with guaranteed QoS. Therefore, Blockchain-based architecture, SDN-based architecture, and Fog-based architecture are the adopted architectures in contributions of this thesis.

2.5 IoV applications

IoV applications are driven by broader range of vehicle types, advanced AI, and enhanced user experience. IoV applications go beyond traffic management and congestion monitoring in VANET to bring autonomous driving (AutoNet2030 [**29**], EU LSPAUTOPILOT [**30**]), smart city services, environmental monitoring, and applications in other different domains like Internet of UAVs/ Internet of Drones (IoD) and Internet of underwater vehicles. In fact, the IoV can integrate underwater vehicles and drones and this diversity enables applications like underwater exploration, aerial support for ground vehicles, and multi- modal mobility. Generally, the IoV caters to personalized applications covering safety, efficiency, user experience, and environmental sustainability.

- Safety applications

These applications focus on accident avoiding to enhance vehicle and pedestrian safety by providing advanced warnings and assisting in critical situations. They are very time- critical that demand immediate and precise responses. There are several safety applications. To name a few examples, emergency assistance and E-call, emergency vehicle priority, pedestrian intent recognition, lane change warning, auto braking. We point that autonomous driving application falls under safety applications since it aims to avoid risk of human errors and accidents. Autonomous driving application contributes to road safety by including features such as lane departure warning and correction, emergency brake assist, recognition of pedestrian and cyclists, and blind spot detection.

- Efficiency applications

The main goal of these applications is to optimize traffic efficiency, vehicle operations, maintenance, and overall efficiency, contributing to smoother traffic flow and reduced environmental impact. Efficiency applications are less time-sensitive than safety applications but they require high availability and reliability. They should be consistently accessible and accurate, and provide updates. Efficiency applications covers (1) traffic management and vehicle-road collaboration service (e.g., path planning, traffic signal preemption for emergency vehicles, green-wave speed guidance, high precision

map, cooperative intersection management), (2) mobility and multi-modal transport (e.g., smart parking, multi-modal route planning, ride-sharing and carpooling optimization), and (3) vehicle and infrastructure monitoring (e.g., vehicle self-diagnostics, infrastructure health monitoring, predictive maintenance, vehicle surveillance). The autonomous driving spans efficiency aspects with applications like path following, lane change warning, auto breaking, autonomous valet services, and unmanned delivery.

- User experience applications

These applications highlight the potential of IoV to redefine the in-car experience and create enjoyable journey for both drivers and passengers. They have more flexibility in QoS requirements compared to safety and efficiency applications. They can tolerate slight delays or fluctuations in performance without compromising the overall user experience. As different passengers may have varying preferences for the user experience, this variability allows for a broader range of acceptable QoS levels. User experience applications prioritize the comfort and convenience of travelers by offering personalized services. These applications outlines device integration such as smartphones and wearable to enable services between vehicle and personal devices. User experience applications provide services such as (1) personalized navigation (e.g., tourist mode), (2) in-car assistance (e.g, voice command, touch-screen interfaces), (3) vehicle customization (e.g., driving modes, climate control), (4) financial services (e.g., usage-based insurance), (5) on broad-information service (e.g., news and entertainment), (6) service recommendation to drivers and passengers (e.g., nearby restaurants, parking lots, entertainment venues), and (7) over-the-air software updates for vehicles (e.g., issue resolution, application integration). Also, autonomous driving application falls under user experience applications. Some of these applications are driving behavior analysis, long- distance travel service, and virtual reality tours.

- Environmental sustainability applications

This class of applications contribute to create a more Eco-friendly driving behaviors, fostering a sense of environmental responsibility. For example, Eco-Driving coaching is provided to drivers, guiding them to adopt fuel-efficient driving practices. Also, these applications introduce Eco-driving

support through applications like energy-efficient routing driver behavior reports, and driving scorecards. When combined with Eco-friendly vehicles such as electric or hybrid propulsion systems, autonomous driving application can enhance the environmental benefits. For example, electric autonomous vehicles emit zero tailpipe emissions, further reducing the carbon footprint.

2.6 IoV security challenges

Security is non-negotiable for the vehicular network context. Many of attacks in the IoV share scenarios with other IoT applications. However, severity of certain attacks on IoV are heightened due to sensitivity and vulnerability stem from the critical nature of vehicular interactions. For instance, attacks scenarios on the IoV seek to tamper with traffic signals, inject bogus collision information, affect the packet routing (e.g., blackhole, wormhole), or even interfere with autonomous driving functions. Such disruptions could lead to chaotic traffic situations, collisions, vehicle damages, and life-threatening accidents. Thus, the threats are remarkably higher in the IoV. Besides, security becomes more concerning when the V2X communications are used for sensitive scenarios such as the healthcare scenario when V2X communication involves critical interactions with ambulances, medical facilities, and emergency responders. A security threat can disrupt these interactions, delaying potentially life-saving interventions. Therefore, highlighting the security is a key link for the IoV. This aspect is the main concern of this thesis.

In fact, the multifaceted nature of security in the IoV covers authentication, data integrity and tamper-proofing, data confidentiality, access control, resiliency and error detection, availability, privacy, and non-repudiation. Research works on such security requirements most commonly have explored (1) cryptography techniques, (2) identity-based methods, (3) signatures and certificate-based authentication, (4) dynamic access control and privacy preservation techniques, and (5) IDS. More recent works have integrated behavioral analysis and Blockchain technology. The effectiveness of used solutions has been evaluated with regard to different performance criteria (e.g., particular attacks). Besides, as the IoV landscape changes, the IoV leads to additional security requirements tailored to the context for an

adaptive security such as (i) trust management, (ii) aligning security with network topology and architectural design, (iii) collaborative security, or (iv) extra layer security.

- Trust management

It is pivotal to dynamically establish trust levels among IoV nodes and identify untrustworthy nodes. Sources of messages should be trustworthy in order to ensure that malicious information do not reach destination. Furthermore, trust of provided information can be verifiable to ensure data integrity. Assessing trust on provided data can be vital to ensure data integrity. Therefore, trust-based security framework can enhance reliability of the IoV. Trust management becomes even more critical in contexts like autonomous driving (e.g., cooperative perception scenario). Also, trusted third-party involvement such as distributed trusted central Fog nodes can be beneficial in securing the network. For example, trusted central Fog node can analyze traffic patterns and manage trust levels among sub-nodes. Besides, trust establishment is crucial in the context of routing. The trust-aware routing enables more secure data transmission. Pointing that trust-based approaches are highlighted to provide security solutions that are QoS-aware.

- Aligning security with network topology and architectural design

A well-organized architectural design can incorporate safeguards that make it challenging for attackers to manipulate network topology to their advantage. Indeed, aligning security with network topology can improve the efficiency of security processes and response mechanisms. For example, clustering-based topology can contribute to manage nodes group communications more securely. Also, emerging technology-based IoV topology can bring advantage to security solutions. For example, Fog-based topology provides a structured framework for IoV and allows for more localized and responsive security measures. In such topology, specific layers can be assigned to particular security-related tasks. This ensures that security responsibilities are well-defined and distributed throughout the architecture. Moreover, SDN can be coupled with innovative detection techniques to more coordinated responses to security incidents. Likewise, a joint clustering and Blockchain scheme can be implemented for a secure decentralized framework for the IoV.

- Collaborative security

Collaborative defense strategies can be employed to collectively trace the appearance of malicious nodes (e.g, collaborative IDS). For example, Federated learning brings solution for collaborative security. Federated learning is as well a good solution to protect IoV privacy. Collaborative security highlights also the exchange of security information between nodes (e.g., detected spam messages) to collectively block malicious sources.

- Integration of UAVs for an extra security layer

The UAVs are expected to provide flexibility, ubiquitous connectivity, and sufficient coverage in the IoV paradigm. The UAVs can be quickly deployed to designated areas whenever they are needed. UAVs can be integrated into the IoV to provide additional data and situational awareness to ground-based security systems. The UAVs can monitor the behavior of ground nodes; looking for anomalies or suspicious activities.

As we continue to traverse the terrain of IoV security, security measures are becoming increasingly sophisticated and integral to the architecture of the IoV. We believe that trustworthy communication system along with optimized network topology, and adaptive secure routing algorithm are key enablers to boost the IoV. Therefore, we draw attention to trust management, collaborative IDS, and trust-based UAV-aided routing for this thesis.

2.7 Conclusion

The IoV is an important component of intelligent transportation systems. In this chapter, we presented the transition from VANET to IoV. Regarding network model architecture, we stated that distributed and decentralized architectures are reliable to boost the IoV. Furthermore, we presented IoV applications by classifying them into four categories based on the primary objective and functionality of the application. The chapter was concluded by presenting IoV security challenges along with promising security-related directions. To sum up, there is a need for a high-performance, secure, and stable IoV. We believe that trust-based solutions that align with IoV architectural design are very promising for a reliable IoV. The following chapter

provides a review of trust management mechanism in vehicular networks, which lays foundation for our first contribution.

Chapter 3

General background and related works: trust management in vehicular networks

3.1 Introduction

In this chapter, we provide a review for trust management within vehicular networks. We highlight the integrated methods for the trust management process within the vehicular network context. We apply a taxonomy of related works from the aspect of used tools. This helps to pick out the advantages of each applied tool for the trust management process. We start by exposing an overview of basic trust notion and modules. Then, we discuss the research methodology. Afterwards, we present major trust challenges within vehicular networks context. Next, we classify the trust management-based works for the vehicular network context. Finally, we give an overall discussion of the relevance of reviewed works.

3.2 Overview of trust management in vehicular networks

Trust management mechanism is one of potential security solutions that are needed to assist in a reliable IoV deployment. Trust management refers to assign trust degrees to nodes during their interaction. Trust factor is a characteristic measured by a trustor node as a quantified belief, and a trustee node (i.e., host node), to mitigate the bad impact of actions of malicious and selfish nodes. We need to consider misbehaving and selfish nodes jointly. Yet, the segregation between the meaning of these two terms is nevertheless helpful and beneficial. The node which generally aims at intentionally leaving other entities' ordinary behavior is known as a misbehaving node. It consists usually in the willingness to spread disperse and inject falsified, malicious or fake data, or reach a deny services, while transmission. Whereas, a node is termed selfish once it seeks to attempt one's proper interest and looks at obtaining a benefit that can be served at the expense of other nodes. From the intuitive segregation of definitions, we could draw that a malicious node is considered selfish, for instance, once its behavior refers to benefit a rise mutual resources utilization (e.g., bandwidth coverage) while declining sharing its owns (referred rather a greedy node), or its actions are uttered, for example, in prevailing the higher communication quality (called also strategic node). The trust proprieties can be summarized in direct vs. indirect trust, local vs. global trust, dynamic trust, asymmetric trust, subjective vs. objective trust, history-based trust, and context-based trust. The

trust factor can be used as a by-product to improve vehicular networks services like routing [**31**], relay selection and information dissemination [[**32**]-[**33**]]. Trust management-based approaches are becoming increasingly favored by academic and industrial researchers. A significant number of surveys were conducted on trust management in vehicular networks. The major publications are from 2017 (e.g, [[**34**]-[**42**]]. The topic has mainly been explored within the VANET. In fact, different QoS-related performances aspects and requirements like robustness, dynamicity, scalability, autonomy (e.g., auto-configuration, and auto-optimization), complexity, communication overhead, and resource constraints (e.g., energy consumption) should be more considered in trust-based vehicular networks realization. The main intent is to reach better secure Quality-of-Experience (QoE) for IoV services users.

3.2.1 Research methodology of trust-based approaches in vehicular networks

We aim in this chapter to identify, classify, analyze, and synthesis the works on trust management for the vehicular networks context, in order to provide a summary of the works done in this field. First, we establish a strategy for selecting the relevant works. Our literature search comprises the definition of search strings, source bases, and inclusion and exclusion criteria. Furthermore, defining the research questions is significant in reflecting the importance of the related works to be chosen. The research questions to be answered for our study are the following:

- What are the common concepts for trust management in the literature?
- What are the metrics utilized to estimate the trust?
- What are the adopted tools in trust management?
- How we can classify the trust approaches based on the used tools to later interpret their efficiency?

We considered different search criteria as exclusion criteria in the activity of related works selection. Hence, the papers not related to the defined research questions, and did not present scientific contributions on security

aspect in vehicular networks context were excluded. The selected search criteria correspond mainly to the year of publication, the citations number.

3.1.2.1 Search string and source bases

The relevant selection of keywords is crucial to ensure the identification of the vital papers useful to answer the defined research questions. We started by applying the main search terms to extract the preferred results. Initially, we proceeded a broad search for papers on Google Scholar and IEEE Xplore Digital Library using "Trust" + "VANET" + "vehicular networks" as keywords. We received an excellent coverage of related works The amount of obtained papers was about 415 on IEEE Xplore Digital Library. The search results denoted also that "reputation" represents a main term related with "Trust". However, to treat the massive results and find vital new articles, we reformulated the keywords to be "trust management" + "internet of vehicles" + "VANET" + "reputation". This served to provide us enough relevant works. The second set of keywords focused on the adopted tools in trust management within the vehicular context. We used the following search queries: "trust" + "vehicular network" + "internet of vehicles" + "reputation" + ("machine learning" or "Cloud" or "Edge" or "Fog" or "SDN" or "Blockchain")

3.1.2.2 Inclusion and exclusion criteria

As previously mentioned, inclusion and exclusion criteria serve as an important way to extract relevant works. The main conditions to select exhibited related works correspond to the year of publication and the number of citations. The use of demographics filter was to realize and assess the evolution of the trust management concept over time, as well as to include the recent papers in case of multiple works introducing very similar schemes. The rational behind considering the number of citations was that we believe that papers having high citations have more impact and pertinent scientific potential. Nevertheless, for papers published from 2018, this criteria was not necessarily applied, since these works are considered recent. We raised also some questions related to conducted experiments to further decide the papers' quality (e.g., the experiments were properly explained?, Does experiments support the suggested idea in the paper?). With these criteria, we selected the papers having acceptable number of citations (with an average

of 5 citations per paper) in different relevant databases, from 2008 to present.

3.3 Trust management fundamental concepts

Trust management mechanisms are widely deployed to secure network environments. However, this concept has been initially defined in other contexts. Considering the sociology domain, the trust meaning is associated to persons and represents one of the values to build the social relationships. From another point of view, in the psychology discipline, the trust concept represents a relation between two persons the trustor and the trustee. The trustor will believe that the trustee will do exactly what is expected. The trust relationship leads to a security and optimism feeling if it succeeds, otherwise, it leads to insecurity and mistrust feelings. In psychology, trust can be impacted by the life experiences and it cannot be regained if lost. The trust concept has also been so attractive to secure networks within the computer science domain. In this context, the trust is associated to the network entities or nodes. It represents the probability of a node to be honest. When this probability is lower than a threshold, then the node could be considered malicious. This subjective probability can vary from 0 when the node is completely distrusted to 1 when the node is completely trusted. When considering the IoV ecosystem, the trust concept is applied to all the entities composing the IoV environment (i.e., vehicles, devices, humans, infrastructure entities).

3.3.1 Trust properties

The trust may have many properties. To deal with them, we consider the trust as a relationship between a trustor and a trustee. The trustor is the entity that has trust on the trustee. The trustee is the entity that is considered as trustworthy. The properties of trust can be defined as follows:

- Direct: when the trust is derived from the direct relationship between the trustor entity and the trustee entity.

- Indirect: when the trust is derived from the recommendations propagated from different neighbours to the trustor.

- Subjective: when the trust is calculated by relying on a personal opinion of the trustor.

- Objective: when the trust is calculated based on well known parameters about the trustee entity.

- Local: when the trust value is only available for the trustee and the trustor. The value cannot be propagated across the network.

- Global: each entity within the network is assigned to a unique trust score that is known by all the entities of the network.

- Asymmetric: when an entity x give trust on an entity y, nevertheless, y does not give trust on x.

- History-based: when the trust value is derived by relying on past behaviors.

- Context-dependent: Trust is related to some contextual conditions (related to the network environment for example) or events.

- Composite: when the trust is calculated based on different parameters like security, honesty,...

- Dynamic: when the trust value can be updated with time if any change occurs on the parameters (for example the network topology) used to calculate the initial trust value.

3.3.2 Trust metrics

A holistic view on the proposed trust management approaches shows that different metrics are applied (rather in different ways) for trust measurement and evaluation. According to the related state-of-the-art, trust computation includes the following typical metrics.

- Reputation-based metrics

In this case, the trust is calculated based on recommendations given about a specific network node. The network nodes may share the same opinion about a node that is propagated within the network. In this case, we are considering a major opinion or a global feedback about that node.

- Knowledge-based metrics

The trust is derived from a direct or a past experience that a node has or got with a specific node. These metrics can be useful for example to detect the selfish nodes within the network.

- Expectation-based metrics

In this case the node will calculate the trust of another node based on how it is expecting that node behavior. Its expectation may rely on its history with that node, on received recommendations or only on an initial prediction when no previous communication exists with that node.

- Node properties-based metrics

In this case, the trust formula is mainly based on a set of node properties like speed, direction, resource availability, packet dropping rate, etc.

- Proximity-based metrics

In this case, the trust is calculated based on the main parameters of proximity with the considered network node such as the time, the location as well as the distance.

- Environment factors-based metrics

In this case, the trust formula includes some environment parameters or properties like the network density, the weather conditions, the considered network area and its historical context of threat, or the type of node and

network topology (for example the presence of cluster heads), when dealing with an IoV network.

We remind that major trust metrics inherit its properties. From reviewed literature, we notice that reputation, knowledge, and proximity-based factors are the most employed metrics. However, trust metrics are often properly selected based on approach design purpose (or rather according to different criteria, such as accuracy, dynamicity, and required time and resource for computation). We can also give classes to trust metrics, as in [**43**] which identified (1) trust scale class (i.e., trust described by continuous, or discrete values), (2) trust facets class (e.g., trust described by pair, or triplet values), and (3) trust logic class (i.e., trust described by probability, fuzzy values). Besides, trust can be distinguished in different types like blind trust, conditional trust, or unconditional trust.

3.3.3 Trust computation

Trust computation includes different components (see figure 3.1). Common considered modules are briefly explained below. Once trust is established it is managed for the duration of target nodes interactions.

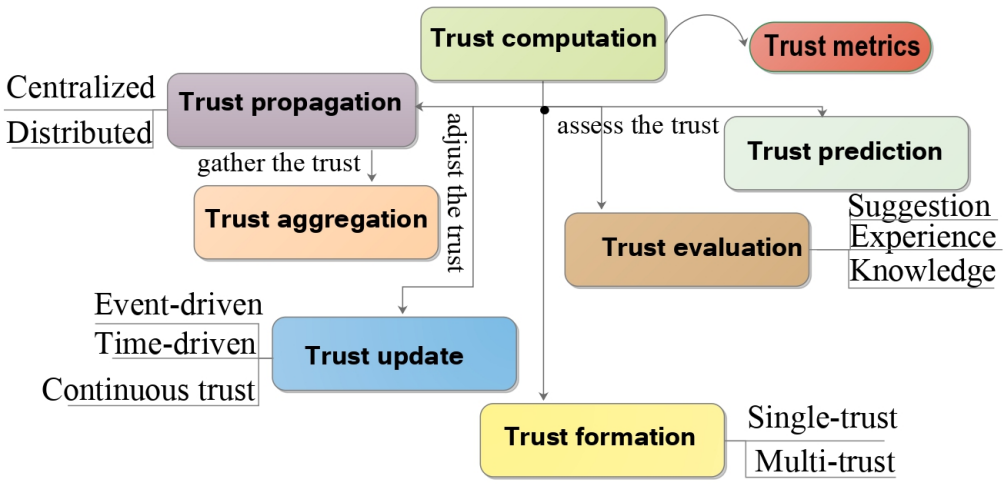


Figure 3.1: General modules in trust computation

- Trust Propagation

Trust propagation module refers to the principle of deriving trust among different communication system nodes, based on generated relationships and pre-existing trustworthiness values while collaboration (e.g., recommendations). Trust propagation is noted as centralized approach where trust is propagated to entities through centralized node or technique and distributed scheme where propagation do not requires central agent [**44**]. Trust transitivity property and also trust fusion are the core factors of trust propagation. Such module provides key benefits. Resources computation cost might be mitigated when measured trust value is propagated over the network, instead of determining each individual entity trust. Moreover, users who are globally trustworthy may command better influence for services.

- Trust Aggregation

Diverse versions about a trust of a node can be propagated through various network paths. When receiving different trust values about a node, the aggregation module aims to define one single value based on an aggregation of received values. Fuzzy logic, weighted sum approaches, and Bayesian model are the major applied techniques for aggregation. Trust aggregation is the principle of composing trust based on a trust path of the different received values.

- Trust Update

Trust update refers to updating trust value, which is a very significant aspect. There are three schemes in trust update. (1) Event-driven trust: node trust value get adjusted after an event occurrence or while transaction making, for instance, when a node is entering, or hereupon a feedback on the quality of a provided service is transmitted for trust aggregation operation. (2) Time-driven trust: concerns when node trust value is applied for adjustment within a determined time period using the aggregation scheme. (3) Continuous trust update: this serves to control one specific node tasks, and consists mainly of protecting integrity.

- Trust prediction

Trust prediction module aims to potentially predict trust relationships among entities based on chosen metrics. In other words, it refers to guess whether a truster node will trust another.

- Trust evaluation

Trust evaluation module includes generally experience, suggestion, and global knowledge parts. Experience is forwardly calculated by requesting the neighbours of the node and gets upgraded at regular intervals in the trust table, then it will be transmitted as a recommendation trust piece. The evaluated trust value is then joined to the global knowledge part. These segments can serve for efficient trustworthiness assessment.

- Trust formation

Trust formation defines the trust formula. To define how the trust will be calculated, we should define the set of trust properties and metrics that should be considered for the trust formula. For that, two trust categories can be defined for the trust formation module:

1. Single trust: In this case the trust formula is almost simple, it is considering only one specific property like direct (when we only consider the previous direct exchanges with the node), indirect (when we consider the different recommendations built about the node), etc. Also, for the single trust, only one metric is almost used for the evaluation. For example, when we consider the direct property, we can use the knowledge-based metrics for evaluating the trust. For the indirect property, the reputation-based metrics are suitable to evaluate the trust.

2. Multi-trust: When the trust formula satisfies more than one property and is evaluated based on more than one metrics type then, we are dealing with multi trust category. Indeed, the multi- trust serves to optimize the trust value and it minimizes the error rate about the calculated value. Considering many metrics helps to get more accurate and reliable trust values.

3.4 Trust management challenges in IoV

Many challenges can weaken the trust management schemes while developed in the vehicular environment. V2X users look for trustworthiness in derived relationships while cooperation. Thus, trust-based approaches are concerned with malicious and selfish nodes. There are some literatures covered trust challenges that target different applications in the vehicular environment [[**37**], [**45**]-[**49**]]. For instance, the authors focused on the adversary-oriented trust management in vehicular networks in in [**49**]. Authors of [**45**] discussed trust-based challenges in vehicular networks. Similarly, authors of [**37**] surveyed trust management in vehicular networks from application and performance perspectives. In nutshell, the major trust challenges are related to the safeguard against attacks on trust and QoS. For instance, good trade-off in terms of trust and privacy is still claimed, since node identity disclosure affects trust relationships, and hence the communication process [**35**]. Trust models threats can be also the outcome of malicious attacks on hardware devices, or end-to-end trust management-enabling technologies like the security issue of 5G integration [**35**]. Examples of major trust-related attacks are mentioned below.

- Bad-mouthing attack

Bad mouthing is a well-known form of reputation-based attacks. It can ruin the good trust reputation of nodes through providing bad recommendations [**46**]. Attacker attempts to send fraudulent trust messages to tarnish the reputation of legitimate nodes. The attacker aims to undermine the accurate assessment of trust.

- Selective misbehavior attack

Attackers deliberately disseminate bogus information to specific network' nodes while behaving normally with others [**46**]. This leads to inconsistent trust evaluation among nodes making it challenging to identify the deception. Reputation systems are effective in such context.

- On-and-off attack

Malicious node switches between periods of normal, benign behavior and sudden malicious activities [**46**]. It starts by behaving normally to evade bad reputation. The primary goal of this attack is to deceive and gain the trust of other nodes before launching malicious activities. For example, a node may behave well, join a platoon of vehicles, and then suddenly disrupt the platoon's communication or manipulate traffic information. Detecting on-and-off attack is very challenging because malicious nodes initially appear as trustworthy and only reveal their malicious intent after gaining trust. However, nodes exhibiting sudden behavioral changes are more likely to breach trust thresholds.

- Time-dependent attack

Time-dependent attack involves nodes that dynamically change behavior over time rather than remaining constant [**46**]. Unlike on-and-off attack, malicious node behavior evolves gradually over time. On-and-off attack involves sudden switches, while time-dependent attack features gradual changes. To mitigate this kind of attack, trust management scheme should use adaptive thresholds that consider the temporal dimension. Temporal analysis is needed to monitor rate at which a vehicle's behavior changes. Abrupt changes in behavior may indicate a potential threat. Also, trust values assigned to nodes can be time-stamped to reflect their behavior at different points in time.

- Selfish attacks

The selfish node behavior comprises malicious activities such as repudiation attack, wherein a greedy node aims to deny communication and to cause the loss of nodes' actions tracking. Self-promoting attack is selfish behavior that can comprise the trust scheme [**46**]. It refers to malicious behavior where nodes act in a non-cooperative manner to gain personal benefits and falsely promote themselves as trustworthy. These nodes prioritize their interests over the cooperation system. Incentive-based trust schemes that provide nodes with motivations or rewards for participating in a more cooperative manner cant prevent such attacks.

- Whitewashing attack

During a whitewashing attack, a node with a previously low trust rate attempts to regain trust and acceptance by adopting a new identity when reentering the communication system [**46**]. The goal is to erase its previous negative history. Assigning relatively low trust values to newcomers nodes can respond to such behavior.

- Tampering trust data attack

The aim of tampering attacks is to deliberately destroy, manipulate, or edit data [**50**]. A tampering trust data attack involves malicious nodes attempting to manipulate or falsify trust-related information or metrics exchanged among nodes. Trust data is crucial for IoV security, as it helps determine the reliability of other entities in the network. Tampering with trust data leads to unsafe V2X communications. The IoV can enhance its defenses against tampering trust data attacks through a Blockchain-based trust ledger. Trust data recorded on the Blockchain cannot be altered without consensus from network participants, making it highly secure against tampering.

3.5 Related works

A search of the literature revealed that trust management approaches are usually categorized into the following common classes: (1) entity-based trust approaches, (2) data-based trust approaches, and (3) combined trust approaches. On the one hand, we can further divide these classes in general in terms of reputation and knowledge, similarity, and utility [**34**]. On the other hand, these approaches can rely on miscellaneous tools (e.g., enabling techniques, and network-advanced architectures) to take their merits for more efficient trust management strategy. Accordingly, we review in the following section some classical examples (e.g., simply inspired by probabilistic logic) of trust management solutions for vehicular communications in each aforementioned class. Next, we present our taxonomy for the recent related approaches, which is based on used tools. We broadly focus on AI and advanced technologies tools. Here, our taxonomy can be viewed as a subclassification of above existing classes. Figure 3.2 draws our review of related works, which consists of presenting our taxonomy for trust manage-

ment approaches in vehicular networks, along with resume tables of the surveyed approaches (Tables 3.1–3.2, 3.3–3.4, 3.5–3.6) in following section 3.5) containing related used trust metrics, applied tools, simulation experiments, and comparative criteria. Figure 3.2 summarizes also the existing trust- approaches in vehicular networks (entity, data and hybrid based).

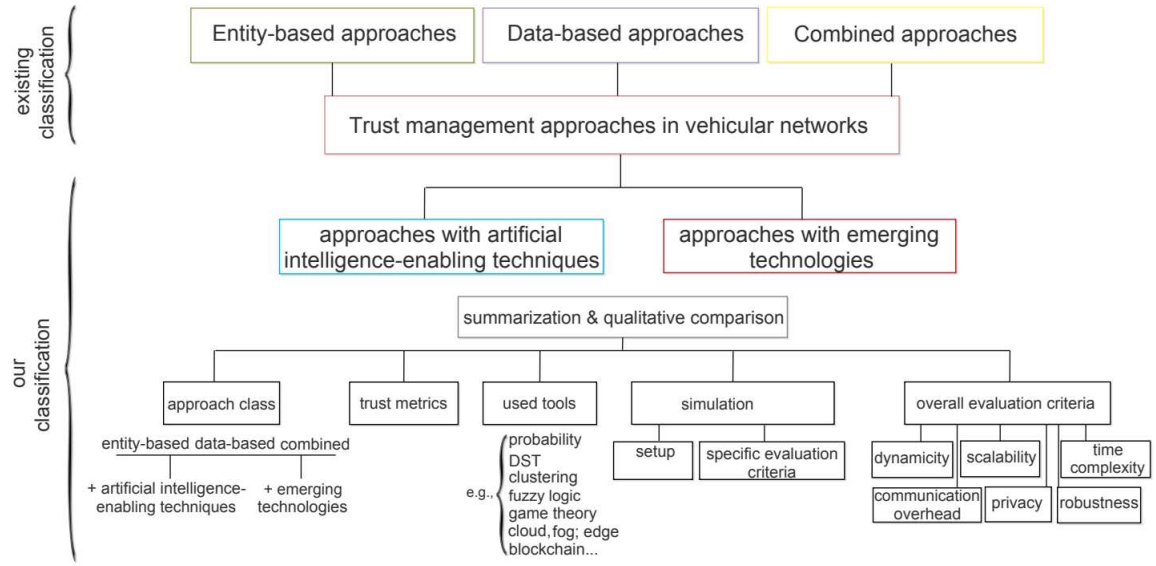


Figure 3.2: Classification and comparison of related works

3.5.1 Existing classification for trust management approaches

The entity oriented approaches consider that the trust is associated to the network' nodes. These approaches aim to assess the trust of the nodes participating in the data exchange within the network. Then, based on the assessment of their trust level, untrustworthy nodes may be excluded from the network or isolated. A part of the existing entity-based approaches in literature are inheriting from the social trust dimension. These approaches are mainly considering reputation-based metrics. Other works are considering a multifaceted trust. Indeed, in addition to the reputation-based metrics, they consider the similarity factor. This latter refers to entities having the same properties. For example, within an IoT network, a vehicle belonging to a specific cluster may consider that the vehicles belonging to the same cluster are more trustworthy than those belonging to other clusters. For this

example, we can rely on the proximity of the node to assign a better trust value to the trustee node. In data-based approaches, trust is linked to the produced message content, which means that these solutions require data authenticity instead entity legitimacy evaluation. Utility is an important aspect to evaluate the data content's trustworthiness. The utility is introduced to refer a specific beneficial act, a worth or usefulness of produced event, in comparison with other actions in same context. Data utility assessment uses often such trust factors: proximity, time, vehicular node role, and occurred event type. Hence, data-based approaches can be distinguished into information oriented methods and event oriented methods. Similarity aspect has been also introduced to assess data trust value. Initially, similarity refers to exchanged data contents coincidence, regarding some parameters like time and closeness. This fact helps in reducing disseminated data amount and ensuring that only useful contents are spread. Nevertheless, we cannot reach typically the assessment of trustworthiness of each exchanged messages part with this model. Moreover, data sparsity represents the major issue for this class. Combined trust management approaches relies on trust of both entity and exchanged data. Entity trust assists in the estimation of data trust value [**51**]; The data content that has been considered to be reliable by trusted nodes is suggested as trustworthy.

3.5.1.1 Entity-based trust management approaches

Authors of [**52**] proposed a reputation-based approach to identify selfish and misbehaving entities. The approach relied on three trust metrics to assess the reputation value of every single vehicle: (1) past and direct entity experiences, (2) surrounding vehicles recommendations, and (3) recommendations from the infrastructure (through available RSUs). Likewise, the proposal considered trust levels (Not trust, +/- Trust, and Trust), and severity levels (high, medium, and low) to be assigned, respectively, to each vehicle and transmitted message, to determinate the received content acceptance. This task was performed through probability rules. The content with "high" severity level is accepted if its issuer reputation score belongs to "Trust" level, while other contents with lower severity level are accepted from entities placed in "+/- Trust" or "Trust" level. An example of similarity-based approach was presented in [**53**] to cope with the injection of false information within VANET safety-related events reports. The similarity rating

was derived through periodic beacons that carried location and speed information. Authors relied also on an echo protocol to achieve trust rating and validate produced reports i.e., supervising the ordinary and anticipated behavior of neighbour vehicles in regard to their reported event. Another multifaceted-agents trust modelling solution for VANET environment was referenced in [**54**]. The trust computation included agents trust maintaining part based on priority notion, and majority opinion-based trusted agents feedback aggregation part. More specifically, the process of trust computation consisted of forming a selected agents list, ordered based on priority metrics, for advice asking/ report. When receiving responses, the requester entity proceeded to majority-based trust calculation. The feedback was followed once reaching majority response consensus, otherwise the requester entity followed the advice of highest trust rate list agent. Similarly, authors in [**55**] designed a multifaceted trust scheme for agents in VANET. The trust values of honest nodes were maintained to demand their related feedback. The authors considered the role, the experience, the majority opinion, and the priority metrics to determine the trustworthiness and ask the proper advisors. The nodes were considered having : (1) authority role, (2) expert role, (3) seniority role, and (4) ordinary role. The number of interactions was taken into account to measure the experience factor. In addition, a forgetting factor was introduced to deal with the behavior changes. The highway platooning scenario was addressed in [**56**]. The reputation was employed to rank the platoon head vehicles. The system model included a server to evaluate the head vehicles' trust. The reputation values were computed by the collection of feedbacks from vehicles. The system applied an iterative filtering to exclude the feedbacks of the malicious user vehicles. The authors of the work [**57**] elaborated a trust inference scheme for VANET to cope with black/grey hole attacks by quantifying subjective trust and recommendation trust. The trust was calculated based on historical interactions, whereas the recommendation trust was derived based on neighbours opinions. Reference [**58**] highlighted the authentication based on trust assessment in VANET. The authentication process involved the evaluation of direct trust; computed from observed behavior, and the estimation of the indirect trust based on the given recommendations. Direct trust values were maintained by authority units. The indirect trust was adopted to allow all the vehicles in the network to validate the new accessing node. The method used correlation coefficient to identify the malicious vehicles and remove

their recommendations. Then, the average of recommendation trusts was calculated to obtain the final recommended trust. Likewise, a reputation-based message authentication for 5G-enabled vehicular was developed in [**59**]. In this model, a trusted authority node was in charge of reputation management to decide about the network access for the vehicles. A vehicle with a low reputation value was not permitted to obtain the credit reference from the trusted authority node to participate in the communication. 3.5.1.2 Data-based trust management approaches Authors in [**60**] were the first to assume that entity-based trust assessment is not enough. They introduced an establishment of data trust suitable for VANET context. Each type of node has a predefined trust value. Reports by each node type on each kind of events were evaluated using default trustworthiness ranking and other trust metrics derived from security status value, as well as dynamic nodes attributes such time proximity and location closeness. In [**61**], the authors introduced an approach to identify rogue node in VANET based on messages similarity. Each vehicle calculated its own flow value based on speed and density parameters correlation, using the Greenshields traffic model. When receiving message (i.e., flow value), the vehicle compared it with its estimated flow measurement. Afterward, the vehicle decided about similarity of received content and its own estimation. Authors in [**62**] introduced a trust management scheme wherein data trust was assessed through computation of confidence and trust values for every received content about particular event. The confidence measurement was based on location closeness, time information, location correctness, and data freshness. In a second step, the method proceeded to trust value measurement on the basis of sender vehicles number and their confidence values. The trust on information in VANET was considered also in work [**63**], where the RSU node was employed to execute the trust establishment, along with the use of ant colony optimization algorithm. This infrastructure-based approach aimed to attenuate the CFD attack through the filtration ability in RSU. The ant colony optimization algorithm integrated observation with feedback information to measure the data trust. The observation factor was defined using the distance from the reported event and the detection range of the vehicle. The RSU gathered and transformed vehicles' reports into evidences to disseminate the trust. An RSU and beacon based trust scheme was further highlighted in [**64**]. The vehicles, with the cooperation of RSUs, built and disseminate the trust values by verifying the plausibility of the beacons and

the reported events with the tanimoto coefficient. The trust building for VANET in [**65**] was completed through location proximity, time closeness, and location verification. The receiver node measured its trust on each reported message. Then, the proposed method sorted computed trust to make the decision about the message. The message validation-based approach in [**66**] consisted to assign the trust according to message similarity, message conflict and route similarity. The routing path parameter was involved to alleviate the fact that the probability of reporting similar tampered messages increases as more the senders share common nodes.

3.5.1.3 Hybrid trust management approaches

Authors in [**51**] aimed to distinguish trustworthy messages in VANET by treating beacon messages to compute entity trust, and verifying event messages and beacon messages plausibility to construct data trust. Cosine similarity was used to assess the trust of beacon sender entity. The data trust was evaluated based on direct trust (beacon sand direct received events), and recommendation trust. In next step, event reputation value was computed to obtain the overall message trustworthiness. In addition, the Dempster–Shafer Theory (DST) was applied to combine transmitted opinions trustworthiness. The acceptance decision was taken according to a trust degree threshold. Reference [**67**] studied a scheme that addresses trust on VANET traffic data and vehicle through behavior evaluation and similarity rating. The DST technique was used for the trust evaluation. The trust of data was calculated based on reported traffic information similarity. Another trust-method combining behavior evaluation and similarity factor was presented in [**68**] for traffic signal control-applications in VANET. The approach enabled the detection of Sybil attack. In [**69**], the authors integrated the particle filters to carry out the plausibility check and estimate the trust of neighbour nodes. They performed the aggregation of the different data in one particle filter per neighbouring node to avoid the duplication. The separate particle filter aimed to achieve a local consistency verification of location-related data in each vehicle. The assessment of the trust relied on analyzed message and sender confidence (history-based). The problem of information cascading and oversampling in VANET was studied in [**70**]. A voting scheme was proposed to decide the opinions of nodes based on the received messages. Each vehicle has different opinion weight according to its closeness

from the reported event. To address traffic message plausibility in VANET, an event-based reputation system was elaborated in [**71**]. The main functions supported in this system were event management, reputation adaptation, event reputation collection, and event confidence list collection. The event reputation score defined the intensity level of a traffic event. The reputation of the vehicle was increased by one once detecting the traffic event. The confidence score indicated the reliability degree of the traffic event. Authors of [**72**] used the context factor to filter bogus messages. The trust was derived through the analysis of the different messages reporting the same event. Each message has to be validated by many vehicles to be regarded as true. The database that maintained the trust values was cleaned periodically by discarding faulty messages. The objective of work [**73**] was to make adaptive decision to enhance trust management in VANET. The decision making scheme was carried out when the time delay exceeded the defined threshold, or in case where the number of received messages overrides the specific threshold. The decision was made according to the trust degree computed through reporting events.

3.5.2 Our classification of trust management approaches

Trust-based approaches include broadly AI-enabling conventional methods like game theory schemes and fuzzy logic method, and machine learning-based schemes. Other some related approaches leveraged other tools like [**74**] which adapted Web of Trust for VANET trust management. More recent related approaches used emerging technologies such as Blockchain, edge computing, and SDN. In this section, we categorize the related works into: (1) trust-based approaches with AI techniques that apply clustering, reinforcement learning and ensemble learning, game theory, and fuzzy logic, and (2) trust-based approaches with emerging technologies that apply Cloud, Fog and Edge computing, Blockchain, and SDN.

3.4.2.1 Trust management with AI techniques

Different machine learning methods have been employed in developing trust-based vehicular networks models, such as clustering, reinforcement learning, ensemble learning and heuristic algorithms.

- Clustering-based trust approaches

Clustering-based trust proposals have shown the potential to address the security issue in Wireless Sensors Networks (WSNs) [[**75**]-[**78**]] and more dynamic network infrastructures like VANET and IoV networks. The creation of trustworthy clusters is an effective method to tackle with communication system security and extend the life of network. In cluster formation driven by trust model, the scheme elects the network node with the highest trust value as a cluster head. This fact can help in enhancing resources system utilization; e.g., by means of service priority-based allocation by the elected head. For most cases, the metric of trust is associated with other factors. For example, the cluster head selection in [**79**] used a composite metric indicating the trust level and the resource availability level. Other factors were combined with the trust metric like the factor of mobility similarity [**80**]. The computation of trust relies mainly on the behavior of the node [**79**], the metric of node reputation [[**81**]-[**83**]] and the metric of communication messages [[**84**]-[**85**]]. Authors of [**80**] suggested a trust management-based clustering and stability for VANET. Trust management considered trust data and trust communication metrics. Stability was referred to vehicle mobility similarity factor. Three phases were required to define nodes cluster role (i.e., cluster head or cluster member). These phases consisted of (1) neighbourhood discovery (with the same direction), (2) cluster head election using a backoff timer solution and (3) cluster stability maintenance. Cluster head selection was based on reputation, mobility and direction similarity metrics. In [**81**], the proposed framework relied on a bio-inspired and trust-based clustering approach to deal with the high overhead problem in WSN based ITS. The trust was associated with two levels. In the node level (normal node and cluster head node), the cluster head computed trust values of its cluster members. Each cluster member measured trust value for its one hop neighbour. The Bat optimization algorithm was applied to select clusters heads. The selection relied on metrics regarding residual energy level, trust level, and number of neighbours. In [**82**], the authors proposed a trust-based cluster head selection in the VANET by using metrics of experience, reputation, and knowledge. RSU nodes were in charge to select the cluster head. The authors of [**83**] proposed a trust-based secure clusters scheme for the VANET. RSU nodes were responsible to establish the clusters and assist to evaluate vehicles trust using velocity similarity. The cluster head maintained a reputation table for vehicles.

Table 3.1: Summary of related works: simple approaches

Class	Ref	Trust metrics	Used tools	Simulation	
				Setup	**Specific evaluation criteria**
Entity-based	[52] 2012	Reputation	Probability functions	Self-developed simulator: number of vehicles, malicious nodes, RSUs, and hops, trust level	-Accuracy of correct received messages -Scalability (+-collusion)
	[53] 2015	Node proprieties (similarity)	-Association rule mining -Echo protocol	SUMO: number of vehicles and malicious nodes ,traffic segment, listening period, trust level	-Success rates (% of vehicles that believed true reports/false reports)
	[54] 2010	Node proprieties +knowledge (multi-faceted)	-Defined formulas -Signature-based	SWANS: number of vehicles and malicious nodes, trust level	-Average speed of nodes (according to the % of malicious nodes) -Detection rate of malicious nodes -Communication ratio
	[55] 2010	Reputation+ knowledge+ node proprieties	-Defined formulas	numb of vehicles and malicious nodes, vehicles speed, trust level	-Effect of malicious nodes on traffic congestion -Effect of role factor on traffic congestion -Effect of knowledge and role factors' combination on traffic congestion
	[56] 2016	Reputation	-Iterative filtering method	Matalab-based: numb of vehicles and malicious nodes, trust level	-Vehicle services quality level -Feedback accuracy level -Trust values measurement -Resilience to badmouth attack and ballot stuffing attack
	[57] 2019	Reputation+ knowledge	-Markov method	Netlogo, SUMO, NS-2: number of vehicles, malicious nodes, and traffic lights, transmission range, speed and length of vehicles, trust level	-Detection rate of malicious nodes -Detection accuracy -Packet delivery rate -Average end-to-end -Number of transmitted control packets
	[58] 2015	Reputation+ knowledge	Correlation coefficient -Signature-based	Matlab-based: number of vehicles, malicious nodes, and recommenders, trust level	-Trust degree distribution -Indirect trust evaluation
	[59] 2019	Reputation	-Elliptic curve method	Simulator not specified, trust level	-Computation cost -Communication cost

Class	Ref	Trust metrics	Used tools	Simulation	
				Setup	Specific evaluation criteria
Data-based	[60] 2008	Proximity (utility)	-DST-based -Signature-based	NS-2: number of vehicles, malicious nodes, affirmative reports, and hops, vehicles distance, vehicles speed, trust level	-Average trust level of malicious nodes -Speed of decision
	[61] 2014	Proximity+ node proprieties +environment factor (similarity)	-Defined formulas -Signature-based	OMNET++, SUMO, VACaMobil: number of vehicles and malicious nodes, traffic segment, average speed, average flow, transmission range, trust level	-Average density (+accident scenario) -Success rate (% of vehicles that received true reports)
	[62] 2014	Proximity	-Defined formulas -Signature-based	SWANS++: number of vehicles, malicious nodes, and reporters, traffic segment, transmission range, trust level	-Fake location detection accuracy -False time stamp detection accuracy -False positives -Overall accuracy of malicious nodes detection -Time scarcity
	[63] 2011	Location+ node proprieties	-Ant colony optimization	NS-3: number of vehicles, road length, vehicles speed, transmission range, expectation of event values	-Data delivery delay
	[64] 2012	Beacon	-Tanimoto coefficient	NS-2, SUMO: number of vehicles, transmission range, beacon time to live, event time to live, vehicles speed, trust level	-Precision, recall -Detection delay
	[65] 2013	Location	-Defined formulas	Simulator not specified: number of vehicles and malicious nodes, trust level	-Effect of malicious nodes on trust -Time complexity
	[66] 2013	Messages similarity	-Defined formulas	Java-based: number of received messages during defined period, trust level	-Effect of conflicting value and path similarity on trust score -Effect of false messages on true messages acceptance -Processing time

Class	Ref	Trust metrics	Used tools	Simulation	
				Setup	Specific evaluation criteria
Hybrid	[51] 2013	-Beacon+event +reputation	-Cosine similarity rule -Signature-based	NS-2: number of vehicles and malicious nodes, vehicles speed, traffic segment, beacon+Event time to live, trust level	-Attacks detection rate -Misbehaving vehicle rate -Detection delay
	[67] 2015	Reputation+ knowledge+ environment (similarity)	-DST-based -Cosine similarity rule	GloMoSim: number of vehicles and malicious nodes, traffic segment, transmission range, vehicles speed, vehicles placement, trust level	-Precision and recall of detection -Communication overhead
	[68] 2016	knowledge+ node proprieties	-Defined formulas -Stochastic cellular	Automata model: number of vehicles, malicious nodes and data, vehicles speed, vehicles distance, trust level	-Accuracy of malicious data detection -Average delay of vehicles with malicious data
	[69] 2012	Knowledge+ location	-Particle filter	-Filter area size, vehicles speed, trust level	-Trust and confidence values with radar area violation -Runtime of particle filer -Accuracy of trust values measurement
	[70] 2014	Knowledge+ location	-Defined formulas	NCTUns: number of vehicles and malicious nodes, path loss mode, antenna options	-Percentage of incorrect decisions
	[71] 2009	Reputation+event	-Fibonacci number function	NS-2: number of vehicles, transmission range, event, vehicles speed, trust level	-Average accumulation speed of event reputation/confidence values
	[72] 2011	Context	-Defined formulas -Signature-based	VNSim: number of vehicles and malicious nodes, trust level	-Effect of malicious nodes on trust values measurement
	[73] 2014	Reputation+event	-Decision making process	NS-2: number of vehicles and malicious nodes, transmission range, Time to live, vehicle velocity, trust level	-Detection accuracy -Decision delay

Table 3.2: Qualitative comparison of related works: simple approaches

Ref	Dynamicity	Scalability	Time complexity	Communication overhead	Robustness	Privacy
[**52**]2012	partially	partially	partially	not available	partially	no
[**53**]2015	partially	partially	not available	not available	no	no
[**54**]2010	yes	partially	partially	medium	partially	no
[**55**]2010	yes	yes	not available	not available	yes	no
[**56**]2016	partially	not available	not available	not available	yes	no
[**57**]2019	yes	not available	simple	partially	yes	no
[**58**]2015	yes	not available	not available	not available	yes	partially
[**59**]2019	partially	not available	simple	partially	yes	yes
[**60**]2008	yes	partially	simple	medium	partially	no
[**61**]2014	yes	yes	complex	not available	partially	no
[**62**]2014	yes	yes	partially	medium	partially	partially
[**63**]2011	partially	not available	partially	not available	yes	no
[**64**]2012	partially	not available	simple	not available	yes	yes
[**65**]2013	yes	not available	simple	not available	yes	partially
[**66**]2013	yes	not available	simple	not available	yes	no
[**51**]2013	yes	yes	partially	not available	yes	partially
[**67**]2015	yes	partially	simple	low	yes	no
[**68**]2016	yes	partially	complex	not available	partially	no
[**69**]2012	yes	yes	partially	not available	yes	no
[**70**]2014	yes	not available	not available	not available	partially	partially
[**71**]2009	yes	not available	not available	not available	yes	no
[**72**]2011	partially	not available	not available	not available	yes	no
[**73**]2014	partially	not available	simple	not available	yes	no

The reputation of vehicles were utilized to derive the trust of exchanged messages. In [**84**], VANET nodes provided opinions about data trust using role and experience metrics. In [**85**], the authors proposed a VANET clustering algorithm. The trust management scheme used the formed clusters and the credibility of exchanged messages to deal with vehicles' reputations. The estimation of message credibility used proximity of location and time as well as forwarders number metric. The cluster maintenance was specified using an inference system. The proposal was evaluated through formal validation.

- Reinforcement learning-based trust approaches

Reinforcement learning-inspired trust management solutions are used in general to adjust the evaluation (i.e., decision-making) strategy, and help entities to get maximal reward [[**86**]-[**88**]]. Authors of [**89**] proposed a collaborative IDS for VANET through ensemble learning. Each vehicle trained

the local classifiers and shared its knowledge on-demand. The performance of the shared classifiers was used as trust factor. Other heuristic algorithms-based trust proposals such as neural network and Support Vector Machine (SVM) can be found in [[**90**]-[**91**]].

- Fuzzy logic-based trust approaches

As trust is determined through approximation (i.e., exchanged data might be inaccurate, incomplete, imprecise), some existing works elaborated their methods by mean of plausibility checking as a suitable solution for tackling uncertainty, and measuring data and source accuracy [[**92**]–[**93**]]. For example, authors in [**92**] conducted the trust management through fuzzy logic-based method. Every node added a unique encrypted ID to its submitted messages. By this way, receiver node verified the message source node. Three behavior aspects were presented to conduct the trust estimation: cooperativeness, honesty, and responsibility. The honesty metric was referred to honest forwarded packets percent. The responsibility corresponded to trustee node work regarding event reports detection. The three metrics were assessed for each neighbour, converted to fuzzy values, and applied to fuzzy rules and defuzzification step for final trust level computation.

- Game theory-based trust approaches

Game theory methods have been applied in the trust management for vehicular networks, as they represent an effective tool for nodes behavior analysis ([[**94**]-[**100**]]). The work [**94**] established an approach to help vehicles defining the trust of other entities (reputation-based) for better messages legitimacy assessment. The authors applied the certainty factor theory to quantify vehicle trust. Direct reputation data were gathered and stored in a history communication table, as usual from direct interactions, and indirect reputation was built using neighbours feedback (experience-based trust rate) and RSUs recommendations. fuzzy C-means clustering was also applied for indirect-reputation establishment. Then the uncertain deductive theory was employed to combine both computed scores. Moreover, the evaluation of the received messages was achieved through attribute-weighted K-means algorithm. The authors aimed to achieve cooperative behaviors authors through an incentive scheme. The game model involved nodes clusters

(normal, selfish, and malicious), nodes adopted strategies (i.e., willingness for receiving, forwarding, or releasing data), and payoff computation (based on reputation). The authors of [**95**] addressed trust within IoV through evolutionary game theory. Their idea was to simulate the dynamical protection process within a reputation-based trust running example, under an evolutionary game framework, by modelling misbehaving nodes attacking strategies, to define its effectiveness. Reputation scores were assigned for both vehicles and traffic related event messages. In terms of trust computation, a punitive reduction was applied on reputation values when receiving false reports or removing sent messages. The deception intensity factor was taken into consideration when deploying the trust game model. it was referred to node deceptive behavior and defined the false reports sending strength. Every malicious node decision is influenced by others. The work [**96**] presented a scheme for securing the IoV based on cooperative game theory. The proposal adopted the hedonic coalitional model for the vehicular trustworthy coalitions formation. Vehicle trust was established through Bayesian inference method based on experience assessment from direct interactions. Whenever a vehicle received an event message, the content of this message was compared to the real event state using incomplete beta function. The algorithm utilized vehicles trust and preference relation parameters to form the final coalition. Each vehicle made its decision for moving between coalitions according to its utility, by applying the shifting rule. In [**97**], an attacker and defender trust game approach was conducted using Nash equilibrium. Betweenness centrality, majority opinion, and node density metrics were used to derive the trust. Betweenness centrality was referred to the number of times when the node crossed route between nodes pair, and majority opinion relied on event trust and nodes type trust. In [**98**], authors introduced a trust approach that relied on signaling game for the VANET. They used the Spence's model in order to filter out dishonest nodes. Signal values were allocated to each node according to perceived behavior. Also, inciting rewards were proposed to encourage nodes cooperation. The eviction of dishonest nodes depended on the credit count (e.g., node's resources). Authors of [**99**] presented an incentive model to deal with selfish nodes in VANET. The credits functions were used to handle nodes' accounts. The credit took into account the reputation metric. Authors of [**100**] proposed a clustering and game theory based multi layered IDS. They employed a lightweight classifier to identify malicious nodes. In addition, they used a

Vickrey Clarke Groves-based incentive structure to promote the vehicles' participation in the head election procedure. Reputation scores were maintained by the RSUs to assess the trust of the cluster head.

3.4.2.2 Trust management with emerging technologies

We introduce in this sub-subsection some of the recent trust-based approaches that leverage Cloud, Fog, Edge, Blockchain and SDN to secure vehicular networks. The common connect in these works predominantly moves around enhancing performance and security.

- Cloud computing-based trust approaches

The benefits of Cloud were exploited in the implementation of trust mechanism within the vehicular network context in many papers (e.g, [[**101**]-[**103**]]. The work [**101**] built a three-layer Cloud-based trust management framework for vehicular social network scenario. The network architecture included (1) a central Cloud layer (server clusters group), (2) a road-side Cloud layer as a trust manager, and (3) a vehicular Cloud layer to support vehicles resources utilization. Accordingly, the trust was managed at three levels: (1) the global trust manager, linked with the central Cloud layer, wherein all vehicles profiles were recorded, (2) the domain trust manager, linked with the road-side Cloud layer; this level conducts trust degrees computation, and (3) the overall trust degree evaluation level, which was associated with the vehicular Cloud layer. In [**102**], the authors suggested an interdependent strategic trust approach for autonomous vehicles within a Cloud-based environment. They used flipit game to maintain the interactions at Cloud services. In [**103**], the authors addressed the trust computation in VANET-Cloud using DST method They applied fuzzy analyzer to decide the level of trusted and distrusted vehicles. Then, they implemented an algorithm to give reward or penalty for messages senders.

- Fog/Edge computing-based trust approaches

Fog/Edge computing has been leveraged in trust-based approaches in recent works such as [[**104**]-[**106**]]. The work [**104**] relied on bidding price-based approach for guaranteeing trusted Fog service transaction in rural area. The

registration to infrastructure-based Fog node was required for each vehicle client to conduct Fog service transaction (through certificates). During registration, vehicle client deposited digital currency for bidding. The trust was built to supervise vehicles activities. The trust computation relied on transaction on rural area and transaction with infrastructure-based Fog node. In [**105**], the authors suggested to ensure trust in the implementation of an Edge-based vehicular environment, wherein Edge computing servers undertook the work of executing local reputation management requests. Local authorities nodes scheduled Edge servers to promote trust building. Each vehicle uploaded reputations of its one-hop neighbours to the nearest local authority. These values were also stored simultaneously in a global reputation base. The work [**106**] presented a trust approach that used edge nodes for securing VANET. The upper layer of the communication system consisted of trusted authority and Cloud server, and the lower layer was represented by Edge nodes and vehicles. The trust on both sender and message were assessed using fuzzy rules. Plausibility (location verification-based), experience, and vehicle type were applied to calculate the trust score. Besides, the work used the k-nearest neighbours algorithm and the Cuckoo filter to deal with the NoS condition and the volume of generated data, respectively.

- Blockchain-based trust approaches

Blockchain is having an increasing interest for trust management in vehicular networks. Blockchain's immutability and trust-building capabilities find significant utility in vehicular networks since trust and data integrity are critical. While both of the trust management mechanism and the Blockchain have been individually applied to secure the vehicular networks, their combination has been more extensively explored in the context of the IoT. Research and practical implementations in the IoV context have been limited compared to the IoT. We refer some works on trust management that adopted Blockchain in the context of vehicular networks [[**107**]-[**111**]]. In [**107**] the trust and Blockchain based approach involved rating generation and uploading phase, trust value offsets computation phase (node proprieties-based), miner election, new blocks generation phase, and consensus application phase. Receiver node evaluated messages credibility based on Bayesian inference rule. Next, it uploaded computed rates to

Table 3.3: Summary of related works: approaches with AI techniques

Ref	Class	Trust metrics	Used tools	Simulation	
				Setup	**Specific evaluation criteria**
[**79**] 2019	Hybrid	Knowledge+ node proprieties	Clustering-based: -defined formulas	Matlab-based: cluster size, number of malicious nodes and hops, trust level	-Trust composite metric value with malicious nodes
[**80**] 2018	Hybrid	Reputation+ node proprieties (similarity)	Clustering-based: -defined formulas	Omnet++: cluster size, number of malicious nodes, vehicles speed, vehicles distance, traffic segment, trust level	-Cluster head duration -Cluster head election time -Rate of dishonest vehicles elected as cluster head -Packet delivery rate
[**81**] 2018	Entity-based	Knowledge	Clustering-based: -bio-inspired algorithm -bat optimization method	Matlab-based: number of vehicles, energy factor, transmitter amplifier, data aggregation energy, packet length, percentage of nominated cluster heads, trust level	-Network lifetime -Average residual energy -Average trust of cluster head
[**82**] 2020	Entity-based	Reputation+ knowledge+ experience	Clustering-based: -defined formulas	Omnet++, SUMO: number of nodes, vehicles speed, traffic segment, transmission range, transmission rate, position of RSU, mobility model, size of packet, trust level	-Average cluster duration -Average cluster head lifetime -Overhead -Throughput -Performances against Sybil attack and wormhole attack -Energy consumption
[**83**] 2020	Hybrid	Reputation+ location proximity	Clustering-based: -defined formulas	Veins; OpenStreet-Map: number of vehicles, vehicles speed, traffic segment, transmission range, trust level	-Cluster stability -Packet Data Ratio -Reputation of honest and malicious vehicle
[**84**] 2013	Hybrid	Knowledge+ node proprieties	Clustering-based: -defined formulas	C-based: percentage of authority roles, average number of vehicles per cluster, traffic segment, maximum distance for trust opinion, vehicles speed, trust level	-Percentage of messages as spam -System evolution time
[**85**] 2020	Hybrid	Reputation + location proximity + forwarders number	Clustering-based: -defined formulas	Formal validation	verification of soundness and completeness

Ref	Class	Trust metrics	Used tools	Simulation	
				Setup	Specific evaluation criteria
[89] 2020	Entity-based	Knowledge	ensemble-learning-based: -defined formulas	NSL-KDD, SUMO:number of vehicles for training and testing, transmission range, vehicles mobility, vehicles speed, trust level	-Detection rate of malicious nodes -Accuracy of malicious nodes detection -False positives -False negatives
[92] 2017	Entity-based	Reputation+ knowledge	Fuzzy logic-based: -defined formulas	NS-2, SUMO, MOVE: number of vehicles and malicious nodes, vehicle speed, transmission range, traffic segment, trust level	-Correlation behavior -Detection accuracy without collusion -Detection accuracy with collusion
[94] 2019	Entity-based	Reputation	Evolutionary game theory-based: -fuzzy C-means clustering -certain factor -Attribute weighted K-means algorithm	MobiSim, NS-2:number of vehicles, malicious nodes, vehicles speed, transmission range, traffic segment, trust level	-False alarm rate -Missed detection rate (message identification) -Accuracy rate of decision making -Throughput -Forwarding rate -Packet delivery delay -Cooperative nodes ratio
[95] 2019	Entity-based	Reputation	Evolutionary game theory-based: -defined formulas	number of vehicles and malicious nodes, group distribution, utility of groups, overall utility, vehicles speed, traffic segment, trust level, payoff	-Nodes strategy changes -Average growth rate of overall utility
[96] 2019	Entity-based	Knowledge	Cooperative game theory-based: (hedonic) -Bayesian inference	Matlab-based: number of vehicles and malicious nodes, coalitions partition, trust level	-Rate of compromised decisions -Rate of false reports in coalitions -Computational time
[97] 2017	Hybrid	Reputation+ knowledge+ node proprieties	Game theory-based: (Nash equilibrium) -probability functions	Matlab-based: number of vehicles and malicious nodes, transmission range, traffic segment, traffic type, trust level, payoff	-Retransmission attempts rate -Throughput -Data drop rate
[98] 2014	Entity-based	Reputation+ knowledge	Signaling game-based: (Job market signaling) -markov chain -signature-based -probability functions	NS2-34, SUMO, VanetMobisim: number of vehicles and malicious nodes, vehicles speed, transmission range, traffic segment, data rate, trust level	-Detection rate of malicious nodes -Percentage of false positives -Detection delay -Average ratio of corrupted data -Reception ratio with selfish nodes
[99] 2013	Entity-based	Reputation+ Knowledge	Signaling game-based: -defined formulas	NS-2, VanetMobiSim: number of vehicles and malicious nodes, traffic segment, transmission range, vehicles speed, trust level	-Detection rate of malicious nodes -False positives -Average ratios of received data and corrupted data

Ref	Class	Trust metrics	Used tools	Simulation	
				Setup	Specific evaluation criteria
[**100**] 2018	Entity-based	Reputation	Game theory+ clustering-based: -neural network -Vickrey Clarke Groves method	NS-3, SUMO: number of vehicles per cluster, vehicles mobility, propagation model, transmission range, trust level	-Detection rate of malicious nodes -False alarm rate -Average cluster membership duration of vehicles -Intrusion detection traffic volume -True positives -False positives -False negatives

Table 3.4: Qualitative comparison of related works: approaches with AI techniques

Ref	Dynamicity	Scalability	Time complexity	Communication overhead	Robustness	Privacy
[**79**]2019	partially	yes	not available	not available	partially	no
[**80**]2018	partially	yes	partially	medium	partially	no
[**81**]2018	yes	yes	partially	not available	partially	no
[**82**]2020	yes	yes	partially	medium	yes	no
[**83**]2020	yes	yes	not available	not available	partially	no
[**84**]2013	yes	yes	partially	medium	yes	no
[**85**]2020	yes	yes	not available	not available	not available	no
[**89**] 2020	yes	not available	not available	medium	yes	no
[**92**]2017	yes	yes	not available	not available	partially	no
[**94**] 2019	yes	not available	simple	medium	yes	no
[**95**] 2019	yes	not available	not available	not available	yes	no
[**96**] 2019	yes	yes	partially	medium	yes	partially
[**97**] 2017	yes	not available	not available	partially	yes	no
[**98**] 2014	yes	not available	partially	medium	yes	no
[**99**] 2013	yes	yes	not available	medium	yes	no
[**100**] 2018	yes	yes	not available	partially	yes	partially

RSU. The second phase consisted in using weighted aggregation to obtain involved vehicles trust value offsets and pack them into a candidate block. In [**108**], the authors combined a Blockchain-based trust management with a privacy-preserving scheme for an effective conditional privacy. Their proposal relied on trusted authority node to trace dishonest vehicle identity with the pseudonym in Blockchain. Likewise, the authors of [**109**] used the Blockchain to implement an anonymous reputation management in order to deal with privacy in VANET. Blockchain was placed to preserve privacy-authentication, during trust estimation. Different components were defined in the proposal such as (1) the certificate authority which managed certificates (the actions of this component were registered into the Blockchain), (2) the law enforcement authority which was responsible for vehicles' registration, supervising, and reputation assessment, (3) the certificates Blockchain (i.e., distributed certificates ledger), and (4) the messages Blockchain. Once being received, the certificate validity was verified, then, proofs of presence and revocation within concerned Blockchains were put. This procedure was referred to the anonymous authentication algorithm.

In [**110**], the Blockchain was used along with deep learning algorithm to manage the trust in vehicular networks. Each vehicle evaluated the messages received from the neighbouring vehicles. The vehicle reported the

identified untrustworthy vehicles to the nearby RSU. The authenticity of the report and the identity of the vehicle were verified using the Blockchain. The trust credentials of malicious vehicles were revoked by the RSU. In [**111**], the regional federated learning was proposed to enhance the security in Blockchain-enabled IoV. The vehicles maintained local learning models, and a reputation scheme was designed to guarantee the trust of vehicles that participate in the regional learning. Furthermore, the Blockchain technology has been associated with clustering scheme to solve the shortcomings within vehicular networks. Clustering techniques can assist to scale Blockchain-based IoV [**112**]. Also, the use of the Blockchain with the clustering has showed to be an effective solution to preserve security and privacy [**113**], and energy consumption [**114**]. However, the use of the two mechanisms together with the trust management was mainly explored in the context of IoT environment [[**115**]-[**120**]].

- SDN-based trust approaches

Some trust-based works have been concentrated on SDN incorporation such as [[**121**]-[**132**]]. The work [**121**] investigated the impact of SDN on VANET security. The authors supported the application of SDN within different vehicular scenarios, and stated that the SDN can be instrumental in managing vehicular networks when using the trust factor. Authors of [**122**] highlighted the need of trust management vis-a-vis SDN-enabled vehicular networks. They stated that the SDN is able to better understand the nodes' behaviors (e.g., sudden rise in the trust score of a node which is historically associated with low trust score). In [**123**], the authors aimed to identify the malicious vehicles in SDN-based VANET using the trust factor. They proposed a double security check by means of trust based detection algorithm and malicious vehicle detection scheme. The RSUs undertook the work of trust calculation by verifying the license plates of the vehicles. The work [**124**] introduced a misbehavior detection system under the SDN structure. The main tasks of the control plane included: (1) vehicular clusters formation, (2) Watchdogs election, (3) trust assessment, (4) Sybil attacks detection, and (5) security parameters adjustment. Cluster head selection relied on trust and mobility factors. The Watchdog was deployed to supervise the surrounding vehicles and transmit its reports to the local SDN controller. Local SDN controllers were responsible for monitoring clusters members and determines their trust values. Also, regional SDN controllers were defined to

monitor the local SDN controllers and compute their trust scores. The computation of trust relied on the interactions of the vehicle with the local SDN controller and the Watchdogs. The trust management has been also applied along with SDN for routing task in many works e.g., [**125**][**[126**]. In [**125**], the authors used trust management with the SDN for routing selection in vehicular networks. The trust of nodes relied on the data packet forwarding ratio and the control packet forwarding ratio. In [**126**], the authors introduced an hybrid SDN-based geographic routing protocol. The routing process applied trust management and encrypted schemes. The nodes of the network were clustered and each cluster head represented a semi-centralized controller. The selection of cluster head relied on map factor. The trust was computed using past interactions. The work [**127**] employed deep reinforcement learning and trust management for routing in SDN-based VANET environment. The proposal consisted of path learning and trust establishment process. The deep reinforcement learning was deployed into a centralized controller. Honest nodes aimed to discover the highest path trust score in order to establish data transfer. The trust level was assessed using the packets forwarding ratios. In other works, the SDN has been combined with Blockchain for the trust management in vehicular networks. The authors of [**128**], introduced a consortium Blockchain-based trust approach for vehicular SDN. The trust was estimated based on the rating recorded in the distributed ledger. The rating was referred to road-relevant messages. The computed trust value was used for the resource allocation on the control plane of the SDN. In [**129**], the authors presented a trust management approach for SDN-enabled 5G-VANET. The data plane consisted of vehicles, RSUs and the 5G base stations. RUSs and 5G base stations were controlled by a centralized SDN controller. With the support of the Blockchain, the RSUs verified the realness of the traffic data using the location proximity metric. In [**130**], the Blockchain was incorporated with the SDN and the Fog to manage the VANET network. The Fog technology was introduced to avoid frequent handovers. The devices in the Fog zones were SDN-enabled. Moreover, the SDN control plane included a Blockchain layer. The Blockchain was designed to support a reputation-based data propagation among the connected peers. The trust feature was also considered in [**131**] in order to improve the throughput of Blockchain-SDN-enabled VANET. The SDN controllers were responsible to collect vehicles trust. The trust values were generated from the history of direct interactions. These values were then

sent to a control layer operating in the distributed Blockchain manner.

3.6 Discussion

Based on the reviewed works, we can conclude that an effective trust management scheme is required in vehicular networks to satisfy users applications' expectations. Diverse criteria as requirements could be defined for the evaluation of the proposal efficiency. These criteria are mostly based on the challenges within the advanced vehicular environment. We summarize the above reviewed approaches (in both presented classifications) in the following Tables 3.1–3.6, wherein we recap the approaches classes, the chosen trust metrics, the used tools and the selected simulation performance parameters, to enable a qualitative comparison. It is worth reminding that the existing classification in trust that consists of entity-based, data-based and combined classes can serve as a core abstract classification. We mentioned that our taxonomy could be an associated sub-classification, and thus we conducted in Tables 3.1, 3.3, and 3.5 the assignment of both classes to the reviewed approaches in subsection 3.4.2. Tables 3.1-3.2 present the simple approaches of trust management in vehicular networks. Tables 3.3-3.4 depict the approaches of trust management in vehicular networks with AI-enabling techniques. Tables 3.5-3.6 list the approaches of trust management in vehicular networks with emerging technologies. Moreover, we selected the following overall criteria to further illustrate the difference between reviewed works: (1) dynamicity, (2) scalability, (3) complexity, (4) communication overhead, (5) robustness, and (6) privacy.

- Dynamicity

From Tables 3.2, 3.4, and 3.6, we can see that a lot of referred approaches have satisfied dynamicity criteria (i.e., mobility patterns dependence, low infrastructure dependence, dynamic metrics, fast trust update, ...). Usually, data-based trust models are more dynamic than entity-based trust models (e.g., infrastructure recommendation less-based to extract global trust, where there is no long interactions between nodes due to high speed.

Table 3.5: Summary of related works: approaches with emerging technologies

Ref	Class	Trust metrics	Used tools	Simulation	
				Setup	**Specific evaluation criteria**
[**101**] 2017	Entity-based	Reputation+ knowledge	Cloud-based: -defined formulas	Performance Evaluation Process Algebra: number of vehicles and virtual vehicular machines, rate of requesting service, trust level	-Resource utilization -Capacity planning for trust calculation -Queue length of trust computation -Throughput -Response time
[**102**] 2018	Entity-based	Reputation+ Knowledge	Cloud-based: -flipit game -signaling game	Matlab-based: vehicle position +angle, trust level	-Probability of controlling cloud services -Vehicle dynamics -State trajectories
[**103**] 2019	Hybrid	Reputation+ knowledge+event	Cloud-based: -DST-based -fuzzy rules	Java-based: number of vehicles, trust level	-Response time -Basic probability task of vehicles -Outcome of fuzzy analyzer -Trust value change (reward+penalty)
[**104**] 2019	Entity-based	Node properties	Fog-based: -signature-based -bidding price	number of vehicles and malicious nodes, traffic segment, trust level, payoff, bidding price	-Transactions number (according to bidding price, and payoff) -Attacks number
[**105**] 2017	Entity-based	Reputation	Edge-based: multi-weighted subjective logic	-Urban area: number of served nodes and malicious nodes, traffic segment, vehicles speed, trust level	-Average reputation value of malicious nodes -Detection rate of malicious nodes -Resource budgets of served nodes -Utility of served nodes
[**106**] 2020	Hybrid	Knowledge+ node properties+ location	Edge-based: -fuzzy logic -k-nearest neighbour algorithm -cuckoo filter	NS-2, SUMO, MOVE: number of vehicles and malicious nodes, traffic segment, transmission range, vehicles speed, vehicle type, trust level	-Precision and recall -Overall accuracy of malicious nodes detection -Communication overhead
[**107**] 2018	Entity-based	Node properties	Blockchain-based: -Proof-of-Work+ Proof-of-stake -SHA-256 -Bayesian inference	Matlab-based: number of vehicles and false reports, packet size, vehicles distance, trust level	-% of unfair ratings -Trust value offset -Generation time of offset blocks -Transmission latency

Ref	Class	Trust metrics	Used tools	Simulation	
				Setup	Specific evaluation criteria
[**108**] 2019	Entity-based	Reputation+ knowledge	Blockchain-based: -Proof-of-work+ Practical Byzantine Fault Tolerates -logistic regression	Python+ Golang-based: number of vehicles, malicious nodes and updating requests, vehicles speed, transmission range, trust level	-Evaluation of announcement protocol (average computation time) -Trust value change -Detection rate of malicious nodes -False detection rate -Average latency of consensus algorithm
[**109**] 2018	Entity-based	Reputation+ knowledge	Blockchain-based: -Proof-of-work -signature-based -SHA-256	-Numb of vehicles, trust level	-Storage and transmission overhead
[**110**] 2020	Entity-based	Node properties	Blockchain-based: -deep learning (feedforward neural network)	NS-2, SUMO: number of vehicles and malicious nodes, vehicle speed, vehicles placement, transmission range, trust level	-Precision, Recall of malicious nodes detection
[**111**] 2021	Entity-based	Reputation	Blockchain-based: -regional federated learning algorithm -signature-based	Number of vehicles and malicious nodes, trust level	-Model accuracy rate under reputation/non reputation selection -Convergence of knowledge price
[**123**] 2018	Entity-based	Node properties	SDN-based: NP-completeness	OMNET ++, Modeler: number of vehicles and malicious nodes, traffic segment, trust level	-Throughput -Average end-to-end delay
[**124**] 2020	Entity-based	Reputation+ Knowledge	SDN-based: -clustering scheme	Veins, SUMO: number of vehicles and malicious nodes, transmission range, traffic segment, trust level	-Detection rate of malicious nodes -False positives
[**125**] 2016	Entity-based	Node properties	SDN-based: -defined formulas	OPNET: number of vehicles, traffic segment, trust level	-Average end-to-end delay -Throughput -Rate of sent messages
[**126**] 2020	Entity-based	Knowledge	SDN-based: -clustering scheme -signature-based	NS-2.34, VanetMobiSim: number of vehicles and malicious nodes, transmission range, vehicles speed, traffic segment, medium capacity, trust level	-Average end-to-end delay -Packet delivery ratio

Ref	Class	Trust metrics	Used tools	Simulation	
				Setup	**Specific evaluation criteria**
[**127**] 2020	Hybrid	Knowledge+ node properties	SDN-based: -deep Q-learning -Markov decision process	TensorFlow: number of vehicles, learning rate, trust level	-Convergence performance -Expected transmission count -Expected transmission count delay
[**128**] 2020	Entity-based	Knowledge	SDN+ Blockchain-based: -Proof-of-stake+ modified Practical Byzantine Fault Tolerance -signature-based	Python-based: number of vehicles and malicious nodes, vehicles distance, number of reference set, message group, trust level, number of paths, virtual network average lifetime	-Path mapping -Prediction accuracy -Transactions number -Transaction confirmation time
[**129**] 2019	Data-based	Location+ time	SDN+ Blockchain-based: -Proof-of-stake -signature-based -SHA-256	OMNeT++, crypto++ library: number of vehicles and malicious nodes, traffic segment, vehicles speed, transmission range, trust level	-Accuracy of detection -Processing time of blocks -Transaction transmission delay -Video encryption time overhead
[**130**] 2019	Entity-based	Reputation	SDN+ Blockchain+ Fog-based: -Practical Byzantine Fault Tolerance -clustering scheme -signature-based	NS-3: number of vehicles, packet size, traffic segment, vehicles speed transmission range, trust level	-Packet delivery rate -Transmission delay -Processing time of blocks
[**131**] 2019	Entity-based	Reputation	SDN+ Blockchain- based: -redundant Byzantine Fault Tolerance -duelling deep Q-Learning -signature-based	TensorFlow: number of vehicles, packet size, block size, traffic segment, trust level	-Throughput -Convergence performance

Table 3.6: Qualitative comparison of related works: approaches with emerging technologies

Ref	Dynamicity	Scalability	Time complexity	Communication overhead	Robustness	Privacy
[**101**]2017	partially	not available	partially	not available	not available	no
[**102**]2018	yes	not available	not available	not available	yes	no
[**103**]2019	partially	yes	partially	not available	yes	no
[**104**]2019	yes	yes	not available	medium	yes	no
[**105**]2017	yes	yes	partially	medium	yes	no
[**106**] 2020	yes	yes	supposed to be simple (e.g., local computation)	medium	yes	partially
[**107**]2018	yes	yes	simple	medium	yes	no
[**108**]2019	yes	not available	partially	not available	yes	yes
[**109**]2018	yes	not available	partially	medium	yes	yes
[**110**]2020	yes	yes	not available	not available	yes	no
[**111**]2021	yes	not available	not available	not available	yes	no
[**123**]2018	yes	not available	partially	medium	yes	no
[**124**]2020	yes	yes	not available	not available	yes	no
[**125**]2016	yes	not available	partially	medium	partially	no
[**126**]2020	yes	yes	partially	medium	yes	partially
[**127**]2020	yes	not available	partially	medium	partially	no
[**128**]2020	yes	not available	partially	medium	yes	no
[**129**]2019	yes	not available	partially	medium	yes	no
[**130**]2019	yes	not available	partially	medium	yes	partially
[**131**]2019	yes	yes	partially	not available	yes	no

- Scalability

Scalability has also been in the interest of the previously-discussed approaches. Some works succeeded to persevere network performances regardless network size and traffic density (e.g., better detection and communication ratios with high density, stable network results with high malicious nodes number).

- Complexity

It is important to consider the complexity criteria for performance evaluation (e.g., complexity in term of time). For example, it is crucial to quickly derive accurate trust value; almost instantly for time critical-based applications. Most reviewed approaches showed, for instance, that the delay of malicious nodes detection remains proportional to network density, besides, the delay can increase mainly with indirect trust computation.

- Communication overhead

The lower communication overhead ratio is, the more efficient the network is (e.g., lower bandwidth use; cost-effective network; accordingly, fast and better real response time, etc.). However, this parameter has been not considered in major discussed approaches (only [**67**] showed good result).

- Robustness

The basis of trust in vehicular networks is to distinguish dishonest nodes and ensure reliable network. We identified reviewed approaches as (1) partially robust, and (2) robust (i.e., the proposal is partially robust when attacks are very slightly addressed).

- Privacy

Privacy criteria is obviously important when conceiving trust-based approaches. Yet, we can notice from Tables 3.2, 3.4, and 3.6 that the privacy preservation is lacking (except in ([**108**][[**109**]).

To sum up, the QoS criteria that we have selected to interpret the efficiency of the reviewed works in Tables 3.2, 3.4 and 3.6 (section 3.5) have not been totally satisfied. Cited approaches have not drawn greater attention to QoS requirements. Consequently, the quality of trust-based services needs to be improved (e.g., reliable and accurate delivery of data in time). Scalability, complexity, communication overhead, and robustness must get more interest to maintain system performance when trust management approaches are deployed. Furthermore, well built trust models will be based on entities, on exchanged data and on environments constraints. In this case, hybrid solutions should be applied. Moreover, different properties and metrics should be considered. Besides, fast update and decay of trust value may be an important issue to deal with in future studies; in fact, each stored trust value is subject to be fast updated or decayed every time a node makes interactions with other entities, thus it is essential to define an appropriate update of the stored trust values (that emphases mainly the trustee node computational capacities) or a lifetime (as nodes are not capable to store all the contacted nodes trust values). AI-enabling techniques have comparatively assisted in intuitive trust scores generation, approximation reasoning

(e.g., fuzzy logic-based), better decision selection (e.g., Q-learning-based), malicious actions reducing (e.g., clustering- based; election of honest cluster head, game theory-based; incentive mechanism), and nodes selfishness coping (e.g., cooperative game-based), but also have faced some QoS concerns; mainly related to communication overhead and time complexity. The clustering techniques help a lot in designing reliable an scalable trust management framework since the trust management is decentralized. However, these techniques can be improved when used with the association of the emerging technologies such as the Blockchain or the SDN. These techniques can facilitate the coordination between the different cluster heads and bring better traceability of the trust management process. Also, the integration of the AI techniques in trust models aims mainly to define the trust formula. When considering the whole IoV ecosystem, a distributed approach of the intelligence on the different components of the IoV environment could be very useful to optimize the proposed model (e.g, collaborative learning of trust level). Thus, integrated the federated learning approaches when building trust management approaches could has an impact on its robustness since the role of different IoV entities could be different (humans, devices, infrastructure entities), and thus their trust formula could be based on different metrics and different parameters. The trust management in emerging technologies is in the core of future works interest for the IoV. These technologies can assist in providing QoS and network architectures that ensure more scalable and efficient trust management; e.g., trust data can be immutable and stored with resiliency and traceability with the Blockchain. Table 3.6 clearly tells about robustness performances of Blockchain-based trust approaches. Fog technology can offer localized processing within the trust scheme. Also, trust need to be built at different levels within emerging architectures like Fog, Edge or SDN-driven environment. Regarding, Blockchain-based trust management approaches, some possible issues related to consensus, blocks generation delay, forks, and power consumption should be considered to more satisfy the QoS. From another hand, let us recommend conceiving lightweight trust management frameworks characterized by a low energy consumption. Indeed, applying the trust management within an IoV environment leads to a communication overhead and thus increases the time complexity. This is becomes more critical when considering real time applications that are delay sensitive. For that, these parameters have to be taken into account for reliable performance evalua-

tion. Hence, we should consider more the energy efficiency of trust models. The evaluation of energy efficiency becomes more important in the green-IoT deployment' context when defining a green communication across the IoV ecosystem. Emerging technologies can be used to improve this parameter since they propose innovative schemes to improve the methodologies of energy efficiency that should be considered during the process of trust building.

3.7 Conclusion

This chapter discussed efficiency of proposed trust-based works for the vehicular environment. Although there have been many works, further improvements are needed. Generally, three aspects for improvement can be considered: implementing efficient trust assessment scheme, considering the QoS aspect, and conceiving optimized architecture. In this thesis, we focus on research direction of trust with emerging technologies. Both SDN and Blockchain technologies are important to trust management in the IoV ecosystem. Besides, they are integral to an optimized IoV architecture. They help establish, verify, and maintain trust among nodes while ensuring reliability of communications. In chapter 4, we will deal with security in the IoV through a trust and Blockchain based approach.

Chapter 4

Trust management approach for the IoV

4.1 Introduction

In this chapter, we introduce our mechanism of trust management to secure the IoV. The trust mechanism can improve the trust of network, and the rate of malicious nodes can be reduced when collaborative nodes and exchanged messages are trustworthy. We argued that a major weakness of reviewed related approaches is that the performances of QoS were not considered simultaneously during the security mechanism evaluation. In this chapter, we introduce a trust based approach with Blockchain technology enabled. In the first, we suggest a centralized trust scheme. Then, we present an extension to integrate with our approach and enhance performance needed to cope with the QoS. For that, we opt for decentralized trust scheme, wherein clustering technique was applied to construct our security framework.

4.2 Blockchain and trust-based approach for the IoV

4.2.1 Proposed approach overview

As presented in previous chapter (sub-subsection 3.4.2.2), the trust management mechanism and the Blockchain have been utilized to reinforce security of vehicular networks ([[**107**]-[**111**]]). However, the application of these two techniques together has been mainly carried out for the IoT context. Compared to related works such as ([[**107**]-[**111**]], we introduce a more adaptable approach for the IoV that tailors trust management to meet the QoS. In our approach, we highlight the network part of the IoV ecosystem. We define a trust management and Blockchain based mechanism that takes into consideration the IoV characteristics of components mobility diversity. On one hand, the trust management mechanism reinforces the nodes reliability. It helps in making informed decisions about data sharing and network participation. On the other hand, the Blockchain technology will provide features of traceability property and transparency. Blockchain with its inherent security features, can complement trust management mechanisms to enhance the security of the IoV ecosystem.

- We present a centralized trust based scheme using multi-trust; i.e., two metrics of trust: reputation and location.

- We apply our proposed approach on a two-level Blockchain based architecture. Trust values are stored in local Blockchain assigned to a specific geographic zone. Then the RSU node defined to be responsible for specific geographic zone, can share the values of trust with other RSU nodes in order to propagate and update the level of trustworthiness of the IoV communication system.

Detailed description of our proposal are given in the upcoming sections. Subsections 4.2.2 and 4.2.3 present an overview of our architecture and threat model, respectively. Subsection 4.2.4 describes our proposed trust management mechanism. Performances evaluation are given in subsection 4.2.5.

4.2.2 Proposed architecture overview

Figure 4.1 illustrates the IoV environment architecture. Our IoV network is two level Blockchain based architecture. It involves vehicles, RSU nodes, trusted authority nodes, and ITS centers. These entities interact through different wireless communication types forming V2X communications. We assign two types of vehicles: ordinary vehicles and edge vehicles. Role of each network component is defined as follows:

- Ordinary vehicles

The ordinary vehicles are considered to carry out the a local basic processing of data. Their performances are not very high and they do not contribute much in the overall network continuity.

- Edge vehicles

The edge vehicles are those vehicles with strong resources and computational capabilities, compared to the ordinary vehicles. Edge vehicles are fitted with cache memory that can employ intelligent caching strategies based on usage patterns and relevance (e.g., frequently accessed data can be prioritized). This can offer severaladvantages in terms of reducing latency, improving data availability, and optimizingbandwidth usage. The edge nodes handle trust computation. The edge nodes are considered to semi-trusted.

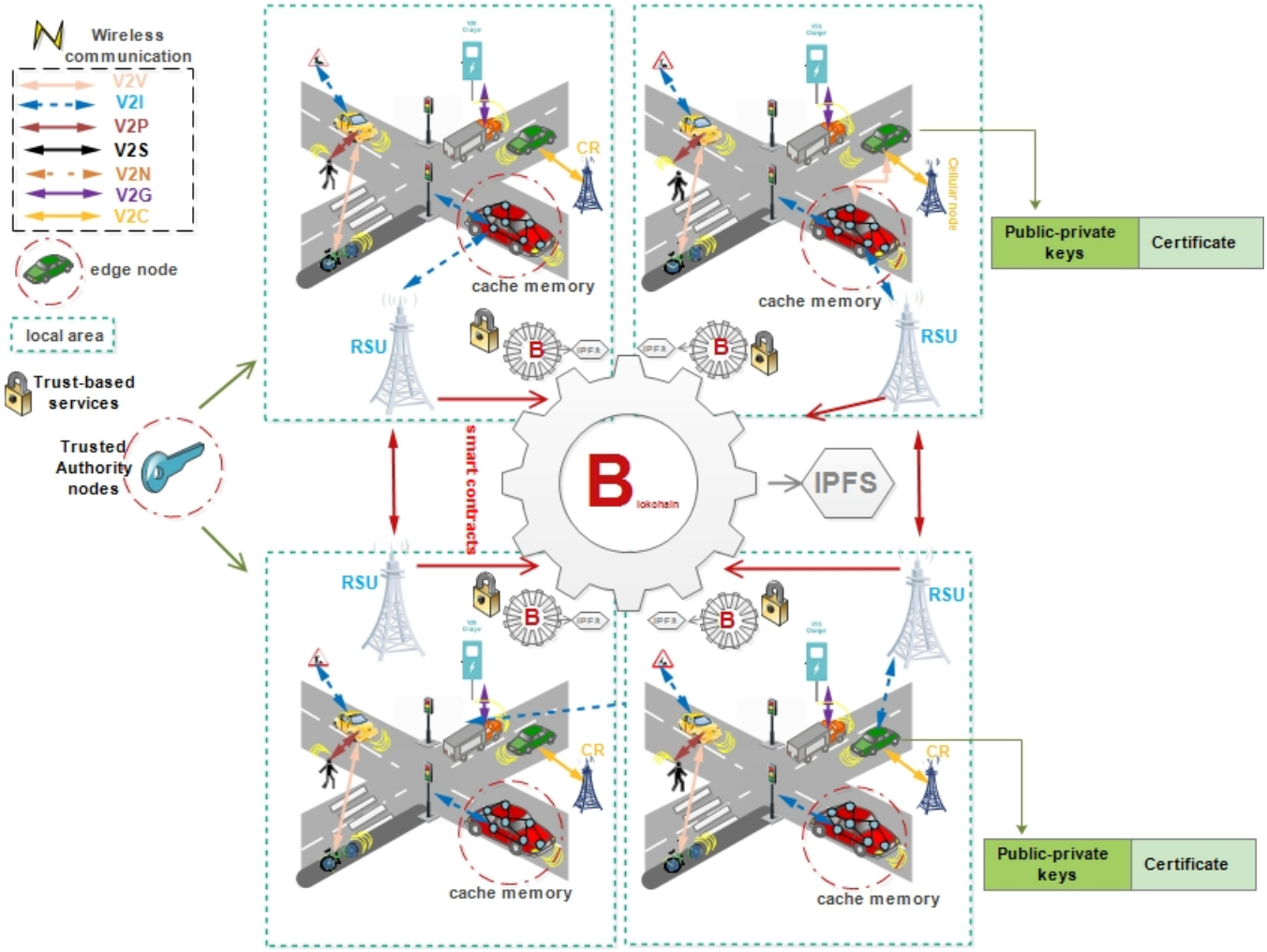


Figure 4.1: Proposed architecture

- Trusted authority nodes

The trusted authority nodes are defined to authenticate the nodes of the network. Each authority node is responsible for a specific geographic zone. The authority node is an independent, reliable, and highly secure entity. It generates communication system parameters and saves real identities of nodes belonging to its area. That let the authority nodes to perform and track out the registration of network nodes through deliverance of certifications. The services of authority nodes are released and regulated by a variety of legal, financial, technical, and structural. The trusted authority nodes are assumed to be enriched with high powered computational resources and sufficient storage, as well as they can not be compromised.

- RSUs

The RSUs are deployed to provide information about road traffic and infrastructure assistance.

- Interplanetary File System (IPFS)

The IPFS is employed to achieve distributed storage of trust information.

Figure 4.2 summarizes the roles of these network components. To reinforce trust between the network entities, we adopt a Blockchain-based IoV architecture model. We consider two types of Blockchain: local Blockchain and global Blockchain. Local Blockchains are well-suited to the dynamic nature of the IoV. Managing trust at the local level is particularly beneficial for real-time QoS improvement.

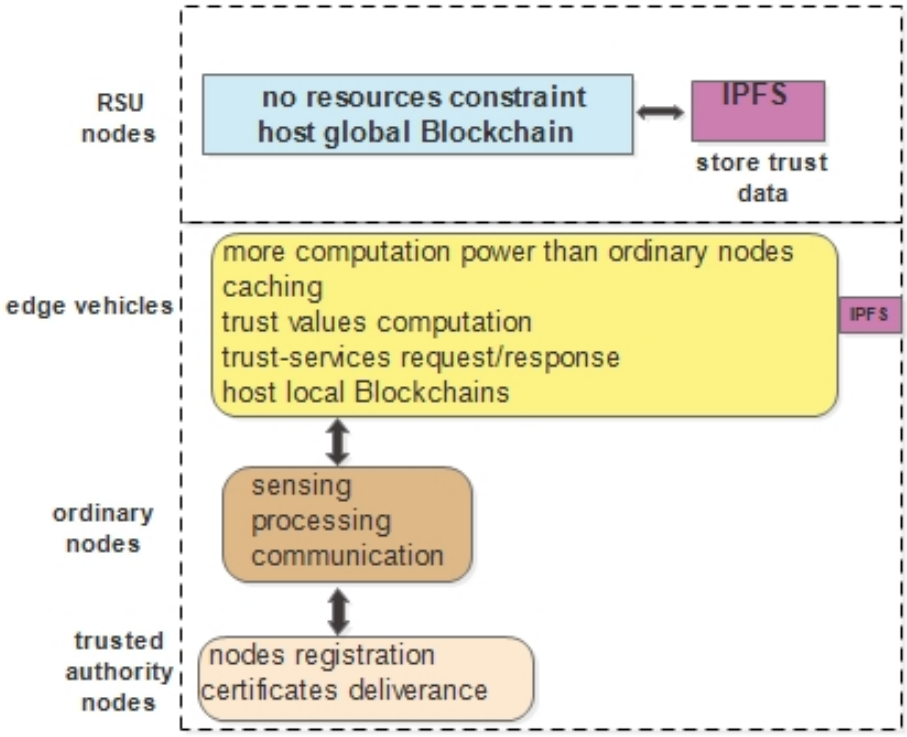


Figure 4.2: Figure 4.2: Basic network model components

1. Local Blockchain

The bottom layer of the proposed network model involves the edge vehicles to build the local Blockchains. Thus, the local Blockchains are associated with different geographical zones to support the local trust management. These local Blockchains are employed to save the trust data of local nodes (i.e., nodes belonging to small zones). The local Blockchains are envisaged to guarantee low-delay when holding time-sensitive target data like trust data. By assigning local Blockchains to specific zones, trust management can be tailored to the specific needs of each zone, further optimizing QoS for local conditions.

2. Global Blockchain

At the upper layer of the proposed network model, the RSU nodes host a global Blockchain for trust. The RSU nodes are bound to upload the local Blockchains into the global Blockchain. This global chain is also employed in the trust management mechanism to maintain the computed trust of nodes and to keep it updated with observed nodes behavior. Hence, the global Blockchain will serve to draw the overall trust level of the communication system. By maintaining a transparent and tamper-resistant history of trust, the global Blockchain can enhance overall network security and reliability. Such layered structure facilitates the recognition of malicious nodes. In next subsection, we give the attack types that our security framework can countermeasure.

4.2.3 Threat model

Regarding our IoV network model, the majority of vulnerabilities are related to malicious edge vehicles, unfair recommendations (from edge vehicles and RSUs), false or unavailable traffic information and falsified trust data. Using these vulnerabilities, attackers can reach serious network performance deterioration. Relying on malicious edge vehicles will affect the network performances. The unfair recommendations result in generating unreliable reputations that will affect the trust information. The perceived traffic information from ordinary vehicles, edge vehicles, and RSUs are liable to be falsified deteriorating road safety. Likewise, the derived trust data is not immune from tampering. Hence, the complete security of the network is

compromised. The main attacks that we consider within our approach are as follows:

- Bogus information attack

For example, a malicious vehicle finds that the road is congested after a vehicle crush, however, it reports false messages to nearby vehicles claiming that the road is clear. In our trust management process that we detail in subsection 4.2.4, the assessment of message credibility along with the incentive algorithm on message sender limits the possibility of the malicious vehicle to send false messages.

- Tampering trust data

A malicious vehicle might try to perform, delete or modify the trust values to hamper the integrity of the network. In such scenario, it is important that consequences on the overall network security are minimized. To avoid such attack, we propose a Blockchain based architecture. Using the Blockchain technology, such attack is extremely difficult. It is almost not possible for a malicious vehicle to tamper with the trust data saved in the Blockchain [**133**].

- Bad-mouthing attack

For example, the malicious vehicle as recommender deliberately gives a negative recommendation to decrease the trust score of the vehicle that sends the message. In our proposed framework, to avoid such behavior, the message credibility is verified to derive the final trust value and apply an incentive algorithm for vehicles participating in the recommendations process.

4.2.4 Proposed trust management process

The phases of our proposed trust management mechanism comprises (1) computation of local trust by the edge vehicles, (2) local mining scheme and blocks generation, (3) upload of computed trust in local Blockchains, (4) global mining scheme and blocks generation, and (5) upload of computed trust in global Blockchain. The computation of local trust is performed by

the edge vehicles. The edge vehicles host local Blockchains to save local computed trust. We denote V=(V_1, V_2, ..., V_i) the set of i ordinary vehicles, E=(E_1, E_2, ..., E_k) the set of k edge vehicles; $1 \leq k$, RSU=(RSU_1, RSU_2, ..., RSU_u) the set of u deployed RSUs; $1 \leq u$, and TA=(TA_1, TA_2, ..., TA_y the set of y authority nodes; $1 \leq y$. As aforementioned, the trusted authority (TA) is responsible of generating and managing certificates of RSUs, group key update, partial keys of vehicles and the tracking and revocation of malicious vehicles. The TA is also responsible for generating system parameters, and the registration of vehicles. It is the only entity that can track the real identity of vehicles. It is assumed that the TA will never be compromised. Each geographic area is controlled by a TA. The vehicle V_i ($V_i \in V$) has a registration and authentication phase to join the IoV network.

- Registration and authentication phase

The TA is responsible for the authentication phase within its specific geographic zone. Authentication guarantees legitimate network' entities. Ensuring that entities within the IoV can verify each other's identities before engaging V2X communications and exchanging sensitive data is essential. The TA represents the architecture PKI (Public Key Infrastructure) that ensures security and privacy through the authentication mechanism based on an asymmetric cryptography. Indeed, it is responsible for issuing the certifications and the public keys of the vehicles belonging to its range as well as managing their identities. When a vehicle enters a new geographic zone, it should be authenticated with the TA. Once the vehicle is registered, the TA generates a certificate and a pair of public and private keys that will be used by the vehicle to communicate securely. The certificate contains the public key with specific attributes such as ID and is signed by the TA private key. By this way vehicles are registered as valid participants. We denote the pair of key for the node V_i as (Pb_i^m, Pv_i^m; $1 \leq m$); m is the cardinality of the key pair. The TA delivers a certification for V_i including information about V_i public key, certification issuer (IF_y), signature of issuer (Sg_y), and certification expiration date (Ep_i). To reinforce the security of the network, the delivered certification includes also the V_i reputation value (R_i).

$\text{Cerf}_{i,y}^m$= (Pb_i^m, Sg_y, IF_y, Ep_i, R_i) (4.1)

The vehicle will use its private and public keys to sign all its messages before sending them. When the vehicle wants to send a message, it will attach its certificate and digital signature using the private key with the message. The receiver of the message has to check the attached certificate using the public key of the TA received when keys were issued. Once the certificate is verified, the sender public key will be obtained and used to check the message's digital signature. The message sender is authenticated if both the certificate and message signature verification succeed.

- Local trust management process

To manage the trust, we propose a centralized process. The trust information is deduced from the metrics of reputation and credibility of received message (the value of trust $\in$ [0, 1]). Basically, each sender is associated with an initial trust value of 0.4, deduced from the conducted simulations. A very low initial trust value impacts negatively the communication process. A high initial trust value increases the risk of the on-off attack where the malicious node starts behaving well and then changes to a malevolent behavior. An ordinary vehicle can query about the trust information of a sender node by requesting the closest edge vehicle E_k. The requested E_k determinate the trust of the target sender node using the metrics of reputation and location. The computed trust of j sender nodes ($R = [R_1, R_2, ..., R_j]$) are maintained in the IPFS of the E_k. In figure 4.3, we depict the process of local trust management.

1. Calculation of reputation

The reputation is computed by the edge vehicle using the metric of knowledge along with the recommendations from surrounding edge vehicles. Each edge vehicle maintains a recommendation value about each node with whom it had at least an exchange. The trust value of each node is then updated based on its behavior and its honest cooperation. The edge vehicle E_k calculates the reputation of a node V_j based on the previous saved reputation value, the average of the reputation values sent by the set of the edge vehicles and the average of the reputation values sent by the RSUs that possess a reputation value about that node. The calculation of the reputation is defined using the following formula:

$$R_j^t = \alpha R_j^{t-1} + \beta \frac{\sum_{h=1}^{|E|} Rec(h,j)}{|E|} + \delta \frac{\sum_{h=1}^{|RSU|} RecRSU(h,j)}{|RSU|} \quad (4.2)$$

where $R^t(j)$ represents the reputation assigned by edge vehicle E_k to the sender V_j at the current time t, $R^{t-1}(j)$ is the last reputation value of V_j saved in the IPFS, $Rec(h,j)$ and $RecRSU(h,j)$ denote the recommendations about the sender V_j from the edge vehicles and the nearly RSUs, respectively. α, β, and δ are the defined weights, respectively, for the three parameters considered in Eq.(4.2), reputation value of the node, recommendations from edges vehicles and recommendations from nearly RSUs. We mention that α+β+δ= 1.

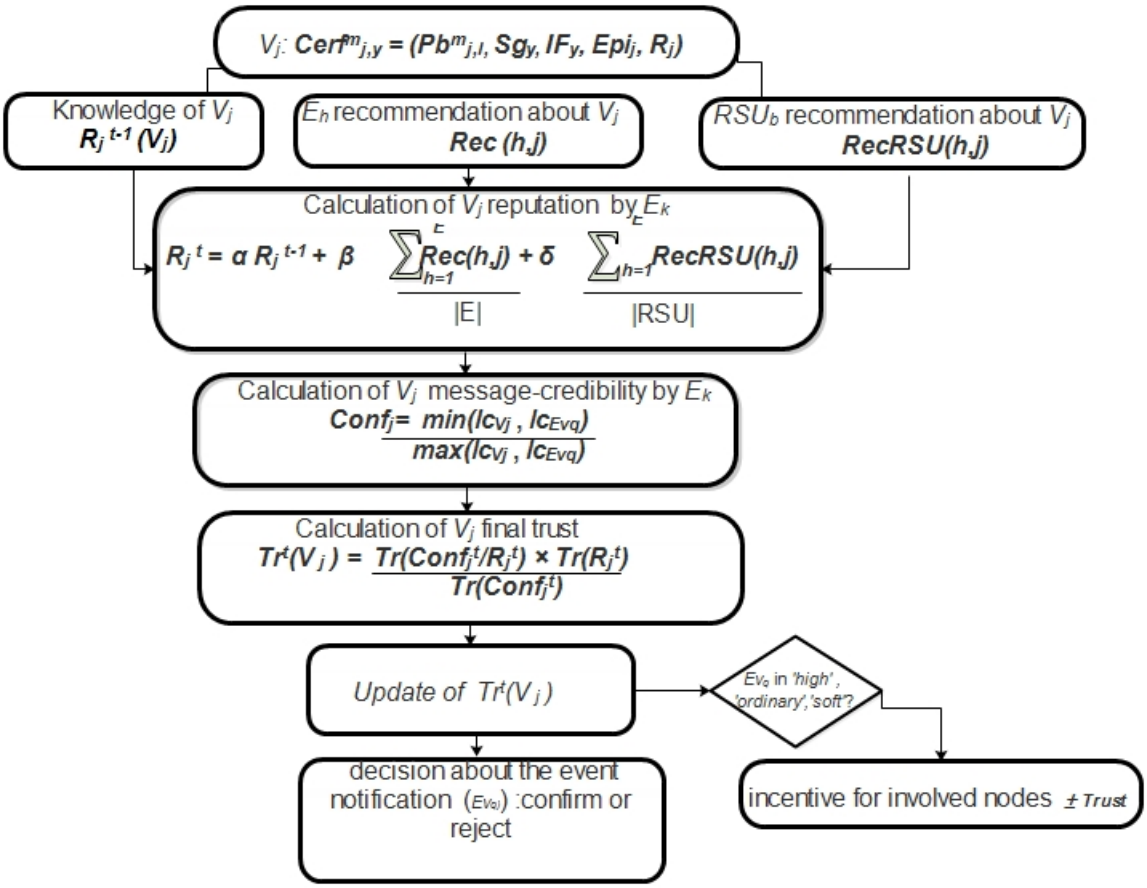


Figure 4.3: Process of local trust management

2. Calculation of message-credibility value

To calculate the trust value of a node, we are based on the reputation value as described above and on the calculation of the message-credibility value. Within our IoV network, when an event occurs (i.e. accident, traffic congestion, etc.), the nodes who want to report the event broadcast an informative message containing their position and the event position. In our model, when a node receives a message informing it about a specific event, it should check the credibility of that message. Thus, it contacts the edge node to know about the sender reputation value. After estimating the reputation value of the sender V_j, the edge vehicle E_k evaluates its belief about the credibility of the event Ev_q using the location metric. In fact, the location proximity constitutes an important factor for message credibility assessment as the reported content is likely to be more accurate when the sender V_j is placed close to the location of the event Ev_q . Let $Conf^t(j)$ be the credibility value of the event Ev_q sent by V_j. The edge vehicle E_k establishes the $Conf^t(j)$ value through calculating the closeness of V_j location to Ev_q as follows:

$$\text{Conf}^t(j)=\frac{min(lc(V_j),lc(Ev_q))}{max(lc(V_j),lc(Ev_q))} \quad (4.3)$$

where $lc(V_j)$ and $lc(Ev_q)$ refer, respectively, to the positions of the sender V_j and the event Ev_q. We point that $|lc(V_j) - lc(Ev_q)|< \xi$lc. ξ represents the accepted threshold for the locations difference. The value 0 will be assigned to $Conf^t(j)$ where the opposite of the marked condition takes place. Concerning the reported messages, the TA defines a service level agreement (SLA) that classifies the messages types based on the importance level of the occurred event. The defined SLA can be shared with the edge vehicles when required. The three categories of the messages are defined as follows:

- high level: refers to emergency messages and crucial reports, such as accident notification or roadworks notifications.

- normal level: refers to informative messages like notifications about driving status and weather conditions.

- soft level: refers to less important reports like non-safety messages related to the driver comfort, for example messages informing about Kiosks or some rest areas or coffee shops.

3. Calculation of the global trust value

The last step of the trustworthiness rating is to calculate the total trust score of a node V_j using the reputation attribute along with the message-credibility. At a specific time t, the edge vehicle E_k calculates this score using the following Bayes rule:

$$Tr^t(V_j) = Tr^t(R^t(V_j)/Conf^t(V_j)) = \frac{Tr(Conf^t(V_j)/R^t(V_j)) \times Tr(R^t(V_j))}{Tr(Conf^t(V_j))} \tag{4.4}$$

where $R^t(V_j)$ indicates V_j reputation at the specific time t, and $Conf^t(V_j)$ defines the credibility related to the event message sent by V_j at the specific time t. The Bayes rule is used because the calculation of the trust value at a specific time depends on the previous behavior of the node and the events that occurred in the past. The Bayes theorem allows the expression of inter-dependent events based on previous events and considering new observations. When many event messages are reported, the calculation of the global trust value of a node is based on the geometric average value of the established trust scores for each event using the following formula:

$$Tr^t(R^t(V_j)/Conf^t(V_j)) = \sqrt[|event|]{\prod_{k=1}^{|event|} Tr^t R^t(V_j)/Conf^t(k)} \tag{4.5}$$

where $|event|$ indicates the number of reported events from V_j.

To encourage the cooperation within the network, the edge vehicles apply an incentive algorithm to recompense or punish the nodes that respectively cooperate positively or negatively through sending correct or erroneous messages notifying specific events. The incentive algorithm (see algorithm 1) is based on the importance level of the sent messages. When the importance level of the message is higher, the recompense or the punishment value increases otherwise it decreases. A node is recompensed when it sends two messages that have been confirmed as correct. It is punished when they notify two events that will not be confirmed. Thereafter, the

incentive for V_j is executed by comparing its final generated trust to the threshold Thr_{ts}.

Input: R_j^{t-1}: old reputation value of V_j
$Thrs_{tr}$: trust threshold
W_q: weightiness level of the event E_q
θ= 2 , ϑ, η: incentive coefficients
Output: updated trust value T_j^t of node V_j
if W_q = *'high'* **then**
  if $T(R_j^{t-1}/Conf_j^t) \geq Thrs_{tr}$ **then**
    $T_j^t = T(R_j^{t-1}/Conf_j^{t-1}) \times \theta$
  end
  $T_j^t = T(R_j^{t-1}/Conf_j^{t-1}) \div \theta$
end
if W_q = *'ordinary'* **then**
  if $T(R_j^{t-1}/Conf_j^t) \geq Thrs_{tr}$ **then**
    $T_j^t = T(R_j^{t-1}/Conf_j^{t-1}) \times \vartheta$
  end
  $T_j^t = T(R_j^{t-1}/Conf_j^{t-1}) \div \vartheta$
end
if W_q = *'soft'* **then**
  if $T(R_j^{t-1}/Conf_j^t) \geq Thrs_{tr}$ **then**
    $T_j^t = T(R_j^{t-1}/Conf_j^{t-1}) \times \eta$
  end
  $T_j^t = T(R_j^{t-1}/Conf_j^{t-1}) \div \eta$
end
Return T_j^t;

Algorithm 1: Incentive algorithm for cooperative nodes

To recap the process, each node that joins the network should be in the first place authenticated through the trusted authority nodes to confirm the legitimacy of its identity. Each node disposes of a digital certificate. The edge vehicles manage the trust in the network. Whenever a node receives a report (step 2), it can request the trust score of the sender from the nearest edge vehicle (step 3). This later proceeds to the calculation of the trust score to exhibit the required value to the requesting node. During this fourth step, the edge vehicle checks firstly its cache memory. It takes the pre-determined trust score of the sender to calculate the new one, once available. If the old score is not provided by the cache, the edge vehicle communicates with the IPFS to retrieve the required trust score (if available). The trust estima-

tion process consists of the calculation of the reputation and the message-credibility. The edge vehicle exhibits the required value to the requesting node. The updated trust score is then preserved in the local Blockchain. Further, the edge vehicle keeps the frequent requested trust scores in its cache. The local mining consensus is performed by the edge vehicles. Finally, an incentive is placed for the involved nodes (i.e., senders, recommenders and edge vehicles). Such algorithm can encourage the vehicle nodes to be cooperative and collaborate in a honest way. Likewise, the local Blockchains are uploaded into the RSUs to be added within the global chain.

4.2.5 Simulation study and results

4.2.5.1 Simulation environment and metrics

We conducted the evaluation of our framework to validate its effectiveness. We implemented our proposal in Python environment under Ubuntu-18.04.5 machine. We created the python script that contains the APIs for the Blockchains implementation using the Ethereum Blockchain. To build the network module, we defined the different types of nodes in the network; including the edges nodes to upload the trust calculation code which calls the Blockchains APIs. We simulated nodes' attributes, attack instance, and node distribution to generate synthetic data and create a data frame using Python. We incorporated malicious nodes sending false accident reports (Bogus information attack). For a V2V scenario, we assumed that the network nodes are already grouped into local areas according to their locations. Each local area zone was defined with 2% of RSUs, 2% of trusted authority entities, 10% of edge vehicles, and 86% of ordinary vehicles. Furthermore, we built our network model with an assumption of having maximum of 40% of malicious vehicles. These selected percentages can seem realistic as the majority of the IoV entities are regarded to be likely trustworthy. The maximum nodes density was kept at 150 nodes, the vehicles' speed was defined at 40 km/h, and the reported events were considered as critical. To start the procedure of trust score calculation, we assigned the value 0.4 as the default reputation value for each ordinary node, and we set the threshold for the trust score at 0.7. Also, we considered that the nodes that provide recommendations are located within maximum 2 hops from the requesting node. Supposed that the trust score estimation will mostly consider the knowledge

metric (e.g., in case of unreachable nodes that provide recommendations). So, we defined the value of the weight α at 0.5, while we initialized β and δ at 0.3 and 0.2, respectively. At this step of evaluation, we point that the nodes that provide recommendations were considered honest. We varied the traffic density from 10 to 150 vehicles, and the percentage of malicious vehicles was as well changed from 10% to 40%. The different variables values used in our simulation tests are provided in Table 4.1. We evaluated the performances in terms of malicious node detection, trust calculation time, and communication overhead.

Table 4.1: Simulation parameters

Parameters	Values
Maximum nodes density	150
Percentage of edge vehicles	10%
Percentage of RSUs	2%
Percentage of authority nodes	2%
Average of vehicle speed	40 km/h
Maximum percentage of malicious vehicles	40%
Trust threshold	0.7
Weights of the reputation value function	α=0.5 β=0.3 δ=0.2

4.2.5.2 Simulation results and evaluation

To evaluate our proposal performances, we are based on a detection performance (Recall metric), and runtime properties analysis (trust calculation time and communication overhead). The Recall metric is related to the proportion of correctly recognized malicious vehicles (True Positive) among all the existing malicious vehicles (True Positive + False Negative).

$$Recall = \frac{TP}{TP + FN} \tag{4.6}$$

The trust calculation time refers to the duration it takes to compute trust values for nodes. This metric measures the computational efficiency of the proposed trust management scheme. It evaluates how quickly the approach can assess and update trust scores for nodes. The communication overhead refers to the additional data exchanged between nodes due to the implementation of the trust management scheme. This metric assesses the impact of trust-related communication on the network's efficiency. Our proposal showed good detection performance. Figure 4.4 illustrates the out-

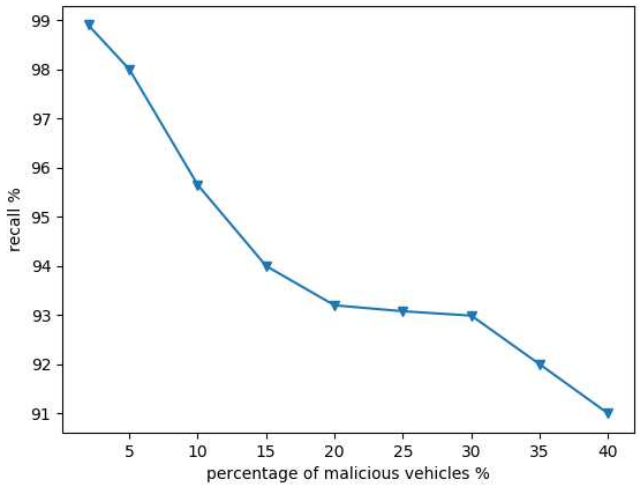


Figure 4.4: Recall rate

comes of the recall rate. The vehicles that sent false events yielded weak trust score. At this point, it is worth mentioning that a higher number of malicious recommenders leads to easier damage of network. From figure 4.4, the detection rate deteriorated with a high percentage of malicious vehicles. Nevertheless, detection performance is considered good with about 91%. Regarding time complexity performance, figure 4.5 shows that our proposal reached a good calculation time performance during the trust management process. The spending time of calculating the trust was dependent on the network density (nodes number). From figure 4.5, we can see that the increase of network density implied an increase in the trust calculation time. This is because in a denser network, there are more nodes actively communicating with each other, which leads to an increase in interactions that the trust management scheme must process. However, the calculation time performance of our proposal is satisfactory. Figure 4.6 illustrates the communication overhead ratio that we considered on the trust process. We evaluated this metric over the nodes density. We can notice that when the rate of malicious vehicles is about 40%, the overhead is not very high.

4.2.6 Summary

Our Blockchain based trust management framework showed good performance results in terms of detection rate of malicious vehicles, duration of establishing trust and the communication overhead related to the trust man-

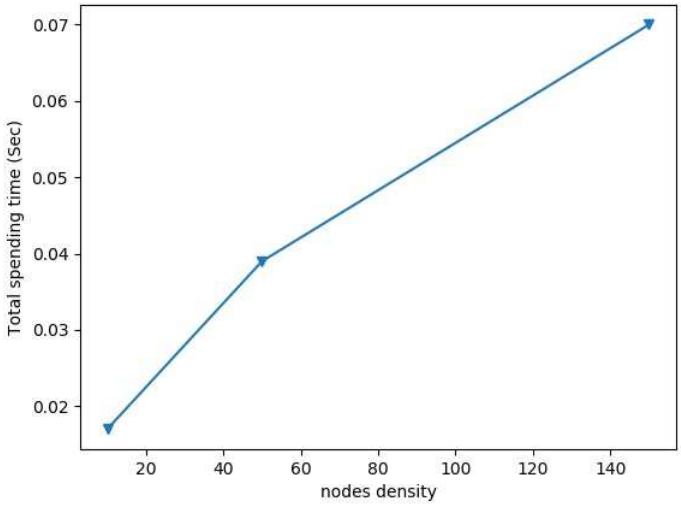


Figure 4.5: Trust calculation time

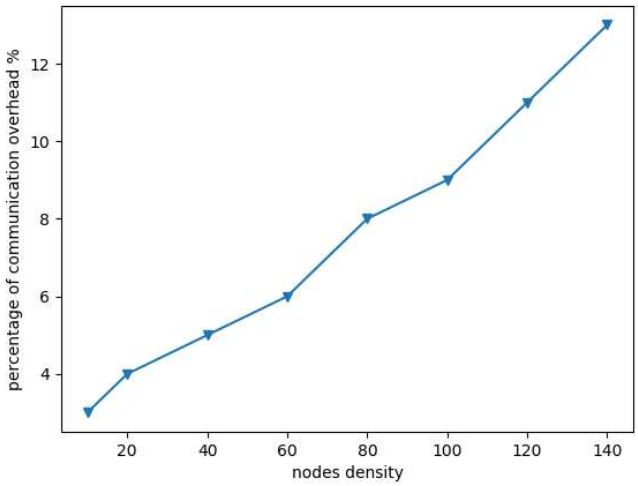


Figure 4.6: Communication overhead

agement. However, it was based on a centralized trust based approach. Thus, its performance is decreasing when dealing with effective scalability. Indeed, the performances quickly decrease when the network scalability is increasing. Moreover, when an attack occurs, it is easy to be propagated within the whole IoV network since the trust values are calculated with a centralized approach. In addition, since the IoV network is characterized with a high mobility, a centralized trust based approach can lead to the interruption of the trust calculation process when some events occur within the network or when some nodes leave the network. Furthermore, we did not focus on

the energy consumption of the proposed model. In the next section, we propose an extension to enhance the highlighted limitations. We make extensive simulation and evaluate the extension.

4.3 Blockchain and trust-based clustering approach for the IoV

4.3.1 Proposed approach overview

We mitigate the centralization issue, in our work, by relying on a decentralized trust process. More importantly, unlike the existing security solutions that may be not suitable for the IoV characteristic, mainly mobility and scalability. As discussed in previous chapter (subsections 3.4.2.1, 3.4.2.2), trust management approaches have used the Blockchain technology as well as the clustering techniques to enforce security within the vehicular context. However, none of these approaches have exploited both techniques simultaneously to improve the network security taking into account the network performances. The simultaneous use of the trust management, the Blockchain, and the clustering techniques within the same framework for the IoV environment is not very well explored. In this section, we are based on a decentralized trust and Blockchain based clustering approach to achieve good performances when dealing with scalability and mobility. As we aim to define a security approach that takes more into account the QoS of the system, we are interested on the network continuity. We introduce a clustering mechanism to construct our framework. We define our clustering process based on different parameters such as the trust value, the safety distance, and the energy factor. Indeed, the IoV is a pervasive environment characterized by a high mobility. Our framework is interested to ensure a continuity of the services through including the energy metric to define the network topology. We propose to extend our framework in section 4.2 based on the following key points:

- We define a new network topology based on a classification of the different nodes. The vehicles within the IoV network will get a different role depending on their parameters and the activities that they perform.

- We propose a decentralized trust management process that will be applied on the nodes classification. We keep the reputation and the message credibility metrics in our proposed process are the reputation and the message credibility.

- We propose a new trust based clustering approach to improve the IoV network performances. The different phases of the clustering process (the cluster formation, the cluster head selection and the cluster maintenance) are based on the trust management process, the safety distance and the energy metrics.

- We focus on the energy metric within the IoV network. First, we consider the energy as a metric to define our clustering approach. Second, we consider the energy consumption as a metric to evaluate our new proposed framework. The energy metric is important to consider since it directly impacts the network continuity. Moreover, it defines a compromise between the network security and the network QoS.

- We reconsider the two layer based Blockchain architecture. The exchanges between the network nodes and the Blockchain are updated since we introduce a new clustering approach. The local Blockchain is assigned for different clusters to proceed and save a local trust management values. The global Blockchain serves to exhibit the global trustworthiness of the communication system through the RSUs.

The rest of the chapter is organized as follows. Subsection 4.3.2 describes our architecture overview. Subsection 4.3.3 details our proposed clustering scheme and trust management process. The performance evaluation results are provided in subsection 4.3.4.

4.3.2 Proposed architecture overview

The architecture of the IoV network model is shown in Figure 4.7. The architectural model that we consider is based on a classification of the different IoV nodes. Similarly in subsection 4.2.2, the IoV network is composed of numerous entities. The IoV entities (vehicles, RSUs, ITS centers and trusted

authority nodes) use different wireless communication technologies to establish V2X communications. We assign the role of ordinary vehicles, edge vehicles, trusted authority nodes, and RSUs to the participating nodes. The edge vehicles handle trust-related requests of ordinary vehicles. They are assumed to have more computational power than ordinary vehicles. The trusted authority nodes are deployed to undertake the work to register the nodes. The RSUs are those static nodes that share road-related information. Also, we reconsider the two layer based Blockchain architecture. The edge vehicles build local Blockchains which will be added in the global Blockchain by the RSUs. From the QoS point of view and to enhance the IoV network performances, we organize our IoV network on clusters. We propose a clustering approach of the vehicle nodes based on the velocity and the distance similarity. Hence, local Blockchains are assigned for different zones to support local trust-based clusters. The details of the clustering process are given in subsection 4.3.3. From a security point of view and to build a secure framework, we propose to manage our clusters based on a trust management process.

4.3.3 Proposed Clustering Scheme

In this subsection, we define a clustering approach to associate it with our proposed trust management scheme in order to enforce security within the IoV. The building of our clustering architecture is based on three main phases: (1) the cluster formation; (2) the cluster head selection and (3) the cluster maintenance.

4.3.3.1 Cluster formation and node joining process

Each geographic area, controlled by a trusted authority, is divided on a set of clusters. The clusters are formed based on the velocity, the direction and the distance similarity. The nodes having similar parameters, in terms of velocity and direction, and belonging to the same area form one cluster. The trusted authority is responsible for broadcasting periodically a cluster formation request to select cluster heads. The cluster head selection is described in next sub-subsection. During the whole process, the cluster heads communicate with the different trusted authorities and can ask for informa-

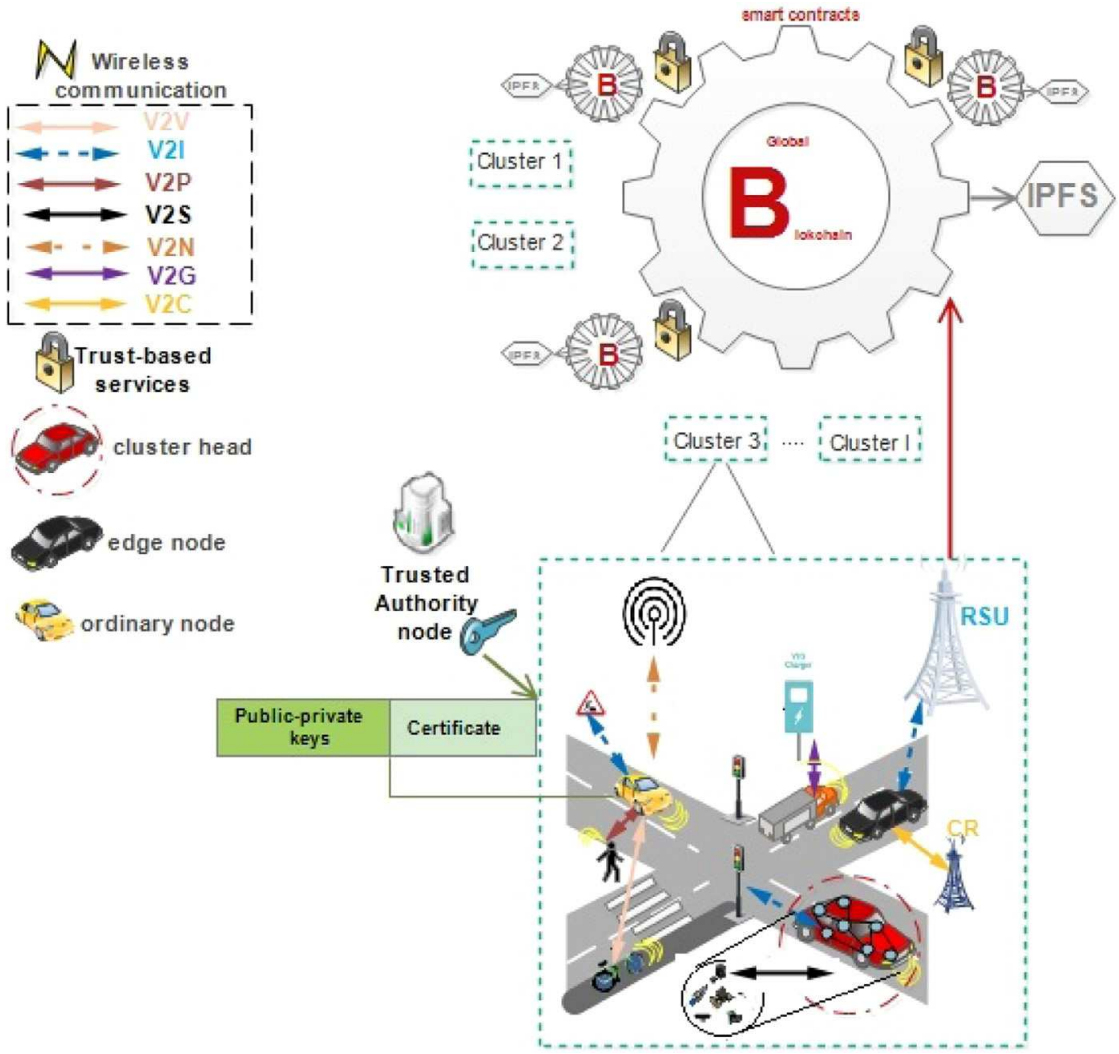


Figure 4.7: Proposed architecture

tion that concern the vehicles belonging to their clusters. When a vehicle joins a geographic area, it should respect two phases: (1) registration and authentication phase, and (2) cluster joining phase.

- Registration and authentication phase

As in section 4.2, the trusted authority (TA) performs the authentication phase. In our scheme, the trusted authority is based on the trust management process to evaluate if a vehicle could be safely registered or not. When a vehicle requests to register in the network, the trusted authority validates and approves its request based on the trust value associated to that vehicle. The registered vehicle is associated with a certificate and a pair of public and private keys from the trusted authority ($Cerf_{iy}^{m}(Pb_{i}^{m}, Sg_{y}, IF_{y}, Epi_{i},$

R_i))(see Eq.(4.1).

- Cluster joining

Once the authentication process succeeds, the vehicle can start the second phase and join the nearest cluster. For that it broadcasts a cluster joining request. The request contains its delivered certificate and other parameters indicating its velocity, position and direction. To check that request, the cluster head will use the trust authority public key that it received when it registered and authenticated with the trust authority. Moreover, the cluster head checks the reputation value of the sender that is included in its certificate. If the trust value of the sender is equal to zero, the cluster head can consider that the vehicle could be non trusted and cannot join the cluster. Moreover, the cluster head will check if the request sender satisfies the cluster properties. When the vehicle properties correspond to the properties of a specific cluster and the reputation value is positive, the cluster head sends a positive response to confirm that the vehicle can join the cluster and its certificate to prove that it is a honest node. The vehicle sending the initial request checks the cluster head identity using its certificate. If this authentication phase succeed, then the vehicle sends a confirmation response of joining that cluster. Hence, the vehicle will be able to participate in the cluster local trust management process. Here, we note that the cluster head selection process is described in next subsection. Later on, the vehicle trust value will be updated by the cluster head depending on its behavior until leaving the cluster. When the vehicle leaves the cluster, the cluster head shares the updated vehicle trust value with the RSU and the trusted authority of its geographic zone. Figure 4.8 summarizes the two phases of the cluster formation process.

4.3.3.2 Cluster head selection

Initially, each trusted authority is responsible of one geographic area. To initiate the clustering scheme, the trusted authority initiates the cluster head selection process. The request is only sent to the edge vehicles since the trusted authority knows the identity of all vehicles belonging to its geographic area. The request contains the properties of the clusters to be formed, i.e. the size of the cluster. We base our cluster head selection process on

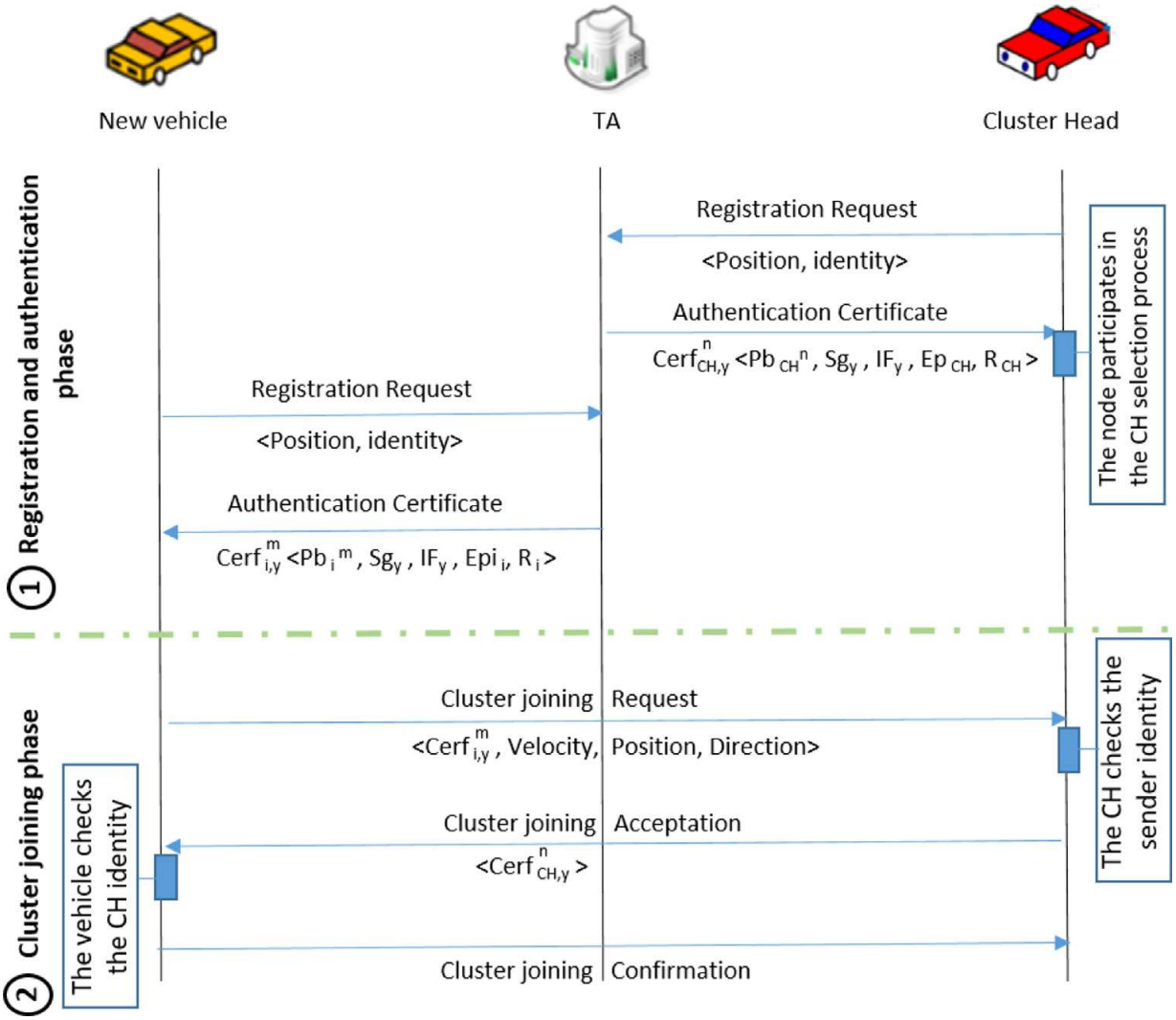


Figure 4.8: Cluster formation process

three parameters: the trust value, the safety distance and the resource indicator. Our choice of these parameters is based on enforcing both security and QoS of our IoV network. The trust value helps to improve the security within the cluster. The safety distance and the resource indicator ensure the network continuity and stability. A safe inter-vehicle distance will assist to avoid collisions. Safety distance is the minimal inter-distance that keeps two vehicles in safety driving situation while they move and apply braking at maximal capacities. Hence, the selection of cluster head that is situated in a safe distance from the following vehicle allows to avoid rear-end collision and prevent the damage of cluster topology. The resource indicator is composed of three parameters related to the vehicle: the storage capacity, the computing capacity and the charging capacity. The charging capacity (battery level) is mandatory when the vehicle is electric. Indeed, the electric

vehicles may face the discharging problem. Hence, to minimize the energy consumption and so minimize the discharging problem, we have to implement lightweight algorithms when integrating security within the vehicles. If the algorithm complexity is low, then its energy consumption will be less. When the vehicle is not electric, the charging capacity is fixed at the maximum value. Let us consider C an index of the computing capacity, S an index of the storage capacity and B an index of the battery charging. The resource indicator $EN(V_i)$ of the node V_i is calculated as follows:

$$\mathrm{EN}(\mathrm{V}_i) = C + S + B \quad (4.7)$$

The cluster head has to manage the cluster members joining and the trust management within the cluster. For that, the cluster head should belong to the edge nodes that are characterized by high performances. Thus, an edge vehicle can become cluster head once the following criteria are satisfied: (i) the edge vehicle' trust meets the required threshold, (ii) the edge vehicle respects a suitable safety distance from the other nodes, (iii) and the edge vehicle has sufficient residual energy. The cluster head will be the node having the maximum value of the following formula:

$$\mathrm{Selec}(\mathrm{V}_i) = \Upsilon \times \mathrm{T}(\mathrm{V}_i) + \Psi \times \mathrm{Saf}(\mathrm{V}_i, V_j) + \Omega \times \mathrm{EN}(\mathrm{V}_i) \quad (4.8)$$

where Υ, Ψ, and Ω are respectively the weights for the trust value, the safety distance, and the resource indicator. We note that $\Upsilon+\Psi+\Omega= 1$. These weights are defined as the relevance of scores for each matched selection criteria, i.e., as the proportions of cluster head attributes' significance to the required selection criteria. The edge vehicle having the highest $Selc$ value is picked to be the cluster leader. To determine the best values of Υ, Ψ, and Ω that optimize the network performances, we conducted a set of tests during our performance evaluation phase described in subsection 4.3.4. The performances evaluation has been performed considering different values for these three weights. The selected values for these weights are presented in sub-subsection 4.3.4.2 The trusted authority communicates continuously with the cluster heads through sending them a "HELLO Cluster Head" requests. When a cluster head do not respond, the trusted authority initiates

the cluster head selection process. Moreover, the associated parameters of different cluster heads may vary with time. For example, the trust value of an edge node may increase or decrease. Hence, to keep an updated clustering scheme, the trusted authority broadcasts periodically requests to re-elect the cluster heads. Figure 4.9 and algorithm 2 illustrate the cluster head selection process.

Input: E=(E_1, E_2, ...,E_k): set of k edge nodes of the cluster l
Input: Thr_{ts}, Thr_{EN}, $Safdis$: thresholds
Output: Selected cluster head CH_l for cluster l
$CH_l \leftarrow$ trusted-authority;
$T_{Ch} \leftarrow 0.4$;
for ($i \leftarrow$0 to $|E|$) **do**
  if (R_i>=Thr_{ts}) $\wedge$(EN$_i$>=Thr_{EN}) $\wedge(Saf(i,j) >=$Safdis)**then**
    $Selc(i) \leftarrow \Upsilon \times$R$_i$ +$\Psi \times$Saf(i,j)+$\Omega \times$EN$_i$;
  else
    $Selc(i) \leftarrow$ 0 ;
  endIf
  if $Selc(i)$ > T$_{Ch}$**then**
    $T_{Ch} \leftarrow i$;
  endIf
endFor
Return T_{Ch};

Algorithm 2: Cluster head selection

4.3.3.3 Cluster maintenance

During this phase, the topology of the IoV network is updated depending on the events that may occur within the network. To maintain the clustering scheme updated, the trusted authority communicates continuously with the cluster heads. Indeed, it broadcasts periodically a "Hello Cluster Head" requests to probe the existence of different cluster heads. When a response is not received from a specific cluster head, the trusted authority can deduce that cluster head left its geographic area. In this case it triggers the selection cluster head process. Even though all cluster heads are still remaining in the trusted authority geographic area, the trusted authority sends periodically requests to re-elect the cluster heads within the IoV network. The update of the cluster heads is mandatory since the trust values of different nodes may change based on the trust calculation process. Indeed, the cluster head

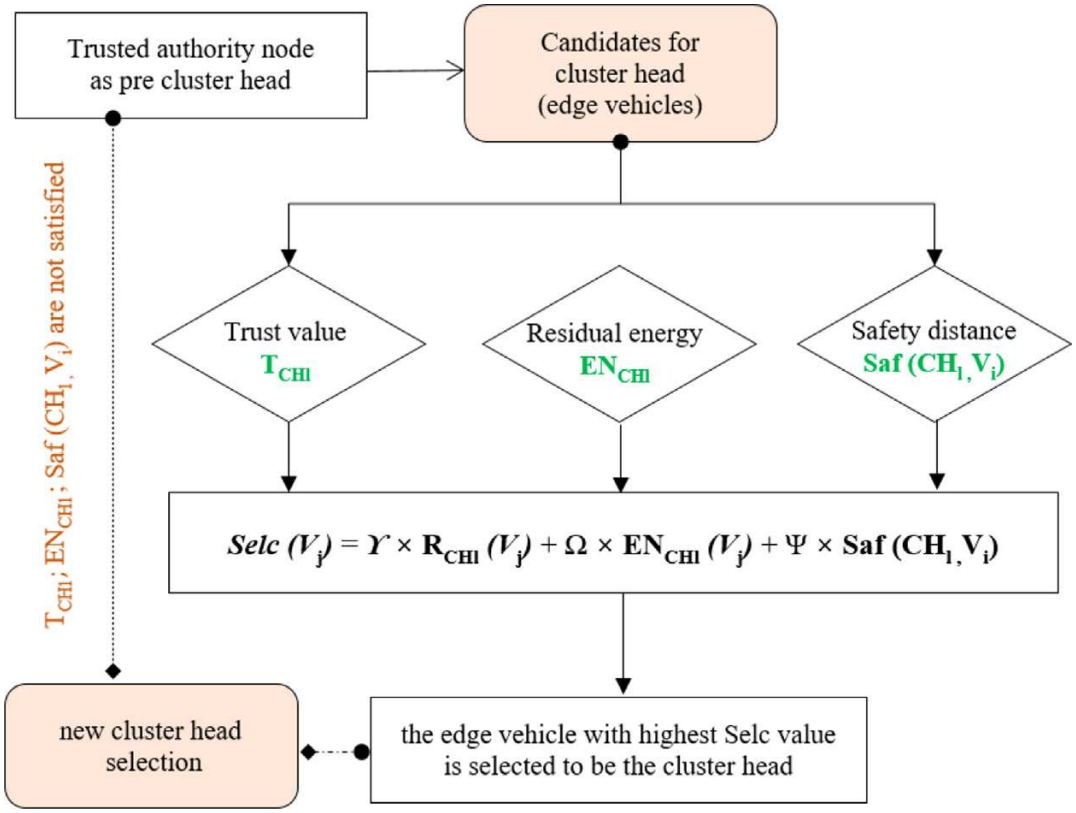


Figure 4.9: Cluster head selection process

selection is based on the trust value among others. Hence, it is mandatory to always check that the elected cluster head has the highest trust value. Within the network, some nodes may be far from all clusters ranges. These free nodes continue sending registration requests until they join a cluster. Moreover, a node has the possibility to change from one cluster to another based on its direction and velocity. In addition, when two cluster heads are too near to each other, they will merge during the next round of the cluster head selection process. Indeed, when the trusted authority triggers the cluster head selection process, these two nodes will exchange messages about their positions. The calculation of cluster properties will lead to consider them within the same cluster. The distance separating them will be less than the range of a cluster. The cluster head having the highest trust value will be the leader in the merged cluster. The other respective cluster head will become ordinary member in that cluster. After the merge of clusters, some related members may become outside the signal range of the new cluster head. These members will retake the state of free nodes. Figure 4.10 summarizes the cluster maintenance process.

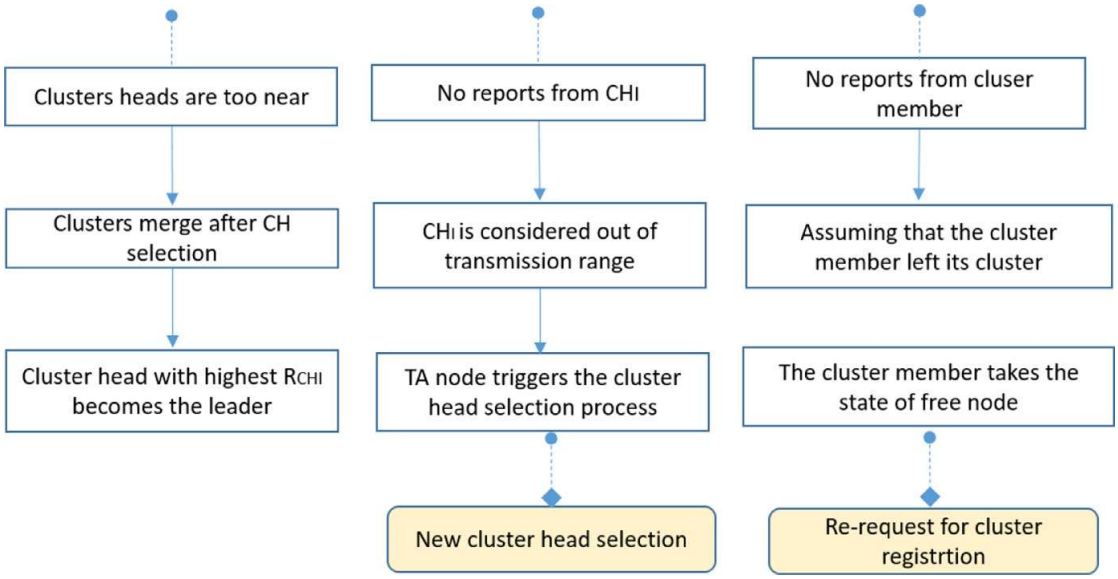


Figure 4.10: Cluster maintenance process

Our proposed framework is based on the clustering scheme that we detailed in above. For that, let us assume that our IoV network is composed of l clusters (C_1, C_2, ..., C_l). Each cluster C_l has a cluster head CH_l and is composed of a set of V ordinary vehicles and a set of E edge vehicles. To manage the trust, we propose a decentralized process applied on clusters. Initially, the selection of the cluster head is based on the trust value, the resource indicator, and the safety distance metric. As the nodes are grouped into clusters, the cluster heads manage the communications. The trusted clusters heads are used to manage the trust (see figure 4.11). Each node that joins the network should present its trusted identity to the cluster head. Each cluster head manages the trust level of its cluster nodes. Each cluster head assigns to all cluster nodes the initial trust value of 0.4. Every cluster node can query the trust of the sender node from its cluster head (CH_l). CH_l computes the reputation of cluster member based on behavior and honest cooperation within the cluster. The cluster head saves the computed trust in the local Blockchain. The cluster heads are part of the edge nodes. Accordingly, the local mining consensus is performed by the edge nodes. Upon being created, the local Blockchains are added within the global Blockchain by the RSUs. This serves to exhibit the global trustworthiness of the communication system. As a reminder, the trust value is calculated based on a reputation value and a message credibility metric. The reputation value is derived from old reputation and recommendations from edges nodes and RSUs (Eq.(4.2)). Next, the cluster head CH_l evaluates

the message-credibility. The computation of message-credibility is based on location proximity metric (Eq.(4.3)). Finally, the cluster head CH_l computes the global trust score to assess the trust in the network. The Eq.(4.3) and Eq.(4.4) are applied to derive the final trust score. Also, the process of cluster trustworthiness assessment involves an incentive for engaged nodes (i.e., senders, recommenders and cluster heads). The clusters heads give reward or penalty to motivate nodes for cooperation (see algorithm 1).

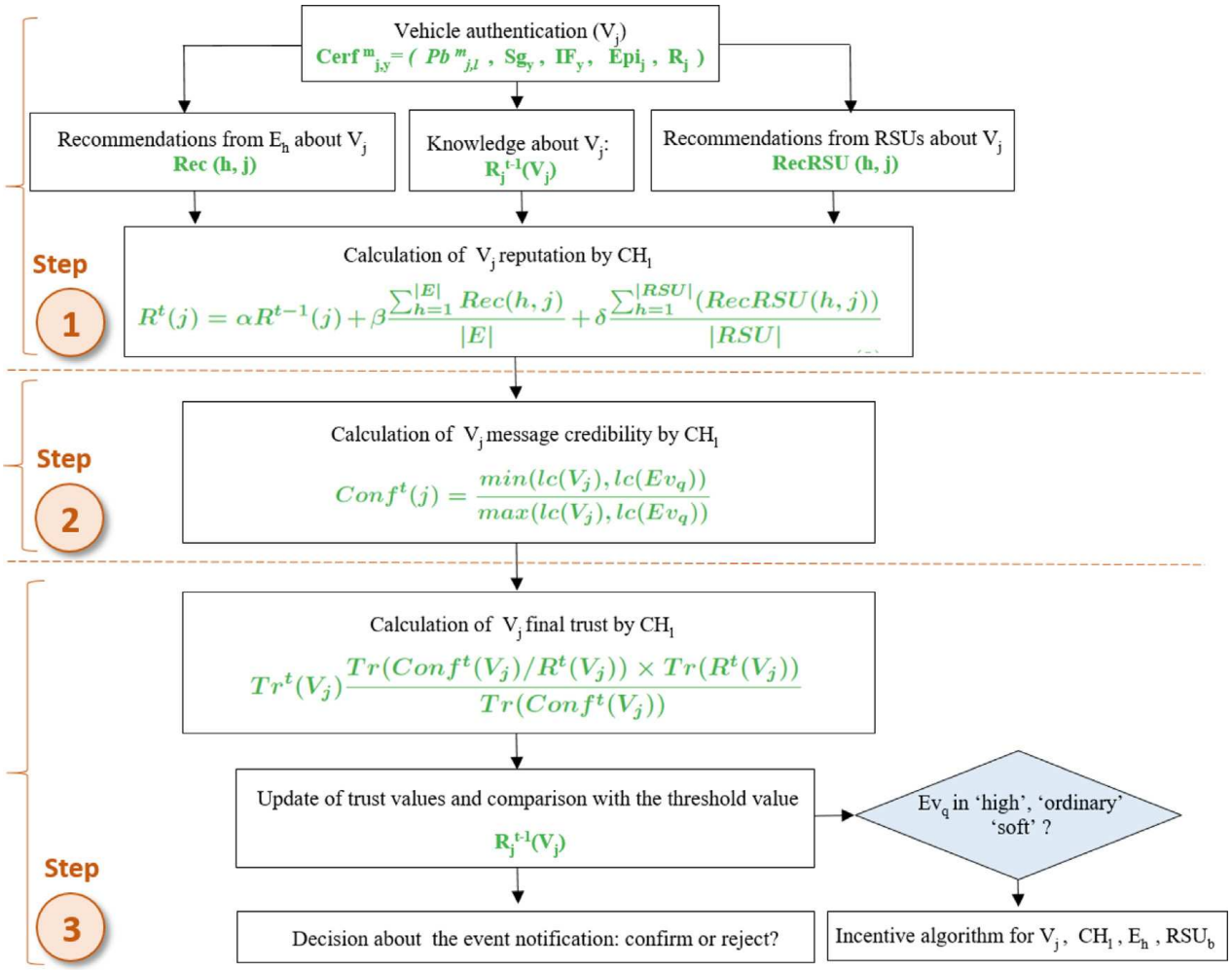


Figure 4.11: Process of cluster trust management

Figure 4.12 recaps the general phases of the trust management process with clusters.

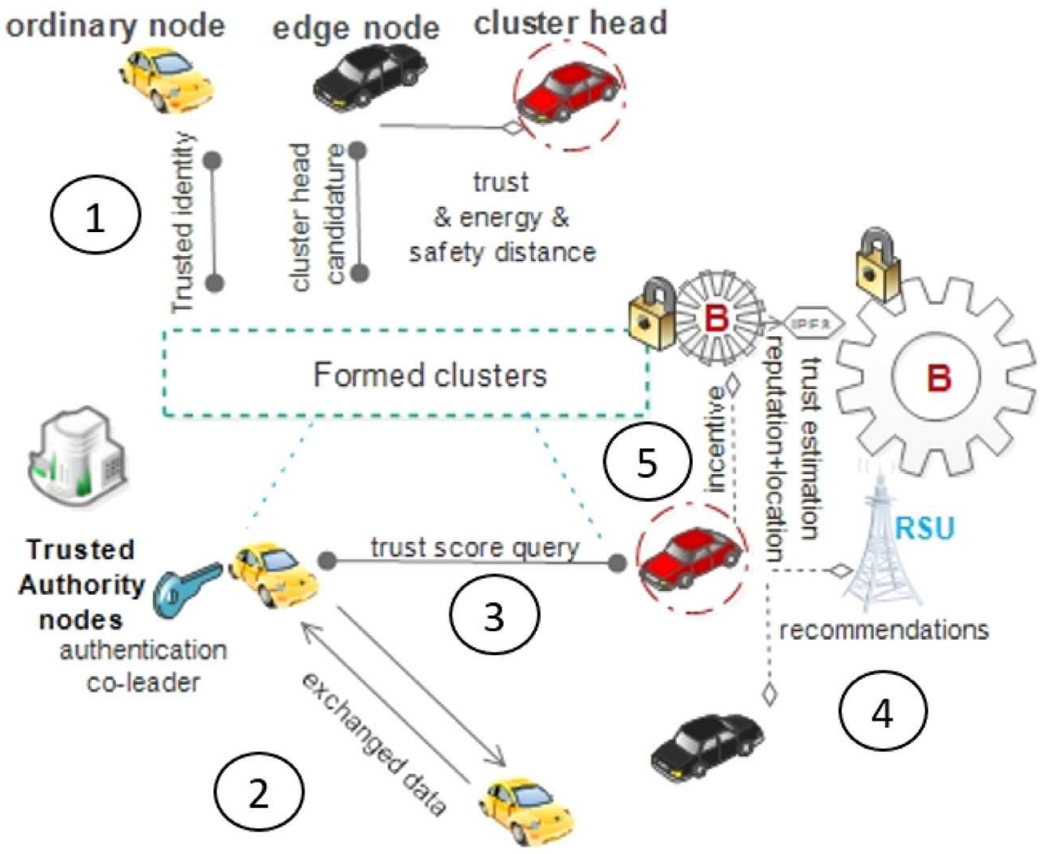


Figure 4.12: General phases of trust management with clusters

4.3.4 Simulation study and results

4.3.4.1 Simulation environment and metrics

To evaluate our framework performances, we conducted many simulation tests. For that, we used the Python environment and the network simulator NS-3. NS-3 is a discrete simulator of events that targets networking analysis. The simulator SUMO is also used to consider the mobility aspect of the vehicles, and generate traffic scenarios. SUMO is known to create traces that mimic real-world traffic. It supports the modeling of pedestrians, bicycles, passenger cars, trucks, busses, trains and even ships. To do so, we installed the NS-3 dependencies and we created the python script containing the Blockchains APIs (Ethereum Blockchain). Similarly to the conducted simulation in subsection 4.2.5, we identified the different types of nodes. We created node instances in the NS-3. We defined the Blockchain nodes in Ganache and made API calls to the Ganache nodes. The mobility traces generated by SUMO were imported into the NS-3. Figure 4.13 summarizes

our simulation environment. We simulated in the NS-3 Bogus information attack scenario . We assumed that each local area zone included 2% of RSUs, 3% of trusted authority entities, 10% of edge vehicles, and 85% of ordinary vehicles, with maximum of 50% of malicious vehicles. We conducted the evaluation for a highway road scenario, where the maximum nodes density was defined to be 400 nodes. The vehicles varied their speeds between 90 km/h to 150 km/h, and the reported events were assumed to be critical. In order to begin the calculation of the trust value, we initialized the reputation value for each sender at 0.4, also, we defined the threshold of the trust at 0.7. The maximum number of hops from the queried recommenders was considered to be 2 hops. We assumed that the recommenders are trusted. Besides, as we consider the importance of knowledge factor (associated to the weight Υ) in the calculation of the reputation value, we initially set the value of Υ at 0.5. Ψ, and Ω were initialized at 0.3 and 0.2, respectively. We assessed the performance of our proposal under various scenarios, considering different traffic densities ranging from 10 to 400 vehicles and varying the percentage of malicious vehicles from 20% to 50%. Table 4.2 summarizes the simulation parameters.

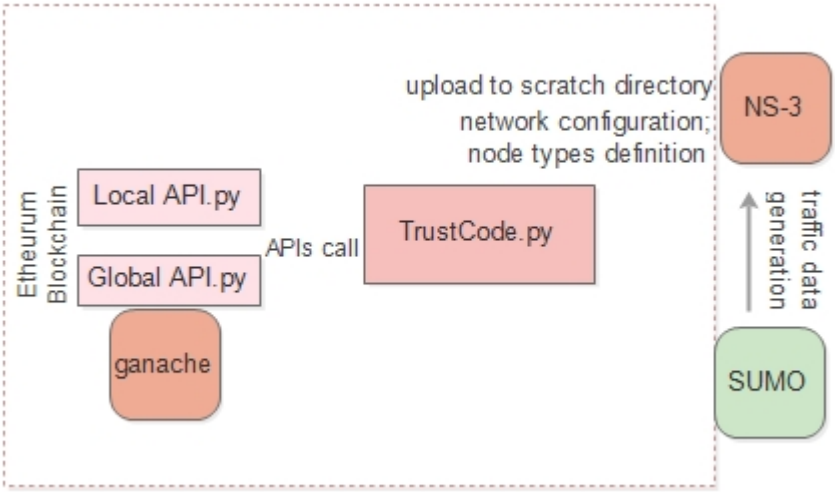


Figure 4.13: Simulation environment

4.3.4.2 Simulation results and evaluation

To evaluate our proposal performances, we are based on a set of metrics that highlight the main outcomes of our scheme.

Table 4.2: Simulation parameters

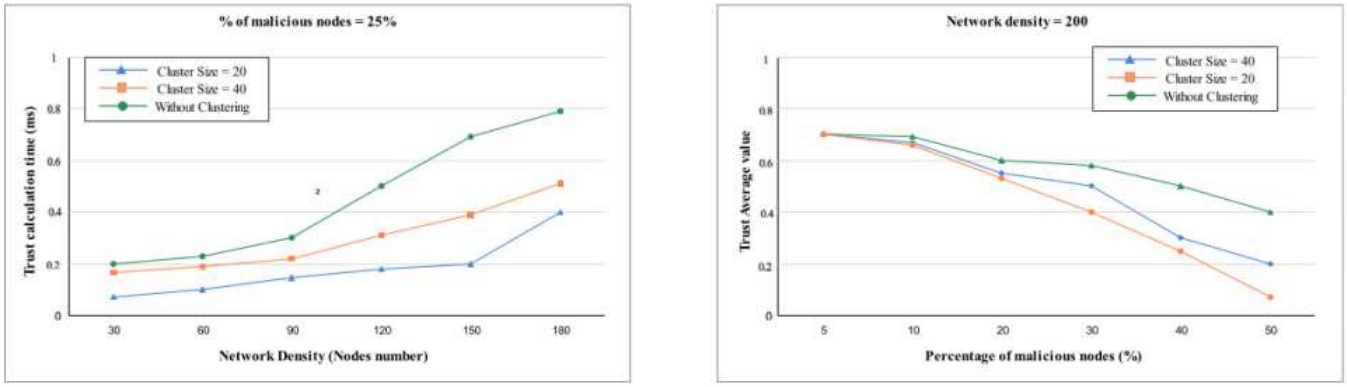
Parameters	Values
Maximum nodes density	400
Percentage of edge vehicles	10%
Average of vehicle speed	110 km/h
Maximum percentage of malicious vehicles	50%
Trust threshold	0.7
Initial values for weights of the reputation function	Υ=0.5 Ψ=0.3 Ω=0.2
Transmission range	350 m
Size of packet	50 bytes
Simulation time	150s
Mac protocol	IEEE802.11p

- Trust scheme analysis

The description of a decentralized trust model shows better results when measuring the spending time for calculating trust. This metric is closely depending on the cluster size. From figure 4.14(a), we can deduce that when the network density is increasing, the clustering scheme is greatly enhancing the convergence of the trust calculation process. We can notice that the trust calculation process is too long when we do not consider the clustering scheme. It may reach 0.8 ms when the network density is about 200. However, when considering clustering, the trust calculation time is about 0.4 ms. We can also notice that the trust calculation time decreases when the size of a cluster is smaller. Hence, we conclude that a decentralized trust management approach is more adequate mainly when the network density becomes significant. Moreover, we evaluated the trust value average considering an increasing percentage of malicious vehicles in figure 4.14(b). We can notice that the trust average value decreases when the malicious vehicles percentage increases. The clustering scheme with smaller size shows a better reaction to attacks. Indeed, when the managed area is smaller, the detection rate is better. When a malicious node is detected the trust average value decreases quickly to limit the propagation of the erroneous messages. We note that for these measurements, we only consider the trust calculation process. The duration of forming the clusters is not included.

- Detection performances analysis

To evaluate the detection performances, we are based on the Recall (Eq.(4.6)) and Precision metrics. The precision metric is related to the proportion of



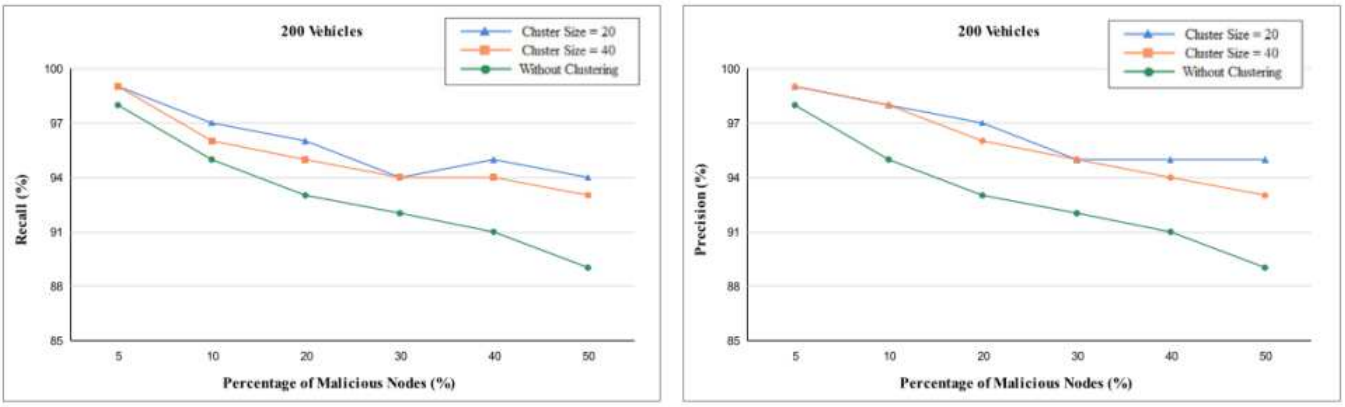
(a) Trust Calculation Time (b) Trust Average Value

Figure 4.14: Trust approach evaluation

correct positive identifications of malicious vehicles (True Positive) among the total number of vehicles (True Positive + False Positive).

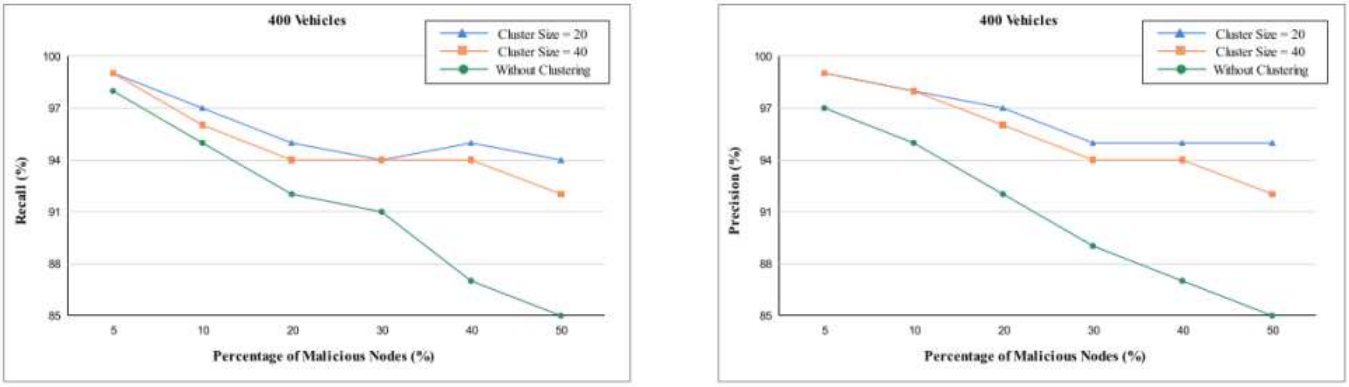
$$Precision = \frac{TP}{TP + FP} \tag{4.9}$$

The performances evaluation of our proposal shows that the clustering scheme reinforces the attack detection rate when considering the recall (figure 4.15(a)) as well as the Precision (figure 4.15(b)). Indeed, when clustering is not considered, the erroneous messages of malicious nodes will be largely propagated within the whole network. Small clusters help to restrict the erroneous messages beyond that cluster. Moreover, the benefit of such approach is more significant when the network density is increasing. Applying trust within clusters helps also to enhance the attack detection performances. Figures 4.15(a) and 4.15(b) show that the recall and precision rates are at least about 94% when applying the clustering scheme with a cluster size of 20 and a network density of 200 nodes. However, they are reaching about 89% when the clustering scheme is not applied. To evaluate the scheme performances when considering scalability, we calculated the Recall and the Precision with two different scenarios where the density is about 400 nodes (figures 4.17(a) and 4.17(b)). These rates are quickly deteriorated when the network density increases and the clustering scheme is not applied. For example they can reach less than 85% when the network density is about 400 nodes and the malicious vehicles rate is about 50%.



(a) Recall Rate (b) Precision Rate

Figure 4.15: Detection performances: network density = 200 nodes



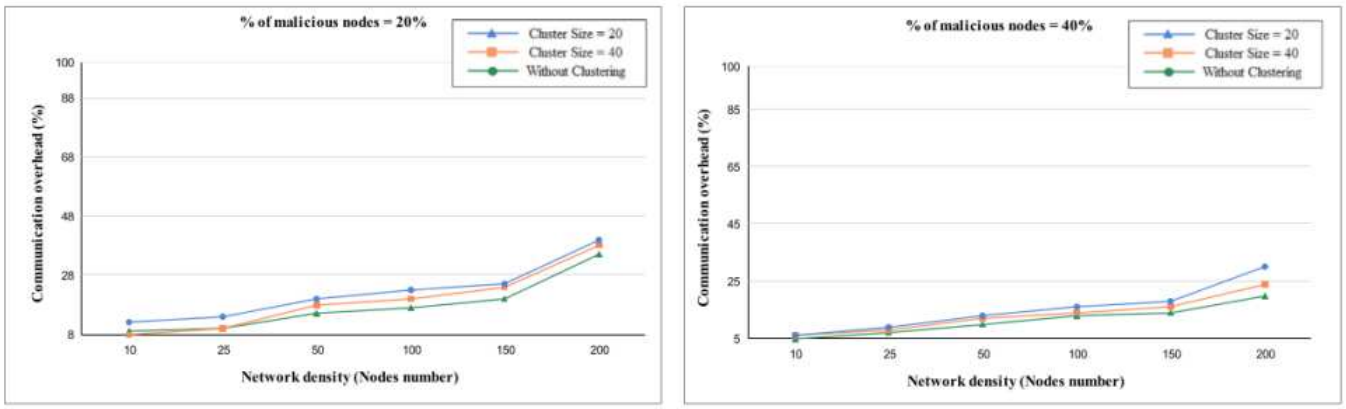
(a) Recall Rate (b) Precision Rate

Figure 4.16: Detection performances: network density = 400 nodes

- Runtime properties analysis

To evaluate the cost of our clustering scheme, we measured the overhead that the clustering scheme may add on the different communications and compare it to the scheme without clustering. Here, we are considering the overhead on the whole process (trust + clustering) for all communications. Figure 4.17 shows that the clustering scheme add an overhead to different communications. This overhead is mainly due to the different phases of the clustering process. However, we can notice that the added overhead is not too much significant compared to the exchanged data. When we consider a higher malicious vehicles rate, the detection mechanism helps to detect these nodes. Hence, the density of honest nodes decreases. As a result, we

can notice that when the rate of malicious vehicles is about 40%, the overhead is still not very high when we consider the density of 200 nodes. From another side, we consider the throughput metric. The throughput is defined as the amount of packets exchanged successfully during a particular time frame. The evaluation of the throughput is a significant perspective to interpret the runtime performance. Figure 4.18 displays the enhancement in the throughput when considering clustering and particularly when the cluster size is smaller. Indeed, the clustering scheme helps to detect malicious vehicles and to eliminate them and so they will not negatively impact the rest of the network. However, we can deduce that the increase of the percentage of the malicious vehicles affect the network throughput since the malicious nodes will participate in dropping and deviating messages. When the clustering scheme is not applied, a malicious node can act on different zones of the network and so deteriorate the throughput.



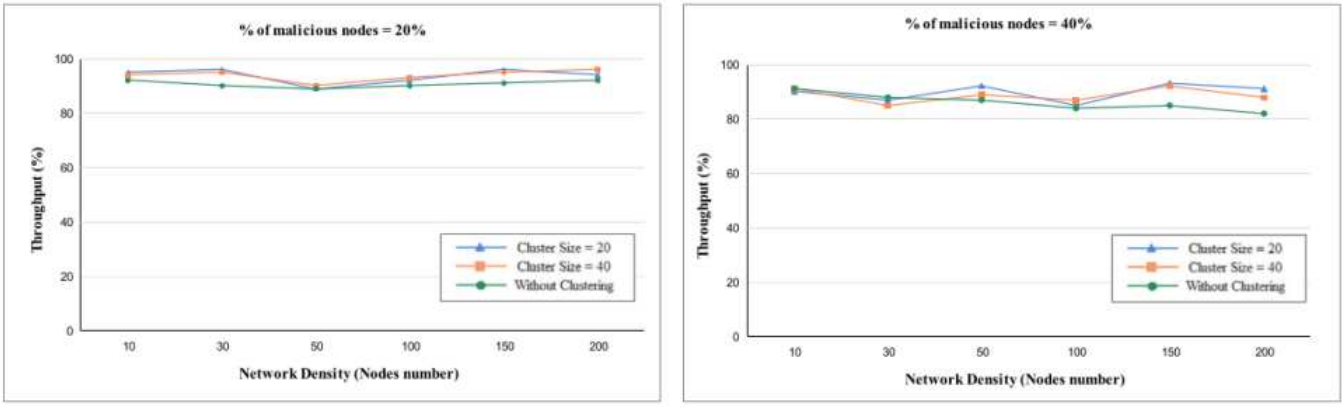
(a) Overhead: % malicious nodes = 20%

(b) Overhead: % malicious nodes = 40%

Figure 4.17: Communication overhead

- Clustering scheme complexity

In our proposal we aim to propose a lightweight scheme. For that, we considered the clustering technique and the use of the Blockchain technology. A lightweight scheme can be easily adapted to an energy dependent environment like for example the electric vehicles. When the calculation complexity of a scheme is low, its energy consumption is also low. We note that we consider the time complexity of the scheme. It means the computational complexity that describes the amount of computer time it takes to run an



(a) Throughput: % malicious nodes = 20%

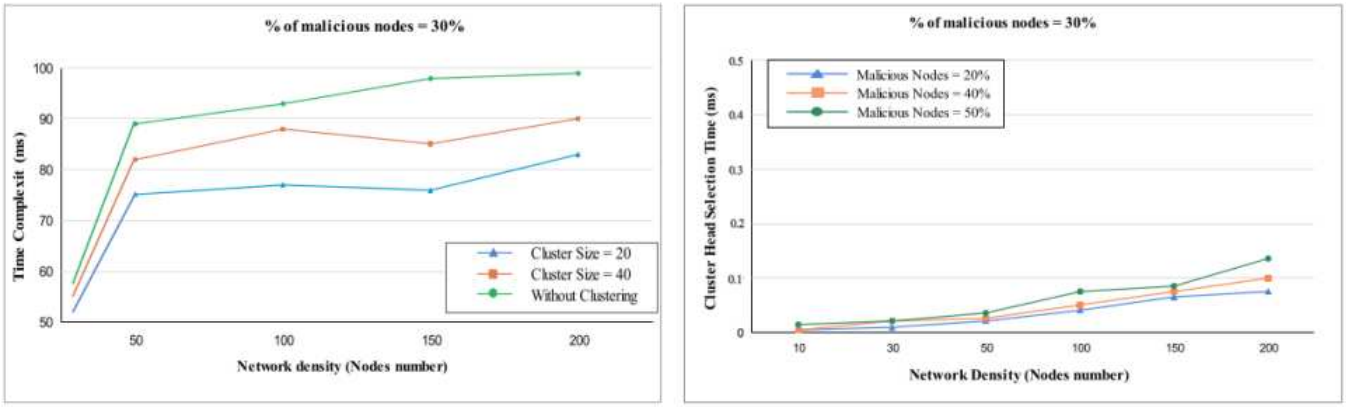
(b) Throughput: % malicious nodes = 40%

Figure 4.18: Throughput

algorithm. To evaluate the complexity of our approach, we considered the global complexity of the whole approach, including the trust management and the clustering phases (see figure 4.19(a)). For the tests, we calculated the complexity time (mseconds) depending on the nodes number. Considering a cluster size of 20, we notice that the time complexity is about 65 ms when the network density is 30 nodes. For the same scenario, the time complexity reaches 80 ms when the network density is 200 nodes about. We can also note that the time complexity is lower when considering the clustering scheme, specifically with smaller cluster size. Moreover, we calculated the complexity of the cluster head selection phase that depends on the variation of the trust management process (see figure 4.19(b)). When the cluster head selection phase takes less time to be executed, the time complexity of the whole scheme is better. For that, we evaluated the execution time that may take the cluster head selection phase depending on the network density. We can notice that even when the network nodes number is about 200 nodes the execution time of the cluster head selection phase is low. Thus, it does not deteriorate the whole complexity scheme. We can also deduce that for both cases, the clustering schemes have a lower complexity. Indeed, the decentralized approach based on clusters helps to decompose the computing complexity. It also gives better results when the clusters have a small size.

- Network stability evaluation

To evaluate the impact of our security scheme on the network performances, we consider the network stability metric for two scenarios: with 30% of ma-

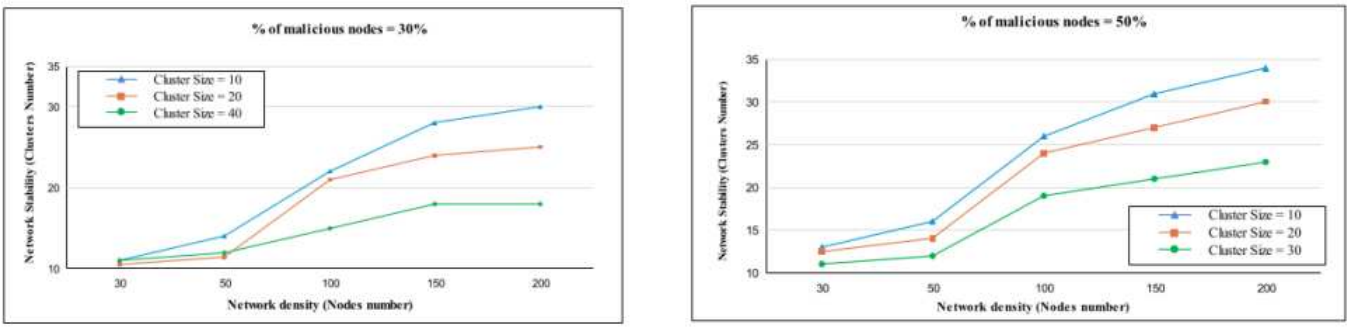


(a) Complexity of global scheme

(b) Cluster Head Selection Complexity

Figure 4.19: Scheme complexity

licious vehicles and with 50% of malicious vehicles. Indeed, the vehicles mobility can impact the network connectivity. For that, the proposed clustering scheme has the advantage to ensure the network continuity and to enhance the network stability. The number of formed cluster is related to two criteria. The first criteria is the vehicle mobility. The second criteria is the percentage of malicious vehicles. In figures 4.21(a) 4.21(b), we present the average number of formed clusters during the simulation period depending on the network density. We can notice that when the percentage of malicious vehicles increases, the network stability is almost the same. For example, when the network density is about 200 nodes, we can reach 30 and 34 formed clusters for respectively 30% and 50% of malicious vehicles. Thus, we can deduce that our trust-based detection scheme does not impact the network stability since the clusters number is not much increasing when the rate of malicious vehicles rises. To compare the performances of our network stability, we considered the scheme presented in [**82**]. The comparison is based on the average cluster duration. Indeed, such a metric is very significant for the vehicular networks stability because forming a cluster increases the energy consumption which is not very suitable mainly for electric vehicles. It is calculated as the average of the duration of all formed clusters during the complete simulation process. To calculate that metric for our scheme, we considered the same parameters indicated in [**82**]. The nodes transmission range is about 200m, the transmission rate is 8 Mbps and the network density is about 120 nodes. Figure 4.21 shows the obtained results for the compared schemes. We can notice that our approach proposes better performances since its average cluster duration (280 s) is better



(a) Network Stability: 30% malicious nodes (b) Network Stability: 50% malicious nodes

Figure 4.20: Network stability

than the one of StabTrust scheme (250 s). Later on, we performed the same tests with a higher network density for our scheme (300 nodes). The obtained average cluster duration is about 260 s. For that network density, the result for StubTrust is not provided in [**82**]. However, we can notice that the obtained result is still good even when we increase the network density for our scheme proving that our scheme ensures a very satisfying network stability.

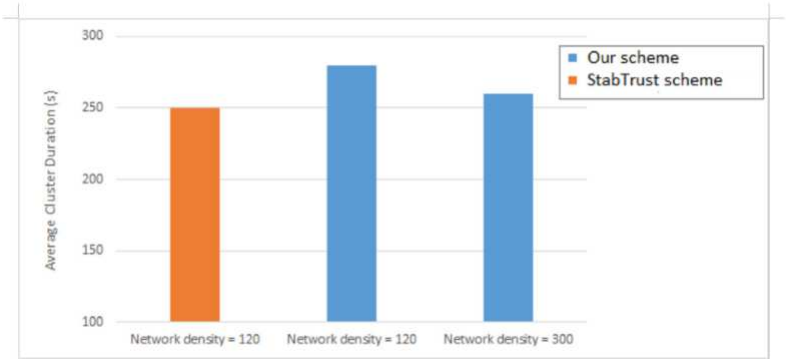


Figure 4.21: Average cluster duration comparison

- Weights evaluation for cluster head selection phase

The cluster head selection phase is based on the three weights Υ, Ψ, and Ω. To fix the corresponding value to each of them, we conducted a set of tests to calculate the detection rate based on the variation of these three parameters. For the first tests, we fixed the simulation parameters to those presented in Table 4.2. We considered a network of 400 nodes with 10% of

malicious vehicles, the average of vehicles speed is about 110 km/h and the transmission range is 350 m. The result of these tests is represented in figure 4.22. The figure shows the variation of the detection rate based on the variation of Υ, Ψ, and Ω. Each round represents a test. For each round we can see the corresponding value for the three weights as well as the detection rate result. We can notice that the minimum detection rate is reached when one of the three weights has a very low value (< 0.1). For example, when Υ=0.3, Ψ=0.7 and Ω=0.3 the detection rate is about 80%. However, the maximum detection rate is reaching its maximum when the three weights are in $[0.2, 0.6]$. For example, when Υ=0.4, Ψ=0.3 and Ω=0.3 the detection rate is about 98%. Moreover, we performed another set of tests considering two other different network configurations. Within these configurations, we varied the network density as well as the percentage of malicious vehicles. Our aim is mainly to check how the three parameters Υ, Ψ and Ω behave in a different network configuration. For that, we considered two other configurations: (1) configuration 1: nodes number = 200, % of malicious vehicles = 30%; (2) configuration 2: nodes number = 300, % of malicious vehicles= 20% . The results of these tests is presented in figures 4.24(a) and 4.24(b). Based on these Figures, we can notice that the good detection rates are still achieved when the three parameters are belonging to $[0.2, 0.6]$. When one of these parameter value is too low ($<0,2$) or too high ($> 0,8$) the detection rate is at its minimum value. These tests helped us to select the good values of Υ, Ψ and Ω to fix for our scheme. All our performance evaluation scenarios have been tested with the following values: Υ=0.4, Ψ=0.3 and Ω=0.3.

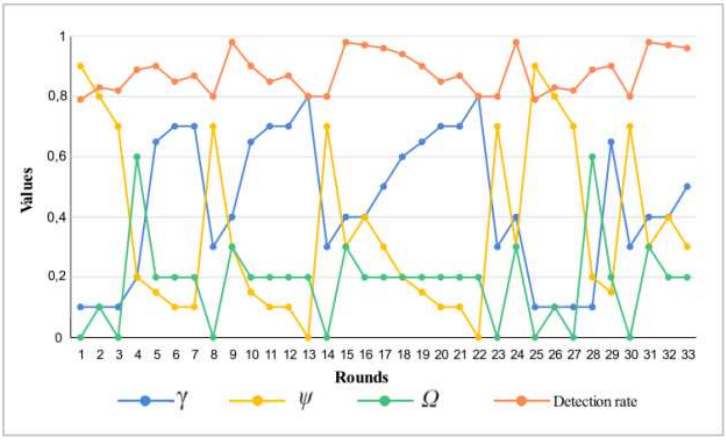
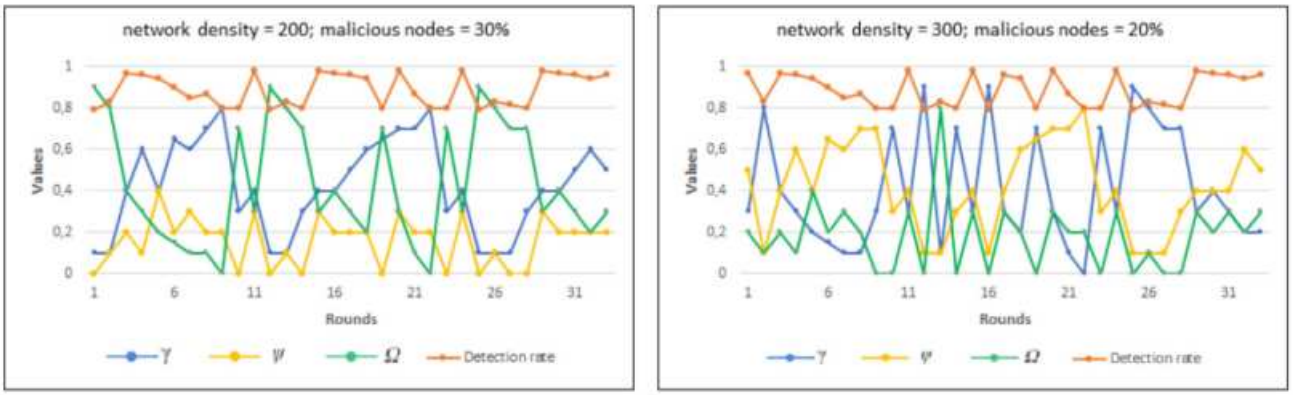


Figure 4.22: Weights evaluation for header selection



(a) Network configuration 1 (b) Network configuration 2

Figure 4.23: Weights evaluation for different network configurations

To recap our contribution, the combination of trust management mechanism, clustering scheme, and Blockchain offers a robust approach to meet several important IoV security requirements. These requirements include authentication, access control, integrity, adaptive security, resiliency and availability. Indeed, the authentication phase ensures that nodes joining the network are legitimate. By requiring nodes to pass this phase, the system can verify the authenticity of participants, enhancing authentication. The use of Blockchain ensures the data integrity. In our approach, the trust-based access control is implied. The nodes with higher trust are more likely to grant access to the cluster, while nodes with lower trust may have restricted access. Data integrity here refers to how much an edge vehicle can ensure security and user trust data accuracy. Trust data in Blockchain are distributed across multiple nodes, which reduces the single point of failure risk. Even if some nodes become unavailable or compromised, the trust data remains accessible from other nodes. By continuously assessing trust levels and factoring them into cluster head selection and node admission, the system adapts to changing conditions, which is essential for adaptive security.

4.4 Conclusion

This chapter proposed a trust management approach for the IoV. With the aid of this proposal, the participating nodes are able to judge the received messages as either true or false. In the first, we implemented a centralized

trust management process that we applied on two-level Blockchain architecture. The trust values were calculated using reputation and message credibility metrics. These values were generated by the nodes and maintained in two levels Blockchains. Then, we applied the clustering for a decentralized trust management process to better consider the QoS metric while enforcing security. The formation of the clusters and the selection of the cluster heads focused on the trust aspect along with the energy and the safety distance factors to create reliable trustworthy clusters. A number of experiments were carried out in order to assess the efficiency of both approaches. The simulation results indicated that the formation of trustworthy clusters using Blockchain reinforces the IoV reliability. Our trust and Blockchain based clustering approach can play a vital role in the improvement of the security of the network, and finally the maintenance of the main QoS requirements in terms of continuity and scalability. In chapter 5, we investigate the trustworthiness along with machine learning techniques and SDN. We will aim to protect the IoV network using collaborative IDS that relies on trust-relate metrics, federated learning, and SDN architecture.

Chapter 5

Trust-based collaborative IDS for the IoV

5.1 Introduction

The previous chapter introduced a trust management solution, wherein trust values were assigned based on predefined trust metrics. The trust management scheme evaluated the trust of nodes through establishing a baseline threshold of trustworthiness, and a Blockchain was utilized to securely store these trust levels. The primary goal was to create a secure and transparent environment where nodes within the IoV could be evaluated and trusted based on interactions. The Blockchain was leveraged to guarantee the immutability and the integrity of trust data. Our work in this chapter targets the trustworthiness' dimension of behavior learning from node-properties. This dimension of trustworthiness relates to learn how nodes behave over time and whether their behavior exhibits anomalies. Anomalies in these behaviors indicates malicious activity. We decide the trustworthiness of nodes based on learned behaviors within the network. Therefore, we rely on an IDS that learns from node behavior through monitoring trust-related metrics to identify anomalies in trust behavior. The IDS operates within an SDN-based IoV architecture. As indicated in chapter 3, the SDN is important to trust management in the IoV ecosystem; an SDN-based IoV architecture provides flexibility and adaptability in managing network. In this architecture, we introduce a collaborative IDS that is based on learning trust features. We organize this chapter as follows. Section 5.2 focuses on IDS' related works. Section 5.3 presents the proposed IDS: collaborative IDS based on trust and SDN. Section 5.4 proposes enhancement for the proposal: collaborative IDS based on trust, SND, and clustering.

5.2 Related works

Various IDS models have been introduced within the vehicular network context. The major of proposed IDSs used the machine learning algorithms. Besides, there has been a notable shift in research towards collaborative IDS [**89**]. Compared to single point IDSs, collaborative IDSs encourage distributed detection. Recently, related works have been started to implement collaborative IDSs based on the federated learning technique [**12**]. The federated learning technique is a decentralized cooperative method that permits data training on multiple entities in order to form a shared training

model while maintaining the local data private. On another hand, trust-based IDSs are gaining attention from researchers. Trust-based IDSs aim to improve the accuracy of malicious node detection by incorporating trust metrics into the detection process. These metrics evaluate the trustworthiness of nodes based on various aspects. However, it is noteworthy that there is limited research on trust-based collaborative IDS within the IoV context. This subsection places a more pronounced emphasis on IDS related works. It highlights the recent works related to trust-based IDSs, federated learning-based IDSs, and IDSs that leveraged emerging technologies-driven architectures. In the first, we present the recent approaches for IDS, then, we refer trust-based IDS related works.

5.2.1 Recent related works on IDS

The major of proposed IDS-based works exploited the advantages of machine learning techniques. Besides, emerging technologies- driven architectures have been deployed to support IDS-based works. Table 5.1 summarizes the main contributions of these works.

5.2.2 Trust-based IDSs

Some of the recent works on the IDS have taken into account the advantage of trust metrics in the detection process. However, very limited works have been proposed under the vehicular networks context. Table 5.2 recaps recent works on trust-based IDS. Summarizing the present subsection, most of proposed IDSs in the vehicular networks context relied on centralized learning (e.g., [[**135**]-[**140**]],[**90**] [**147**]). Centrally training the detection model requires significant data collection, which may raise inaccurate or incomplete detection concerns. Thus, cooperative learning for collaborative IDS is more suitable over VANET and IoV, enabling continuous accuracy evolution and flexibility. The cooperation among nodes in the IoV improves the detection accuracy ([**89**][**141**][**142**][**144**] [**146**][**148**][**149**][**150**]). Indeed, collaborative IDS gathers data from multiple nodes, which provides a more comprehensive view of network behavior by communicating

Table 5.1: Summary of recent related works on IDS

Ref	Main contribution	Implementation tools and Datasets
[**134**] 2018	collaborative IDS for privacy-preserving in VANET: -ADMM, empirical risk minimization algorithm, dual variable perturbation.	Dataset: NSL-KDD
[**135**] 2021	machine learning-based IDS for VANET: -Random Forest, coresets-based clustering	Dataset: CICIDS2017
[**136**] 2021	multi-tiered IDS for intra and inter IoV: -decision tree technique, Random Forest technique, extra trees technique, extreme gradient boosting technique -stacking ensemble model, BO-TPE, k-means, BO-GP	Dataset: CAN-intrusion CICIDS2017
[**137**] 2021	an oversampling strategy to reach IDS on imbalanced datasets for intra and inter IoV: -outlier detection -heuristic genetic algorithm	Network simulators: UNSW-NB15 , CICID, S-2017, ROAD, Car-Hacking, CAN-intrusion
[**138**] 2015	feedforward neural network-based IDS for self-driving cars: -proportional overlapping scores method -fuzzification method	Network simulators: NS2, SUMO, MOVE, NAM
[**139**] 2021	deep neural network-based IDS for VANET: -computation of reconstruction error to classify anomalies	Nvidia jetson nano emulator
[**140**] 2018	IDS based on hierarchical growing neural network for VANET: -semicooperative extraction of location information -two-step confirmation scheme to identify abnormal messages	Network simulators: NS2, SUMO
[**141**] 2021	federated learning-based collaborative IDS: -CNN algorithm -training global model across central server	Dataset: IDS2017
[**142**] 2020	federated learning-based collaborative IDS: -long short-term memory algorithm -training global model across central server	Dataset: SEA
[**143**] 2020	Fog-based approach to offload IDS tasks for VANET: -multiobjective optimization method based on genetic algorithm	Network simulator: PowerTutor
[**144**] 2021	Blockchain and federated learning-based collaborative IDS for ITS: -encoder-decoder modules -RSUs as miners nodes -Delegated Byzantine Fault Tolerant consensus	Datasets: TON-IoT, Car-hacking

Ref	Main contribution	Implementation tools and Datasets
[**145**] 2019	Blockchain-based geographical IDS for the IoV: -micro-Blockchains to construct local on-demand IDS -control plane over micro-Blockchain to configure IDS -back propagation neural network	Dataset: Keras library
[**146**] 2020	SDN-based collaborative IDS for the IoV: -multi-discriminator generative adversarial network -SDN controllers as local learners -multi-layer perceptron	Datasets: KDD99, NSL-KDD

Table 5.2: Summary of trust-based IDSs

Ref	Main contribution	Implementation tools and Datasets
[**89**] 2020	collaborative IDS for VANET: -ensemble learning to train local classifiers -performance of local classifiers as trust factor (entity-based trust process)	Dataset: NSL-KDD, Netowrk simulator: SUMO
[**99**] 2018	multi layered IDS thatis based on game theory for VANET: -clustering scheme, reputation scores to assess the trust of clusters heads -the selection of cluster head used Vickrey Clarke Groves	Network simulators: NS3, SUMO
[**90**] 2018	SVM-based IDS for VANET: -response module holding a shared trust score table (reputation-based (entity-based trust process) -alarm from response module if trust score reached zero	Network simulators: NS2, MOVE, SUMO
[**147**] 2021	trust management-based IDS for VANET: -knowledge-based (current and past behaviour) (entity-based trust process) -DST, Bayesian learner	Own simulator
[**148**] 2020	markov-based reputation collaborative IDS for VANET: -Non-dominant Sorting Genetic algorithm -Non-Linear Programming optimization scheme -Hidden Generalized Mixture Transition Distribution -multilayer perceptron	Network simulators: NS2, SUMO, March, Aegean WiFi Intrusion
[**149**] 2021	trusted Blockchain and federated learning-based collaborative IDS for vehicular networks: -multi-layer perceptron -a mask noise model upload scheme -trust-based incentive mechanism	Dataset: KDDCup99
[**150**] 2021	challenge-based IDS in SDN-driven architecture: -reputation-based (entity-based trust process) -the trust management component refers to send a challenge periodically investigating the trust-worthiness of other nodes -SDN controllers contributed to collect data and provide network status	Open vSwitch, POX controller, Snort

experiences between nodes. The diversity of data assists in identification of malicious activities which may be invisible to a particular node, and thereby can lead to enhanced detection accuracy. Besides, collaborative IDSs capture the distributed and dynamic nature of the IoV constructing cooperation over the network. They can adapt to changing conditions in the IoV by incorporating data from different parts of the network. As the IoV network scales, adding more IDS nodes to the collaboration is relatively straightforward. This can treat inaccurate detection owing to extensive changes in nodes locations. Furthermore, in collaborative IDS, even if a subset of nodes is compromised, the collective knowledge from the entire network can still help in identifying malicious activities. Nevertheless, machine learning-based collaborative IDSs can have implications for privacy and communication overhead due to the exchange of data and model updates. However, the extent of the overhead can be handled through strategies like optimal number of IDS nodes, data preprocessing, and optimized communication protocols. The fundamental barrier of privacy preservation can be treated through federated learning-based IDSs ([**141**][**142**][**144**]). Overall, federated leaning-based IDSs are more efficient and privacy-preserving approach in the IoV, where data volume and privacy are critical considerations. Regarding IDSs that were applied on emerging technologist-based architecture, SDN-based IDSs and Fog-based IDSs assisted to alleviate some QoS concerns ([**146**][**143**]). The SDN encouraged flexible and granular management of detection process in the IoV while offering large-scale architecture optimization. Besides, arranging SDN controllers into the IDS enforced security policies in the IoV [**146**]. In view of the significance of Blockchain-based IDS, the development of a computationally efficient consensus that maintains the real-time requirements of the IoV can be a bright direction to profit the full potential of the Blockchain for IDSs. On the other hand, we concluded that the integration of trust metrics in the detection process of malicious behaviors helps to enhance the detection performance. Trust-based IDS adapted to changing network conditions by updating trust metrics based on real-time behavior ([**99**] [[**90**]-[**150**]]). These mentioned points inform strategies for further enhancement in works on IDS for the IoV. The simultaneous use of trust-based IDS, federated learning and SDN-based IDS within an IoV framework is not very well explored. In this chapter, we take advantage of the collaborative nature of the IoV and explore federated learning in software-defined IoV to design a collaborative trust-based IDS. The

IDS utilizes the collaboration among local SDN controllers to jointly implement the trust- based detection model.

- By focusing on specific zones, the SDN can allow for efficient resource allocation, ensuring that IDS nodes have the necessary resources (e.g., energy, bandwidth) to participate in collaborative detection. This resource management optimizes the performance of the IDS.

- The SDN will allow for efficient network segmentation, malicious activity in one segment can be contained under an SDN structure, preventing it from affecting the entire network. Also, the SDN controllers can focus the detection on specific regions or nodes types.

- The local decision-making based on trust metrics calculated at the SDN controllers aligns well with the distributed nature of the IoV.

- By processing data and making decisions locally, SDN-based, the IDS can reduce detection latency which is crucial for real-time threat detection.

- Federated learning will allow SDN controller to leverage global knowledge from other controllers while retaining the ability to adapt to local conditions. In SDN, security policies can be adjusted in real-time, and collaborative IDS can benefit from this dynamic adaptation to respond to changing network conditions.

- SDN and federated learning will allow for easy scalability of the collaborative IDS.

- The distributed nature of federated learning and SDN architecture will enhance fault tolerance. If one SDN controller fails, others can continue to operate the IDS.

- Federated learning aligns with data privacy regulations.

- Trust-based IDS analyzes the dynamic behavior of nodes, taking into account their interactions, communication patterns, and adherence to network protocols. This behavioral analysis helps identify anomalies or deviations from expected behavior. Also trust-based IDS provides contextual insights into network behavior. It can dynamically update trust metrics based on real-time behavior. This ensures that trust evaluation adapts to changing conditions.

5.3 Collaborative IDS based on trust and SDN

5.3.1 Proposed IDS overview

Our trust and SDN based collaborative IDS constructs the detection model based on local detection by SDN controllers. Each SDN controller assesses the trustworthiness of assigned network nodes. The trustworthiness relates to how nodes behave over time and whether their behavior exhibits anomalies that could indicate malicious activity.

- We present a two-tier IDS based on the federated learning technique in order to reinforce the recognition of malicious activities that may be invisible for a single particular node.

- The first tier (local detection) is responsible for analyzing network behavior locally. The second tier (global detection) aggregates information from the first tier and makes decisions based on a broader view of the communication system.

- SDN controllers have greater visibility of network status. They carry out local detection of dishonest nodes (first tier), and distribute security policies to underlying IoV nodes. The Cloud server, on the other hand, serves as the entity responsible for aggregating local detection models and facilitating collaborative learning (second tier).

- The IDS decides about the trust of nodes. It relies on node properties-based metrics to identify behaviors and determine whether observed behaviors align with trustworthy behaviors. The SDN provides global

view of network behavior and nodes activities, and the detection process leverages this visibility to detect anomalies.

Details about the proposed IDS that combines trust, federated learning, and SDN are provided in the upcoming sub-sections.

5.3.2 Proposed architecture overview

The proposed network architecture for the IDS is illustrated in figure 5.1. The network model includes the IoV nodes, the federated learning model, and the SDN controllers constituting the control plane. The Cloud server corresponds to the upper layer, while the SDN controllers and the IoV nodes are placed in the bottom layer. The Cloud server acts as part of the control plane, enabling collaborative IDS through federated learning.

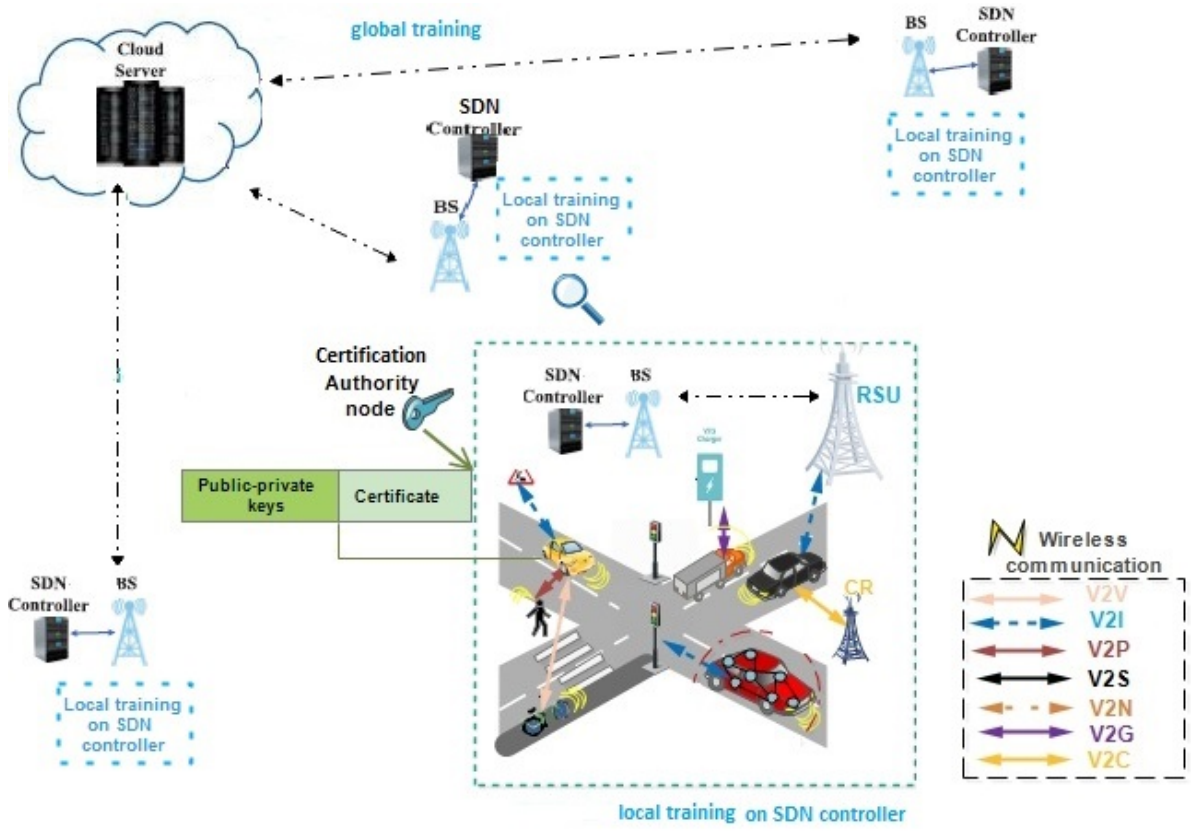


Figure 5.1: Proposed framework for IDS

- Data plane

Smart vehicles, RSUs, trusted authority nodes, and switches are associated with the data plane as they generate, exchange, and process data. They are monitored by SDN controllers for unusual behavior.

- Control plane

SDN controllers refer to the core of control plane in the SDN network. They are deployed at base stations, each responsible for managing a segment of the network. The SDN controllers are fully trusted.

- Ordinary vehicles

The ordinary vehicles are sensor-equipped entities capable of sensing their environment, making real-time decisions, and communicating with other network nodes Their capabilities includes safety, connectivity, and enhanced driving experiences.

- Authority nodes

The authority nodes serve mainly to register the nodes of the network through certificates. IoV entities and SDN controllers use these certificates to facilitate secure communication channels and ensure that authorized nodes are part of the network. Also, the trusted authority nodes can provide valuable information about the legitimacy of network nodes, which can be used as an input for intrusion detection running on SDN controllers.

- RSUs

The RSUs are controllable by SDN controllers. An RSU can receive data from vehicles within their range and relay it to the appropriate SDN controller. Also, the RSUs can aggregate data from multiple vehicles, providing additional context and environmental data for a consolidated stream of information to the SDN controllers. They can control traffic signals and signs in coordination with the SDN controllers to optimize road traffic.

- The SDN controllers

SDN controllers are the central brain in the SDN-based IoV architecture. The controllers are in charge of controlling the overall behavior of the network. The SDN controllers determine the network rules and apply them to the data plane. They provide control and management of network resources, including RSUs and vehicles. We propose that the SDN controllers perform local intrusion detection and monitoring functions. They analyze real-time data from RSUs and vehicles within their coverage area to identify anomalies in network behavior. They transmit local detection models to the Cloud. Besides, the SDN controllers manage data flow between the local network nodes (vehicles, RSUs, trusted authority nodes) and the Cloud server. They determine how data is routed to prioritize traffic and ensure that data from the different nodes reaches the Cloud server efficiently.

- Cloud server

The Cloud serves for global anomaly detection and model aggregation from SDN controllers. The Cloud server can overseen the collaborative learning process to generate comprehensive security reports and collaborate with the SDN controllers for coordinated responses to security threats.

5.3.3 Threat model

Anomalies in trust behavior manifest as suspicious or unexpected activities that may indicate a breach of trust. These anomalies can take various forms. The main anomalies that can be handled by the proposed IDS are as follows:

- Inconsistent data forwarding

Nodes that inconsistently forward data, or drop packets are exhibiting trust anomalies. For instance, a spike in packet drop rate can indicate malicious behavior (e.g., blackhole attack). Likewise, nodes that drop packets related to security incidents to fail reporting incidents may be considered untrustworthy since cooperating in security protocols are essential trust-building behaviors. Metric like packet behavior deviation can be highly effective in detecting inconsistent data forwarding.

- Driving pattern and platooning anomalies

When vehicles are traveling in close proximity, they tend to maintain relatively consistent speeds to avoid collisions. In platooning scenarios where vehicles travel closely, if a vehicle behaves erratically within a platoon (e.g., abruptly changing speed), it may signal a security threat. The driving pattern metric can help to monitor platooning operation.

- Frequency of interactions

Trust anomalies can include nodes that either excessively or minimally interact with others. For example, close vehicles with similar speed are expected to interact more frequently. This is because vehicles in close proximity tend to communicate more for various reasons, such as collision avoidance, cooperative driving, or sharing traffic information. Excessive interaction frequency between vehicles that are very close may indicate unnecessary flooding of messages, which can be a sign of malicious activity. Likewise, an abrupt reduction in interactions between closely spaced vehicles may also be indicative of an anomaly. For instance, if a vehicle suddenly stops communicating with nearby vehicles, it can suggest a trustworthiness issue, especially within scenario of cooperative driving or information sharing. Vehicle proximity and similar speeds related metrics can contribute to detect such anomalies.

- Unusual communication patterns

Trust anomalies may involve nodes engaging in atypical communication with unknown nodes or accessing restricted areas of the network. For instance, a vehicle communicating with numerous unknown nodes may be behaving anomalously. Metric related to stranger node can be valuable in detecting unusual communication patterns.

- Unusual access points

Trust anomalies may involve nodes connecting to unauthorized access points (e.g., cellular base stations, Wi-Fi hotspot in urban areas or specific zones). This behavior raises suspicion. Monitoring nodes' connections to access

points is essential for detecting unauthorized access. If a node starts connecting to access points that it has never connected to before, or it attempts to connect to unauthorized network access points, this behavior can be detected through stranger node-related metric.

5.3.4 Proposed IDS Process

The IDS model is divided into two key modules. The first module is the derivative module. This module's primary function is to compute valuable attributes from the collected data. These derived attributes are prepared in the preprocessing step, and will be used as inputs in the classification module which represents the decision-making part of the IDS. Under the federated learning paradigm, this module employs machine learning algorithms to assess the input features and classify nodes as either malicious or non-malicious based on learned patterns. The IDS model starts by extracting valuable information from the data in the derivative module. Then, this processed data is passed on to the classification module, which makes the final decision on whether a node should be considered malicious or not. The IDS model consists as follows. The IDS makes local decisions at the level of SDN controllers. Each distributed SDN controller is responsible for managing a specific network zone encompassing IoV nodes such as vehicles, RSUs, and trusted authority nodes. These SDN controllers collect and analyze data within their managed zone. The SDN controllers detect the trustworthiness of local nodes. The local models of trustworthiness detection, created and maintained by the SDN controllers, are then aggregated into a global detection model . This model is a comprehensive overview of network trustworthiness. The Cloud server acts as the aggregator for these local detection models using federated learning. Federated learning allows SDN controllers to share local detection models with the Cloud without directly sharing sensitive data. When a threat is confirmed, SDN controllers can take local actions to mitigate the threat and initiate adaptive security responses. These responses can be tailored to the specific context of the threat (e.g., isolate malicious nodes from the network, increased monitoring, reroute network traffic, or alerting nearby vehicles to take precautionary measures).

In fact, the IDS builds upon trust metrics. It takes into account metrics related to node-properties to decide whether node can pose a security threat

within the IoV network. Moreover, the IDS can benefit from environment factors-based trust metrics for threat contextualization and tailored security responses. Environment factors-based metrics target analyzing the circumstances surrounding an event or behavior. This includes factors like traffic density, time of day, and nodes types. For example, if a sudden traffic slowdown is detected during rush hour in a critical traffic zone, it may indicate a security breach or accident. The global view of environment factors allows for situational and context-aware threat detection. This contextualization helps prioritizing security responses, e.g., the SDN controllers can implement stricter security measures during known high-risk hours or in specific geographical areas. Furthermore, the SDN controllers can maintain historical threat data to recognize patterns over time for proactive security. This historical context helps in assessing whether a detected anomaly is a one-time event or part of a recurring threat pattern. Besides, the global detection model can rank detected anomalies based on their severity for immediate response and resource allocation. Indeed, high-severity anomalies may require more resources (e.g., bandwidth and computational power) for analysis and response. By ranking anomalies, the IDS can allocate resources according to the perceived threat level. The following subsection details the used trust metrics into the detection process.

5.3.5 Trust Features as Inputs for IDS Training

Anomalies in trust behavior signify deviations in how nodes establish and maintain trust within the IoV network. In the collaborative IDS implementation, the SDN controllers as local IDS nodes leverage trust-related metrics to detect anomalies in the IoV network. The SDN controllers utilize the derivative module to extract features that reflect trust of nodes. Different metrics can be used to estimate the trust (subsection 3.2.2 of Chapter 3). These metrics involve node properties-based metrics that assess trust based on the characteristics of individual nodes within the network. We consider trust metrics related to node properties for our IDS as follows.

- Stranger node

A node is considered as stranger if it did not registered at the authority node, or once it has no history of interactions with the authority nodes.

The stranger node metric evaluates the node trust based on its registration status within the network. A non-registered node is considered as untrustworthy. The SDN controllers have granular topology overview to respond to stranger nodes. The SDN controllers can monitor the registration of nodes within their managed area, and use this metric for early suspicion of non-registered nodes. The SDN controllers can compare the source nodes of incoming packets with a list of registered nodes at the authority nodes. Also, the SDN controllers can monitor the traffic pattern of nodes attempting to access restricted parts of the network to identify stranger nodes. Besides, trust can be influenced by interaction with stranger nodes. For example, nodes that interact with stranger nodes are suspicious. The SDN controllers can analyze the traffic patterns of nodes within their managed zones to observe unusual communication patterns, such as a registered node engaging in frequent interaction with an unregistered node. The SDN controllers can subject these nodes to further scrutiny and monitoring. While stranger node metric may not contribute in recognizing dishonest nodes in a complete certainty as it can show normal behavior, it raises a high level of suspicion to initiate timely investigation. This metric can enhance the agility of threat mitigation. The SDN controllers can trigger real-time response when stranger nodes are detected, such as isolation or traffic restriction. Moreover, SDN controllers can utilize this metric to make context-aware decision and prioritize responses. For example, if a non-registered stranger node suddenly appears in a critical part of the network, it will be flagged as a potential security threat. This severity may be used to prioritize security response. From another hand, monitoring strange nodes metrics is beneficial to ensuring proper resource allocation in SDN. The SDN can allocate different levels of access and resources based on authentication status. Authenticated nodes receive priority access and resources, while unauthenticated nodes are subject to limitations.

- Driving Pattern Deviation

Driving pattern metric evaluates the trustworthiness of nodes based on their behavior related to speed propriety. This metric focuses on variations in a vehicle's speed to identify unusual driving patterns that may indicate an anomaly. Anomaly can indicate reckless driving. This metric involves monitoring the speed of a vehicle as it moves along a road segment by calculating the deviation of a vehicle's current speed from the expected or average speed

for that specific road segment. This calculation can be based on historical speed data for the same road segment. Beside, incorporating vehicle proximity and similar speeds can enhance accuracy in monitoring driving behavior. Close vehicles that exhibit predictable and consistent speed patterns may be considered more trustworthy, while those with irregular or suspicious speed patterns could be viewed with lower trust. When vehicles are in close proximity, it is expected that their speeds will be relatively similar to maintain safe distances and avoid abrupt speed changes. If vehicle's speed deviates significantly from the speeds of nearby vehicles in a similar lane or vicinity, it can raise suspicion. For example, if a vehicle has an erratic speed behavior in one area but not in others, it can be a sign of localized malicious activity. The SDN controllers can collect data from various parts of the IoV to assess driving patterns comprehensively. They can use this metric to analyze the behavior of vehicles and identify deviations from expected speed pattern.

- Packet behavior deviation

This metric assesses individual node behavior in terms of how it handles packets. The rate at which a node successfully delivers packets can be an indicator of its reliability. Node that consistently sends and receives a similar number of packets is expected to behave predictably. When a node deviates from this expected behavior by either excessively dropping packets or modifying packets attributes, it is considered an anomaly. The SDN controllers have granular visibility into node behavior within their managed area. By taking advantage of real-time flow monitoring features, the SDN controllers can keep track of flow, analyze flow statistics, and determinate number of dropped packets. Also, the SDN controllers can further manage flow rules to log information about dropped packets. Packet dropping rate D_i for the node i can be expressed as:

$$D_i = R_i \, S_i \quad (5.1)$$

where R_i denotes the dropped packets at the node i, and S_i is the number of successfully sent packet. Each SDN controller keep track of packet dropping rate for local nodes to decide their trust status.

5.3.6 IDS training

Both the SDN controllers and the Cloud server host lightweight learning algorithms fesigned for training the IDS. Each SDN controller deploys local trust learning models in its segment of the network. These local models are trained to detect anomalies within their respective zones. The local models on the SDN controllers are implemented to be aggregated in a global model. This creates a global detection model that benefits from insights gathered from all the local zones. Each local SDN controller contributes to the global detection model without sharing sensitive data directly. The Cloud server act as a central point for model aggregation and hosts the global detection model. The shared detection model is assumed to become more efficient as the time goes by and adapt to the dynamic nature of network. Following the federated learning technique, the SDN controllers communicate with the Cloud during the model training over a predefined number of rounds. In each training round, the process unfolds as follows.

- Local trust learning models

Each distributed SDN controller independently creates and trains its local model. This training is done based on the data and behavior patterns observed in its respective zone. Once the local training is complete, each SDN controller sends to the Cloud its local model.

- Global model aggregation

The Cloud server aggregates the local models to build an improved global model. This global model benefits from the collective knowledge of SDN controllers. During this aggregation step, the Cloud combines these local models while considering weights related to performance of SDN controller (e.g, latency, throughput). This step aims to ensure that contributions from different zones are appropriately weighted to create an effective global model.

- Weighted aggregation

The weights determine the influence of each local model on the formed global model. The weighted aggregation can ensure that the global model

reflects relevant insights from the SDN controllers. We adjust weights based on performance-related factors at each SDN controller: (i) SDN controller with more available resources (e.g., CPU, memory) can receive higher weight, (ii) SDN Controller that provides consistent and stable performance over time (iii) SDN controller that performs local detection at a high rate or with low latency may be associated to higher weight, (iv) SDN controller with lower error rates in local detection and more accurate local model may be favored, (v) SDN controller managing critical segment may be considered as more trusted and receive higher weight. Furthermore, weights can be adjusted based on other factors like data heterogeneity. Indeed, the SDN controllers may have different amounts of data or data with varying levels of relevance. Weights allow the aggregation server to consider this heterogeneity. Local models from SDN controllers with abundant data can be given higher weights. After aggregating the local detection models, the Cloud server redistributes the global detection model (GM_r) with new parameters (GM_{r+1}).

$$GM_{r+1} = \frac{1}{C \sum_{i=1}^{C} W_r^i LM_r^i} \tag{5.2}$$

where W_r^i is the weight of the local detection model LM_r^i of the SDN controller (i) at round r, and C is the number of the SDN controllers.

- Enhancing the global model

The Cloud server enhances the global model through learning new patterns. The new global model is redistributed to the SDN controllers. Upon receiving a copy of the global detection model, the SDN controllers load their locally updated datasets to train their models. After the training round, the SDN controllers transmit back the resulted local models to the Cloud.

- Iterative process

This iterative process of local training, aggregation, model enhancement, and model distribution continues over several rounds, enabling the global model to evolve and adapt dynamically to the network.

5.3.7 Simulation study and results

5.3.7.1 Simulation environment and metrics

We conducted the simulation of our IDS to evaluate its performances. We implemented the SDN-IoV architecture using Python. For the deployment of SDN network topology, we used the Mininet WiFi; an OpenFlow-enabled network emulator. In fact, OpenFlow is one of the most widely used protocol standard for building SDN applications [**151**]. The simulator SUMO was applied to generate mobility of vehicles. We considered a network topology consisting of two SDN controllers. Each SDN controller communicated with 4 RSUs, 4 authority nodes, and 150 vehicles. A random node as host is defined to represent the cloud server side. We defined an ordinary node to represent the cloud server side that aggregates local detection models to simulate the federated learning-based detection. Simulation was conducted for a highway road scenario. We assumed that the local area zone of the SDN controller included 4 RSUs, 4 authority entities, and about 142 vehicles. Maximum of 40% vehicles out of the vehicles were defined to be malicious. The vehicles varied their speed between 90km/h and 150km/h. Table 5.3 summarizes the used parameters in the performance evaluation.

Table 5.3: Simulation parameters

Parameters	Values
Simulation time	1500s
Maximum nodes density	150
RSUs	4
Authority node	4
Average of vehicle speed	90km/h to 150km/h
Maximum percentage of malicious vehicles	40%
SDN controllers	2
packets dropping threshold	15
Hyperparameters	batch size of 32
	10 rounds
	7 epochs
	0.5 learning rate

Trust-related features were integrated to build the detection process. We considered the features of driving pattern (speed deviation), packet dropping rate of nodes, and strange node interaction. Scenarios of nodes that intentionally drop packets, and nodes that are not registered were created in the Mininet WiFi. Also, the suspicious driving was simulated with a

large difference of speeds between close vehicles under a same traffic situation. The SDN controller collected data from network nodes to extract the considered features for detecting untrustworthy behaviors. They gathered data by capturing network packets and queried flow statistics from the SDN switches to derive packets dropping number and stranger nodes. The vehicle' driving pattern was captured from the data generated by SUMO. Regarding the learning algorithm, we trained the Random Forest algorithm, the 1-dimensional CNN algorithm and the 1-dimensional RNN algorithm for comparison purpose. Such algorithms produced good results as in [**136**][**141**]. Therefore, we selected these different algorithms to understand their performance under the federated learning structure. Besides, we use a lightweight CNN and RNN learning models to alleviate the complex learning operations. We compared the detection performance in terms of recall, precision, and F1 score.

5.3.7.2 Simulation results and evaluation

Recall and precision measures were calculated using Eqs.(4.6)(4.9). The F1-score combines rates of precision and recall. It can provide a concise evaluation of detection performance. The proposed IDS led us to a recall value of 99.04%, a precision value of ±99.30%, and a F1-score of ±99.17% after 10 rounds, as shown in figures 5.2, 5.3, and 5.4, respectively. Such values confirm the relevance of training features under the federated learning paradigm. Compared to the SDN-based collaborative IDS proposed in [**146**], our proposal showed more efficiency according to detection performance. Our proposed IDS learned properly with the Random Forest, the CNN, and the RNN even with few rounds. Recall value, precision value and F1-score remain respectively over 95.20%, 95.61%, and 95.40% with 4 rounds. We point that the CNN algorithm outperformed the Random Forest algorithm and the RNN algorithm in term of precision rate (see figure 5.2). The Random Forest algorithm dominated slightly the CNN algorithm during first training rounds in term of recall rate (see figure 5.3). The algorithms of CNN and Random Forest exhibited very close detection performance, outperforming slightly the RNN algorithm.

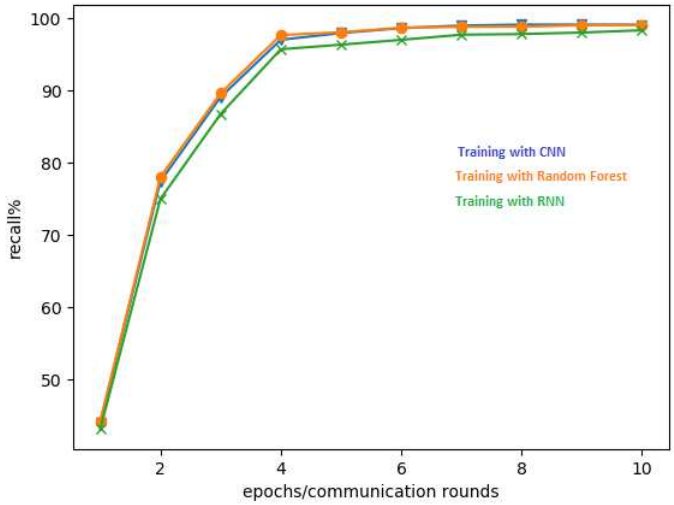


Figure 5.2: Recall rate

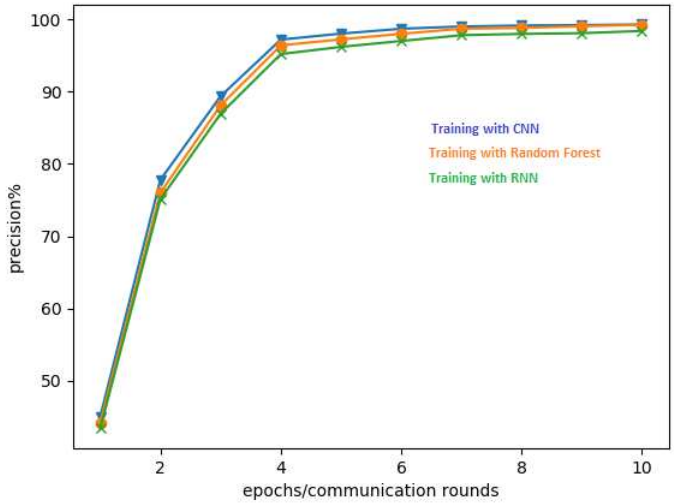


Figure 5.3: Precision rate

5.3.8 Summary

We adapted SDN for flexible collaborative detection of dishonest activity in the IoV. The SDN provides global view of network topology, and collaborative IDS leverage this visibility to detect anomalies. We considered an SDN-IoV architecture that allows for distributed detection within each SDN controller's zone while maintaining global data privacy through federated learning. The SDN controllers contribute in detection model training along

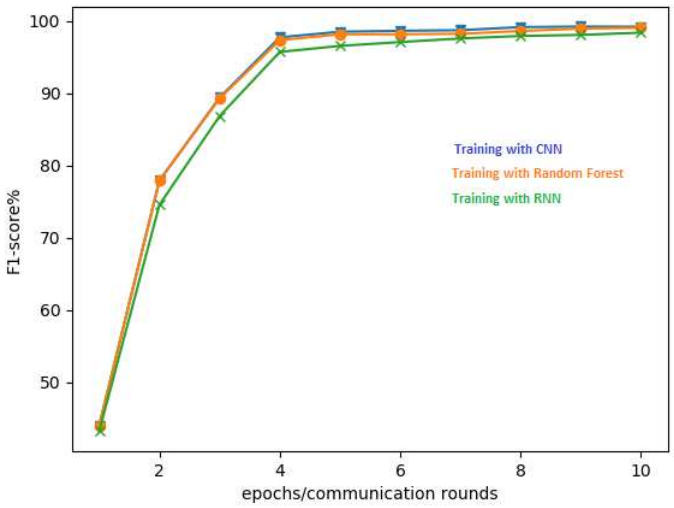


Figure 5.4: F1-score

with the Cloud server. The Cloud server was used to handle large-scale federated learning tasks, and facilitate collaborative learning. The detection process relied on trust metrics to make decisions about the trustworthiness of nodes. The performance of the IDS was evaluated using metrics of recall, precision, and F1-score. In the following section, we aim to enhance the IDS by highlighting the QoS.

5.4 Collaborative IDS based on trust, clustering, and SDN

5.4.1 Proposed IDS overview

This section suggests to improve the above presented IDS through allowing more local trust-decision making detection. As stated we combined the SDN, the federated learning technique and the trust-related metrics to exploit the benefits of these tools and perform the detection process. We presented a centralized data handling, where the Cloud provided a central point for analyzing local detection models from multiple SDN controllers and handling federated learning task. To enhance the IDS, we are interested on near-real-time detection and more local control. We introduce a clustering mechanism to construct the IDS framework. The local zones of the SDN controllers will be sub-divided into clusters, wherein clusters heads operate with the local SDN controller to detect untrustworthy nodes. This can provide more

granular detection and enhance time responsiveness of the IDS. Besides, we adopt the multi cluster head concept and apply the trust factor criteria for cluster head selection. The framework for the proposed IDS in section 5.3 is extended according to the following points:

- We present an IoV network topology that is based on clustering the nodes within the local zone of the SDN controller. The nodes in each local area zone are clustered for more highly manageable network.

- We introduce a more local detection process that is applied on the nodes clustering. We reconsider trust-related features for the detection process.

- We assign the task of local intrusion detection to reliable clusters heads, while the global detection model will be placed on the SDN controller.

- We apply the concept of multi-cluster head in a particular cluster for more reliable detection. We optimize the cluster heads selection by using the dolphin swarm concept [**152**]. We use the dolphin swarm concept because it can provide good features for the selection of cluster heads like division of labour and cooperation, echolocation, and information exchanges. The defined cluster head will act like a dolphins and scan their nearby nodes to detect untrustworthy behaviors.

- We reconsider the layered collaborative IDS that uses federated learning. The

- local detection is conducted by trusted cluster head. The global detection is hosted on the SDN controller.

5.4.2 Proposed architecture overview

We outline in figure 5.5 the proposed framework architecture for the IDS. The framework consists of three main modules: SDN controllers layer, cluster layer, and end IoV nodes. The cluster layer and the IoV nodes are associated to the data plane. The IoV nodes layer is made up of smart ordinary

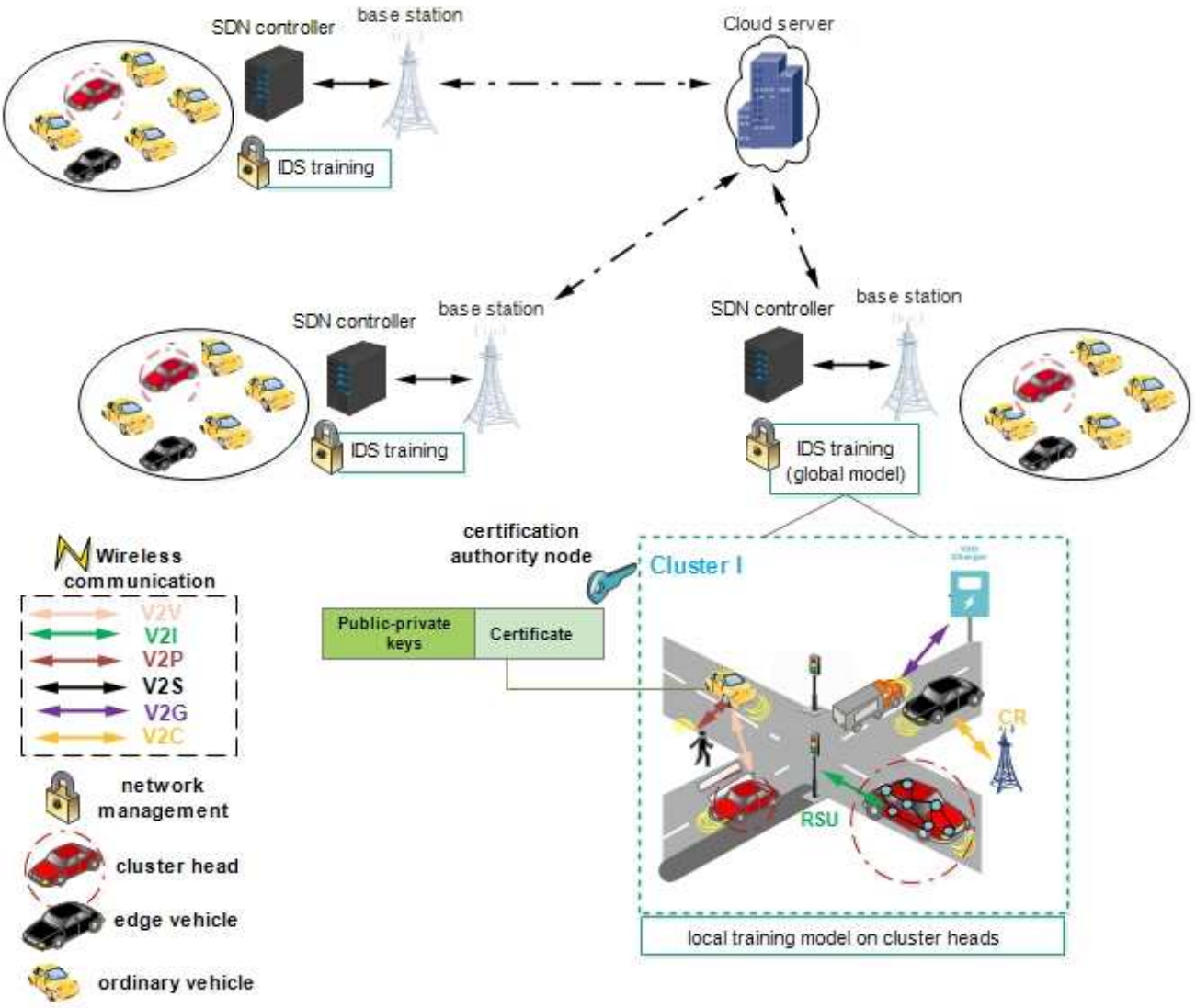


Figure 5.5: Proposed framework for IDS

vehicles, RSUs, authority nodes, and other IoV devices. The cluster layer consists of edge vehicles with enough resources compared to ordinary vehicles. The control plane is defined by the SDN controllers. The Cloud server in the architecture serves for global perspective of the entire IoV network. We reconsider the defined role of each component in subsection 5.3.2: (i) the ordinary vehicle enables environment sensing, communication, and basic local data processing, (ii) authority nodes role is primarily related to the issuance of certificates and the authentication of network nodes, (iii) the RSUs provide environmental data optimize road traffic, (iv) edge vehicles can be requested from ordinary vehicles to handle some processing and comput-

ing tasks for the IoV. Edge vehicles are mainly responsible for assuming the role of cluster head. The selection process identifies edge vehicles with the necessary capabilities to perform localized anomaly detection. Edge vehicle as cluster head facilitates communication between the nodes within respective cluster and the SDN controller, (v) SDN controllers define the network policies and inform OpenFlow switches about forwarding rules. Each SDN controller manages the respective part of the IoV network.

- Layered collaborative IDS

To reinforce the local network management and facilitate quicker recognition of dishonest nodes, we cluster the nodes of each local area zone of SDN controller to form sub-local zones. We assign cluster heads to perform local detection, while the SDN controller host the global detection model. These cluster heads can be akin to sub- controllers within the larger SDN framework. The details of the clustering process and the detection process are provided in the following subsection. From the QoS point of view, clustering nodes within the different local zones managed by SDN controllers creates smaller more manageable network segments. The SDN controllers can oversee clusters, which reduce the complexity of network management. Besides, the distribution of detection task among cluster heads reduces the detection load on individual SDN controller. This localized management simplifies control plane operations from the SDN controller. This allows the network can make better use of available resources and acknowledge the QoS. Each cluster head focuses on its assigned cluster, making detection more efficient and responsive to local threats. Furthermore, the clustering ensures improved scalability. The network can easily accommodate an increasing number of nodes without compromising performance. Moreover, the introduction of multi-cluster heads further balances the detection workload. In densely clusters, multiple cluster heads collaborate to detect dishonest nodes, ensuring that no single point becomes a bottleneck. From a security point of view, we build the local detection on trusted clusters heads. Cluster-based monitoring allows for more local detection, thereby improved region-specific awareness. Each cluster operates under the supervision of its cluster heads. Each cluster has its cluster heads responsible for local detection. This approach enhances the ability to detect untrustworthy behavior that may be specific to a particular segment of cluster.

5.4.3 Proposed Clustering Scheme

We associate a clustering scheme to our proposed IDS to enforce the network security and to more consider the QoS. We organize the nodes into clusters to help SDN controller in managing the IDS. The clustering architecture comprises registration stage, cluster formation and cluster head(s) selection.

5.4.3.1 Cluster formation and node joining process

The SDN controller is deployed to manage and govern the network elements of a particular zone. It is considered that the SDN controllers are fully trusted. Let us assume that each geographic area, controlled by an SDN controller is composed of u clusters ($C_1, C_2, ..., C_u$). Each cluster C_u involves a set of IoV nodes; ordinary vehicles, edge vehicles, RSUs grouped based on the distance and the velocity similarity, and it can have one or more cluster heads. All cluster heads communicate with the respective SDN controller. The authority node initiates a periodic request to form a cluster and select its head. The cluster heads are elected from edge vehicles based on trust and resource metrics. The authority node act as super cluster head in cluster C_u, and can provide the selected cluster head(s) with information that concern the nodes belonging to respective cluster C_u. The IoV nodes should be authenticated to join clusters.

- Authentication

When a vehicle want to join a geographic area, it should register the authority node. The authority node generates certificate at the time the node is registered (we follow the authentication process adopted in section 4.2 from chapter 4). A non-authenticated node must be confirmed by the trusted authority before being allowed to be a cluster member.

- Cluster joining

A free node can join the nearest cluster as it authenticated with the authority node. The free node broadcasts request for cluster joining. The cluster joining request involves node certificate and other parameters related to node speed and position. The cluster head verifies the cluster joining request by

using public key contained in the free node' certificate. The node can join the cluster if it satisfies the cluster properties. The node behavior will be monitored by the cluster head until leaving the cluster. To maintain the topology of SDN local zone, the authority node continuously communicates with the cluster heads. The authority node considers that the cluster head left its zone, when it do not hear from it periodically. In addition, the authority node periodically verifies whether the cluster heads are maintaining the required criteria. The supervision of cluster head is mandatory since its trust may get changed. The periodic check compels clusters heads to behave well in order to keep their role. The authority node can communicate with the respective SDN controller and shares its available evaluation of cluster heads trustworthiness status. The role of cluster manager is retrieved from cluster head with updated low trustworthiness value (i.e., below the threshold). Figure 5.6 illustrates the general phases of the proposed IDS.

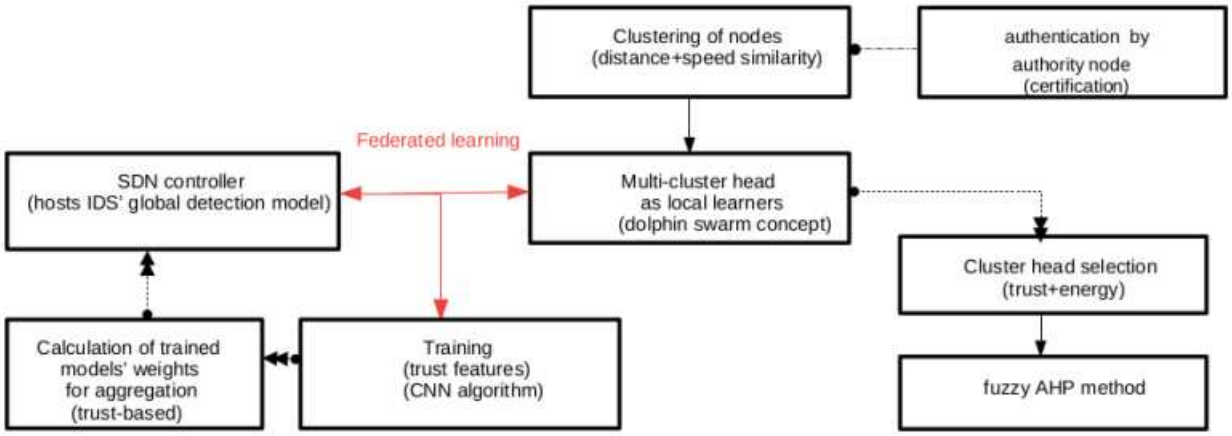


Figure 5.6: General phases of the proposed IDS

5.4.3.2 Multi cluster head selection

The selection process identifies edge vehicles with the necessary capabilities to perform as clusters heads. The authority node initiates the phase of cluster head selection. The authority node sends to the edge vehicles a request containing properties of the cluster that will be created; i.e. cluster size and maximum number of cluster heads. The authority node as pre cluster head undertakes the role of verifying the trustworthiness of edge vehicles. In this phase, the edge vehicles are evaluated to become cluster heads in cluster C_u

based on their trust and resource indicator.

- Selection metrics

By considering both trust and resource indicators, we strike a balance between security and network performance for our IDS framework. The trust value boosts security within the cluster, while resource indicator helps keeping the network stable. If a cluster head does not fit trust criteria, then the feasibility of attacks will remain high and increase over time. Malicious cluster head can change cluster configuration to perform attacks such as blackhole. Besides, trusted cluster heads are expected to make more accurate decisions when it comes to detect malicious behavior. On another hand, cluster head with sufficient resources, such as residual energy, computing and storage are needed to better handle detection and other management tasks without compromising the QoS. The cluster head is selected from edge vehicles. The edge vehicle can be elected for cluster head role if it meets trustworthiness criteria in terms of knowledge, communicativeness, and energy.

1) Knowledge: The knowledge is determined based on past experience that an edge vehicle got with other edge vehicles. It can be quantified by evaluating positive experiences over total experiences according to trust compliance criteria. For example, positive experiences with an edge vehicle can be derived from valid events reported by the edge vehicle (e.g., validation of events can be based on location proximity).

2) Communicativeness: The communicativeness of an edge vehicle can be assessed based on its dynamic interactions within the network' area zone. This metric is related to cooperativeness trust-related metric. Communicativeness can be determined by analyzing the number of encountered nodes throughout the edge vehicle's lifetime in the network. A higher 1communicativeness value indicates a more cooperative behavior. This signifies a willingness to collaborate rather than act selfishly.

3) Energy: We consider an energy metric when selecting a cluster head. This choice is vital for ensuring the effective operation of the clustering architecture. When a cluster head with insufficient remaining energy is chosen, it tends to exhaust its power quickly. A minimum level of energy is required for cluster head to achieve its tasks. In our IDS, we apply the multi

cluster head concept which can handle strict energy requirement. However, the cluster head is responsible for various tasks such as collaborative local detection cluster management, and communication with the SDN controller. Hence, this metric is used to more strengthen the selection decision. Considering the above trustworthiness-related metric, the decision on the selection of cluster head follows Fuzzy Analytic Hierarchy Process (Fuzzy AHP) rule. The Fuzzy AHP is a method that accommodates uncertainty in decision-making. The Fuzzy AHP is beneficial when dealing with subjective trust. This approach helps to assign weights to the criteria used for selecting a cluster head. A criterion with higher weight has greater importance compared to others criteria for the decision-making process. The steps of Fuzzy AHP for cluster head election are the followings:

1) Hierarchical structure creation

The structural hierarchy Construction involves determination of criteria (knowledge, communicativeness, energy and knowledge. These criteria are used to select proposer clusters heads for the IDS.

2) Construction of pairwise comparison matrix

For each criterion, a pairwise comparison matrix is created. This matrix helps to determine the relative importance of one criterion compared to the other. For example, we can assesses the importance of knowledge compared to energy. Assuming m criteria, the pairwise comparison of criterion x with criterion y builds a matrix A_{mm} where a_{xy} is the relative importance of x regarding y. In A_{mm}, a_{xy} = 1 when $x = y$ and a_{yx} = $1/a_{xy}$.

$$\mathrm{A}_{mm} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mm} \end{pmatrix} \quad (5.3)$$

3) Normalization of comparison matrix

The values in the comparison matrix are normalized for a common scale. This step helps in making proposer comparison between criteria.

D_{xy} = a_{xy} / $\sum_{y=1}^{m} a_{xy}$; x≤m,y≤m(5.4)

4) Construction of weighted normalized matrix

After normalization, the weighted normalized matrix is calculated. This matrix incorporates the relative importance of each criterion, as determined in the pairwise comparison.

W_x = $\sum_{y=1}^{m} D_{xy}$ / m; $x \leq$m(5.5)

$$W = \begin{pmatrix} W_1 \\ W_2 \\ \dots \\ W_m \end{pmatrix} \text{(5.6)}$$

5) Calculation of eigen vector and row matrix

The eigenvector represents the final weights of the criteria.

$E = nthrootvalue / \sum nthrootvalue$(5.7)

row matrix= $\sum_{y=1}^{m} a_{xy} \times e_{y1}$ (5.8)

6) Calculation of maximum eigenvalue $_{max}$

The maximum eigenvalue is calculated for each criterion. This value helps in ensuring the consistency of decision-making phase.

λ_{max} = $rowmatrix$ / E (5.9)

7) Calculation of consistency index and consistency ratio

Consistency index is computed along with the consistency ratio to assess the reliability and consistency of the pairwise comparisons and weight assignments. This step validates the soundness of the decision.

CI= (λ_{max} - m) / (m - 1) (5.10)

CR=CI/RI(5.11)

where RI is a randomly consistency index. We perform the defuzzification on the Fuzzy output by centroid method. By the end of Fuzzy AHP defuzzification process, a ranking value is assigned for the target edge vehicle. If the edge vehicle' ranking is greater than the defined trust threshold, this edge vehicle can be appointed as leader in the cluster. In the contrary case, the edge vehicle is marked as ordinary cluster member. This helps in having secure and battery-powered clusters heads.

- Multi-cluster head concept

A multi cluster head scheme is implemented where a cluster can have more than one single cluster head. We consider that the cluster heads number depends on cluster density and intra-cluster topology. The multi-cluster head concept permits to deploy the federated learning structure. The multi cluster heads helps mainly in the case when one cluster head fails or experiences issues; other cluster heads can take over the responsibilities and form the back up to maintain cluster stability. So that the cluster do not dies out so quickly and the entire cluster becomes useless. Also, by having multiple cluster heads within a single cluster, the processing load for cluster managing and local detection is distributed among these heads. This can handle the computational workload. Besides, the distribution of the detection workload across multiple cluster heads allows for better energy utilization. In addition, multi-cluster heads can minimize communication overhead between clusters. Cluster heads can aggregate and filter information locally, and share only relevant findings with the respective SDN controller. This alleviates the communication overhead between clusters and SDN controller across the network. From perspective of collaborative learning, multi-cluster head allows for (i) diverse detection models and improved decision fusion for SDN controller, (iii) localized anomalies detection. Indeed, each cluster head within a cluster may develop a slightly different local detection model based on its subset of nodes and local observations. Multiple cluster heads increases the diversity of local detection models at the

SDN controller. The SDN controller that host a global detection model can aggregate decisions from multiple cluster heads, each representing a different perspective instead of relying on a single cluster head's output. Besides, a single cluster head may have limited visibility into the overall cluster behavior. The aggregation of local detection models from multi-cluster head can reduce the bias introduced by individual cluster head. Moreover, the impact of localized anomalies is reduced with multi-cluster heads. For example, if each cluster head acts as a single point of failure, the entire cluster's detection capability may be compromised by a single anomaly.

1) Pre-cluster head selection

Authority node acts as initial cluster head in cluster C_u. For the first round the selection considers the location proximity. The authority node assesses the nearby edge vehicles based on trustworthiness level. The criteria for the selection of the edge vehicle is that it should exceed the trustworthiness threshold, so that it can efficiently monitor the C_u members and provide the local detection model. Initially the trusted authority node can get information about number of nodes, and energy level and location information of edge vehicles through communication with the SDN controller. The authority node can broadcast energy status of edge vehicles at regular intervals to cluster members. The authority node starts to monitor the trust of nearby edge vehicles using metrics of knowledge and communicativeness. The authority node communicate with these nodes to gather trust information. The knowledge regarding the number of confirmed events reported by edge vehicle can be obtained by receiving event reports from edge vehicle. Data about communicativeness can be captured within the certificate. The edge vehicle calculates the trustworthiness level of the nearby edge vehicles. The edge vehicle with highest trust level is picked to be a cluster leader. Then, the authority node acts as a super cluster head and it can coordinate the selected cluster heads. The authority node further transmits the estimated trustworthiness of selected cluster head to the corresponding SDN controller.

2) Multi-cluster head selection process

We utilize the dolphin swarm behavior to optimize the selection of cluster heads. The dolphin swarm algorithm simulates the biological characteristics of dolphin echo location, division of work and cooperation. This algorithm

can help cluster head accomplishing convenient actions to choose workmate. The multi-cluster head selection follows the dolphin swarm algorithm concept. The edge vehicle selected by the authority node is subject to act like a dolphin to optimize the selection of other cluster heads in its cluster C_u. The criteria for the selection of another cluster head are based on two factors: (i) near edge vehicle to the current cluster head, (ii) maximum trustworthiness level among the nearby edge vehicles. The nearby edge vehicle with high trustworthiness level is nominated as another cluster head. Trust of this new cluster head will be sent to the SDN controller. Transmitting the trust of elected clusters heads to the SDN controller is beneficial for the weighted aggregation of local detection models (i.e., local detection model from cluster head with highest trust will gain higher importance). As the properties of formed cluster are known, we put a limit for selection process when the maximum number of cluster heads is reached. Since there is need to take over more cluster heads, the current cluster head (as dolphin) scans its nearby edge vehicles based on trustworthiness. If a scanned edge vehicle fulfills the trustworthiness requirement, it can be taken as another cluster head in C_u, and join the dolphins' swarm. The newly appointed cluster head in C_u becomes dolphin and contributes to maintain its cluster security. It can also scan its nearby edge vehicle to elect other trustworthy manager in C_u. The elected cluster heads in C_u train the behavior of their members to support the joint learning of malicious nodes detection. In doing this, a swarm of dolphins by communication and cooperation of labor will provide trustworthy clusters in the local zone area. Figure 5.7 summarizes the cluster heads selection process.

5.4.4 Proposed IDS Process

The collaborative IDS uses trust-related features as inputs for training. We reconsider two of the trust-related features in section 5.3. The IDS extracts these features from the derivative module. Then, it forwards them to the classification module to distinguish between honest and dishonest nodes. The metrics used to refer to trust-related features involves driving pattern deviation and packet behavior deviation. As a reminder, driving pattern deviation assesses trust in nodes based on speed behavior. This metric calculates a vehicle's speed deviation from expected averages for a road segment, considering proximity and similar speeds for accuracy. Cluster head mea-

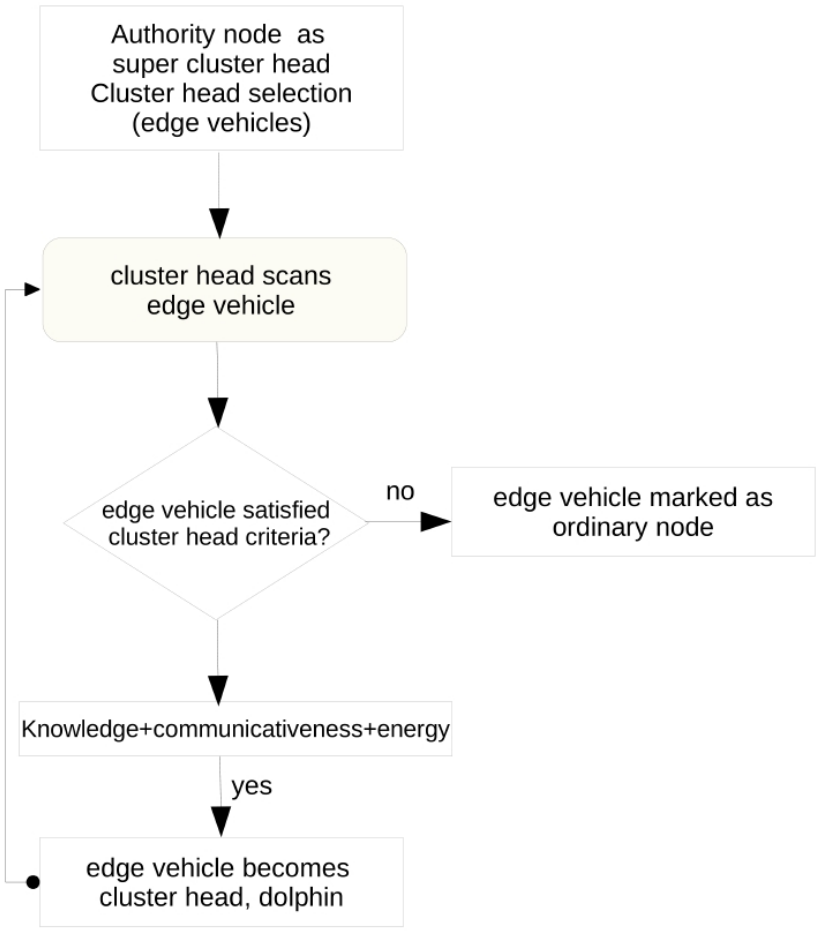


Figure 5.7: Cluster heads selection

sures this metric for their members. Since each cluster has some properties including similar speed profile, the members deviating from this profile will be flagged as dishonest. Cluster head can communicate with local RSU to obtain speed-related information for its cluster members, or it can apply map-based speed assistance to estimate its members speeds. The packet behavior deviation metric measures node reliability by tracking packet delivery consistency and flagging anomalies such as excessive drops. The cluster head can monitor the traffic it sends to a member and the traffic it receives from that member. Any missing packets in the received traffic that were expected to be sent by the member can be considered dropped. Besides, cluster head can leverage the flow statistics available in the local SDN switch. Cluster head can periodically query these flow statistics for each node within its

zone to identify anomaly of excessive packet drops. As aforementioned, the framework for the federated learning based IDS assumes the division of the network into local zones, wherein each local zone is associated with an SDN controller and may be divided into smaller clusters. The clusters can operate with multi cluster head. These cluster heads are selected from the edge vehicles to contribute in the collaborative learning for dishonest nodes in the SDN-based IoV network. Each cluster head is assumed to be local IDS node and perform the local training of the detection model. Each cluster head verifies its members for maliciousness by applying a lightweight learning algorithm. The local detection models are aggregated in the corresponding SDN controller. The local SDN controller creates a global detection model based on the insights derived from the formed local clusters. Initially, a global detection model is hosted on the local SDN controller to be shared and improved over multiple rounds of learning. The clusters heads get the global detection model at the beginning of a communication round. Each cluster head creates its own local model for a predefined number of epochs. These models learn from the data in their respective zone of clusters. Once the local training is done, the clusters heads send to the SDN controller the local models. The SDN controller combines these models to create an improved global model. The aggregation of local detection models uses weights decided based on the trust of clusters heads to give more importance to some models. The SDN controller adjusts the importance of each local model in creating the global model using trustworthiness-related data. This weighted aggregation ensures that more trustworthy clusters heads have more influence in global model update. The SDN controller improves the global detection model through learning new patterns. The updated global detection model is redistributed to clusters heads. This process is repeated over several iterations and communication rounds to refine the global model and make it adapted to network change.

5.4.5 Simulation study and results

5.4.5.1 Simulation environment and metrics

This sub-subsection evaluates the performance of the proposed IDS. The simulation was done in Python using Mininet WiFi to deploy the SDN controllers. The SUMO was utilized to generate the mobility pattern for the

network. Table 5.4 shows the simulation parameters in this evaluation. We built our network model with one SDN controller connected to one switch, and assumed to manage a specific area zone. The maximum traffic density in this zone was set to 300 nodes with 15 edge vehicles, 4 RSUs, and 4 authority nodes. The vehicles were assumed traveling on urban street with maximum speed of 50km/h. The percentage of malicious ordinary vehicles was defined to 40%. The nodes of the area zone were clustered based on distance and speed similarity. The cluster size was bounded by 50 nodes. The maximum number of cluster heads was 2. The clusters heads performed the local training for their sub-zones. We simulated packets dropping and speed deviation behavior. The speed profile was obtained through SUMO. The flow statistics from the SDN switch were leveraged to derive packets dropping number. We trained 1-dimensional CNN for a lightweight learning process. We adopted the 1-dimensional CNN since it outperformed Random Forest and RNN in the previous simulation. The performance evaluation were based on detection performance, runtime properties, and clustering scheme analysis. The training was conducted with batch size of 32, 10 communication rounds with 7 training epochs each, and learning rate of 0.5. The performance evaluation were based on detection performance, runtime properties, and clustering scheme analysis.

Table 5.4: Simulation parameters

Parameters	Values
Simulation time	1500s
Maximum nodes density	300
Percentage of edge vehicles	15
RSUs	4
Authority node	4
Average of vehicle speed	50km/h
Maximum percentage of malicious vehicles	40%
Cluster size	50
Maximum cluster heads number	2
SDN controller	1
Packets dropping threshold	15
CNN hyperparameters	batch size of 32
	10 rounds
	7 epochs
	0.5 learning rate

5.4.5.2 Simulation results and evaluation

The impact of the clustering scheme on IDS performances is evaluated regarding detection performance, cluster stability, and cluster head selection time.

- Detection performance

After training 10 communication rounds, the performances are quite satisfying, with a recall value of 99.18%, a precision value of ±99.40%, and a F1-score of ±99.30%, as shown in figures 5.8, 5.9, and 5.10, respectively. Values of recall, precision and F1-score did not decline, respectively, over 88.15%, 88.27%, and 88.19% after the 3rd round. For example, compared with [**135**] that used Random Forest, our CNN-based learning outperformed slightly in term of detection performance. Recall and precision rates resulted in increasing the F1-score of about 0.5% (see figure 5.11). This highlights the contribution of trustworthy clusters heads in cooperative learning. Trustworthy clusters heads as local learners are capable to provide accurate detection with the SDN controller. Also, multi cluster head contributed to boost the local detection.

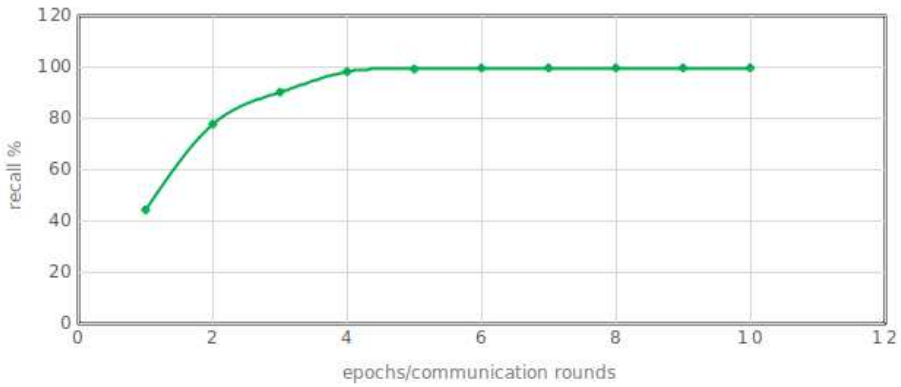


Figure 5.8: Recall rate

Detection time refers to the taken time for malicious nodes detection. The lowest the detection time is, the more secure the network is. The IDS is considered efficient one it id less time-consuming. Figure 5.12 depicts the detection time of our IDS over different density of nodes. The detection time increased with the increase of nodes density. The IDS performed with

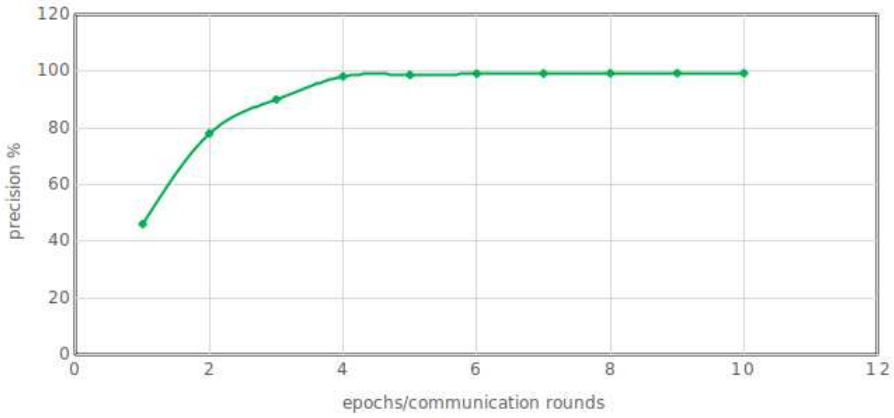


Figure 5.9: Precision rate

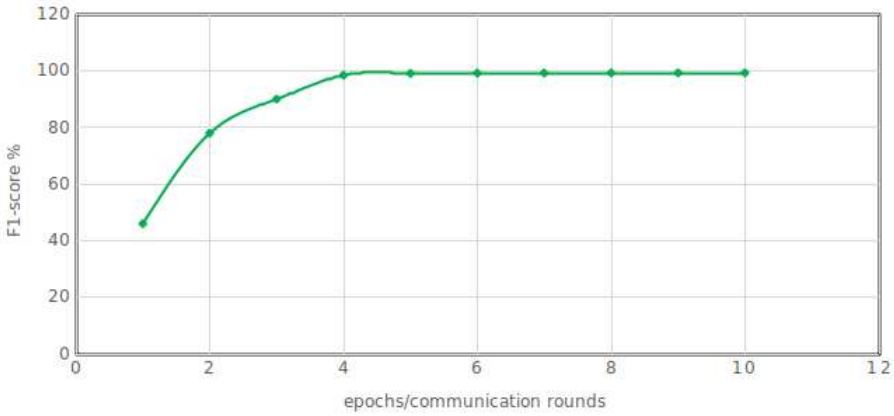


Figure 5.10: F1-score

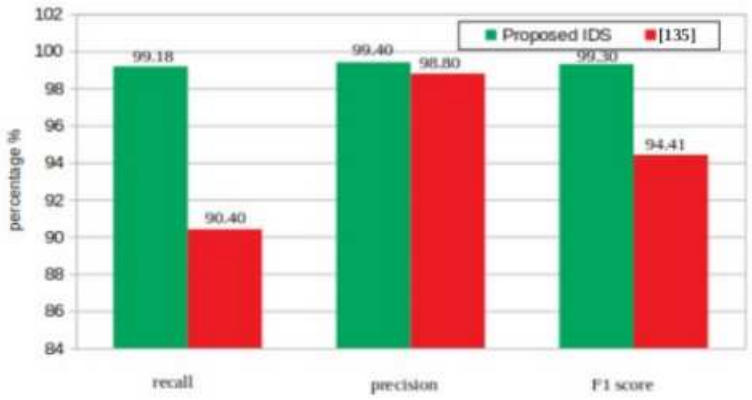


Figure 5.11: Detection performance comparison

a detection time of 71 ms. This was expected as the technique of federated learning impacts usually the detection time. Still, the IDS is not considered very costly in term of detection time regarding the detection performance it

showed in comparison with non-federated learning-based works.

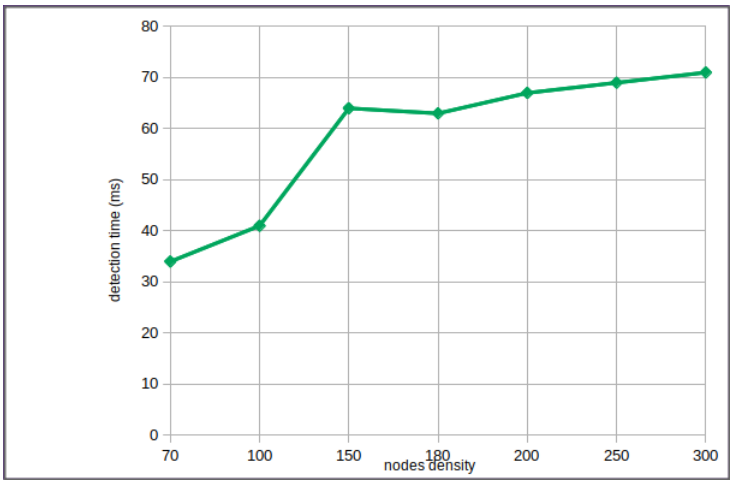


Figure 5.12: Detection time

- Cluster stability

Cluster stability can depend strongly on nodes density. We evaluated cluster stability in term of average cluster lifetime. Lifetime is the average time that a cluster maintains its state, and it is directly related to that of the cluster head. Figure 5.13 reveals the average duration of cluster while varying the traffic density from 100 to 300 nodes. By increasing the nodes density, the probability of a member joining another cluster increases, and the average duration of cluster head decreases, leading to disturb cluster stability. Compared to [**100**], we notice that our clustering scheme has good cluster stability of about 82%, since cluster can be kept relatively stable through having multi-cluster head. The average cluster stability for [**100**] was about 75.

- Cluster head selection time

The time for cluster head selection is showed in figure 5.14. The selection procedure yielded a time of about 24 ms. It increased slightly with change in nodes density. This is ascribed to the simplified optimization of the selection operation. Also, the selection procedure used a lightweight mechanism that calculates values using Fuzzy AHP.

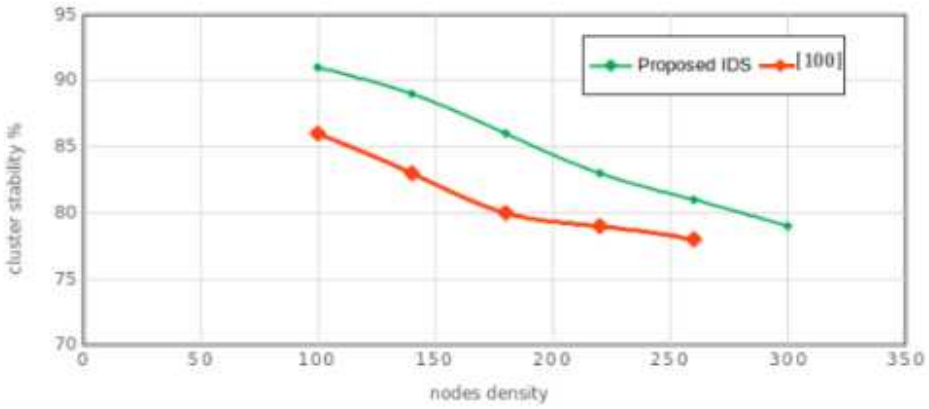


Figure 5.13: Cluster stability

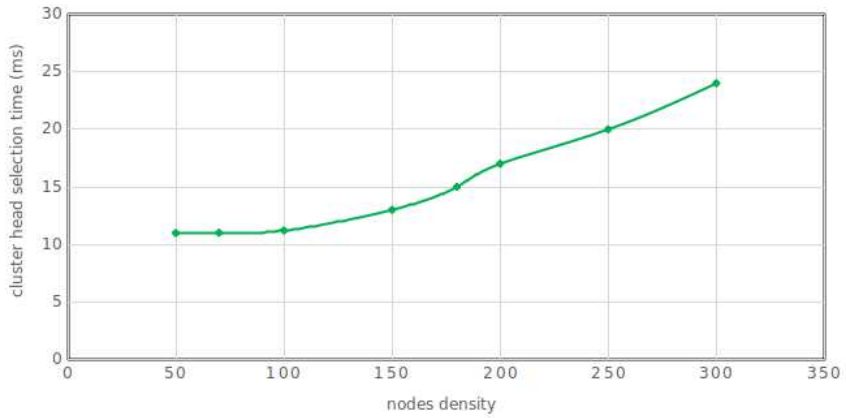


Figure 5.14: Cluster head selection time

5.5 Conclusion

This chapter introduced a collaborative IDS that uses federated learning, and trust-related metrics under SDN-IoV architecture. By combining these techniques, the IoV can benefit from efficient anomalies detection. The SDN tailored the IoV to enable collaborative detection of dishonest nodes and dynamic adaptation with better data privacy under the federated learning paradigm. The IDS framework allowed for distributed detection and localized responses while maintaining global threat awareness through the federated learning. In the first, the SDN controllers were designed as local IDS nodes that share their detection models with an upper layer represented by Cloud server. The detection process used node-properties related trust metrics (stranger node, packets dropping, speed profile). Then, we integrated the clustering for more local decision making and better IDS responsive-

ness. The clusters heads undertook the role of local IDS nodes and the corresponding SDN controller acted as detection models aggregator. We used the multi cluster head concept to boost the local detection and the cluster stability. Besides, we integrated the trustworthiness factor for the local IDS local, wherein the selection of clusters heads relied on trust and energy aspects, and was optimized through following dolphin swarm concept. The SDN along with the trustworthy multi cluster head permits the IDS to satisfy the security requirements of adaptive security, resiliency and availability. The evaluation results demonstrated the integration of trustworthy clusters heads as local IDS nodes enhances the detection performance along with providing satisfying QoS-requirement profile in terms of reliability and scalability. In the next chapter, we address IoV security along with trust management and UAVs technology from the routing perspective. Our interest will be in selecting reliable paths for data transmission.

Chapter 6

Trust-based UAV-aided routing protocol for the IoV

6.1 Introduction

Effective routing is a critical necessity for IoV applications. There are very few works that we mentioned in section 3.4.2 from chapter 3 that addressed the routing in vehicular networks with the trust management (such as [**67**] [**126**]). In fact, the routing in the IoV can be challenging due to disruption/disconnection in network. To overcome this challenge, the UAVs are the suitable candidates to strongly support the routing in the IoV. The UAV can bridge ground routing gap in the IoV and assist in data transmission during ground link breakages due to variations of traffic density or disconnectivity. The UAV-based relaying task is an important way to enhance the performance of IoV. The optimal relay-node selection is one of prominent technologies of IoV. The core element of the UAV relay-based routing is the cooperation. Ground source node can transmit the data to intermediate UAV which relays the data to the target destination node. A suitable selection of UAV relay can assure a reliable delivery of data in the IoV in case of unstable ground routing. Besides, it is worth acknowledging that to maximize the potential of UAV-aided routing in the IoV, UAV can be complemented with other technologies such Fog computing. The UAV_Fog brings to the IoV a geographical distribution and awareness of local environment and node mobility, and hence, an optimized UAV relaying. The notion of trust is crucial in the implementation of IoV routing protocols. Therefore, trust factor can be applied for a secure IoV-UAV-aided routing. In this context, we organize the reminder of this chapter as follows. Section 2 provides the recent related works. Section 3 gives an overview of the proposed routing. Section 4 presents the IoV-UAV_Fog concept. Section 5 describes the proposed IoV-UAV_Fog architecture. Section 6 details our UAV-aided routing. Section 7 provides the used metrics for the proposed routing. Section 8 gives the simulation results. Section 9 concludes the chapter.

6.2 Related works

In IoV-UAV routing protocols, vehicles collaborate with existing UAVs to ensure that transmission pathways are reliable. Routing and relay selection are the two chief considerations. The routing is the properly designed process to realize correct establishment of communications between ground nodes and

UAVs. Qualitative comparison between related works on UAV-aided routing and relaying mechanisms for the IoV is provided in Table 6.1. The reviewed routing protocols have demonstrated the benefit of the UAVs to support the routing within IoV. However, these proposals have shown some limitations with respect to their key designing metrics, outstanding features, and functioning principles. The selection of routing metrics is a top priority. The OoS awareness must be considered for good routing design. As seen, QoS can be measured by monitoring factors like packet loss, latency, available bandwidth, and signal strength. Also, nodes security is a crucial factor in the UAV-aided IoV routing. Another aspect should be focused is the network configuration including UAV deployment. The flight pattern of the UAVs should be assumed or drawn so that the impact of the UAV deployment can be derived.

6.2.1 Synthesis of Related works on UAV-aided Routing for IoV

In the following, this subsection discusses the main routing aspects to consider based on lessons learned from the studied related works. These aspects include network and IoV nodes attributes, QoS,security, routing mode, routing algorithm, and routing infrastructure.

- Network and IoV nodes attributes

Contextual information (e.g., traffic conditions, road hazards, or other relevant data) can influence the selection of optimal paths. The density was considered in most of the works in order to adapt to the dynamic nature of vehicular environment. Such metric is quite suitable for real-life implementation. By continuously monitoring and updating density information, routing algorithms can dynamically adapt the selected paths. Despite this, we apply the density information only for the initialization of the UAV-aided routing. The density does not directly influence the selection of best path in the context of UAVs as relays. Besides that, most of proposals derived paths based on vehicle' mobility information (i.e., speed, location, and direction), which is practical for estimating connectivity among nodes, mainly in the context of ground routing. Proximity information plays a crucial role in selecting UAV relays that are strategically positioned to reach destination and accommodate its trajectory.

Table 6.1: Related works on UAV-aided routing for IoV (Urban VANET' use case)

Ref	Main goals of the routing protocols	Features and Configuration of the network
[**153**] 2012	To optimize flooding between vehicles and UAVs	-Swarm-inspired approach was used for selection of Store-Carry-and-Forward (SCF) relay nodes. -The SCF selection took into account historical forwarding of nodes, speed and movement angle between UAV and ground node
[**154**] 2022	To recover up to two hops from the issues arise during the communication due to distance between vehicles and congestion	-Each node created an allied table using the vehicles in the forwarding zone. -The best node was selected from the allied node table in an end-to-end route
[**155**] 2022	To build an UAV-aided routing protocol between vehicles, and between UAVs themselves	-The UAVs elected the cluster head. The cluster head selection took into account speed, position and trust value of vehicles (direct interactions and recommendation-based). -The ground path selection for took into account traffic density information, trust of vehicle, and distance from destination node. The path selection between UAVs took into account the shortest path to the destination. -A path maintenance strategy was integrated to discover alternative path for the UAVs
[**156**] 2021	Trust-based UAV-aided routing between vehicles and UAVs, and between UAVs themselves	The UAV-aided routing was applied in case of congestion and ground connection issue. -The path selection between UAVs took into account the trust of UAVs (monitoring of forward and backward packets). -The routing between the UAVs used the greedy method. -The vehicle-UAV path selection took into account traffic information and energy level of UAV. -The path selection was based on control packets.
[**157**] 2020	To detect and repair broken paths between vehicles by using UAVs	-The proposed protocol adopted the structure of the ClouDiV algorithm. -A proactive routing was applied by each data center to detect new paths. -A reactive routing was applied by vehicles to find the nearest data center as an intermediate node.

Ref	Main goals of the routing protocols	Features and Configuration of the network
[**158**] 2019	Routing protocol based on improved flooding between vehicles and UAVs	-The UAVs were used to discover paths when the ground network is fragmented. The UAVs used the greedy method. -The route discovery used control packets with static size. -A prediction method was used to assess the expiration time of discovered path. -The path selection took into account the connectivity degree, the density, the delivery delay, and the expiration time of discovered path. -A path maintenance strategy was integrated to recover path failure.
[**159**] 2020	Flooding-based routing protocol between vehicles and UAVs	-The path discovery was initialized where there was need to establish a path. Each path is represented by a succession of zones. -The discovery process was initialized to get the maximum of paths towards the destination.
[**160**] 2019	Energy-efficient routing protocol	-A robust backbone of UAVs is deployed to proceed the reactive routing. -The UAVs are used to monitor the traffic and detecting incidents on the roads. The UAVs maintains monitoring tables of the traffic density. -The UAVs with a low residual energy level are excluded from data transmission. -The path discovery uses broadcast of route packets. -The path selection takes into account density of vehicles and fluidity degree of road segment. The near best path is generated based on the traveling time towards the area of interest.
[**161**] 2022	Energy and delay aware relay selection protocol	-The traffic followed a poisson stochastic process. The vehicles were clustered using K-means. -The UAVs adopted "decode and forwarding" relaying method. The cluster head relayed data from vehicles to UAVs.

Ref	Main goals of the routing protocols	Features and Configuration of the network
[**162**] 2021	Relay selection for Optimized Link State Routing (OLSR) protocol	-The selection of cluster heads and relays was based on Gale-Shapley matching game. -The selection took into account QoS (bandwidth and connectivity) and reputation. The QoS was computed using Bayesian function. -An incentive was applied for cluster-members to adjust trust of nodes.
[**163**] 2020	Qos aware relay selection protocol	-The UAVs assisted in routing by using the storage-carry-forward method. -The relay selection took into account link QoS and forward capacity metrics. -The relay selection used the multi-objective optimization problem to maximize link QoS and forward capacity of node.
[**164**] 2019	Mobility and energy aware relay selection protocol	-The UAVs were used as relays to transmit data to mobility service center. -UAVs locations were determined using particle swarm optimization. -The path selection used the mixed integer non-linear program to minimize UAVs consumed energy and UAVs residual energy.

- QoS

Careful consideration is given to QoS, which is crucial when designing routing protocol. Delay and timeliness can facilitate time-bounded routing. Certain contributions use the connectivity factor and the bandwidth as metric for path selection, which added great benefit to communication quality. Also, energy metric was applied in some works. The energy consideration effects UAV hovering time and helps to reduce packet drops. We will consider the energy pattern of the UAVs in our proposal. However, the energy optimisation is out of the scope of our work.

- Security

Security metric is absent in the majority of proposals. As such, the proposals are vulnerable. The security of nodes is one of the bottleneck of the routing. It is rife with problems that degrade routing. If a path node is compromised for any reason, the network is affected. During routing process, the nodes should be aware of presence of dishonest nodes. The dishonest nodes may have effects other than non-QoS routing. It is recognized that a security-based evaluation introduces additional costs. Nonetheless, it is feasible to integrate low-cost security mechanism like trust management. Only three works ([**155**][**156**][**162**]) proposed to insert the trust attribute. An incentive was used to build faster responses to malicious nodes in [**162**].

- Routing mode

Most of cited works used the UAVs as path planners, but the data transmission was through ground nodes themselves. The UAVs assisted in decision-making by providing guidance on the efficient path, but they did not actively participate in routing. Noting, that the use of ground nodes for routing can conserve UAV energy. However, the use of ground nodes as hop or relay when ground communication is poor can be problematic to implement. The ground communication is more affected by channel disturbances. The ground paths may be easily failed due to traffic, collisions, mobility factors, obstacles/geographical barriers, or even energy depletion. The condition of the road may lead to temporary and undisciplined interaction that prevents regular delivery of messages for V2X. An UAV_Fog can solve communication issue by acting as relay regarding its characteristics to strengthen

connectivity, coverage, flexibility, availability, and cost-effectiveness. Upon considering ground connection status, the UAV relaying with optimization helps the ground nodes to make and maintain constant communication. This can give best achievable network performance for the IoV that uses cooperative UAV relaying.

- Routing algorithm

The reviewed protocols were based on reactive UAV-aided routing. The UAVs worked on-demand for routing, wherein paths discovery was initiated when needed. This type of routing algorithm can save bandwidth. The flooding was adopted to initiate the path discovery. The flooding is useful because it ensures data delivery to all of the nodes that are reachable from a source node. However, the flooding leads to high overhead. To limit the overhead, certain works used control packets that had a fixed size. A selective flooding can alleviate the overhead challenge. Also, certain works used the greedy algorithm which served to obtain local optimum path node. The optimization was used to handle path decision-making. Optimization allows to consider multiple objectives and constraints simultaneously, enabling to find the path that maximizes performance metrics. By formulating the path selection as an optimization model, the works explored different trade-offs and found optimal solution. Besides, it is worthy to introduce maintenance strategy. The maintenance can involve additional processes such reconfiguration of the network topology to improve its fault tolerance, resilience, and overall performance.

- Routing infrastructure

In some works, the UAVs were exploited for monitoring (traffic, malicious activities) along with the routing. The UAVs were assumed to exchange information and strengthen the communication system. Yet, the UAVs were not fully exploited in the works. The UAVs can be used as relay and also data processing center. The segmentation of the network into zones helped in the identification of disconnected segments. The studied works assumed that the UAVs are connected to ground nodes. However, the communication model for the UAVs was not well-described (i.e., communications for Vehicle-to-UAV and UAV-to-UAV). The works did not produce dedicated ar-

chitecture for UAV-IoV routing. The introduced network models did not provide an advantageous option for better routing. The works did not assumed the case for best UAV placement to achieve the routing. The works should introduce a regular Vehicle-to-UAV communication for more efficient routing. It is worthy to include an organizing relay forwarding and an assured delivery mechanism. An assignment of UAV-zone that considers optimal number of UAVs and free-collision leads to routing optimization. In terms of proximity and timeliness, ground nodes that are closer to the UAV relays can establish faster and more reliable communication links, leading to improved data transmission performance.

Summarizing this section, an UAV-aided routing is an effective solution for the IoV. We focus on targeting some limitations in the reviewed works. Indeed, the reviewed works cover partial objectives in routing. We particularly pinpoint metrics and routing infrastructure limitations. As disadvantages, many works considered QoS and nodes attributes for routing decision, but trust for UAV is neglected. An UAV-aided routing protocol should be not vulnerable to malicious UAVs. We conclude that there is a need to design an UAV-aided routing protocol that provides QoS and security for data delivery and facilitates the IoV network. Thus, we present an organized UAV relay-based routing model for the IoV. Our solution exploits the UAVs in both relaying and monitoring (i.e., trust monitoring) components. Trough composite routing metric, optimization, and maintenance strategy, we believe to achieve reliable routing protocol.

6.3 Proposed routing overview

From the conducted review of the related works, we introduce our UAV-aided routing protocol for the IoV. The overall routing solution involves the following points.

- Producing an hierarchical architecture to merge IoV with UAV and Fog to facilitate the UAV-aided IoV routing. The UAVs in the proposed solution act as Fog nodes.

- Dealing with the deployment of the UAVs to maximize the network benefits and exploit the placement of UAVs. The distribution of UAV is

in preference following an UAV-area assignment that considers coverage and learning capabilities.

- Defining connection pattern in the IoV-UAV to shape the routing strategy.

- Producing an On-demand UAV-aided routing strategy for the IoV in case of ground routing instability. The UAVs are involved as relays, forming the data paths. The routing strategy is subject for selective limited flooding with optimized greedy to decide about relay/next-hop relay. The routing strategy involves maintenance procedure.

- Providing the model for best UAV relay selection. The selection of UAV relay considers to aggregate composite QoS and trust metrics to set up the best path discovery. Optimization based on constraints for QoS and trust is performed. Among available UAV relays, the best UAV is chosen to be part of path to forward data to the destination.

- Exploiting the trust management to add security aspects and do not excessively burden network nodes, considering their resources. The trust learning with punishing capability is incorporated to deal with the malicious activity in the network.

6.4 IoV-UAV_Fog in nutshell

Our routing strategy is applied in a hierarchical architecture based on the UAV_Fog. Each layer in the network model has its predefined tasks and offers services to the other layers. We adopt a basic Fog computing network model. The Fog computing enables a service continuum in close proximity to IoV end users and scale the IoV. The IoV-UAV_Fog network model can improve the QoE for IoV end users (i.e. extended range coverage, broadband connectivity, localized services, low latency, scalability...). Regarding data routing, the UAV_Fog can act as the platform for data delivery in an optimum manner. The UAV_Fog can play a major role in selecting reliable data routes with guarantee of security and QoS factors. The IoV-UAV_Fog model can support reliable reactive data delivery, location awareness, high mobility

support, and fast computational speed. For instance, in case of pedestrian accident, the UAV_Fog is plotted to reach the optimal data routing to the health car services (i.e., secure and fast delivery of accident data to nearest ambulance and hospital). Hence, the IoV-UAV_Fog network model can be more valuable in critical scenarios. Moreover, such framework model is suitable for the IoV when multicasting special events. For example, if a terrorist attack occurs, emergency messages have to be properly diffused to authorities to intervene immediately and also to other nodes to divert from the area for their safety. In such scenario, the IoV-UAV_Fog is quite promising for multicast routing. In upcoming sections, we detail our IoV-UAV_Fog-based routing solution. Section 6.5 introduces the proposed IoV-UAV_Fog architecture. Section 6.6 details our proposed routing strategy. Section 6.7 presents the performance evaluation of our proposal.

6.5 Proposed IoV-UAV_Fog architecture

Before describing the UAV-aided routing, we point out details about the network architecture that we are based on. Next, we show the assumptions associated with the IoV-UAV_Fog. Also, this section deals with the IoV-UAV_Fog elements' setting.

6.5.1 Network model

Figure 6.1 illustrates the envisaged IoV-UAV_Fog network model. We discern four tiers in the architecture: ground IoV nodes, local/semi-central UAV_Fog nodes (i.e., mist computing layer), central UAV_Fog layer, and 5G Cloud server. The partitioning of UAVs into mist Fog computing (i.e., lightweight Fog nodes) and central Fog is used to gain Fog service quality. We describe the key roles of as follows.

- Ground IoV nodes

Ground IoV nodes represent terminals deployed at the edge and bottom layer of the network including different types of entities such as smart vehicles, RSUs, passengers, control nodes (e.g., local trusted authority nodes) and other infrastructures. The ground IoV nodes establish a local network

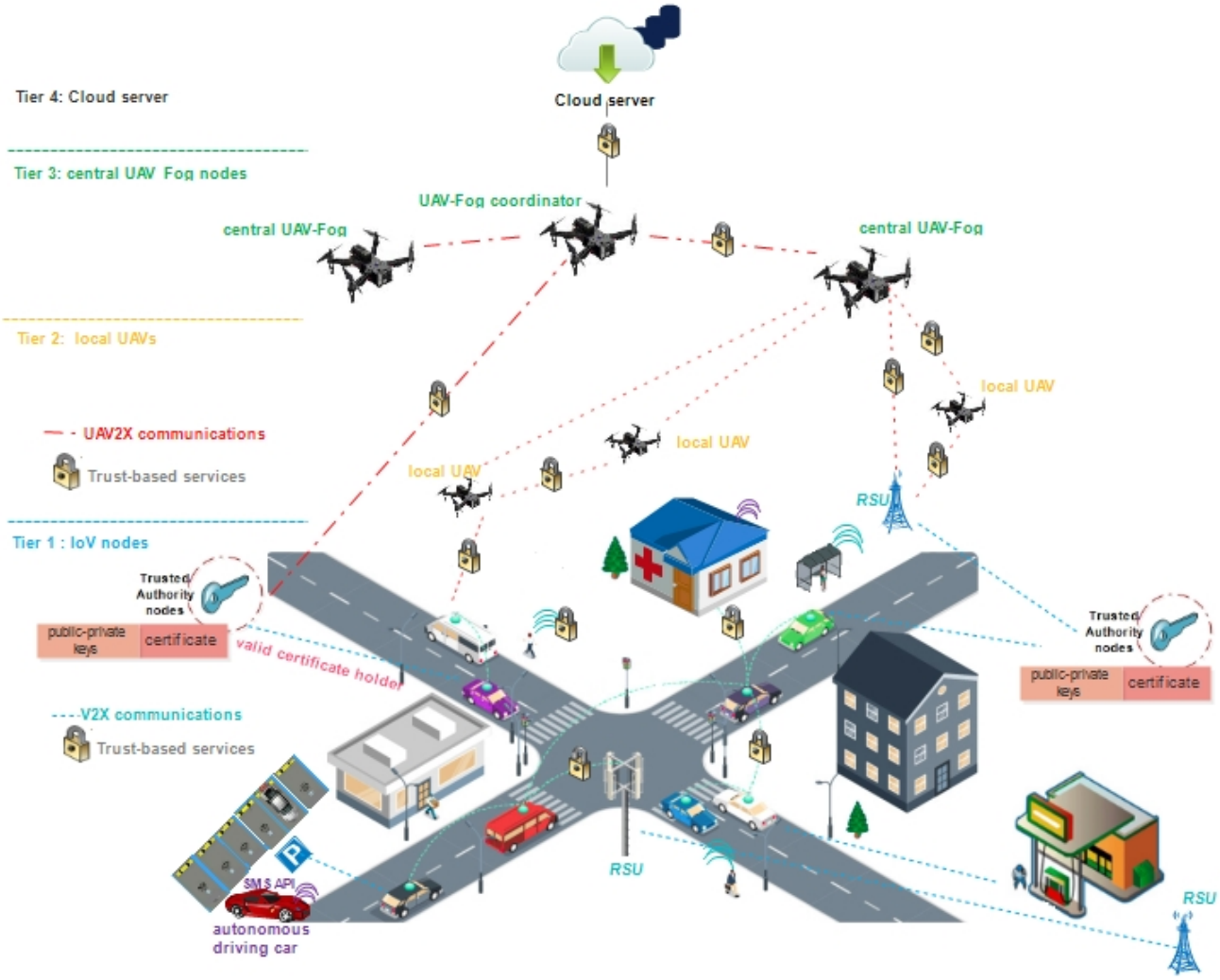


Figure 6.1: Proposed architecture

with the UAV_Fog. The ground IoV nodes can communicate with the UAVs via APIs (e.g, RESTful) to request data routing.

1) RSU

The RSUs are deployed to serve multiple purposes to V2X communications for several specific scenarios: relays, broadcasters of information (e.g., events, V2X ground link' quality), and learners of vehicles' trust. We consider the deployment of RSUs to aid optimal data relay.

2) Control node

The control node is meant to be a trusted authority node that redacts

V2X authorization and monitors RSUs behavior and cooperation pattern. The control nodes can be categorized as super ground nodes with respect to RSUs. The control node tracks periodically trust and functional status of RSUs in its area.

- UAV_Fog layers

UAV_Fog layers are composed of multiple UAV nodes. The UAVs are deployed to be fully exploited. The UAVs follow UAV-area assignment. The UAVs perform both planned tasks (e.g., traffic situation analysis) and on demand actions (e.g., data routing). The UAVs are linked with the RSUs for cooperative awareness of traffic and dispatch of nodes' security information.

1) Central UAV_Fog nodes

Central UAV-Fog nodes are located between the Cloud and the local UAVs to implement computing and storage functions. The central UAV_Fog will be able to view the historical data about the IoV network and develop V2X-related recommendations. The central UAV_Fog nodes control and monitor the local UAVs. Besides that, the central UAV_Fog is in charge to build the multicast routing of sensed particular events. In fact various events monitored by the IoV sensors require routing to a set of nodes (e.g., road maintenance, poor road condition, case of parade or strike, suspicious vehicle rushing to a specific location...). So that we leverage the central UAV_Fog-assisted resources to alert on specific events via multicast routing. The central UAV_Fog integrates the multicast-related sub-tasks like multicast requests classification, scheduling, and joining multicast session.

2) Local UAVs

Local UAVs are connected directly to the central UAV_Fog. The local UAVs are considered as first access point choice for the ground IoV nodes to communicate the central Fog. Nonetheless, ground IoV nodes like RSUs and authority node may make direct connection with the central UAV_Fog nodes. That being so, central UAV_Fog nodes can serve the ground IoV nodes either with assistance of local UAVs or directly. Given the high volume of requests, the local UAVs can balance ground IoV nodes' routing demands on central Fog. They can accept continuous requests from the ground IoV

nodes and refine the final demands to the related central UAV_Fog nodes, so that split the balance load. The local UAVs maintain a "forwarding class" in which the recent path up to frequent destination is cached. In fact, "forwarding cache" can be useful in scenario of traffic with low mobility travelling nodes (e.g, within zone of high V2P interaction which requires low-speed like residential area or school zone). The forwarding cache can serve to quickly have recent available path to frequent local destination but not fast moving destination.

3) UAV_Fog coordinator

The UAV_Fog coordinator manages the central UAV_Fog nodes. This node acts as an authority control node for the UAVs. The UAV_Fog coordinator deploys the relay UAVs based on coverage and sensing capabilities, wherein it tracks performance and updates the area assignment according to the performance criteria. From another hand, the UAV_Fog coordinator can perform V2X data pre-processing for effective multicast routing. The dissemination of event data to destination nodes is preceded with an event weight-based classification. The sensed events data are pre- processed to favor urgent event to be informed. The pre-processing categorizes the sensed data into normal and critical branch on the basis of event importance. These branches decide on the multicast of the event data by the appropriate central UAV_Fog. Once the event is found to be critical, the UAV_Fog coordinator generates an immediate multicast routing request to the central UAV_Fog.

- Cloud server

The Cloud server is running for big data processing uploaded from the central UAV_Fog nodes to provide vehicular services. The Cloud layer allows the data to be available for use 24 × 7. We assume a trust-based IDS server to further take placement inside the Cloud server. The UAV_Fog coordinator and the control nodes can always interact with the Cloud and provide bi-directional communication to open trustworthiness evaluation. In doing that, the Cloud server observes the trust of UAV_Fog coordinator and control nodes for overall network security performance. The Cloud layer can be implemented using a customized application including different panels such admin, Fog, and IoV user. The log into can be via tablets, smartphones, laptops, or vehicle navigation system. The overall network maintenance can

be placed on the Cloud server. Pointing that the Cloud server does not contribute in the routing procedure.

- V2X communications

The communication between the network' nodes may be achieved using 4G/LTE, 5G, 6G/ B6G, WiMAX, WI-FI/DSRC/WAVE, Bluetooth, and ZigBee. These technologies permit to the IoV nodes to submit requests in the Fog tier. The inter-network-tier connection can be done through WI-FI, LTE, 5G, 6G or WiMAX for long coverage. For intra-network-tier connection (e.g., over short distance/platoon communication), we can situate Bluetooth or ZigBee. The IoV-UAV_Fog enables comprehensive collaboration and bidirectional data flow between nodes (i.e., Ground-to-Aerial (G2A) and Aerial-to-Ground (A2G) communications). We interest in the following bi-directional communication links: (i) V2V, (ii) V2R, (iii) Vehicle-to-local UAV (V2U), (iv) RSU-to- central UAV_Fog (R2F), (v) local UAV-to-local UAV (U2U), (vi) local UAV-to-central UAV_Fog (U2F). V2U, U2U, and U2F links are highlighted when data routing on the ground is not possible/difficult. The formed communication links are subject to trust factor. Also, communications between authority node and central UAV_Fog or UAV_Fog coordinator are taking place within authentication and trust services. In our proposal, the UAV nodes perform drones-oriented routing mission in case of ground connection failure or difficulty in destination reachability. The local UAVs are the core for the data routing. The central UAV_Fog is mainly responsible for the multicast routing.

Each group of local UAVs matches to one central UAV_Fog, following the preferred assignment of UAV nodes in the network areas. The coverage area of the central UAV_Fog includes primarily its own semi-central UAVs. The IoV nodes connecting to this set of local UAVs are considered in the coverage area of that central UAV_Fog. Besides, we assume that here is no overlapping area when there is a semi-central boundary UAV within the transmission range of two central UAV_Fog nodes; since each semi-central UAV connects to its central UAV_Fog base. However, the semi-central boundary UAV can be connected to the foreign adjacent central UAV_Fog base if needed. (details are given in subsection 6.6.2). The central UAV_Fog is fully aware of UAV_Fog- data from local UAVs and other central UAV_Fog nodes. Each group of local UAVs nodes sends to its central UAV_Fog base sensed

information about ground IoV nodes (e.g., location and direction). Every UAV_Fog (i.e. semi-central or central) has local database with data samples containing information in each link such trustworthiness and connectivity (e.g., traffic density, line-of sight (LoS), available bandwidth, latency). Each UAV_Fog can use its database to define optimal routes quickly. In addition, the central UAV_Fog maintains a general database containing information about the entire geographical zone. The central UAV_Fog saves area data either from its own or from semi-central UAV and RSU. This means that the central UAV_Fog has completed knowledge about semi central UAVs, ground IoV nodes and links that found in its geographical area.

6.5.2 Assumptions

Let us state the main assumptions about the setting of the IoV-UAV_Fog elements. The IoV-UAV_Fog network implies that each user must have fetched and validated certificate before initiating communication. The network is assumed to insert servers of encryption and certificate-based authentication at the local control node. We assume that we place a secure trusted UAV_Fog coordinator. The coordinator is deemed to be valid legitimate and trustworthy UAV server in the assumption. With programmable authority, the essential management operations for the UAV_Fog network including UAVs authentication pass through the UAV_Fog coordinator. In addition, we assume that the UAVs can be turned on/off based on energy level. Central UAV_Fog nodes and UAV_Fog coordinator are fitted with high-capacity reloadable batteries supplied permanently through the resources of the UAV compared to local UAVs. For convenience, we assume that each UAV maintains at least a single connectivity (direct or indirect) to allow exploiting UAV as relay. There is at least one direct connection path between head node and local sub-node: (i) C2F (UAV_Fog Coordinator- to-central UAV_Fog), and (ii) U2F (local UAV-to-central UAV_Fog). Also the RSUs are assigned to at least local direct (i) R2A (RSU-to-Authority) and (ii) R2F (RSU-to-central UAV_Fog) connections. Likewise, the UAV_Fog Coordinator preserves at minimum a direct C2A (UAV_Fog Coordinator-to-Authority) connection. The UAV_Fog coordinator deploys a management contract through some criteria to manage and pilot the central UAV_Fog nodes. From active actions on the central UAV_Fog nodes like assignment of local UAVs and on/off-status change, the contract serves passively the on- demand data routing. The contract is also

adaptable for the central UAV_Fog node to assign optimal local UAVs for local geographical area. The setting of the IoV-UAV_Fog' elements is described in the next subsection; wherein we highlight local UAV-area assignment. As aforementioned in above subsection, we consider to set up the trustworthiness between and within the four network tiers to instill confidence in IoV services. Both ground and aerial layer consider the trust to maintain cooperative relationships between communicating nodes. The head nodes (e.g., RSUs and central UAV_Fog nodes) are trusted with a favorable level of security to monitor sub-nodes' trust. The RSUs are assumed to be workers of federated learning in the bottom layer to identify malicious vehicles. The RSUs upload their local trust learning models to the local control node. Each central UAV_Fog, as semi-trusted node, builds-up a trust table for the local UAVs to be exhibited. The evaluation of local UAVs' trust contributes in distinguishing reliable relay nodes for the on-demand data routing. As we proceed we will take into account the trust criteria to apply for selection of optimal data paths. The tables of trust of local UAVs and ground IoV nodes are assumed to be shared between RSUs and central UAV_Fog and exhibited to other involved nodes. In doing that, malicious vehicles and distrusted UAVs can be flagged and avoided over the routing process. Additionally, we suppose that from time to time the UAV_Fog coordinator exchanges with the control nodes to keep an eye on trust of sub-nodes (i.e., central UAV_Fog nodes and RSUs). This can promote the trust recommendations' integrity.

6.5.3 IoV-UAV_Fog elements' setting

We draw the UAV_Fog nodes close to the ground IoV nodes. This permits to facilitate traffic awareness, and reduce response latency and processing. Ground network and airspace are splitted into X large leaf zones. Each of these leaf zones is in turn subdivided into Z sub-zones/local areas, considering majorly the intersections as illustrated by dashed lines in figure 6.2. The segmentation of the network into local areas can provide link breakages location. The local division helps to undertake data path maintenance. The network is divided into Z local areas whose set is denoted Z=1,...z,...Z. Area-ID is used to distinguish the area wherein a target node is located. The area-ID includes information like matched head nodes, geographical coordinates of head nodes, and traffic conditions. It helps to guide the routing algorithm. Besides, under multicast routing scenario, the Area-ID enables

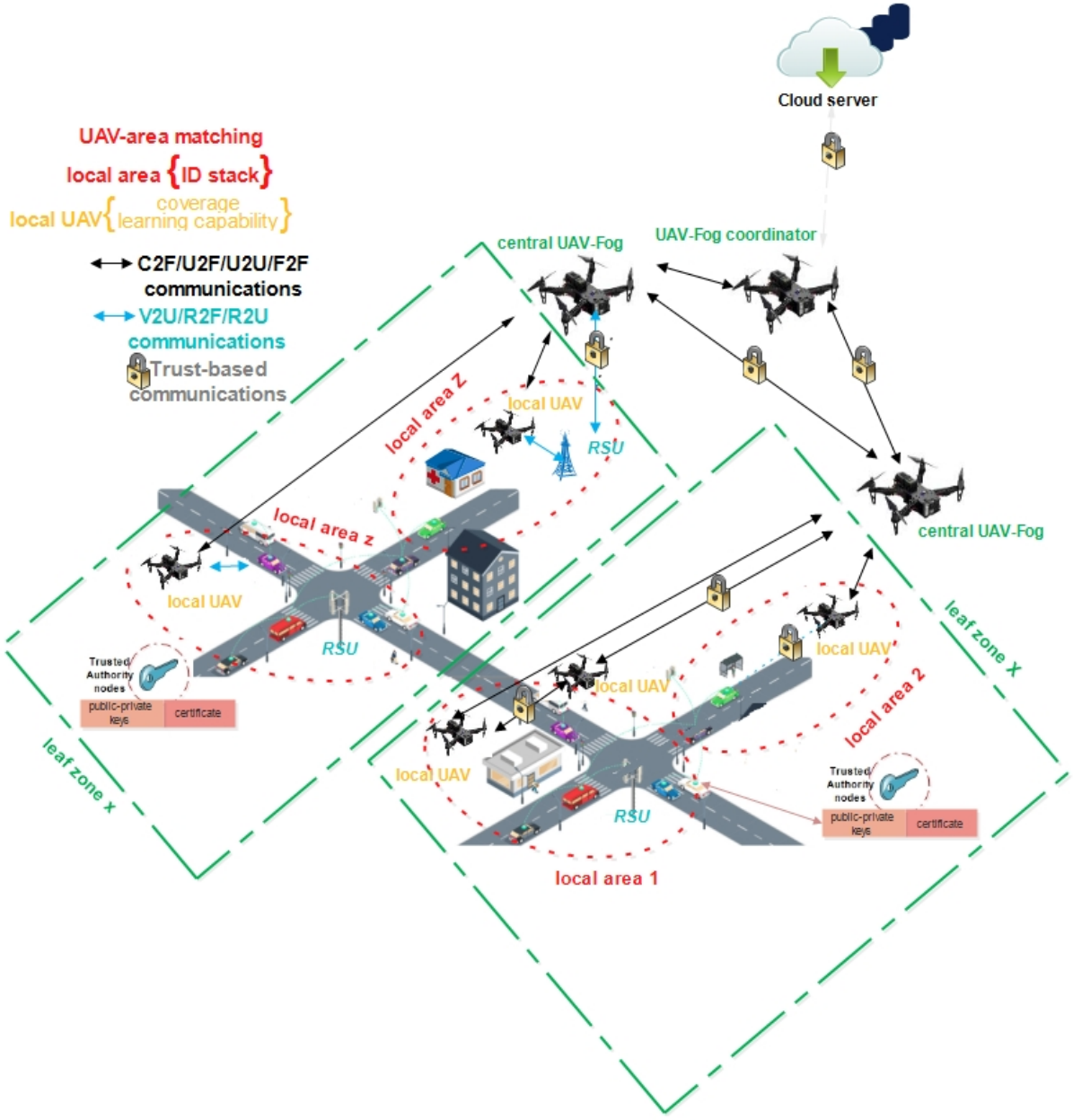


Figure 6.2: IoV-UAV_Fog elements' setting

to reach correct zones for packet delivery. We intend to make reliable deployment of IoV-UAV_Fog elements which links the maximum number of disconnected IoV nodes. In fact, maximizing the UAV coverage implies maximizing the number of linked nodes and covered areas.

The IoV-UAV_Fog model requires the number of control nodes, RSUs, central UAV_Fog nodes, and local UAVs to be defined. This is due to the role and qualification needs of such nodes for network access and commu-

nication. The optimal deployment of RSUs refers to a limited number of RSU nodes with maximum coverage. However, we place the RSUs arbitrary since the RSU deployment problem is out of the scope of our work. We install fixed RSUs at traffic light intersection since it will be passed by vehicles. To avoid the complexity of having variable densities of RSUs, we suppose that the maximum number of located RSUs within/nearby to the target zone is $|RSU|= U$. The control nodes are also placed arbitrary. The number of authority nodes is set to be lower than the RSUs, reflecting the hierarchical structure of IoV-UAV_Fog network.

In addition, positioning of UAVs affects IoV-UAV_Fog' performance. It is argued that using UAVs to serve ground nodes anywhere and at any time is critical. However, UAV-area assignment is an important observation to consider to effectively serve the ground nodes, and avoid broken UAVs links. We do not specify the mobility model of UAVs and the relative speed difference between UAVs and vehicles, but we consider that each UAV should be assigned in a certain zone. Each large leaf zone is covered by one central UAV_Fog. The central UAV_Fog is mainly located around the sub-zones of high traffic flow or at the sub-zones of traffic accident-prone. The central UAV_Fog distributes groups of local UAVs to cover the sub-zones/local areas. Each group of local UAVs covers a certain local area. The ground IoV nodes in a geographical local area connect with the matched group of local UAVs. We force each UAV to remain within a containing area to ensure that ground nodes remain in coverage. The network can extend easily that an UAV swarm is assigned for each local area z (i.e., according to IoV density).

The set of ground IoV nodes involves vehicles $V = V_1^1$,...,V_n^z,..., V_N^Z, RSUs $RSU= RSU_1^1$,..., RSU_u^z,..., RSU_U^Z, and authority nodes $TA= TA_1^1$, ..., TA_a^z, ...,TA_A^Z. The vehicle V_n in local area z ($V_n \in$V^z) is positioned at $x_{V_n}^z \in$R^3. The vehicle V_n is located at $Dn(V_n, V_i)$ distance from destination node V_i within z ($\forall$Vi$\in$V$_N$, $V_i \neq V_n$), where $Dn(V_n, V_i)= \|x_{Vn} - x_{V_i}\| < \infty$, i.e., the ground nodes are inter-accessible. Nodes V_n and V_i are considered disconnected when there is at least one area z separating them, where $|V^z| = 0$ (i.e., V^n is empty). In area z, the group of J local UAVs is denoted UAV_J=1, ..., UAV_j,..., UAV_J. UAV_J surrounds its base (i.e., corresponding central UAV_Fog) and it is ready to be located in chosen local area. Denote C_j the central UAV_Fog of UAV_j located at $y_{Cj} \in$R^3. The distance between C_j and local area z is marked as $Dn(C_j) = \|y_C^j - x_{V_{n*}}^z\| < \infty$, where V_n^* is important

to cover in area z (e.g., Police vehicle or rescue vehicle). The local UAVs under consideration are low-attitude UAVs. The IoV node may maintain line-of sight (LoS) link (or at least a reasonable non-LoS (NLoS)) with the UAV with which it communicates. In fact, different obstructions like buildings, bridges or high trees may lie between IoV node and UAV, causing the non-LoS (NLoS) connectivity. LoS conditions generally leads to better performance as it provides a clear and unobstructed path, reduced interference, minimal signal attenuation, fewer signal reflections, and shorter propagation path. The LoS scenario results generally in higher Signal-to-Interference-plus-Noise Ratio (SINR) and strong signal quality. G2A and A2G links can be connected, if their respective LoS meet/exceed the given thresholds. The LoS depends on distance between UAV_j and IoV node V_n^z, and environment variables; i.e., communication surrounding such as blockage and density of obstacles. The probability of LoS is given by [**165**]:

$$PLoS(\theta_{UAV_j^{V_n^z}}) = \frac{1}{1 + \alpha exp(-\beta(\theta_{UAV_j^{V_n^z}} - \alpha)'} \tag{6.1}$$

where $\theta_{UAV_j^{V_n^z}}$ is the elevation angle between UAV_j and ground node V_n^z. α and β are constants determined by environment characteristics (e.g., dense urban, urban, or sub-urban). From Eq.(6.1), the probability of NLoS occurrence is as $PNLoS(\theta_{UAV_j^{V_n^z}})$= 1 - $PLoS(\theta_{UAV_j^{V_n^z}})$. We assume that each UAV hovering over a given area can cover multiple ground nodes and establish a successful communication with a certain LoS. Therefore, the UAVs should be properly dispatched in a way to operate long-term communication relays that intelligently link disconnected ground nodes. The proper UAV-area assignment provides an efficient coverage strategy, preserves connectivity during the flight period, prohibits frequent link breakage and ensures the integrity of the upper layer (UAV_Fog service) by getting around of disruptive collision between UAVs and fairly exploiting available capabilities on each UAV. The UAV-area assignment uses the information about coverage capability and quality of sensing as function of maliciousness learning. An UAV with coverage and intelligent sensing capabilities enables strong observation, real-time data evaluation, and better data transmission rates.

- Coverage capability

The coverage capability considers distance and cost of sensing. We term $A^{UAV_j,z}$ the coverage assignment of UAV_j in area z; $A^{UAV_j z}$= $\{a_{V_n,V_i}^{UAV_j,z}$, $\forall$V$_i$

$\in V_N\ V_i \neq V_n\}$, where a_{V_n,V_i} = 1 means that UAV_j is tasked to fly through the area between nodes V_n and V_i, and a_{V_n,V_i} = 0 implies otherwise. The sensing distance $S^z_{UAV_j}$ covered by UAV_j under assignment $A^{UAV_j z}$ is expressed as [**165**]:

$$S^z_{UAV_j} = \sum_{V_i \in V_N, V_i \neq V_n} a^{UAV_j,z}_{V_n,V_i}\ S^z_{V_n,V_i} \quad (6.2)$$

Let $\rho^z_{UAV_j} = \frac{1}{|V^z|} \sum_{V_i \in V_N, V_i \neq V_n} a^{UAV_j,z}_{V_n,V_i}$ the proportion of UAV_j coverage in area z, where $0 \leq \rho^z_{UAV_j} \leq 1$. The proportion of coverage regards the moving IoV nodes towards center of UAV_j coverage and the moving IoV nodes towards edge. However, we state that the UAV_j can fly as well to and from its base; i.e., C_j. Hence, the total moving distance by UAV_j: $Ts^z_{UAV_j}$= $S^z_{UAV_j}$ + $S^z_{C_j}$, where $S^z_{C_j}$ is the transition distance to C_j. The traveled distance is energy consuming. The energy consumed to cover the distance of sensing and transition takes part in UAV_j' on/off-status change. The energy cost for sensing area z can be expressed as [**165**]:

$$PW^n_{UAV_j} = \frac{Ts^n_{UAV_j}}{sp_{UAV_j}} pow_{UAV_j}\ PW^n_{UAV_j} = \frac{\rho^n_{UAV_j} S^n_{UAV_j} + S^n_{C_j}}{sp_{UAV_j}} pow_{UAV_j} \quad (6.3)$$

where sp_{UAV_j} is UAV_j' average velocity; with propulsion power consumption pow_{UAV_j} ($pow_{UAV_j} = pf_{UAV} sp^3_{UAV_j} + \frac{pr_{UAV}}{sp_{UAV_j}}$; where pf_{UAV} and pr_{UAV} are powers related to skin friction and drag force, respectively).

- Learning capability

An UAV with a strong learning capability can actively monitor and gather data on nodes behavior, traffic patterns, and historical performance to be used to train detection model. A strong learning capability can lead to higher detection rate of malicious nodes. Here, we accord the learning capability to time and energy cost. The UAV_j learns trust of its UAV group' members and ground nodes over K training iterations in reasonable policy sets. The UAV_j as learning node generates decision-making on optimal trusted data route. The UAV_j trains its model $W^K_{UAV_j,z}$ using the sensing data to

minimize a μ-convex loss function with L-Lipschitz gradient ($\phi_{UAV_j,z}$) up to the target accuracy O^* defined by the UAV_Fog coordinator to derive trust parameters' update and UAV_j' assignment. A higher value of O^* indicates deviation from the optimal value. Here, we have 0 < O^* < 1, implying that the model $W^k_{UAV_j,z}$ is trained to approach optimally for every training step; i.e., to minimize computation time. The UAV_j completes the trust training in area z after K= $\frac{o}{1-O^*}$ iterations, where o= $\frac{2L}{2\mu\zeta}$ and ζ is optimization parameter ($0 \leq \zeta \leq \frac{\mu}{L}$). The computation time is taken as:

$$\mathrm{LT}^z_{UAV_j} = K(\frac{U_{UAV_j} * Z_{UAV_j} * \rho^z_{UAV_j} * q^z_{UAV_j} * log_2(1/O^*)}{lt_{UAV_j}}) \quad (6.4)$$

where Z_{UAV_j} is CPU cycles to handle one sample data, $q^z_{UAV_j}$ is UAV_j' data samples from $Q^z_{UAV_j}$ local database, and $\rho^z_{UAV_j} q^z_{UAV_j}$ is unit of samples sensed by UAV_j. lt_{UAV_j} is UAV_j' computation capacity in CPU cycles. The energy for learning is taken as:

$$\mathrm{LE}^z_{UAV_j} = K(\iota * \mathrm{Z}_{UAV_j} \, ^* \, \rho^z_{UAV_j} * q^z_{UAV_j} * U_{UAV_j} * log_2(1/O^*) * lT^2_{UAV_j}) \quad (6.5)$$

where K is the iterations number, ι is the effective switched capacitance, $U_{UAV_j} log_2(1/O^*)$ represents minimal number of required iterations to reach O^*; where U_{UAV_j} = $\frac{2}{(2-L\delta)\delta\mu}$, and δ is size of optimization step (δ < $\frac{2}{L}$).

The UAVs are dispatched after finding the optimal UAVs number and the well-suited locations.

- Optimal number for UAVs

The proportion of coverage (ρ) is used to find optimal number of UAVs for coverage of area z (up to predefined coverage threshold th_{cov}). The optimal number of UAVs to cover area z is derived following algorithm 3.

Input: ρ, location of IoV nodes $X = x_{V_1^z}, ..., x_{V_n^z}, ..., x_{V_N^z}$, coverage threshold th_{cov}
Output: number of UAVs for area z
do
  if (ρ < th_{cov}) **then**
    add a new UAV (UAV_J) in area z
    update list of assigned UAVs in area z
  endIf
while (ρ < th_{cov})

Algorithm 3: optimal number of UAVs in area z

- Suited location of UAVs

Initially, a first UAV_j is located at $loc_{UAV_j} \in \mathrm{R}^3$ in the middle of the area z. The UAV_j can fly over different locations to sense information along the boundary of area z. Increasing the distance $Dn(V_n, V_i)$ between source node V_n and destination node V_i impacts significantly on $\rho^z_{UAV_j}$ (proportion of UAV_j coverage in area z), and accordingly cost of sensing ($PW^z_{UAV_j}$). Deploying the corresponding number of UAVs in z provides an efficient coverage strategy for involved IoV nodes. A location-constraint for UAV_j is applied on crossing boundary of matched area z ($bound_z$).

$\mathrm{Pen}_{boundzUAV_j}$= 0 when 0 $\leq \mathrm{loc}_{UAV_j} \leq bound_z$, Υ otherwise (6.6)

Such constraint is significant for suited location of the UAV. Besides, for possible collisions between UAVs, the UAV_j avoids collision from aspect of getting penalty every time it takes action resulting in going over the safety distance.

$\mathrm{Pen}^z_{colUAV_j}$= Λ when $Dz(UAV_j, UAV_J) < saf$, 0 otherwise (6.7)

where $Dz(UAV_j, UAV_J)$ is the distance between UAV_j and UAV_J, and $saf \leq Dz(UAV_j, UAV_J)$ is safety distance.

Considering Eq.(6.7), the UAV_j adapts its trajectory to prevent UAV-to-UAV collision in z. The UAVs are allowed to make small adjustments to their

locations to optimize their proximity to the destination for higher quality link when proceeding direct communication link. Additionally, in order to facilitate monitoring and addressing collision, when a collision occurs the involved UAVs can transmit to the central UAV_Fog a collision report containing relevant details, such as the time of collision, the UAVs involved, and the location of the incident. These reports enable central UAV_Fog to identify any patterns or recurring collision instances. Besides, the UAV-area assignment is assumed to involve use of density-based thresholds. The density of ground nodes will indicate the availability and distribution of UAV relays. Placing the UAVs based on IoV density can optimize the routing. Higher IoV density areas may offer more opportunities for cooperative communication and UAV relaying. We consider that the optimal UAV-area choice would be to assign the higher performance UAV to denser area due to sensing cost considerations. The highest trusted- capable UAVs are positioned in the highest dense areas. Paths passing through areas with higher ground nodes density are more likely to have potential relay enabling routing over areas with higher demand for data transmission. The UAVs placed in less traffic area and observed to have less trust level area are although assumed to maintain minimum trust threshold. Stating that, the UAV-area assignment provides the basis for the trustworthiness of the UAV_Fog service. We assume that higher performance UAVs are more effective and efficient in densely populated areas due to their superior capabilities and the associated costs of sensing and coverage. An optimization method can be involved to relieve the UAV-area assignment. For example, the grey-decision calculation can be performed to sort the UAVs with multi-objective decisions. Moreover, given that the UAV-area preference is based on performance criteria, an update for the assignment is triggered (e.g., remove of UAV_j from z /re-matching with other area/change of UAV_j position). The central UAV_Fog observes the status of its local UAVs and can update the assignment over the density-based thresholds. Once the connection links seems unreliable/absent, the UAVs topology can be modified. The UAV keeps the area as a matching zone when the connection link is reliable. The update is expected to guarantee proper coverage and reliable ongoing UAV-aided routing.

6.6 Proposed UAV-aided routing

This section describes the proposed UAV-IoV aided routing. This section covers routing process, path selection metrics, and path selection problem formulation. The routing paths are created through cooperation with UAVs, according to connection status of ground IoV nodes and flow of ground traffic. The path decision considers trust and QoS factors. A path maintenance is included, saving the path/path back to the source once the decided path is broken. The rules for path decision and path maintenance are given below. Then, preliminaries and main process of the routing protocol are detailed. We describe the routing functioning and the route packet format. Next, we provide a brief description of the multicast routing process. After that, we present the metrics for optimal path selection, and finally we used multi-objective optimization to formulate the relay selection problem.

6.6.1 Initialization of UAV-aided routing

We now have the network' elements in place to describe our routing strategy. The core content of our routing strategy is the selection of optimal relay nodes (i.e., succession of relays forming the selected path when it is needed). Each node is assumed to be smart enough to sense any event at every point in its communication range. It generates a data packet to be sent containing axis coordinates, timestamps and ID for that event. In the "sense-and-send" scenario, the node V_n wants to send event information to node V_i. We assume that based on the trust learning phase, an IoV node can distinguish whether target node is malicious. Nodes flagged as misbehaving will be isolated either locally by its neighbor or globally, by deleting it from data routing protocol repository and ignoring its messages. The suggested routing strategy presupposes that it is to security layer to decide whether to interact with malicious nodes. On this basis, we claim that the node V_i can avoid malicious nodes by means of shared trust information. The aid from UAV to route an event in area z is according to ground traffic analysis an ground connection status. The UAV-aided routing is either initiated by source node V_n or through recommendation from head nodes (e.g., central UAV_Fog, RSU). More details of UAV-aid initialization are presented as follows.

The source node V_n starts an appraisal of the traffic situation in order to reveal the possibility of reliable routing through ground nodes. In the case when V_n self-estimates a traffic congestion or finds that the terrestrial vehicular network is full of obstacles, the UAV will be exploited as relay. When the source node self-estimates that quality of ground route from it to the destination does not meet QoS and security constraints, the UAV is applied for the routing. In case when there is a ground-based communication failure, or the estimated route to destination has large probability to fail the QoS, assistance from UAV is carried out. The links failure in ground can cause intermittent routing paths and increased end-to-end delay. Assuming that the source node V_n has a latency requirement i.e, a message has to be successfully received within time period, if the total task delay is well beyond the delay constraint for the source node V_n, it initiates a help-request to UAV. In other words, the source node that fails to receive information from target node in a given time period, can make request to UAV-based routing. As well, if the available route to the destination is too likely to be very long in terms of hop count, the use of UAV would be more appropriate to fulfill security. Likewise, if the possible ground route to the destination is not secure, that is to say the route must relay through malicious nodes and possibility that it is compromised is very high, the source node avoids ground routing. As it is possible to notice malicious nodes from trust table, the source node can predict ground link' security, and takes alternate solution for secure routing. From another hand, UAV' assistance can be recommended for routing an event by RSU or central UAV_Fog. As we assume that head nodes (i.e., RSUs, central UAV_Fog, UAV_Fog coordinator) are tuned to sub-tasks like analysis of traffic characteristics and link quality status, these nodes can exhibit their predictions to IoV nodes to assist in planning reliable data route. Such operation offers a route control module to keep up-to-date information about path QoS. Parameters of communication link quality can be obtained through "link monitoring unit" that makes measurement of route QoS at predefined time interval. Pointing that link quality changes noticeably over a relatively large interval of time (about 10 minutes). The measurement of the link quality is updated after an interval of 10 minutes [**166**]. Regarding traffic analysis, for instance the estimation of ground traffic jam can be performed cooperatively to reach a consensus decision on traffic situation. An estimation part is conducted locally on RSU, and the other part is processed by UAV_Fog (central UAV_Fog or UAV_Fog coordi-

nator). The head nodes at road intersection (i.e.,RSU and central UAV_Fog) are relevant to make better prediction and assessment of traffic congestion' level. The head nodes can reliably quantify traffic intensity and alert road users. We stress that the collaborative estimation is launched once a traffic congestion is locally detected. This claims that no overhead is needlessly added when traffic situations are normal. If the traffic situation estimation exceeds the congestion threshold th_{cong}, the RSU activates the cooperative traffic data exchange. Likewise, a learning model can be trained to predict traffic jam and density based on historical data or real-time observations. The head node estimates local traffic condition and feeds estimated values into the congestion detection system on UAV_Fog coordinator. With such operation, nodes are able to take appropriate decision since they have the information of interest. Once routing circumstances are defined; e.g., the source node V_n is aware of ground communication status with destination node V_i (e.g., poor link quality or traffic congestion), it decides on UAV-assisted routing to send event information. Details of how the UAV-assisted routing is processed is described in the following subsections.

6.6.2 Preliminaries of UAV-aided routing

The proper UAV-area assignment claims to have trusted-capable UAV relay, however, it does not solve the problem of finding optimal route to destination. A trust check among pair of source and target node is assumed. Each node evaluates the trust level of its target. The path selection is achieved following multi-objective optimization to attain an appropriate balance between trust and QoS metrics. In this subsection we define preliminaries providing the basis for UAVs to discover optimal path through relay(s). The routing preliminaries involve: mode of connection in UAV_Fog network, link breakage/disconnection and alternative path, and post-routing (i.e., complement procedure after routing). We define connection pattern in the IoV-UAV_Fog to shape the routing strategy. Particularly, we outline the mode of communication of central UAV_Fog to other nodes.

Our routing strategy combines selective limited flooding, greedy and multi-objective optimization; considering all available options from the start and finding the optimal one. The optimal path is build through selection of appropriate relay(s). We use the selective limited flooding of packets to simply

try to limit the overhead. The selective limited flooding allows for better targeted packet delivery to a specific group of interested nodes, while taking into account some requirements. We initiate a path discovery by flooding to discover and announce relay' group members. The selective limited flooding with the optimized greedy permits to filter relay candidates for selection problem inputs and avoid failed points. The strategy selects the closest relays that are known for closer location and large access to destination. The k nearest local UAVs (next-hop relay) are determined using a simple Euclidean distance evaluation for the surrounding trusted local UAVs. Based on limited distance to be conditioned (related to sensing distance of local UAV (S^z_{UAV})), the source node V_n can succeed to find the set of suitable relay candidate points. The flooding for j closest relays helps to likely have short path in terms of hop count since packet will be delivered to nearly node to the destination. The node leverages greedy and multi-objective optimization properties to explore best path for routing. The greedy mode involves trying the first option, and if it meets the routing criteria, proceeding with it. If the first option does not satisfy the criteria, the subsequent options are explored sequentially until a suitable one is found. Such mode is often used when the routing criteria are few and not overly complex. By using the hybrid mode, we can leverage advantages of considering multiple criteria and finding global optimum. The greedy principle is adopted for local search of feasible path. On the other hand, multi-objective optimization is involved to refine local decision. Multi-objective optimization focuses on making globally the best or near-optimal path by considering entire problem space and specific objective functions. In multi-objective optimization, the path discovery has plan to find alternative path/post-optimal and path maintenance when there is an instability of the selected path. Pointing that optimization can significantly improve network efficiency. A set of "link test' packets" are used among nodes to investigate availability and performance data. The "link test' packet" can attach V2X information, or instructions in header. Such operation can correspond to a beaconing phase to detect broken link and reporting it, which is beneficial for ongoing path selection. The link test packet is primarily utilized for monitoring, where it will include specific control information allowing the test of performance of communication links between head node and its slave nodes. The request-response interaction allows to filter out the nodes that do not contribute in reliable communication and data delivery. The report of broken links is useful to exclude them

from inputs for optimal path selection session. Each node receiving "link test' request" (LT-REQ) generates a "link test' response" (LT-REP) so that to confirm the communication link. If the timeout (Δ_T) is ended and the node does not hear from its target, a "node' service failure" is reported in order to be avoided as relay node. The node having no LT-REP is noticed to its head node which will decide on sub-node state update to proceed for local route recovery. For instance, the unavailable local RSU for ground nodes and UAVs in area z is reported to control node. The control node decides on the state-update of malfunction RSU. The update can comprise temporary replacement of failed RSU with a local UAV station or may even a virtual RSU (e.g., initialization and synchronization of virtual RSU can be accomplished using coordinates of the failed RSU). Besides, the routing strategy proceeds with the assumption of permanent connections in the network tiers. The permanent connections are exhibited through multicast-routing. The permanent connections can be saved in node' "forwarding class" to be exploited as relay candidate/default alternative path option. For more details, the central UAV_Fog informs the UAVs and the ground nodes about permanent U2F (local UAV-to-central UAV_Fog) and R2F (RSU-to-central UAV_Fog) links. The UAV_Fog coordinator displays persistent C2F (Coordinator-to-central UAV_Fog) and C2A (Coordinator-to-Authority) to central UAV_Fog nodes and authority nodes, respectively. Likewise, the authority node exhibits the persistent R2A (RSU-to-Authority) to RSUs. Moreover, we refer the preliminary we made for forwarding cache. The recent path to most frequent destination (e.g., node which the majority of traffic is routed) can be cached in the "forwarding class" for a time-link-status period (e.g., 10 minutes). Whenever there is a re-need of a route to re-reach a recent target local destination within minutes, the “forwarding class” can be first consulted. The routing discovery process is initialized if there is no route to target destination. Assuming that the path can remain the best over a short period. In fact, that is rare to happen that path quality becomes poor, path point is going to break, or Fog distribution gets updated. We proceed to define the communication way for network tiers. Specifying the connection pattern allows to delineate suitable points for routing and initiate flooding, optimization, and path maintenance. The communication modes are reference points to align a well-defined maintenance strategy, aiding in point of failure identification, backup paths, and smoother maintenance scheduling. In other words, it is through this specification that candidates for relay are strategically pin-

pointed. Given that, a source node will have a global vision of what are the relays to be queried. An assortment of distinct connection modes is given in the following sub-subsection.

6.6.2.1 Modes of connection in IoV-UAV_Fog network

The connection between IoV-UAV_Fog' nodes is done in point-to-point, master -slave manner, or point to multipoint. Head nodes arrange master-slave and head-to-head node communications for figuring reachability index and monitoring QoS of links. The master-slave connection is mainly referred for assigning monitoring. The slave-master connection in a unidirectional fashion primarily relies on direct connection. In this configuration, the direct connection ensures a streamlined flow of commands, information, and task delegation. The slave node can then be directly ordered by its head. Other point-to-point connections encompass notable examples of slave-slave and head-head connections. Ground nodes and UAVs interact via point-to-point and point to multipoint connections. We outline these connections modes (1) locally and (2) when exceeding domain of node, for both connection directions between nodes (i.e., G2A and A2G). In fact, as the connection can transcend the local domain of node, we encounter the need for seamless communication between nodes operating in different areas. The communication way incorporates use of extra relays. The choice between direct and relayed connection follows based on distance-dependent probability of a direct connection. The information on possible points to query such as the k nearest nodes/next-hop relay is used to initiate flooding and optimization. The source node queries trusted k nearest nodes, where k is normalized according to routing protocol's measure of distance/coverage. In addition, the extra relays can act as backup nodes, ready to take over in the event of a failure of primary relays. The extra relays can serve as temporary substitutes during path maintenance. Once the source node is aware that the k nearest nodes seem to be not ideal to provide continuous communication throughout the destination, a backup relay can be used. The head nodes undertake the trace for a path if the destination exits the domain of the slave relay node. The head nodes can be informed about exceed of destination from domain of slave relay node, so that finally the head node will be able to find path if the destination is inside its coverage. In nutshell, the assist from head/super head nodes is conducted in case of (1) link failure/break for

all k nearest nodes, (2) estimation that k nearest nodes/next-hop relay are too far away from the destination, and (3) exceed of local domain of source node. A dedicated extra relay such RSU can be used to help forward data. The RSU is presented as an intermediate node to access the Fog when the source node fails to reach local UAV. The RSU would seek a connection with the central UAV_Fog. Take an example of source vehicle V_n locating best local UAV to route data to destination V_i. In case the vehicle V_n fails to reach target k local UAVs, it requests the assist from the nearest local RSU to reach the Fog service. The RSU provides a central UAV_Fog-connection for the given vehicle V_n. The RSU generates a "local UAV failure report" message to the local central UAV_Fog.

Besides, we look at specific modes of communication with central UAV_ Fog nodes and UAV_Fog coordinator. The coordinator establishes a A2G link via local UAVs. Nevertheless, for a G2A link, the access to the coordinator is through the central UAV_Fog. A direct communication would be perceived only for the authority nodes. Regarding the central UAV_Fog, it operates inter- and intra-area communications. The central UAV_Fog creates an horizontal path for inter-area communication. The inter-central UAV_Fog communication is highlighted in the scenario of cross-domain. The inter communication between central UAV_Fog nodes can be accomplished through UAV_Fog coordinator (C2F) (super head), direct communication (F2F), or semi-central boundary UAV (U2F) (slave of target head). If suitable, the central UAV_Fog can use the boundary UAV which is found in its coverage but matched to target central UAV_Fog. The central UAV_Fog can determine which path would be the optimal among these manners based on distance-dependent probability of a direct connection. Likewise, the central UAV_Fog performs vertically the intra-area communication with the local ground IoV nodes either directly (e.g., with RSU and authority node) or via semi-central UAV. An RSU seeking a G2A connection with central UAV_Fog can resolve it directly or using semi-central UAV or RSU which has permanent connection with the target central UAV_Fog. The central UAV_Fog establishes a A2G link with vehicles via semi central UAV. The central UAV_Fog can request the local RSU to link with the target vehicle when path through semi-central UAV is failed. As aforementioned, our routing strategy suggests that assist from higher-role node is included in the path development step. If a slave node fails to execute the routing request, it forwards that request to the higher layer. We illustrate in the remainder of this subsection the assist

from the central UAV_Fog for routing data in case of cross-domain of semi-central UAV. The central UAV_Fog as a stable relay point can handle quickly most of migration cases of ground IoV nodes and failed link through semi-central UAV. In fact, the high speed of the IoV node implies frequent migration from the domain of semi-central UAV to another, may before delivering response-packet. The central UAV_Fog can correct the broken links inside a target zone. A source vehicle V_n in area z selects the local UAV (UAV_j) to route data to destination V_i. The UAV_j searches in its own dataset to figure if the destination V_i is found in its coverage area and estimates its distance from it. The UAV_j is projected to be smart to predict that destination leaves neighborhood in near future and enter the domain of another local UAV, or that use of next-hop UAV relay is unlikely to be ideal for routing to destination. The UAV_j determines the location of the destination V_i within its coverage map. UAV_j is assumed to have appropriate capabilities for processing. Therefore, searching in the database and distance estimation does not take high time. In case when the destination V_i exits the coverage of the UAV_j, the route request will be forwarded to the immediate central UAV_Fog. At this point, the central UAV_Fog determines the location of the destination V_i and decides the best-suited path to route data from V_n to V_i. If the destination V_i moves out the domain of the central UAV_Fog, then the central UAV_Fog hands the route request to the appropriate node point.

6.6.3 Process of UAV-aided routing

The UAV-aided routing adheres to the defined modes of communication. Figure 6.3 illustrates the overall different phases for UAV-aided routing. The routing strategy includes (1) adaptive flooding using route packet, (2) selection of best relay on the basis of a score using multi-objective optimization, (3) path maintenance when there is an instability of the discovered path, and (4) post-routing phase. We believe that these standards contribute to an efficient UAV-aided routing. Noting that, the described routing strategy can be adapted for ground routing components (e.g., RSU will represent the role of UAV relay during ground routing, an authority node can play the role of central UAV_Fog). To have a quick understanding of the path discovery in UAV-aided routing, route packet is forwarded intelligently to UAV forming the selected routing path. Suggesting that the use of extra/next-hop relay is included in the path development step. Selective flooding with optimized

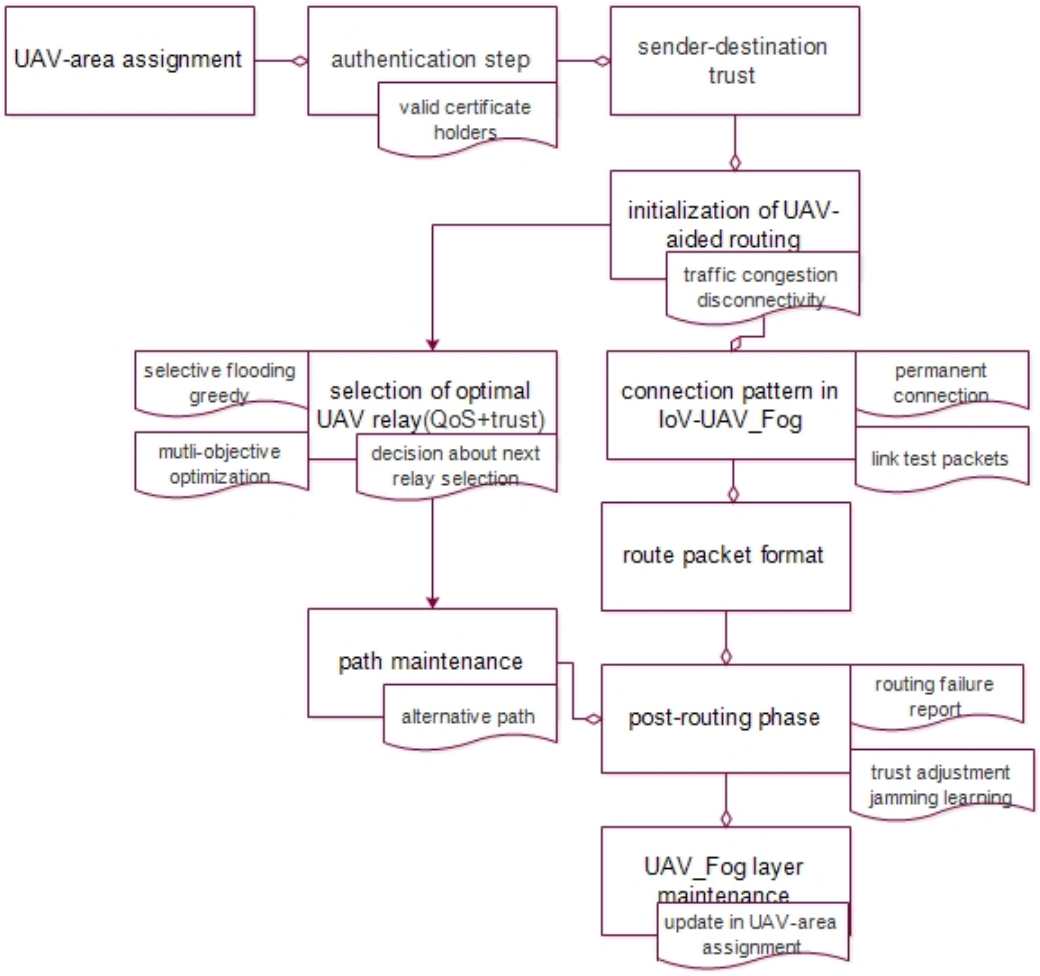


Figure 6.3: General phases of the proposed UAV-aided routing

greedy is applied to select first UAV relay, as well as next-hop relay whenever a directed routing to destination from the first relay may not be the optimal. The routing behavior does not use blindly flooding for finding best path. The selective flooding is applied to find first UAV relay which will calculate the best path to route data to destination. Forming the rest of the path alternates between direct and relayed routing (use of extra/next-hop relay). The routing strategy advocates to gradually build path area segment by area segment in the form of UAVs' succession towards destination. The path decision is made based on criteria combining trust and Qos such as connectivity, delay, and bandwidth. The decision presents a multi-objective optimization maximizing a belief the reflects trust and QoS of relay for routing data. Besides, the routing strategy aims to trace possible paths from the UAV toward destination. We have chosen to adhere modes of connections, use of extra relay and optimization to generate path maintenance and alternative path. When the optimal path disconnects, we can converge to a path which is very close to the optimum without re-initiating the discovery pro-

cess. Figure 6.4 summarizes the process of our UAV-aided routing solution.

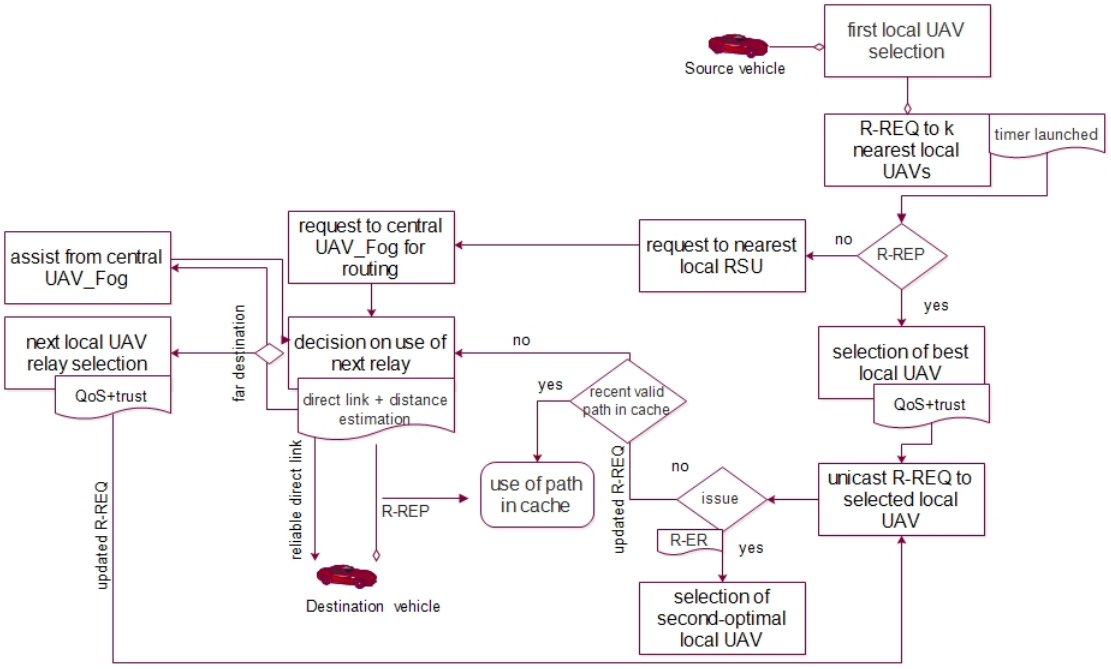


Figure 6.4: Flowchart of the proposed UAV-aided routing

In detail, the delivery of data packet can follow either a direct route or a diverted route mode. It is important to consider the specific requirements and constraints to apply the most appropriate routing mode. The choice of the mode depends on factors like nature of routing criteria, computational capabilities, and trade-off between optimality and forwarding-related computational capabilities. Here, the choice is completed according to estimation of distance from destination and direct link' status. Indeed, a direct link may not be able to guarantee the required routing criteria. For example, a direct link can not be the optimal when network conditions are not favorable due to network configuration/setting, performance considerations, specific use case requirements, security policies, scalability, or other limitations. Therefore, the node should be prepared to assess factors to determine whether a direct link is more desirable (i.e., thresholds on trust and QoS). This is a primitive operation performed by nodes. The direct link gain is first assessed, and then, the choice of proper routing manner is carried. The node can seek a visible direct link to the target. The source is assumed to be able to predict when the link is bad/going to break and take alternate action. The node prioritizes a direct visible reliable link when the distance to the destination node is too small and the node assesses that a direct link is

advantageous and there are no limitations in terms of interference, congestion, security, or limited resources. Otherwise, the node specifies a diverted route using selective flooding and optimization. The flooding is conducted to select best first UAV relay, as well as best intermediate/next-hop relay whenever needed. Each UAV relay is characterized by reputation and QoS metrics. The reputation are evaluated and the malicious candidates are excluded. The QoS is acquired according to the information from coverage and sensing incurred for the UAV. Once the interaction from the first UAV relay is properly achieved, and there is need for an intermediate UAV, the greedy is proceeded to select locally optimal intermediate UAV. The optimization is solved based on maximizing belief function. The higher is the value, better is the selection. A larger value of belief implies a greater node forwarding, due to the larger sensing and computational capabilities. This process is repeated by the selected intermediate UAVs until reaching the destination. The flooding is applied to select intermediate/next-hop relay when the first relay estimates that destination stills too far, or it perceives limitation for a direct link to destination. The flooding generates a "route request" (R-REQ) packet, a "route reply" (R-REP) packet and a "route error" (R-ER) packet. R-REQ packet and R-REP packet carry the information of discovered path, while R-REP packet carries path failure information. Every R-REQ packet has a period to respond (Δ_T), and when the period expires, the response to R-REQ (R-REP) packet is discarded. The source node starts the first UAV relay selection. The candidates for UAV-relay are the k trusted nearest local UAVs that can cover the destination (sub-subsection 6.6.2.1). The source node dismisses reported failed UAVs. The source node floods R-REQ to k nearest local UAVs. When the R-REQ is sent, a timer is launched to have a R-REP. If there is no proper R-REP from all k nearest local UAVs, the source node consults the nearest local RSU to reach the local central UAV_Fog for routing to destination. The nearest RSU is the local RSU with low delay with the source node. It can be explained by RSU_u^{zt}. The closest RSU to source node at $time^t$ can be determined by sending beacon packets.

$$RSU_u^{zt} = arg_{u \in U} mindelay^t_{V_n, RSU_u} \tag{6.8}$$

where U is the set of neighboring RSUs and $delay^t_{V_n, RSU_u}$ is the delay between the node V_n and the RSU at $time^t$. Pointing that, range measurement can be used as an alternative to beacons. The source node can can periodically perform distance measurement or rely on signal strength indicator to identify the nearest RSU. In other respect, the local UAVs that response to the

R-REQ within the timer period Δ_T are considered as inputs for selection of best relay. At this point, the source node starts the data forwarding and send unicastly a R-REQ to the best local UAV. The R-REQ is enclosed in the header of data packet. This R-REQ continues to be forwarded toward destination. In turn, the selected local UAV schedules its transmission and decides on direct transmission to destination or designating a next-hop relay towards destination direction. If the local UAV becomes aware that destination stays far away and that it is not able to provide ideal direct transmission to destination, an intermediate UAV will be queried using selective flooding. The intermediate UAV is determined by iterating over the k UAV relay candidates. The optimization is executed to determine whether an UAV intermediate candidate obeys routing rules combining trust and QoS metrics. The optimization can track the post-optimal solutions. The generated solutions are sorted in the R-REQ packet according to their metric (B) in decreasing order. In other respect, the local UAV opts for a direct transmission to destination as it believes to guarantee a required reliable link quality. As the destination lies in the coverage of the local UAV and the distance from the destination is fairly close, the local UAV is allowed to fly directly towards the destination and disseminates the data packet. This helps to construct a short path in terms of less relays and reduced delay. The route-mode decision process continues until receiving R-REP from destination.

In addition, the maintenance is proceeded if noticing a response failure from the optimal UAV relay. An alternative path is investigated following modes of connection in IoV- UAV_Fog (sub-subsection 6.6.2.1). When the optimal UAV fails, the backup UAV relay is in the second optimal solution. An extra relay (head node) is used in case of issue with possible alternative solutions. A request for assist from the central UAV_Fog is sent in following cases: (1) the UAV relay finds that use of intermediate relay appears to be not efficient (i.e., very distant from destination, (2) case of link break for all alternative relays, or (3) case of exceed of destination from domain of the local UAV. The path maintenance process generates "route error" packet in order to be utilized for Fog service maintenance. The ID stack of the failed relay is copied in the "route error" packet. The "route error" packet is sent to notify the central UAV_Fog about service failure from local UAV. In case of disconnection of path back to the source node, a R-ER packet has to be generated to the central UAV_Fog. The central UAV_Fog undertakes the work to recover the path to the source. The path discovery phase is

completed with a post-routing phase. Indeed, the maintenance in UAV-Fog layer needs to be performed to solve quality and security goals on both IoV user side and UAV-Fog side. The maintenance in UAV-Fog layer can make up the shortcoming of failed path for the UAV-aided routing. Updating the UAV-Fog service is quite important because we need to know about path state when we route data again. The maintenance of UAV-Fog comes in the post-routing phase. It includes report of failed nodes, evaluation of trust on event message, incentive of reputation of involved nodes in path discovery, and update on UAV-area assignment. As such, the destination can assess the trust of received message for message utility check. The destination can rate the message through considering distance between message sender and event location. The destination can accept only the message with high trust and uses the reliability of message to validate the reputation of the message sender. The failed local UAV is traced and reported to central UAV_Fog using R-ER packet. The central UAV_Fog observes coverage (i.e., position and connectivity) and sensing capabilities of failed local UAV to distinguish whether the local UAV is malicious or it is just influenced by the malicious neighboring. If the local UAV is maintaining its performances, the central UAV_Fog suspects a possible jamming from neighboring of local UAV. A neighboring vehicle is mostly considered to be jammer when the last UAV relay fails. If the local UAV fails for two times, it is mostly regarded to be jammer. The trust evaluation is applied as penalty when learning malicious behavior. For example, if a local UAV relay is classified as jammer or it shows malicious behavior (e.g., crossing of matched area), or it fails to perform its routing duties properly (e.g., poor link quality/lose of connectivity, poor learning), it receives a penalty, resulting in a decrease in its reputation value. The trust evaluation initiates an update of UAV-area list with the position of local UAV. If the local UAV deviates from its normal behavior, it will be redeployed because it can no longer provide reliable UAV-Fog service. The redeployment refers to switching on/off status and matching to another area.

6.6.4 Route packet format

The fields of R-REQ are depicted in figure 6.5.(a). The ID fields define coordinates of source node and target destination (node-ID, area-ID (matched head-ID, central UAV_Fog-ID)). The timer field indicates the required period to send back aR-REP packet. The transited segment field corresponds to the

traversed optimal path as a list of used relays toward destination. The transited segment field contains required information to make the reverse path. The transited segment field involves ID stack of used relay (relay-ID, area-ID, and matched super head-ID). A Req- ID field is used to avoid route request duplication in flooding for path discovery process. Initially, the transited segment field is empty. A modification has to be done in the transited segment field to indicate the selected relay. When source node receives R-REP and decides the local UAV relay, it adds the relay-ID stack in the transited segment field (relay-ID, area-ID, matched central UAV_Fog-ID). The selected local UAV will in turn verify ID field and coordinates of destination (area-ID, direction, average speed) to check whether area-ID of destination belongs already to its area, and decide about rest of path. The selected local UAV takes into consideration the timer field. The ID stack of requested intermediate UAV will be enclosed in the transited segment field. The local UAV can have alternative routing paths to destination that are stored in backup path field. The R-REQ continues to be forwarded toward destination. Figure 6.5.(b) illustrates R-REP format. The R-REP includes Rep-ID field, ID fields of source and destination, and relay-ID stack field (relay- ID, area-ID (matched head-ID, central UAV_Fog-ID)). The path point that replies to the R-REQ adds its ID and coordinates to the R-REP packet. The details about failure/break of path toward destination are exported and summarized in R-ER packet (figure 6.5.(c)). As aforementioned, the R-ER packets are exploited for maintenance phase. The R-ER contains failed relay ID stack and area-ID fields (matched head-ID, central UAV_Fog- ID). At the generation of a R-ER, the backup path field is consulted to investigate other path. After that, the central UAV_Fog-ID in the R-ER is consulted to report the failure. We exemplify the routing behaviors of flooding and forwarding as follows. As we explain earlier in subsection 6.6.1, there is cases to initiate the UAV-aided routing in the IoV. The vehicle V_1 located in the area z wants to establish a path towards vehicle V_2 and inform about an event. We assume that the vehicle V_1 trusts the destination V_2 i.e., V_1 experienced exchange with V_1. The vehicle V_1 perceives traffic situation and prediction of ground connection status. The vehicle V_1 requests UAV-aided routing. It floods to k nearest local UAVs to find the first relay. The vehicle V_1 selects the best local UAV (UAV_1) from the local UAVs that respond within the timeout period Δ_T. Here, the vehicle V_1 does not require aid from the nearest local RSU as it received proper response R-REP and found the first UAV relay. The

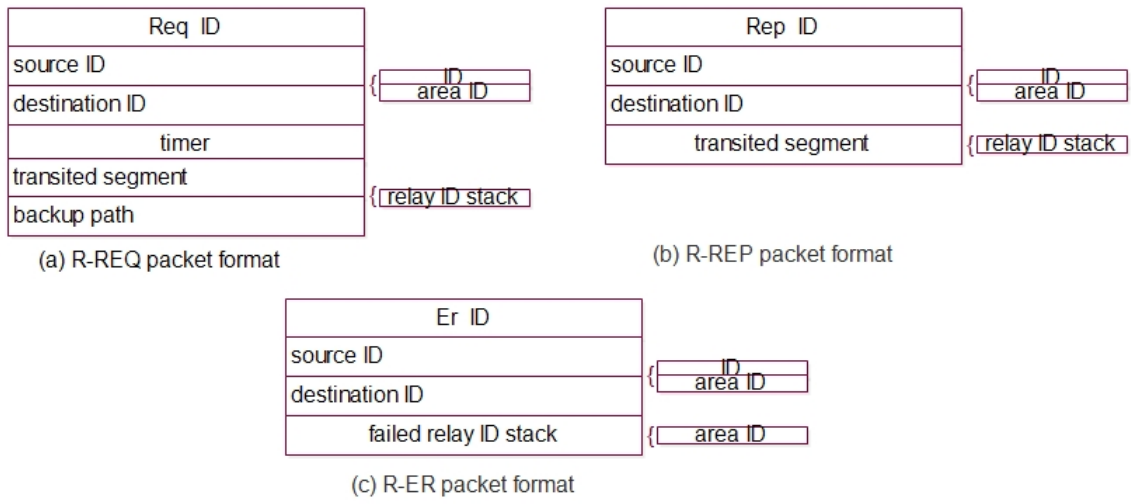


Figure 6.5: Route packet format

vehicle V_1 unicasts R-REQ to the UAV_1 to start the data forwarding. The ID stack of UAV_1 is added to transited segment field in the R-REQ. checks first its forwarding cache about a recent valid path to vehicle V_2. The UAV_1 resolves a path to destination V_2 as no path to V_2 exists in the forwarding cache. Supposing that the vehicle V_2 is located fairly away from the UAV_1. The UAV_1 estimates the coordinates of target destination V_2 using R-REQ. The UAV_1 perceives that V_2 is located in same area z but at distant place. The UAV_1 figures out that it needs support from another local UAV, since a direct routing may not be the optimal. The UAV_1 requests an intermediate local UAV relay to reach V_2. The UAV_1 floods to the k local UAVs located near to destination V_2. Afterward, the UAV_1 selects the best intermediate local UAV (UAV_2) according to trust and QoS metrics. (The UAV_1 does not require aid from the central UAV_Fog as it had proper response R-REP and found the intermediate local UAV relay). The UAV_1 encloses the ID stack of UAV_2 in the R-REQ for a patch back. Finally, the UAV_2 adopts the distance-dependent probability of a direct routing to decide whether to route directly to destination V_2. The UAV_1 has alternative paths (e.g., UAV_3, UAV_4) to be used if the UAV_2 is not obeying the routing rule. If so, the post-optimal local UAV will be automatically selected from the alternative local UAVs in the backup path field of R-REQ. The UAV_1 will send R-ER about the failure of UAV_2 to the central UAV_Fog. By this way, the central UAV_Fog manages the state of UAV_2 parameters and proceeds for maintenance. The intermediate local UAV repeats the process and the R-REQ continues its transition

until reaching R-REP from destination V_2.

6.6.5 Multicast routing

The multicast communication involves dissemination of event detection messages, or delivery of report about node/link test failure or data for monitoring. The central UAV_Fog acts as the root of the multicast communication and is responsible for selecting the optimal path to reach all the desired destinations. The central UAV_Fog constructs the multicast tree that connects it to all the destination nodes. The central UAV_Fog solves the set of optimal paths to the destination nodes. Along each selected path, the central UAV_Fog identifies suitable local UAV relays. The selected paths and intermediate local UAV(s) form the structure of the multicast tree. Each intermediate local UAV becomes a branch node in the tree, connecting the central UAV_Fog (root) to the destination nodes (leaves). Once the multicast tree is formed and the paths are selected, the central UAV_Fog forwards the multicast data along the determined paths. Regarding that various multicast requests may reach the central UAV_Fog, these requests will be served according to priority (e.g., application type: firefighting, ambulance, police). We point that the packet format in multicast routing includes "application type" as additional information. A source node can send a request to the local central UAV_Fog when it wants to send data to a set of destinations. The request involves ID-stack of destinations and application type. The central UAV_Fog verifies whether the target destinations are found in its domain or not. If so, it constructs the multicast tree. Otherwise, it forwards the multicast request to another central UAV_Fog that is matched to same area as destinations. Finally, the central UAV_Fog sends the data to the local UAVs belonging to the created multicast tree. Besides, the central UAV_Fog manages phase of joining an active multicast session. The central UAV_Fog deletes destination node from the multicast session node if it exits the entire IoV.

6.7 Metrics for optimal path selection

The routing protocol uses reputation of relay and connectivity quality. The reputation is assessed to trust metrics; we detail each of the metrics in the

following. The QoS metric is acquired according to information obtained from latency, connectivity belief, and bandwidth. The connectivity belief refers to the probability of keeping the communication link.

6.7.1 QoS metric

The path prefers an UAV relay with strong communication range and signal strength since an UAV outside the range may experience connectivity issues. Therefore, we take into account LoS and SINR to be calculated. Although there is no restriction for the length of a path (it can be long or short depending on the way to go to the end destination), we try to solve a short path that can provision QoS in terms of latency by requesting closest relays. The QoS metric is assessed using below equations.

- Latency

The latency represents the time delay in transmitting the R-REQ. It denotes the expiration time of a R-REQ. The request for UAV-aided routing is completed within the time frame (Δ_T). Let Δ_T and $\bigwedge_T$ the expected time and actual time taken by the UAV relay to reply to a R-REQ. The latency is calculated as

$\text{Latency}_{UAV_j^{V_n^z}} = \Delta_T - \bigwedge_T$ (6.9)

- Connectivity index

The connectivity index considers LoS and SINR to calculate the probability of establishing a high link quality. The UAV relay should have a high probability of keeping connection [**165**].

$$LQI(CL_{UAV_j^{V_n^z}}) = PLoS(\theta_{UAV_j^{V_n^z}}$$

$\varkappa_{UAV_j^{V_n^z}}$ (6.10)

Supposing that $CL_{UAV_j^{V_n^z}} \in 0, 1$ is the link with an UAV_j. When a connection is established with UAV_j, $CL_{UAV_j^{V_n^z}}$ is 1, otherwise 0. $LQI(CL_{UAV_j^{V_n^z}}$

denotes the link connectivity indicator of a $CI_{UAV_j^{V_n^z}}$) with UAV_j given its LoS ($\theta_{UAV_j^{V_n^z}}$)$andSINR(\varkappa_{UAV_j^{V_n^z}})$. The SINR quantifies the strength of the desired signal relative to the interference and noise in the channel. The link connectivity indicator is derived by multiplying the probability of LoS and the SINR to provide a measure of the overall connectivity index and quality. The link connectivity indicator takes into account both the likelihood of having LoS conditions and the relative strength of the desired signal compared to interference and noise. The SINR can be obtained as [**165**]:

$$\varkappa_{UAV_j^{V_n^z}} = \frac{pow^r_{UAV_j^{V_n^z}}}{Npow^2 + IF_J} \tag{6.11}$$

where, $pow^r_{UAV_j^{V_n^z}}$ is the received power from the UAV$_j, and$Npow^2 is the noise power of that particular link. IF_J represents the sum of interference received from other ($J1$) UAVs. By using path-loss given in Eq.(6.1) and the optimal transmit power ($\beth_{UAV_j^{V_n^z}}$), received power($pow^r_{UAV_j^{V_n^z}}$) is calculated as [**165**]:

$$pow^r_{UAV_j^{V_n^z}} = 10log_{10}(\beth_{UAV_j^{V_n^z}}) + \beth_{UAV_j^{V_n^z}} - PLoS(\theta_{UAV_j^{V_n^z}}) \tag{6.12}$$

where $\beth_{UAV_j^{V_n^z}}$ represents the fading parameter. The optimal transmit power is calculated by applying an interference constraint. $CL_{UAV_j^{V_n^z}}$ is the connectivity with the UAV_j. We apply the following constraint [**165**]:

$$\text{CL}_{UAV_j^{V_n^z}}\ gain_{UAV_j^{V_n^z}}\ \beth_{UAV_j^{V_n^z}} \leq Th_{IF} \tag{6.13}$$

where $gain_{UAV_j^{V_n^z}}$ is product of the magnitude squared of the channel gain, and Th_{IF} is the interference threshold.

The optimal transmit power $\beth_{UAV_j^{V_n^z}}$ is given by [**165**]:

$$\urcorner_{min} \leq \beth_{UAV_j^{V_n^z}} \leq \urcorner_{max} \tag{6.14}$$

where $\urcorner_{min}$ and $\urcorner_{max}$ are the minimum and maximum transmit power of an UAV_j.

- Bandwidth

The nodes allocate consciously a percentage of bandwidth for routing. The QoS value increases as the relay contributes with higher bandwidth ratio. The relay starts allocating the bandwidth using Eq(6.10). The required bandwidth from a relay can be calculated as

$$Bd_{UAV_j^{V_n^z}} = \frac{dr_{UAV_j^{V_n^z}}}{log_2(1 + \varkappa_{UAV_j^{V_n^z}})} \tag{6.15}$$

where $dr_{UAV_j^{V_n^z}}$ denotes the demanded data rate from the relay, and $\varkappa_{UAV_j^{V_n^z}}$ is the SINR.

6.7.2 Trust metric

Trust evaluation is applied for selecting routing path along with monitoring the trustworthiness of nodes throughout the routing process. The trust evaluation is continued to adjust the reputation of nodes involved in routing process. The reputation of ground node comes from observation of speed feature and packets dropping, while UAV reputation is derived from UAV learning capability and position change. The trust of local UAVs is exhibited to IoV ground nodes by the central UAV_Fog.

- Learning capability

Learning capability is assessed to adjust trust level. The learning capability is utilized to estimate the detection performance associated with accuracy, time and energy. This can make more informed routing decisions where selecting paths that maximize expected performances. A strong learning capability can contribute to better detection of malicious nodes. The learning capability assists in selecting relevant features for malicious nodes detection. A strong learning capability optimizes the loss function and leads to accurate predictions and lower loss function. In turn, minimizing the loss function during training generally leads to a lower false detection rate. An UAV with strong learning capability in terms of time can process and analyze data in a timely manner, enabling faster decision- making. On the other hand, having

a strong learning capability in terms of time means that the UAV can optimize its energy consumption during the learning and decision- making processes. This capability allows the UAV to make efficient use of its resources while still acquiring and processing the necessary data. By optimizing the learning energy, the UAV can extend its operational time, and ultimately optimizes its energy consumption to reduce the need for frequent recharging. By taking Eqs.(6.4)(6.5) (subsection 6.5.3), we can obtain the learning capability index of a local UAV.

- Position change

The trust level of UAVs incorporates their adherence to assigned positions and avoidance of collisions. The position change of a local UAV can raise suspicions regarding its behavior or intentions. By monitoring the position changes of local UAVs, the central UAV_Fog can detect such behavior. When a local UAV, which is expected to follow a specific assigned location, significantly deviates from its assigned location, it is regarded as suspicious. If the local UAV crosses the boundary of its matched area or consistently collides with other UAVs, it raises suspicions about its adherence to the designated rules. The trust value assigned to that local UAV can be decreased as a consequence. As aforementioned in subsection 6.5.3, we associate penalties with decreasing the trust value (Eqs.(6.6)(6.7).

To recap this subsection, the incorporation of trust and QoS in path selection metrics can enhance the chances of dynamic optimized path selection while minimizing the impact of untrustworthy nodes by adapting to changes in the trustworthiness of participating nodes.

6.7.3 Problem formulation for optimal path selection

The selection of the best path is based on maximizing multi-objective belief function that reflects reputation and QoS. We propose solving the following optimization problem.

- Decision variables

The selection of local UAV relay from a set of available candidates can be represented by discrete decision variables. Since the local UAVs are as-

signed to specific areas, each local UAV candidate can be treated as binary variable indicating whether it is chosen as a relay from that particular area or not. $x(UAV_j)$ is a binary decision variable indicating whether the UAV_j is selected as a relay (1) or not (0).

$$Maximize(B) \tag{6.16}$$

Maximizing the belief B refers to maximize the trust metric and the QoS metric.

$$Maximize(Reputation\_Function(x)), Maximize(QoS\_Function(x)) \tag{6.17}$$

We consider a function that maximizes QoS, represented by a nonlinear equation involving bandwidth and latency, while simultaneously maximizing reputation, represented by a non linear trust model that factors in packet dropping rate, learning capability, and position change. The optimization problem in Eq.(6.17) can be categorized as a MINLP. This formulation can integrate the Non-Dominated Sorting Genetic Algorithm II (NSGA-II) to provide post-optimal solutions. Maximizing Reputation_Function(x) and QoS_Function(x) is subject to the following constraints. **C1** Trust constraint. The selected relay should satisfy a minimum reputation threshold (th_{rep}). The reputation is a composite measure incorporating packet dropping rate, learning capability, and position change that are likely to have nonlinear dependencies on the decision variable. The reputation value is influenced by the individual sub-metrics in a complex manner that may not follow a simple linear relationship. Let $D(x)$ the packet dropping rate of local UAV_j, $Learning(x)$ the learning capability of local UAV_j, and $Loc_change(x)$ the position change metric of local UAV UAV_j. The Reputation_Function(x) can be represented as:

$$Reputation\_Function(x) = \alpha_1 * Learning(x)^{\gamma} + \alpha_2 * exp(-\delta_1 * D(x)) + \alpha_3 * exp(-\delta_2 * (Loc_change(x)) \tag{6.18}$$

where α_1, γ, α_2, $delta_1$, α_3, and $delta_2$ are parameters that control the shape and influence of each reputation sub-metric in the Reputation_Function(x). The learning capability is raised to γ and weighted by α_1. The packet dropping rate and position change are exponentiated and weighted by α_2 and α_3, respectively. For the selected relay UAV_j, $th_{rep} =< x(T_{UAV_j})$ <=Reputation_Function(x).

C2 QoS constraint. The minimum requirements for LoS, SINR, latency, and bandwidth for the selected UAV relay should be specified. **C2-1**. Con-

nectivity constraint that ensures that the connectivity index (derived from LOS and SINR) meets the desired threshold ($th_{connectivity}$). ($x(C) >= th_{connectivity}$). **C2-2**. Bandwidth constraint. Supposing the bandwidth of UAV is represented by a continuous decision variable ($x(Bd)$). The bandwidth must satisfy the demanded data rate from the relay (dr) and SINR ($\varkappa$). **C2-3**. Latency constraint. In some cases, the relationship between latency and a specific decision variable may not be linear. We can capture the nonlinear relationship between latency and bandwidth simply as:

$$log(x(Bd)) <= R_{Latency} * th_{latency} \tag{6.19}$$

where $R_{Latency}$ is the rate of increase in latency as bandwidth decreases. By using the logarithmic function, the latency constraint ensures that the bandwidth of the selected relay is sufficient to satisfy the maximum acceptable latency requirement.

6.8 Simulation study and results

6.8.1 Simulation environment and metrics

The optimization part related to obtaining the optimal UAV deployment in this work is assumed before the simulation starts. The routing approach is evaluated using NS-3. The simulation area was divided into two large leaf areas by defining two central UAV_Fog nodes as separating nodes. Each leaf area encompassed the region between the two separating central UAV_Fog nodes. A number of local UAVs were dispatched in each leaf area forming the local small sub-areas. This division permits to incorporate area- assignment constraint. After assuming the optimal location and number for UAV deployment, the mobility model of UAVs was constructed in NS-3 using random waypoint mobility model. The starting and hovering positions of the UAVs were determined. We assumed that the selected waypoints for UAV movement fall within valid range of the assigned area. The mobility of the vehicles was generated using SUMO. Parameters setup is captured in Table 6.2. Considering an urban scenario, the vehicles number augmented from 50 to 350. The maximum speed of vehicles was set to 50 km/h. The range of communication of vehicle was supposed to 300 m. We distributed two authority nodes supposing that authority node logic is created. Likewise for the UAVs, we assumed that coordinator logic is implemented, and node

is created representing the UAV_Fog coordinator. We used up to five local UAVs to provide coverage to the vehicles in each local sub-area. We assume that the trust values of local UAVs are known between each other. A local UAV flied at a low altitude not surpassing 200 m, with a communication range of 1000 m, while a central UAV_Fog hovered at high altitude long endurance with longest communication range up to 2000 m. Supposing that the ground IoV was congested because of an accident. The UAVs were tested as relay nodes to route accident information to ground vehicles. A predefined quota of vehicles were assigned either as accident event' sender (source node) or accident event' receiver (destination node).

Table 6.2: Simulation parameters

Parameters	Values
Simulation time	1500s
Ground IoV nodes	50-350 vehicles
UAVs	1 coordinator, 2 central UAV_Fog, 10 local UAVs
Vehicle speed	0-50 km/h
UAV speed	0-60 km/h
Vehicle range	300m
UAV range	1000m for local UAV, 2000m for coordinator and central UAV_Fog
Mac protocol	IEEE 802.11p
Packet size	1KB

6.8.2 Simulation results and evaluation

We utilized the following metrics for the performance evaluation.

- Packet delivery ratio (PDR)

PDR represents the number of received packets given total number of sent packets. The more the PDR value is, the better the routing strategy performance is.

- End-to-end delay (EED)

A lower EED delay indicates faster packet delivery and reduced latency.

- Average number of hops

The hops number provides insights into the average path length or distance traveled by packets successfully handed over from the source to the destination.

- Overhead

The overhead ratio corresponds to additional routing packets during packet delivery to destination node. A lower overhead indicates a more efficient routing strategy.

Performances of our proposal are compared to: (1) UAV-assisted routing protocols such in [**156**] and [**158**], and (2) UAV relay-assisted routing protocols, such in [**163**]. Comparison with UAV-assisted routing protocols allows to understand the effect of utilizing UAVs as relays, while the UAV relay-assisted routing protocols are compared as the baseline. Figure 6.6 shows the PDR ratio according to vehicles density. We note that the PDR of all proposals increased with the increase of vehicles density. This is because there is more chances for messages to be sent as the number of nodes is large. Besides, we can see from this figure that the probability of success transmissions is higher through UAV relay than through ground nodes. Our proposal achieves the highest PDR compared to [**156**][**158**] and [**163**]. Indeed, this can be explained by the consideration of the metric of packet dropping to discover the appropriate path. The check of packet propriety of relay allows to minimize the packet loss and improve the transmission to destination node. Also, the use of QoS metric involving connectivity index and bandwidth in path selection contributes to higher PDR. Besides, the exclusion of reported failed relays can ensure a certain reliability of the data transmission. The selective flooding avoids to use relay that fails packet. Besides, we can refer that alternative path and permanent connection constraints can be also an advantageous factor increasing the PDR.

The EED is shown in figure 6.7. Our proposal fluctuates around an acceptable delay value. The delay of our proposal is stable versus the increase of number of vehicles. This is because of the density of ground vehicles has nothing to do with the selection of optimal path, and only UAV with a certain required latency is used to do the routing correctly. Our proposal

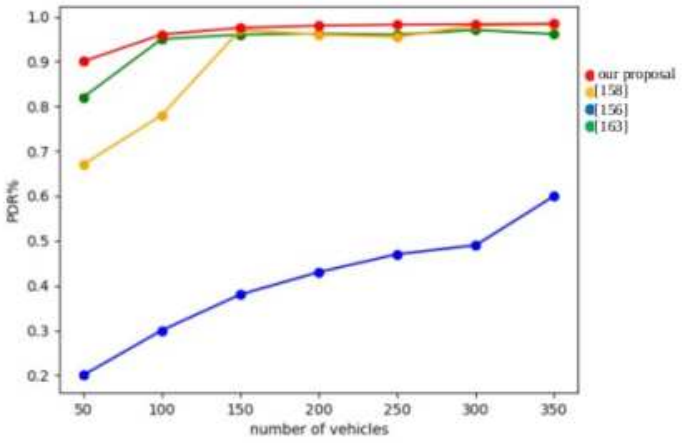


Figure 6.6: PDR

uses timers to raise response constraint and selects the appropriate UAV relay considering the metric of latency. Furthermore, routing strategy that ensures alternative path allows to avoid re-initializing the path discovery path. Also, proximity of UAV relay to source and destination can result in shorter routing paths and reduced transmission delays. By minimizing the distance traveled by data packets, the EDD delay can be effectively minimized. However, as expected, the path selection process in our proposal generates additional extra times. The time taken for the optimization to converge and select the best relay is added to the overall end-to-end. The work [**158**] achieved the lowest delay. This can be explained by the fact of using a prediction method to estimate the expiration time of discovered path. The use of the SCF method in [**163**] can be a disadvantageous factor for the EDD delay. Also, in [**156**][**158**] the packets were sent based on physical mobility of vehicles. Hence, the EDD delay depends highly on the mobility model of vehicles. Instead, UAV as relay can bypass mobility obstacles and provide a data transmission with reduced EDD delay. Pointing that, the EDD delay can be reduced with an optimized "forwarding cache" that allows for faster routing operation. Figure 6.8 depicts the average hops number according variable vehicles density. A larger distance between source node and destination node can cause an increased hops number. The distance between source and destination gets larger when the vehicles number augments. This can lead to a high number of hops. Also, the selection of path points without taking into account the distance separating each path point from destination can lead to an increased number of hops. Figure 6.8 illus-

trates that our proposal minimizes the hops number by avoiding inessential hops. In our proposal, the established paths were short with hop count between 2 and 3 in average. This is due to that our protocol raises the direct link constraint. The use of extra/next hop relay is decided according to estimation of distance from destination and direct link' status. Besides, our protocol assumes the proper location and number of UAV for the delivery of packets which avoids placing inessential UAV relay. A proper UAV coverage facilitates the use of local UAVs for short data delivery and can reduce the hops number. The average hops number in [**158**] was high, wherein it increases almost for the higher densities. This is because the vehicles as relay nodes had limited communication range and therefore they required multiple hops to accomplish the data transmission. Instead, the UAV relay makes easier to control multi-hop forwarding. Distinct observation is distinguished for [**163**], wherein the hops number remains stable under high densities. The outcome of the work [**163**] is close to our proposal. Generally, the hops number increases when increasing the number of UAVs and vehicles. Figure 6.9 represents overhead with different density of vehicles.

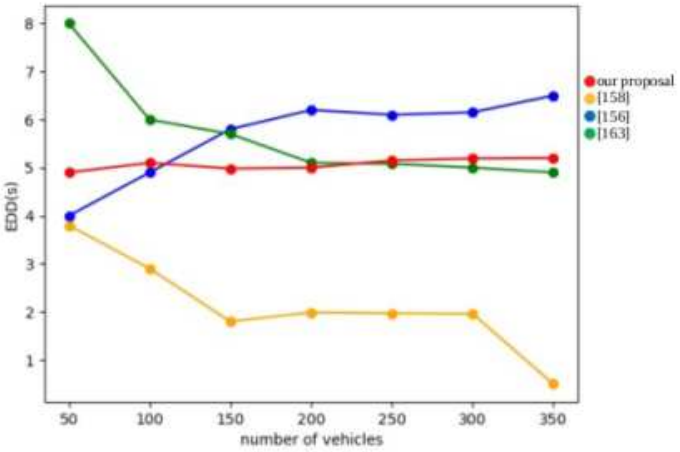


Figure 6.7: EDD delay

Our proposal generates route request, route response, and error packets as necessary control packet to calculate the scores of each relay. The overhead in our proposal is acceptable compared to other reactive protocols. The overhead refers mainly to additional control packets during path discovery and periodical exchange of "link test' packets". However, as a reactive routing protocol, our proposal generates small overhead thanks to the selective limited flooding, and the path maintenance strategy that reduces the flooding during path breakage. The overhead in [**156**] increased with the large

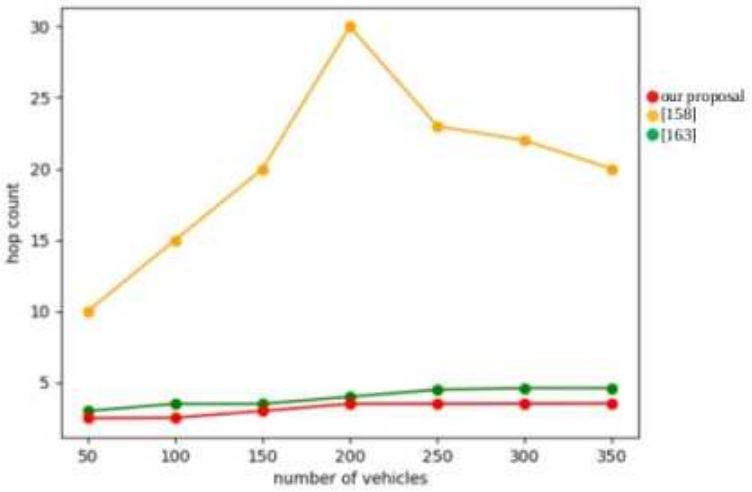


Figure 6.8: Average number of hops

density of vehicles, because the work was based on flooding route packets between all UAVs for path discovery. The overhead in [**158**] was significant due to control packets exchange during path discovery. [**163**] limited effectively the overhead by using the utility function. In our proposal, the overhead of UAV relays is low and stays approximately constant since we used control packets strategy rather than hello packets flooding strategy. The UAV-assisted routing protocols have important overhead particularly when the number of vehicles goes up, which is explained by the overhead associated with routing information dissemination between UAVs and all vehicles. Pointing that an appropriate number of UAV relays compared to vehicles can limit the overhead.

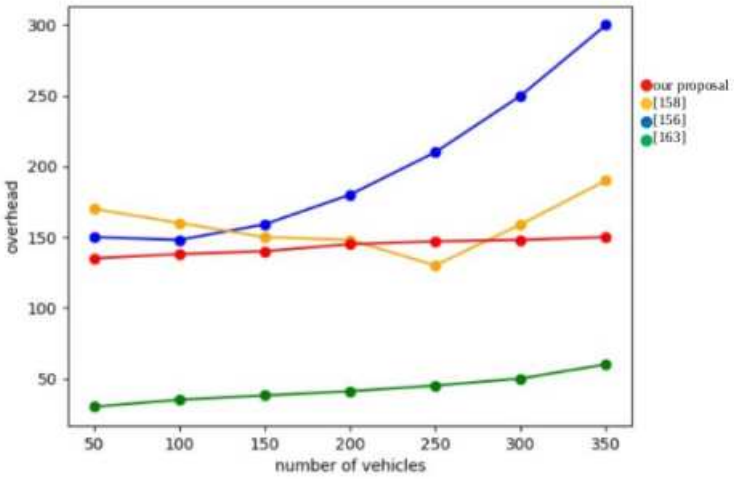


Figure 6.9: Communication overhead

6.9 Conclusion

This chapter takes care of routing in the IoV. We introduced a four layered IoV-UAV_Fog architecture to permit favorable setup for UAV-aided routing. The UAVs were strategically matched with geographic areas. Semi-central UAV_Fog nodes layer and central UAV_Fog layer were the workers of the UAV-aided routing. The routing strategy allowed to choose the optimal UAV relay(s) node forming the best path towards destination using composite QoS and trust metrics. The path maintenance was also defined in this stage permitting for alternative paths. The UAV_Fog-aided routing made use of selective flooding, greedy, and optimization concepts. Simulation results proved the efficiency of our heterogeneous routing model according to PDR, EDD delay, hop count, and overhead.

Chapter 7

Conclusion

7.1 Summary of Contributions

Research works have paid considerable attention for the IoV technologies in recent years. However, the IoV communication system comes with security risks. This thesis addresses the obstacles to providing a secure IoV framework that maintains the QoS while mitigating communication system threats. The proposed solutions leverages trust management along with emerging technologies like Blockchain, SDN, and UAV-Fog computing because of their exciting properties in supporting security and QoS in a critical network such the IoV.

Chapter 2 provided the background knowledge on the IoV network. It also highlighted security challenges. In chapter 3, we conducted a review of existing trust-based approaches in the context of vehicular networks. We provided a taxonomy of reviewed works based on used tools. We shed light on works that applied trust management with AI and emerging technologies. We proposed to rely on trust management with Blockchain, clustering, and SDN for our two first security solutions.

Throughout chapter 4, we built a Blockchain and trust-based approach for the IoV. Our objective was to address the security with focus on trust data integrity while meeting QoS. In the first, we proposed to rely on reputation and location proximity metrics to establish the trust within the network. We applied our trust scheme on two layer Blockchain architecture to preserve trust data. The Blockchain consisted of local Blockchain to store local trust, and global Blockchain to exhibit global trust information. In the second, we extend our solution by applying the clustering to more consider the QoS in terms of continuity and scalability. Our proposal was validated in terms of detection performances and QoS through simulation.

In chapter 5, we proposed an IDS that uses the combination of trust metrics, federated learning, and SDN structure to detect dishonest nodes in the IoV network. We aimed to investigate the collaborative learning of nodes trust in an SDN-IoV. In the first the SDN controllers were the workers of the local detection of dishonest nodes. They were based on node properties-related trust metrics. Afterwards, our work targeted more localized and responsive detection through placing the local detection on trustworthy clusters heads while hosting the global detection model on SDN controller.

Pointing that the formation of trustworthy clusters heads integrated the multi cluster head concept to better consider the stability performance. The results showed good detection and QoS-related performances for our proposal.

Finally, we addressed the IoV security from the perspective of routing in chapter 6. We presented an UAV-aided routing solution that applied the trust metric for a reliable IoV network architecture. We introduced an IoV network model consisted of UAV_Fog relay nodes to provide strong support for the routing. The fundamental idea was to find the optimal path to destination through selecting the optimal UAV as relay node with the goal of maximum trust and maximum QoS through multi-objective optimization. Our routing solution proved to be reliable according to the results obtained regarding PDR, EED, and hop count.

7.2 Perspectives

The thesis' contributions provided solutions for improving the IoV security while considering the QoS. As part of future research, we suggest a few aspects related to trust management that can be investigated to further enhance the presented solutions.

- Considering the global IoV ecosystem

We were mainly based on the vehicles as trustee and trustor entities to define trust mechanism within the IoV environment. An efficient trust management model is recommended to include various dimensions like the human dimension to fulfill the IoV context and lead to realistic models.

- Trust negotiation

We focused in our solutions on a calculation process of the trust. This process can be alleviated through the use of trust negotiation mechanism. Further works can consider the trust negotiation through defining a procedure describing the requirements to reach the required trust level. When a vehicle or a human entity needs to join the IoV network, a negotiation process is triggered in order to achieve a common agreement with the central entity

(for example, the ITS entity or the RSU). The trust negotiation relies on the exchange of a set of credentials in order to consider the entity as trustworthy. Getting a higher trust level requires exchanging more sensitive credentials.

- Trust bootstrapping

(1) Trust bootstrapping (i.e., computing real initial trust value) may need further research; indeed, a random initial trust value was assigned for the newly encountered nodes, yet, the assumed trust value may do not match with the real trust value, hence more research is required to determine the precise initial trust value. (2) Reputation computation can be more handled. Actually the reputation is a broadly used metric. A fast reputation value computation is required. Reputation computation scheme should take into account many different factors that are related to the reputation-based metric.

- Data perception trust

Data perception trust should be more investigated. It refers to data trust collection and pre-processing. Therefore, the inspection of the quality of sensed data is inevitably crucial for supporting IoV trust. Ongoing research need to pay attention to trust properties in the IoV physical perception layer (i.e., sensor preciseness, persistence, data aggregation efficiency). We could additionally take into consideration the hardware platform issues (e.g., sensors security) when embed a trust management solution.

Bibliography

[1] P.P. Ray, A survey on internet of things architectures, J. King Saud Univ.-Comput. Inf. Sci. 30 (3) (2018) 291–319

[2] Sharma, S. (2019, October). Vehicular ad-hoc network: An overview. In 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) (pp. 131-134). IEEE.

[3] M.K. Priyan, G.U. Devi, A survey on internet of vehicles: applications, technologies, challenges and opportunities, Int. J. Adv. Intell. Paradigms 12 (1–2) (2019) 98–119.

[4] Hussain, R., Zeadally, S. (2018). Autonomous cars: Research results, issues, and future challenges. IEEE Communications Surveys Tutorials, 21(2), 1275-1313.

[5] H. Zhou, W. Xu, J. Chen, W. Wang, Evolutionary V2X technologies toward the internet of vehicles: Challenges and opportunities, Proc. IEEE 108 (2) (2020) 308–323.

[6] A. Hbaieb, O.B. Rhaiem, L. Chaari, In-car gateway architecture for intra and inter-vehicular networks, in: 2018 14th International Wireless Communications Mobile Computing Conference, IWCMC, IEEE, 2018, pp. 1489–1494

[7] Hayes, J. (2020). Hackers under the hood: It's been five years since the first reports of car hacking emerged, but despite progress in vehicle protection standards, automotive cyber-security remains on high alert. Engineering Technology, 15(3), 32-35.

[8] Blaze, M., Feigenbaum, J., Lacy, J. (1996, May). Decentralized trust management. In Proceedings 1996 IEEE symposium on security and privacy (pp. 164-173). IEEE.

[9] Khatri, P., Rajvanshi, P. R. (2020). A relative study about mobile ad-hoc network (MANET): Applications, standard, protocols, architecture, and recent trends. In IoT and Cloud computing advancements in vehicular ad-hoc networks (pp. 156-173). IGI Global.

[10] J.B. Kenney, Dedicated short-range communications (DSRC) standards in the United States, Proc. IEEE 99 (7) (2011) 1162–1182, http://dx.doi.org/10.1109/ JPROC.2011.2132790.

[11] Abboud, K., Omar, H. A., Zhuang, W. (2016). Interworking of DSRC and cellular network technologies for V2X communications: A survey. IEEE transactions on vehicular technology, 65(12), 9457-9470.

[12] Du, Z., Wu, C., Yoshinaga, T., Yau, K. L. A., Ji, Y., Li, J. (2020). Federated learning for vehicular internet of things: Recent advances and open issues. IEEE Open Journal of the Computer Society, 1, 45-61.

Abu Talib, M., Abbas, S., Nasir, Q., Mowakeh, M. F. (2018). Systematic literature review on Internet-of-Vehicles communication security. International Journal of Distributed Sensor Networks, 14(12), 1550147718815054.

[14] Garg, T., Kagalwalla, N., Churi, P., Pawar, A., Deshmukh, S. (2020). A survey on security and privacy issues in IoV. International Journal of Electrical Computer Engineering (2088-8708), 10(5).

[15] Sun, Y., Wu, L., Wu, S., Li, S., Zhang, T., Zhang, L., ... Cui, X. (2017). Attacks and countermeasures in the internet of vehicles. Annals of Telecommunications, 72, 283-295. 2019.

[16] Stevens, G., Bossauer, P., Jakobi, T., Pakusch, C. (2017). Second Dashboard: Information Demands in a Connected Car. Mensch und Computer 2017-Tagungsband

[17] Wang, X., Li, Y. (2019). Content retrieval based on vehicular cloud in internet of vehicles. IEEE Transactions on Computational Social Systems, 6(3), 582-591.

[18] Gerla, M., Lee, E. K., Pau, G., Lee, U. (2014, March). Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. In 2014 IEEE world forum on internet of things (WF-IoT) (pp. 241-246). IEEE.

[19] Pasha, M. (2015). Vehicular Cloud Computing: leading towards tomorrow's Internet of Vehicles. Journal of Wireless Sensor Network, 2.

[20] Silva, L., Magaia, N., Sousa, B., Kobusińska, A., Casimiro, A., Mavromoustakis, C.X., ... De Albuquerque, V. H. C. (2021). Computing

paradigms in emerging vehicular environments: A review. IEEE/CAA Journal of Automatica Sinica, 8(3), 491-511.

[21] Zhao, W. (2021, January). A survey on fog computing applications in internet of vehicles. In 2021 2nd International Conference on Computing and Data Science (CDS) (pp. 27-32). IEEE.

[22] Jiacheng, C., Haibo, Z. H. O. U., Ning, Z., Peng, Y., Lin, G., Sherman, S. X. (2016). Software defined Internet of vehicles: architecture, challenges and solutions. Journal of communications and information networks, 1(1), 14-26.

[23] Smida, K., Tounsi, H., Frikha, M., Song, Y. Q. (2019, June). Software Defined Internet of Vehicles: a survey from QoS and scalability perspectives. In 2019 15th International wireless communications mobile computing conference (IWCMC) (pp. 1349-1354). IEEE.

[24] Hakimi, A., Yusof, K. M., Azizan, M. A., Azman, M. A. A., Hussain, S. M. (2021). A Survey on Internet of Vehicle (IoV): A pplications Comparison of VANETs, IoV and SDN-IoV. ELEKTRIKA-Journal of Electrical Engineering, 20(3), 26-31.

[25] Mollah, M. B., Zhao, J., Niyato, D., Guan, Y. L., Yuen, C., Sun, S., ... Koh, L. H. (2020). Blockchain for the internet of vehicles towards intelligent transportation systems: A survey. IEEE Internet of Things Journal, 8(6), 4157-4185.

[26] Azam, F., Biradar, A., Priyadarshi, N., Kumari, S., Tangade, S. (2021, December). A review of blockchain based approach for secured communication in Internet of Vehicle (IoV) scenario. In 2021 Second International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE) (pp. 1-6). IEEE.

[27] Kumar, S., Velliangiri, S., Karthikeyan, P., Kumari, S., Kumar, S., Khan, M. K. (2021). A survey on the blockchain techniques for the Internet of Vehicles security. Transactions on Emerging Telecommunications Technologies, e4317.

[28] Herbadji, A., Goumidi, H., Harbi, Y., Medani, K., Aliouat, Z. (2020). Blockchain for internet of vehicles security. Blockchain for Cybersecurity and Privacy: Architectures, Challenges, and Applications, 159.

[29] De La Fortelle, A., Qian, X., Diemer, S., Grégoire, J., Moutarde, F., Bonnabel, S., ... Sjöberg, K. (2014, September). Network of automated vehicles: the AutoNet 2030 vision. In ITS World Congress.

[30] http://autopilot-project.eu/

[31] H. Xia, S.S. Zhang, Y. Li, Z.K. Pan, X. Peng, X.Z. Cheng, An attack-resistant trust inference model for securing routing in vehicular ad hoc networks, IEEE Trans. Veh. Technol. 68 (7) (2019) 7108–7120.

[32] S. Dahmane, C.A. Kerrache, N. Lagraa, P. Lorenz, WeiSTARS: A weighted trust- aware relay selection scheme for VANET, in: 2017 IEEE International Conference on Communications, ICC, IEEE, 2017, pp. 1–6.

[33] K. Rostamzadeh, H. Nicanfar, N. Torabi, S. Gopalakrishnan, V.C. Leung, A context- aware trust-based information dissemination framework for vehicular networks, IEEE Internet Things J. 2 (2) (2015) 121–132.

[34] F. Ahmad, A. Adnane, C.A. Kerrache, V.N. Franqueira, F. Kurugollu, Trust management in vehicular ad-hoc networks and internet-of-vehicles: Current trends and future research directions, in: Global Advancements in Connected and Intelligent Mobility: Emerging Research and Opportunities, IGI Global, 2020, pp. 135–165.

[35] R. Hussain, J. Lee, S. Zeadally, Trust in VANET: A survey of current solutions and future research opportunities, IEEE Trans. Intell. Transp. Syst. (2020).

[36] R. Iqbal, T.A. Butt, M. Afzaal, K. Salah, Trust management in social internet of vehicles: Factors, challenges, blockchain, and fog solutions, Int. J. Distrib. Sens. Netw. 15 (1) (2019) 1550147719825820.

[37] I. Souissi, N.B. Azzouna, T. Berradia, Trust management in vehicular ad hoc networks: a survey, Int. J. Ad Hoc Ubiquitous Comput. 31 (4) (2019) 230–243.

[38] J.M. Qurashi, Survey on risk-based decision-making models for trust management in VANETs, in: Secure Cyber-Physical Systems for Smart Cities, IGI Global, 2019, pp. 52– 73.

[39] S. Sumithra, R. Vadivel, An overview of various trust models for vanet security establishment, in: 2018 9th International Conference on Computing, Communication and Networking Technologies, ICCCNT, IEEE, 2018, pp. 1–7.

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

[41] Siddiqui, S. A., Mahmood, A., Sheng, Q. Z., Suzuki, H., Ni, W. (2021). A survey of trust management in the internet of vehicles. Electronics, 10(18), 2223.

[42] Deshpande, A. (2021). Review of Effective Trust Management Systems in VANET Environments. International Journal of Grid and Distributed Computing, 14(1), 1771-1780.

[43] K. Govindan, P. Mohapatra, Trust computations and trust dynamics in mobile adhoc networks: A survey, IEEE Commun. Surv. Tutor. 14 (2) (2011) 279–298.

[44] D. Suo, S.E. Sarma, Real-time trust-building schemes for mitigating malicious behaviors in connected and automated vehicles, in: 2019 IEEE Intelligent Transportation Systems Conference, ITSC, IEEE, 2019, pp. 1142–1149.

[45] Sateesh, H., Zavarsky, P. (2020, November). State-of-the-Art VANET trust models: Challenges and recommendations. In 2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) (pp. 0757-0764). IEEE.

[46] Wang, Y., Zen, H., Sabri, M. F. M., Wang, X., Kho, L. C. (2022). Towards Strengthening the Resilience of IoV Networks—A Trust Management Perspective. Future Internet, 14(7), 202. 1771-1780.

[47] Deshpande, A. (2021). Review of Effective Trust Management Systems in VANET Environments. International Journal of Grid and Distributed Computing, 14(1), 1771-1780.

[48] Papakonstantinou, N., Van Bossuyt, D. L., Linnosmaa, J., Hale, B., O'Halloran, B. (2020, August). Towards a zero trust hybrid security and

safety risk analysis method. In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (Vol. 83983, p. V009T09A060). American Society of Mechanical Engineers.

[49] Kerrache, C. A., Calafate, C. T., Cano, J. C., Lagraa, N., Manzoni, P. (2016). Trust management for vehicular networks: An adversary-oriented overview. IEEE Access, 4, 9293-9307.

[50] Bendiab, G., Hameurlaine, A., Germanos, G., Kolokotronis, N., Shiaeles, S. (2023). Autonomous vehicles security: Challenges and solutions using blockchain and artificial intelligence. IEEE Transactions on Intelligent Transportation Systems.

[51] Y.M. Chen, Y.C. Wei, A beacon-based trust management system for enhancing user centric location privacy in VANETs, J. Commun. Netw. 15 (2) (2013) 153–163.

[52] F.G. Mármol, G.M. Perez, TRIP, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks, J. Netw. Comput. Appl. 35 (3) (2012) 934–941.

[53] H. Al Falasi, N. Mohamed, Similarity-based trust management system for detecting fake safety messages in vanets, in: International Conference on Internet of Vehicles, Springer, Cham, 2015, pp. 273–284.

[54] U.F. Minhas, J. Zhang, T. Tran, R. Cohen, A multifaceted approach to modeling agent trust for effective communication in the application of mobile ad hoc vehicular networks, IEEE Trans. Syst. Man Cybern. C 41 (3) (2010) 407–420.

[55] U.F. Minhas, J. Zhang, T. Tran, R. Cohen, Towards expanded trust management for agents in vehicular ad-hoc networks, Int. J. Comput. Intell. Theory Pract. 5 (2010).

[56] H. Hu, R. Lu, Z. Zhang, J. Shao, REPLACE: A reliable trust-based platoon service recommendation scheme in VANET, IEEE Trans. Veh. Technol. 66 (2) (2016) 1786– 1797.

[57] H. Xia, S.S. Zhang, Y. Li, Z.K. Pan, X. Peng, X.Z. Cheng, An attack-resistant trust inference model for securing routing in vehicular ad hoc networks, IEEE Trans. Veh. Technol. 68 (7) (2019) 7108–7120.

[58] A. Zhou, J. Li, Q. Sun, C. Fan, T. Lei, F. Yang, A security authentication method based on trust evaluation in VANETs, EURASIP J. Wireless Commun. Networking 2015 (1) (2015) 1–8.

[59] J. Cui, X. Zhang, H. Zhong, Z. Ying, L. Liu, RSMA: Reputation system-based lightweight message authentication framework and protocol for 5G-enabled vehicular networks, IEEE Internet Things J. 6 (4) (2019) 6417–6428.

[60] M. Raya, P. Papadimitratos, V.D. Gligor, J.P. Hubaux, On data-centric trust establishment in ephemeral ad hoc networks, in: IEEE INFOCOM 2008-the 27th Conference on Computer Communications, IEEE, 2008, pp. 1238–1246.

[61] K. Zaidi, M. Milojevic, V. Rakocevic, M. Rajarajan, Data-centric rogue node detection in VANETs, in: 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications, IEEE, 2014, pp. 398–405.

[62] R.A. Shaikh, A.S. Alzahrani, Intrusion-aware trust model for vehicular ad hoc networks, Secur. Commun. Netw. 7 (11) (2014) 1652–1669.

[63] A. Wu, J. Ma, S. Zhang, RATE: a RSU-aided scheme for data-centric trust establishment in VANETs, in: 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing, IEEE, 2011, pp. 1–6.

[64] Y.C. Wei, Y.M. Chen, An efficient trust management system for balancing the safety and location privacy in VANETs, in: 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, IEEE, 2012, pp. 393– 400.

[65] R.A. Shaikh, A.S. Alzahrani, Trust management method for vehicular ad hoc networks, in: International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, Springer, Berlin, Heidelberg, 2013, pp. 801–815.

[66] S. Gurung, D. Lin, A. Squicciarini, E. Bertino, Information-oriented trustworthiness evaluation in vehicular ad-hoc networks, in: International Conference on Network and System Security, Springer, Berlin, Heidelberg, 2013, pp. 94–108.

[67] W. Li, H. Song, ART: An attack-resistant trust management scheme for securing vehicular ad hoc networks, IEEE Trans. Intell. Transp. Syst. 17 (4) (2015) 960–969.

[68] B. Placzek, M. Bernas, Detection of malicious data in vehicular ad hoc networks for traffic signal control applications, in: International Conference on Computer Networks, Springer, Cham, 2016, pp. 72–82.

[69] N. Bißmeyer, S. Mauthofer, K.M. Bayarou, F. Kargl, Assessment of node trustworthiness in vanets using data plausibility checks with particle filters, in: 2012 IEEE Vehicular Networking Conference, VNC, IEEE, 2012, pp. 78–85.

[70] Z. Huang, S. Ruj, M.A. Cavenaghi, M. Stojmenovic, A. Nayak, A social network approach to trust management in VANETs, Peer-To-Peer Netw. Appl. 7 (3) (2014) 229– 242.

[71] N.W. Lo, H.C. Tsai, A reputation system for traffic safety event on vehicular adhoc networks, EURASIP J. Wireless Commun. Networking 2009 (2009) 1–10.

[72] S. Mazilu, M. Teler, C. Dobre, Securing vehicular networks based on data-trust computation, in: 2011 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, IEEE, 2011, pp. 51–58.

[73] Y.C. Wei, Y.M. Chen, Adaptive decision making for improving trust establishment in VANET, in: The 16th Asia-Pacific Network Operations and Management Symposium, IEEE, 2014, pp. 1–4.

[74] Y. Xiao, Y. Liu, Bayestrust and vehiclerank: Constructing an implicit web of trust in vanet, IEEE Trans. Veh. Technol. 68 (3) (2019) 2850–2864.

[75] A.R. Rajeswari, S. Ganapathy, K. Kulothungan, A. Kannan, An efficient trust- based secure energy-aware clustering to mitigate trust distortion attack in mobile adhoc network, Concurr. Comput.: Pract. Exper. (2021) e6223.

[76] N. Mittal, S. Singh, U. Singh, R. Salgotra, Trust-aware energy-efficient stable clustering approach using fuzzy type-2 Cuckoo search optimization algorithm for wireless sensor networks, Wirel. Netw. 27 (1) (2021) 151–174.

[77] T. Khan, K. Singh, M. Abdel-Basset, H.V. Long, S.P. Singh, M. Manjul, A novel and comprehensive trust estimation clustering based approach for large scale wireless sensor networks, IEEE Access 7 (2019) 58221–58240.

[78] Sahoo, R. R., Sardar, A. R., Singh, M., Ray, S., Sarkar, S. K. (2016). A bio inspired and trust based approach for clustering in WSN. Natural Computing, 15, 423- 434.

[79] A. Mahmood, B. Butler, W.E. Zhang, Q.Z. Sheng, S.A. Siddiqui, A hybrid trust management heuristic for VANETs, in: 2019 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops, IEEE, 2019, pp. 748–752.

[80] S. Oubabas, R. Aoudjit, J.J. Rodrigues, S. Talbi, Secure and stable vehicular adhoc network clustering algorithm based on hybrid mobility similarities and trust management scheme, Veh. Commun. 13 (2018) 128–138.

[81] T. Gaber, S. Abdelwahab, M. Elhoseny, A.E. Hassanien, Trust-based secure clustering in WSN-based intelligent transportation systems, Comput. Netw. 146 (2018) 151–158.

[82] K.A. Awan, I.U. Din, A. Almogren, M. Guizani, S. Khan, StabTrust stable and centralized trust-based clustering mechanism for IoT enabled vehicular ad-hoc networks, Ieee Access 8 (2020) 21159–21177.

[83] A. Kchaou, R. Abassi, S. Guemara, Towards the performance evaluation of a clustering and trust based security mechanism for VANET, in: Proceedings of the 15th International Conference on Availability, Reliability and Security, 2020, pp. 1–6.

[84] J. Zhang, C. Chen, R. Cohen, Trust modeling for message relay control and local action decision making in VANETs, Secur. Commun. Netw. 6 (1) (2013) 1–14.

[85] R. Abassi, A.B.C. Douss, D. Sauveron, TSME: A trust-based security scheme for message exchange in vehicular Ad hoc networks, Human-Centric Comput. Inform. Sci. 10 (1) (2020) 1–19.

[86] J. Guo, X. Li, Z. Liu, J. Ma, C. Yang, J. Zhang, D. Wu, TROVE: A context awareness trust model for VANETs using reinforcement learning, IEEE Internet Things J. (2020).

[87] R. Xing, Z. Su, N. Zhang, Y. Peng, H. Pu, J. Luo, Trust-evaluation-based intrusion detection and reinforcement learning in autonomous driving, IEEE Netw. 33 (5) (2019) 54–60.

[88] S. Guleng, C. Wu, X. Chen, X. Wang, T. Yoshinaga, Y. Ji, Decentralized trust evaluation in vehicular Internet of Things, IEEE Access 7 (2019) 15980–15988.

[89] F.A. Ghaleb, F. Saeed, M. Al-Sarem, B. Ali Saleh Al-rimy, W. Boulila, A.E.M. Eljialy, et al., Misbehavior-aware on-demand collaborative intrusion detection system using distributed ensemble learning for VANET, Electronics 9 (9) (2020) 1411.

[90] E.A. Shams, A. Rizaner, A.H. Ulusoy, Trust aware support vector machine intrusion detection and prevention system in vehicular ad hoc networks, Comput. Secur. 78 (2018) 245–254.

[91] Ga, M., Eb, U. (2021). VANET: Trust Evaluation Using Artificial Neural Network. Advances in Parallel Computing Technologies and Applications, 40, 9.

[92] S.A. Soleymani, A.H. Abdullah, M. Zareei, M.H. Anisi, C. Vargas-Rosales, M.K. Khan, S. Goudarzi, A secure trust model based on fuzzy logic in vehicular ad hoc networks with fog computing, IEEE Access 5 (2017) 15619–15629.

[93] Hasan, M. M., Jahan, M., Kabir, S., Wagner, C. (2021, July). A fuzzy logic- based trust estimation in edge-enabled vehicular ad hoc networks. In 2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE) (pp. 1-8). IEEE.

[94] N. Fan, C.Q. Wu, On trust models for communication security in vehicular ad-hoc networks, Ad Hoc Netw. 90 (2019) 101740.

[95] Z. Tian, X. Gao, S. Su, J. Qiu, X. Du, M. Guizani, Evaluating reputation management schemes of internet of vehicles based on evolutionary game theory, IEEE Trans. Veh. Technol. 68 (6) (2019) 5971–5980.

[96] T. Halabi, M. Zulkernine, Secure Collaboration, Trust-based cooperative game model for secure collaboration in the internet of vehicles, in: ICC 2019-2019 IEEE International Conference on Communications, ICC, IEEE, 2019, pp. 1–6.

[97] M.M. Mehdi, I. Raza, S.A. Hussain, A game theory based trust model for Vehicular Ad hoc Networks (VANETs), Comput. Netw. 121 (2017) 152–172.

[98] N. Haddadou, A. Rachedi, Y. Ghamri-Doudane, A job market signaling scheme for incentive and trust management in vehicular ad hoc networks, IEEE Trans. Veh. Technol. 64 (8) (2014) 3657–3674.

[99] N. Haddadou, A. Rachedi, Y. Ghamri-Doudane, Trust and exclusion in vehicular ad hoc networks: an economic incentive model based approach, in: 2013 Computing, Communications and IT Applications Conference, ComComAp, IEEE, 2013, pp. 13–18.

[100] B. Subba, S. Biswas, S. Karmakar, A game theory based multi layered intrusion detection framework for VANET, Future Gener. Comput. Syst. 82 (2018) 12–28.

[101] X. Chen, L. Wang, A cloud-based trust management framework for vehicular social networks, IEEE Access 5 (2017) 2967–2980.

[102] J. Pawlick, J. Chen, Q. Zhu, iSTRICT: An interdependent strategic trust mechanism for the cloud-enabled internet of controlled things, IEEE Trans. Inf. Forensics Secur. 14 (6) (2018) 1654–1669.

[103] B.K. Chaurasia, K. Sharma, Trust computation in VANET cloud, in: Transactions on Computational Science XXXIV, Springer, Berlin, Heidelberg, 2019, pp. 77–95.

[104] F. Dewanta, M. Mambo, Bidding price-based transaction: Trust establishment for vehicular fog computing service in rural area, in: 2019 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops, IEEE, 2019, pp. 882–887.

[105] X. Huang, R. Yu, J. Kang, Y. Zhang, Distributed reputation management for secure and efficient vehicular edge computing and networks, IEEE Access 5 (2017) 25408– 25420.

[106] S.A. Soleymani, S. Goudarzi, M.H. Anisi, N. Kama, S. Adli Ismail, A. Azmi, et al., A trust model using edge nodes and a cuckoo filter for securing VANET under the nlos condition, Symmetry 12 (4) (2020) 609.

[107] Z. Yang, K. Yang, L. Lei, K. Zheng, V.C. Leung, Blockchain-based decentralized trust management in vehicular networks, IEEE Internet Things J. 6 (2) (2018) 1495– 1505, (01).

[108] X. Liu, H. Huang, F. Xiao, Z. Ma, A blockchain-based trust management with conditional privacy-preserving announcement scheme for VANETs, IEEE Internet Things J. (2019).

[109] Z. Lu, Q. Wang, G. Qu, Z. Liu, Bars: a blockchain- based anonymous reputation system for trust management in vanets, in: 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/ 12th IEEE International Conference on Big Data Science and Engineering, TrustCom/BigDataSE, IEEE, 2018, pp. 98–103.

[110] C. Zhang, W. Li, Y. Luo, Y. Hu, AIT: An AI-enabled trust management system for vehicular networks using blockchain technology, IEEE Internet Things J. (2020).

[111] Y. Zou, F. Shen, F. Yan, J. Lin, Y. Qiu, Reputation-based regional federated learning for knowledge trading in blockchain-enhanced IoV, in: 2021 IEEE Wireless Communications and Networking Conference, WCNC, IEEE, 2021, pp. 1–6.

[112] Amiri, M. J., Agrawal, D., El Abbadi, A. (2021, June). Sharper: Sharding permissioned blockchains over network clusters. In Proceedings of the 2021 international conference on management of data (pp. 76-88).

[113] Q. Mei, H. Xiong, Y. Zhao, K.H. Yeh, Toward blockchain-enabled IoV with edge computing: Efficient and privacy-preserving vehicular communication and dynamic updating, in: 2021 IEEE Conference on Dependable and Secure Computing, DSC, IEEE, pp. 1–8.

[114] V. Sharma, An energy-efficient transaction model for the blockchain-enabled internet of vehicles (IoV), IEEE Commun. Lett. 23 (2) (2018) 246–249.

[115] N. Tariq, M. Asim, F.A. Khan, T. Baker, U. Khalid, A. Derhab, A blockchain- based multi-mobile code-driven trust mechanism for detecting internal attacks in Internet of Things, Sensors 21 (1) (2021) 23.

[116] M. Banerjee, J. Lee, Q. Chen, K.K.R. Choo, Blockchain-based security layer for identification and isolation of malicious things in IoT: A

conceptual design, in: 2018 27th International Conference on Computer Communication and Networks, ICCCN, IEEE, 2018, pp. 1–6.

117 M.T. Hammi, P. Bellot, A. Serhrouchni, BCTrust: A decentralized authentication blockchain-based mechanism, in: 2018 IEEE Wireless Communications and Networking Conference, WCNC, IEEE, 2018, pp. 1–6.

[118] Y. Sun, W. Chen, L. Chen, M. Li, Research on clustering management of power distribution Internet of Things based on trusted blockchain, J. Phys. Conf. Ser. 1748 (5) (2021) 052064.

[119] A. Dorri, S.S. Kanhere, R. Jurdak, P. Gauravaram, LSB: A lightweight scalable blockchain for IoT security and anonymity, J. Parallel Distrib. Comput. 134 (2019) 180– 197.

[120] W. She, Q. Liu, Z. Tian, J.S. Chen, B. Wang, W. Liu, Blockchain trust model for malicious node detection in wireless sensor networks, IEEE Access 7 (2019) 38947– 38956.

[121] A. Di Maio, M.R. Palattella, R. Soua, L. Lamorte, X. Vilajosana, J. Alonso-Zarate, T. Engel, Enabling SDN in VANETs: What is the impact on security? Sensors 16 (12) (2016) 2077.

[122] A. Mahmood, W.E. Zhang, Q.Z. Sheng, S.A. Siddiqui, A. Aljubairy, Trust management for software-defined heterogeneous vehicular ad hoc networks, in: Security, Privacy and Trust in the IoT Environment, Springer, Cham, 2019, pp. 203–226.

[123] H. Vasudev, D. Das, A trust based secure communication for software defined VANETs, in: 2018 International Conference on Information Networking, ICOIN, IEEE, 2018, pp. 316–321.

[124] A. Boualouache, R. Soua, T. Engel, SDN-based misbehavior detection system for vehicular networks, in: 2020 IEEE 91st Vehicular Technology Conference, VTC2020- Spring, IEEE, 2020, pp. 1–5

[125] D. Zhang, F.R. Yu, Z. Wei, A. Boukerche, Software-defined vehicular ad hoc networks with trust management, in: Proceedings of the 6th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications, 2016, pp. 41–49.

[126] L. Alouache, M. Maachaoui, R. Chelouah, Securing hybrid SDN-based geographic routing protocol using a distributed trust model, Adv. Sci. Technol. Eng. Syst. J. (2020).

[127] D. Zhang, F.R. Yu, R. Yang, A machine learning approach for software-defined vehicular ad hoc networks with trust management, in: 2018 IEEE Global Communications Conference, GLOBECOM, IEEE, 2018, pp. 1–6.

[128] N. Zhao, H. Wu, X. Zhao, Consortium blockchain-based secure software defined vehicular network, Mob. Netw. Appl. 25 (1) (2020) 314–327.

[129] L. Xie, Y. Ding, H. Yang, X. Wang, Blockchain-based secure and trustworthy Internet of Things in SDN-enabled 5G-VANETs, IEEE Access 7 (2019) 56656–56666.

[130] J. Gao, K.O.B.O. Agyekum, E.B. Sifah, K.N. Acheampong, Q. Xia, X. Du, et al., A blockchain-SDN-enabled Internet of vehicles environment for fog computing and 5G networks, IEEE Internet Things J. 7 (5) (2019) 4278–4291.

[131] D. Zhang, F.R. Yu, R. Yang, Blockchain-based distributed software-defined vehicular networks: A dueling deep qlearning approach, IEEE Trans. Cogn. Commun. Netw. 5 (4) (2019) 1086–1100.

[132] D. Zhang, F.R. Yu, R. Yang, H. Tang, A deep reinforcement learning-based trust management scheme for software- defined vehicular networks, in: Proceedings of the 8th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, 2018, pp. 1–7.

[133] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system, 2008, (Accessed on: Nov 2019).

[134] Zhang, T., Zhu, Q. (2018). Distributed privacy-preserving collaborative intrusion detection systems for VANETs. IEEE Transactions on Signal and Information Processing over Networks, 4(1), 148-161.

[135] Bangui, H., Ge, M., Buhnova, B. (2021). A hybrid machine learning model for intrusion detection in VANET. Computing, 1-29.

[136] Yang, L., Moubayed, A., Shami, A. (2021). MTH-IDS: A Multi-Tiered Hybrid Intrusion Detection System for Internet of Vehicles. IEEE Internet of Things Journal.

[137] Jin, F., Chen, M., Zhang, W., Yuan, Y., Wang, S. (2021). Intrusion detection on internet of vehicles via combining log-ratio oversampling, outlier detection and metric learning. Information Sciences, 579, 814-831.

[138] Gruebler, A., McDonald-Maier, K. D., Alheeti, K. M. A. (2015, September). An intrusion detection system against black hole attacks on the communication network of self-driving cars. In 2015 sixth international conference on emerging security technologies (EST) (pp. 86-91). IEEE.

[139] Alladi, T., Gera, B., Agrawal, A., Chamola, V., Yu, F. R. (2021). DeepADV: A Deep Neural Network Framework for Anomaly Detection in VANETs. IEEE Transactions on Vehicular Technology, 70(11), 12013-12023.

[140] Ayoob, A. A., Su, G., Al, G. (2018). Hierarchical growing neural gas network (hgng)-based semicooperative feature classifier for ids in vehicular ad hoc network (vanet). Journal of Sensor and Actuator Networks, 7(3), 41.

[141] Ayed, M. A., Talhi, C. Federated Learning for Anomaly-Based Intrusion Detection. In 2021 International Symposium on Networks, Computers and Communications (ISNCC) (pp. 1-8). IEEE.

[142]Zhao, R., Yin, Y., Shi, Y., Xue, Z. (2020). Intelligent intrusion detection based on federated learning aided long short-term memory. Physical Communication, 42, 101157.

[143] Mourad, A., Tout, H., Wahab, O. A., Otrok, H., Dbouk, T. (2020). Ad Hoc Vehicular Fog Enabling Cooperative Low-Latency Intrusion Detection. IEEE Internet of Things Journal, 8(2), 829-843.

[144] Abdel-Basset, M., Moustafa, N., Hawash, H., Razzak, I., Sallam, K. M., Elkomy, O. M. (2021). Federated Intrusion Detection in Blockchain-Based Smart Transportation Systems. IEEE Transactions on Intelligent Transportation Systems.

[145] Liang, H., Wu, J., Mumtaz, S., Li, J., Lin, X., Wen, M. (2019). MBID: Micro- blockchain-based geographical dynamic intrusion detection for V2X. IEEE Communications Magazine, 57(10), 77-83.

[146] Shu, J., Zhou, L., Zhang, W., Du, X., Guizani, M. (2020). Collaborative intrusion detection for VANETs: a deep learning-based distributed SDN approach. IEEE Transactions on Intelligent Transportation Systems.

[147] Alsarhan, A., Al-Ghuwairi, A. R., Almalkawi, I. T., Alauthman, M., Al-Dubai, A. (2021). Machine learning-driven optimization for intrusion detection in smart vehicular networks. Wireless Personal Communications, 117(4), 3129-3152.

[148] Liang, J., Ma, M. (2020). ECF-MRS: An efficient and collaborative framework with Markov-based reputation scheme for IDSs in vehicular networks. IEEE transactions on information forensics and security, 16, 278-290.

[149] Liu, H., Zhang, S., Zhang, P., Zhou, X., Shao, X., Pu, G., Zhang, Y. (2021). Blockchain and federated learning for collaborative intrusion detection in vehicular edge computing. IEEE Transactions on Vehicular Technology, 70(6), 6073-6084.

[150] Li, W., Wang, Y., Jin, Z., Yu, K., Li, J., Xiang, Y. (2021). Challenge-based collaborative intrusion detection in software-defined networking: an evaluation. Digital Communications and Networks, 7(2), 257-263.

[151] Azodolmolky, S. (2013). Software defined networking with OpenFlow. Packt Publishing.

[152] Sharma, S., Kaushik, B. (2021). A survey on nature-inspired algorithms and its applications in the Internet of Vehicles. International Journal of Communication Systems, 34(12), e4895.

[153] Azzoug, Y., Boukra, A. (2022). Enhanced UAV-aided vehicular delay tolerant network (VDTN) routing for urban environment using a bio-inspired approach. Ad hoc networks, 133, 102902.

[154] Chughtai, O., Naeem, M., Khaliq, K. A. (2022). Uav-assisted cooperative routing scheme for dense vehicular ad hoc network. In Intelligent Unmanned Air Vehicles Communications for Public Safety Networks (pp. 199-214). Singapore: Springer Nature Singapore.

[155] Fatemidokht, H., Rafsanjani, M. K., Gupta, B. B., Hsu, C. H. (2021). Efficient and secure routing protocol based on artificial intelligence algorithms with UAV-assisted for vehicular ad hoc networks in intelligent

transportation systems. IEEE Transactions on Intelligent Transportation Systems, 22(7), 4757-4769.

[156] Qureshi, K. N., Alhudhaif, A., Shah, A. A., Majeed, S., Jeon, G. (2021). Trust and priority-based drone assisted routing and mobility and service-oriented solution for the internet of vehicles networks. Journal of Information Security and Applications, 59, 102864.

[157] Bakhtiari, Z., Oskouei, R. J., Soleymani, M., Jalbani, A. H. (2020). Presenting an Effective Method to Detect and Track the Broken Path in VANET Using UAVs. Wireless Communications and Mobile Computing, 2020, 1-12.

[158] Oubbati, O. S., Chaib, N., Lakas, A., Lorenz, P., Rachedi, A. (2019). UAV- assisted supporting services connectivity in urban VANETs. IEEE Transactions on Vehicular Technology, 68(4), 3944-3951.

[159] Sami Oubbati, O., Chaib, N., Lakas, A., Bitam, S., Lorenz, P. (2020). U2RV: UAV-assisted reactive routing protocol for VANETs. International Journal of Communication Systems, 33(10), e4104.

[160] Oubbati, O. S., Lakas, A., Lorenz, P., Atiquzzaman, M., Jamalipour, A. (2019). Leveraging communicating UAVs for emergency vehicle guidance in urban areas. IEEE Transactions on Emerging Topics in Computing, 9(2), 1070-1082.

[161] Mokhtari, S., Nouri, N., Abouei, J., Avokh, A., Plataniotis, K. N. (2022). Relaying data with joint optimization of energy and delay in cluster-based uav-assisted vanets. IEEE Internet of Things Journal, 9(23), 24541-24559.

[162] Abualola, H., Otrok, H., Barada, H., Al-Qutayri, M., Al-Hammadi, Y. (2021). Matching game theoretical model for stable relay selection in a UAV-assisted internet of vehicles. Vehicular Communications, 27, 100290.

[163] He, Y., Zhai, D., Wang, D., Tang, X., Zhang, R. (2020). A relay selection protocol for UAV-assisted VANETs. Applied Sciences, 10(23), 8762.

[164] Ghazzai, H., Khattab, A., Massoud, Y. (2019, September). Mobility and energy aware data routing for UAV-assisted VANETs. In 2019 IEEE In-

ternational Conference on Vehicular Electronics and Safety (ICVES) (pp. 1-6). IEEE.

[165] Oubbati, O. S., Atiquzzaman, M., Baz, A., Alhakami, H., Ben-Othman, J. (2021). Dispatch of UAVs for urban vehicular networks: A deep reinforcement learning approach. IEEE Transactions on Vehicular Technology, 70(12), 13174-13189.

[166] Miao, W., Ding, Z., Tang, H., Zeng, Z., Zhang, M., Zhang, S. (2021). A Seq2Seq Learning Approach for Link Quality Estimation Based on System Metrics in WSNs. IEEE Access, 9, 44207-44216.

Author's Publications

Peer-reviewed international journal articles

- Hbaieb, A., Ayed, S., Chaari, L. (2022). A survey of trust management in the Internet of Vehicles. Computer Networks, 203, 108558.

- Ayed, S., Hbaieb, A., Chaari, L. (2023). Blockchain and trust-based clustering scheme for the IoV. Ad Hoc Networks, 142, 103093.

Peer-reviewed international conference articles

- Hbaieb, A., Ayed, S., Chaari, L. (2021, April). Blockchain-based trust management approach for IoV. In International Conference on Advanced Information Networking and Applications (pp. 483-493). Cham: Springer International Publishing.

- Hbaieb, A., Ayed, S., Chaari, L. (2022, August). Federated learning based IDS approach for the IoV. In Proceedings of the 17th International Conference on Availability, Reliability and Security (pp. 1-6).

SSRN (open access paper)

- Hbaieb, A., Samiha, A. Y. E. D., CHAARI, L. (2021). Internet of Vehicles and Connected Smart Vehicles Communication System Towards Autonomous Driving.

Chapter 8

Annexe: Résumé de la thèse

8.1 Introduction

L'Internet des Véhicules (IoV) applique les concepts de l'Internet des Objets (IdO) pour des systèmes de transport plus intelligents. L'IoV est un réseau complexe composé de véhicules, d'infrastructures, de personnes (conducteurs, passagers, piétons sur le bord de la route) et d'autres dispositifs intelligents connectés au réseau. L'IoV comprend la conduite autonome, la conduite sécurisée, la conduite sociale, les services de divertissement, les applications mobiles intelligentes et les technologies émergentes. Diverses types de communication de véhicule à tout (V2X) existent entre les nœuds de l'IoV pour échanger des données importantes telles que la localisation, la vitesse, et les informations sur le trafic. Ce type de réseaux se caractérise essentiellement par une précision, une fiabilité et une utilité accrues des informations collectées. Cependant, le domaine d'usage et d'application de l'IoV le confront à des exigences très contraignantes d'une part en terme de sécurité et d'une autre part en terme de qualité de service . En effet, les nœuds d'un tel réseau sont très mobiles et peuvent communiquer directement ou à travers l'intermédiaire d'une infrastructure. D'autre part les données qui transitent dans ces réseaux sont de nature sensible et imposent à avoir un degré de confiance très haut concernant les nœuds qui forment ce réseau. En effet, ces données sont utilisées sur la route et peuvent être considérées dans des différents cas d'étude (par example, embouteillage, accident, conditions climatiques particulières, etc.). De ce fait, elles doivent être optimales et fiables pour servir au mieux à améliorer la qualité du trafic, diminuer le taux des accidents, et éviter que ces données soient utilisées d'une façon malveillante. Dans les travaux actuels, il y a plusieurs travaux qui se sont intéressés aux aspects de performances techniques de ce réseau. Ces travaux n'ont pas pris en considération l'aspect de sécurité primordial dans le cadre de ces réseaux. De point de vue sécurité, d'autres travaux se sont intéressés à l'étude de mécanismes de sécurité dans l'IoV mais sans étudier les impacts des solutions proposées sur les performances d'un tel réseau. Il est donc essentiel d'établir un mécanisme de sécurité entre les nœuds de l'IoV pour garantir une sécurité qui tienne compte de la qualité de service. Dans cette thèse nous proposons une gestion sécurisée de ce type de réseau en se basant sur la gestion de la confiance. Nous nous concentrons sur la gestion de la confiance et la partie réseau de l'écosystème de l'IoV. Une approche de sécurité basée sur la gestion de la confiance permet de distinguer les nœuds malveillants et d'atténuer les attaques. Les approches basées sur la gestion de la confiance peuvent assurer la fiabilité et la stabilité du processus de communication dans l'IoV. La gestion de la confiance établit un mécanisme d'évaluation de la confiance selon des politiques spécifiques. Les nœuds malveillants peuvent être identifiés sur la base de leurs évaluations de confiance afin que les nœuds du réseau puissent les éviter et donner la priorité aux nœuds dignes de confiance pour les interactions. Nous abordons la confiance et l'architecture du réseau en utilisant des technologies émergentes. Trois aspects du travail sont proposés dans cette thèse : (1) un cadre de gestion de la confiance, (2) un système de détection des intrusions (IDS) et (3) un protocole de routage.

Tout d'abord, nous proposons une plateforme de sécurité basée sur la gestion de la con-

fiance et la Blockchain pour garantir la fiabilité et la sécurité de l'IoV. L'architecture de la plateforme de sécurité est une Blockchain à deux niveaux qui se compose d'une Blockchain locale pour sauvegarder les valeurs de gestion de la confiance locale, et d'une Blockchain de confiance globale. La Blockchain est utilisée en raison de ses propriétés clés telles que la décentralisation et l'immutabilité. En premier, nous adoptons un processus centralisé de gestion de la confiance. Ensuite, nous étendons la plateforme de sécurité proposé pour mieux prendre en compte la mesure de la qualité de service tout en renforçant la sécurité. Nous nous concentrons sur la mesure de l'énergie car elle affecte directement la continuité du réseau. Nous optons pour une gestion décentralisée de la confiance afin de permettre une plus grande évolutivité et s'adapter à la topologie dynamique de l'IoV. Pour cela, l'approche de confiance applique un processus de clustering.Ensuite, nous proposons une évaluation de la confiance à l'aide d'un IDS basé sur l'apprentissage fédéré pour résoudre le problème de la sécurité dans l'IoV. Nous examinons l'apprentissage collaboratif de la confiance dans le cadre de l'architecture SDN (Software-Defined Networking). Pour atténuer les difficultés liées à l'apprentissage complexe de la confiance, nous utilisons un processus de détection léger. En fait, peu de travaux envisagent l'intégration des métriques de confiance, de l'apprentissage fédéré et du SDN dans un IDS pour une sécurité et une qualité de service conjointes dans l'IoV. L'IDS collaboratif, peut etre une bonne solution pour assurer la sécurité dans l'IoV grâce à son apprentissage distribué. En outre, l'adoption de l'architecture SDN permet un IDS qui peut maintenir la qualité de service. Le SDN apporte certaines solutions aux limites du réseau, ce qui permet d'obtenir de bonnes performances du réseau, telles qu'une gestion souple et granulaire, une allocation efficace des ressources pour faire fonctionner l'IDS et une prise de décision localisée. En outre, nous reconsidérons les métriques de confiance, l'apprentissage fédéré et le SDN, et procédons à l'amélioration de l'IDS. Nous intégrons le clustering pour améliorer la détection, et les performances globales en matière de qualité de service. Nous adoptons le concept de l'algorithme de l'essaim de dauphins pour définir des nœuds qui vont agir comme des dauphins qui scannent les nœuds voisins pour détecter les comportements malveillants en se basant sur les métriques de confiance.

Enfin, nous examinons l'aspect du routage dans l'IoV, en prenant en compte le facteur de confiance. Notre objectif est de trouver une solution de routage qui permette de trouver un compromis entre la sécurité et la qualité de service. Nous proposons une solution de routage réactif assisté par drone. Pour cela, nous produisons un cadre IoV-UAV_Fog pour faciliter le routage assisté par drone. Le routage proposé consiste à sélectionner le drone optimal servant de relais pour la transmission des données vers le nœud de destination. La sélection du relais optimal prend en compte une métrique composite qui maximise à la fois la confiance et la qualité de service.

8.2 Contributions

Les contributions de cette thèse sont résumées comme suit :

• Revue de la littérature sur la gestion de la confiance dans les réseaux véhiculaires. Nous présentons les travaux existants basés sur la gestion confiance pour les réseaux véhiculaires, tout en couvrant les différents aspects de la gestiono confiance (modules de confiance, propriétés de la confiance, métriques de la confiance, et défis liés à la gestion de la confiance). L'état de l'art permet de positionner notre travail de thèse.

• Une plateforme de sécurité basée sur la gestion de confiance et la Blockchain pour valider la fiabilité du système de communication IoV. Nous évaluons la proposition pour valider son efficacité. Nous proposons ainsi d'appliquer le clustering pour une plateforme de sécurité plus évolutive afin d'obtenir une bonne qualité de service. La formation de clusters avec la sélection des têtes de clusters est basée sur le processus de gestion de la confiance, la distance de sécurité et la métrique de l'énergie. L'évaluation des performances de l'approche proposée est réalisée pour valider son efficacité.

• Un système de détection des intrusions (IDS) basé sur les métriques de confiance, le SDN, et l'apprentissage fédéré pour addresser la malveillance des nœuds dans l'IoV. L'objectif est de détecter les anomalies tout en maintenant la QoS. L'IDS apprend le comportement des nœuds à l'aide de métriques liées à la confiance pour décider si le comportement observé est une anomalie. Nous appliquons un processus de détection léger dans lequel les contrôleurs SDN sont les nœuds de détection locaux. Nous présentons également une extension de l'IDS afin d'améliorer le processus de détection en appliquant le clustering. L'évaluation des performances de l'IDS est fournie.

• Un protocole de routage pour l'IoV assisté par drone. La solution de routage proposée prend en compte la confiance et la qualité de service. La solution de routage consiste à sélectionner de drones comme relais pour former le meilleur chemin vers le nœud de destination lorsque le routage par les nœuds terrestres n'est pas fiable. Les performances de la solution de routage sont évaluées.

8.3 Approche basée sur la gestion de la confiance pour l'IoV

8.3.1 Approche basée sur la confiance et la Blockchain

Dans cette thèse, nous nous concentrons sur la gestion de la confiance avec les technologies émergentes. La technologie Blockchain est importante pour la gestion de la confiance dans l'écosystème IoV. La Blockchain fait partie intégrante d'une architecture IoV opti-

misée. Pour cela, nous addressons la sécurité dans l'IoV par le biais d'une approche basée sur la gestion confiance et la Blockchain. La Blockchain, avec ses caractéristiques de sécurité inhérentes, peut compléter les mécanismes de gestion de la confiance pour améliorer la sécurité et la fiabilité de l'écosystème IoV.

8.3.1.1 Architecture proposée

Nous proposons une approche centralisée basée sur la confiance qui utilise deux métriques de confiance: la réputation et la localisation. Nous appliquons l'approche proposée à une architecture basée sur une Blockchain à deux niveaux. La couche inférieure de l'architecture proposée du réseau IoV comprend des "véhicules edge" qui constituent les Blockchains locales. Les Blockchains locales sont utilisées pour stocker les valuers de confiance des nœuds locaux (c'est-à-dire les nœuds situés dans de petites zones). En effet, les Blockchains locales peuvent permettre de réduire les délais tout en conservant ces données sensibles au temps. Au niveau supérieur de l'architecture, les RSUs hébergent la Blockchain globale. Les RSUs communiquent pour ajouter les Blockchains locales à la Blockchain globale. La Blockchain globale sert à sauvegarder les valeurs de confiance calculées et à les mettre à jour en fonction des comportements observés. Elle est déployée pour montrer le niveau de confiance globale dans le réseau.

8.3.1.2 Processus de gestion de confiance

Le processus de la gestion de confiance est défini comme suit. Chaque nœud doit être authentifié par les nœuds de l'autorité afin de confirmer la légitimité de son identité pour rejoindre le système de communication IoV. Chaque nœud dispose d'un certificat numérique. Les "véhicules edge" gèrent la confiance dans le réseau. Lorsque un nœud reçoit un message, il peut demander la valuer de confiance de l'expéditeur au "véhicule edge" le plus proche. Ce dernier procède au calcul de niveau de confiance afin de fournir la valeur requise au nœud demandeur. Le processus d'estimation de la valeur de confiance consiste à calculer la réputation du nœud expéditeur et la crédibilité de son message envoyé.

• Calcul de la réputation: La réputation du nœud est calculée en se basant sur la métrique de connaissance et des recommandations fournies par les autres "véhicules edges" et les RSUs qui sont proches. Chaque "véhicule edge" conserve une valeur de recommandation pour chaque nœud avec lequel il a eu au moins un échange. La valeur de confiance de chaque nœud est ensuite mise à jour en fonction du comportement et de la coopération honnête. Le "véhicule edge" (Ek) calcule la réputation d'un nœud Vj en se basanr sur la valeur de réputation sauvegardée précédemment, la moyenne des valeurs de réputation envoyées par l'ensemble des "véhicules edge", et la moyenne des valeurs de réputation envoyées par les RSUs.

• Calcul de la valeur de crédibilité de message Au sein de notre réseau IoV, lorsqu'un événement se produit (accident, embouteillage, etc.), les nœuds qui souhaitent signaler l'événement diffusent un message d'information contenant leur position et la position de l'événement. Lorsque un nœud reçoit un message concernant un événement spécifique, il doit vérifier la crédibilité du message après avoir estimé la valeur de réputation de l'expéditeur. La crédibilité du message est estimée en se basant sur la métrique de localisation. En fait, la proximité de l'emplacement constitue un facteur important pour l'évaluation de la crédibilité d'un message, car le contenu rapporté est susceptible d'être plus précis lorsque l'expéditeur Vj est situé à proximité du lieu de l'événement Evq.

• Calcul de la valeur de confiance totale La dernière étape de l'évaluation de la confiance du nœud expéditeur Vj consiste à calculer la valeur de confiance totale. Le "véhicule edge" Ek calcule cette valeur en tenant compte des valeurs de réputation et de crédibilité du message à l'aide de la règle de Bayes. La règle de Bayes est utilisée parce que le calcul de la valeur de confiance à un moment donné dépend du comportement antérieur du nœud et des événements qui se sont produits dans le passé. Le théorème de Bayes permet d'exprimer des événements interdépendants sur la base d'événements antérieurs et en tenant compte de nouvelles observations.

Aprés avoir calculé la valeur de conficance totale du nœud expéditeur Vj, le "véhicule edge" communique cette valeur au nœud demandeur. La valeur de confiance calculée est ensuite conservée dans la Blockchain locale. Enfin, un algorithme d'incitation est appliqué pour récompenser ou punir les nœuds concernés (c'est-à-dire les nœuds expéditeurs, les recommandeurs et les "véhicules edges"). L'algorithme d'incitation est basé sur le niveau d'importance des messages envoyés. Lorsque le niveau d'importance du message est plus élevé, la valeur de la récompense ou de la punition augmente. Cet algorithme peut encourager les nœuds à être coopératifs et à collaborer de manière honnête. De même, les Blockchains locales sont ajoutées à la Blokchain globale par les RSUs afin d'avoir une vision globale sur le niveau de confiance dans le réseau.Nous évaluons les performances de notre proposition en terme de rappel, de temps de calcul de la confiance, et de surcharge de communication. Notre proposition basée sur la gestion de la confiance et la Blockchain a montré de bons résultats en termes de taux de détection des véhicules malveillants, de durée d'établissement de la confiance et de surcharge de communication liée à la gestion de la confiance. Cependant, nous avons opté pour une approche centralisée de gestion de confiance. Par conséquent, les performances diminuent lorsque le nombre de nœuds dans le réseau augmentent. De plus, lorsqu'une attaque se produit, elle peut se propager dans le réseau IoV puisque les valeurs de confiance sont calculées avec une approche centralisée. En outre, étant donné que le réseau IoV se caractérise par une grande mobilité, une approche centralisée basée sur la confiance peut conduire à l'interruption du processus de calcul de la confiance lorsque certains événements se produisent au sein du réseau ou lorsque certains nœuds quittent le réseau. Pour cela, nous proposons une extension afin d'améliorer les limitations mises en évidence.

8.3.2 Approche basée sur la confiance, la Blockchain, et le clustering

Nous introduisons une gestion de confiance décentralisée avec le clustering et la Blockchain afin d'améliorer les performances du réseau en matière d'évolutivité et de mobilité. Nous nous intéressons à la continuité du réseau pour prendre davantage en compte la qualité de service du réseau.

8.3.2.1 Architecture proposée

Nous proposons d'étendre notre cadre de sécurité en définissant une topologie de réseau basée sur une classification des différents nœuds. Le processus décentralisé de gestion de la confiance est appliqué à la classification des nœuds. Nous conservons les métriques de réputation et de crédibilité des messages dans le processus que nous proposons. Nous reconsidérons aussi l'architecture de la Blockchain à deux niveaux. Les échanges entre les nœuds du réseau et la Blockchain sont mis à jour puisque nous introduisons une approche de clustering. Les différentes phases du processus de clustering (la formation des clusters, la sélection des têtes de clusters et la maintenance des clusters) sont basées sur le processus de gestion de la confiance, la distance de sécurité et la métrique d'énergie. Il est important de prendre en considération la métrique de l'énergie car elle a un impact direct sur la continuité du réseau. Cette métrique peut définir un compromis entre la sécurité et la qualité de service du réseau.

8.3.2.2 Processus de clustering et de gestion de confiance

Un nœud émet une demande d'adhésion au cluster. Cette demande contient son certificat d'authentification et d'autres paramètres indiquant sa vitesse, sa position et sa direction. Pour vérifier cette demande, la tête de cluster vérifie l'authentification et la valeur de réputation du nœud demandeur. Si la valeur de confiance de ce nœud est inférieur au seuil de confiance, la tête de cluster peut considérer que le nœud n'est pas digne de confiance et qu'il ne peut pas se joindre au cluster. Lorsque la phase d'authentification réussit; et les propriétés du nœud correspondent aux propriétés du cluster ainsi que la valeur de réputation satisfait le seuil, la tête de cluster envoie une réponse positive pour confirmer que le nœud peut rejoindre le cluster. Concernant les têtes de clusters, le nœud de l'autorité entame le processus de leur sélection. Les "véhicules edge" sont les candidates pour les têtes de clusters. Les têtes de clusters sont séléctionés sur base des paramètres de valeur de confiance, la distance de sécurité et l'indicateur d'énergie. Ces paramètres permettent d'améliorer la sécurité au sein des clusters et d' assurer la continuité et la stabilité du réseau. Le "véhicule edge" ayant la valeur la plus élevée liée aux paramètres de séléction est choisi comme tête de cluster. Chaque tête de cluster attribue à ses membres une valeur de confiance initiale.

Chaque membre du cluster peut demander la valeur de confiance d'un autre membre au nœud de tête de cluster. La tête de cluster calcule la réputation de ses membres sur la base de leur comportement et de leur coopération honnête au sein du cluster. Les têtes de clusters ajoutent les valeurs de confiance calculées dans les Blockchain locales. Une fois créée, les Blockchains locales sont ajoutées à la Blockchain globale par les RSUs. Le nœud de l'autorité envoie périodiquement des demandes de réélection des têtes de clusters. La mise à jour des têtes de clusters est importante car les valeurs de confiance peuvent changer. Il est donc obligatoire de vérifier que la tête de cluster élue a la valeur de confiance la plus élevée. En outre, lorsque deux têtes de clusters sont trop proches, elles fusionnent lors du prochain tour du processus de sélection de tête de cluster. La tête de cluster ayant la valeur de confiance la plus élevée sera la tête de cluster fusionné. Les autres têtes de clusters concernés deviendront des membres ordinaires de ce cluster. Nous évaluons les performances de détection, les propriétés d'exécution, la complexité du schéma de clustering, et la stabilité du réseau. Les résultats montrent que notre approche satisfait davantage la qualité de service et la sécurité du réseau IoV en comparant avec la première proposition.

Pour récapituler, nous avons présenté une approche de sécurité basée sur la gestion de confiance pour le réseau IoV. Dans un premier temps, nous avons mis en œuvre un processus centralisé de gestion de la confiance que nous avons appliqué à une architecture Blockchain à deux niveaux. Le processus de gestion de la confiance a évalué la confiance des nœuds à l'aide des mesures de réputation et de crédibilité des messages. La Blockchain a été utilisée pour garantir l'immuabilité et l'intégrité des données de confiance. Ensuite, nous avons appliqué le clustering pour un processus de gestion de confiance décentralisée afin de mieux prendre en compte la qualité de service tout en renforçant la sécurité. La formation des clusters et la sélection des têtes de clusters ont pris en compte l'aspect de la confiance afin de créer des clusters fiables. L'évaluation des performances a été menée afin de valider l'efficacité de notre proposition. Les résultats indiquent que la formation des clusters fiables à l'aide de la Blockchain renforce la fiabilité de l'IoV. Notre approche de clustering basée sur la confiance et la Blockchain peut jouer un rôle essentiel dans l'amélioration de la sécurité du réseau IoV et le maintien des principales exigences de qualité de service en termes de continuité et d'évolutivité.

8.4 Système de détection d'intrusion collaboratif basé sur la gestion de la confiance pour l'IoV

8.4.1 Système de détection d'intrusion basé sur la confiance et le SDN

Nous abordons la gestion de la confiance dans le réseau IoV selon l'angle de l'apprentissage du comportement des nœuds. Cela consiste à analyser le comportement des nœuds dans

le temps afin d'identifier les anomalies ou les écarts par rapport au comportement attendu. Nous nous appuyons sur un IDS collaboratif basé sur l'apprentissage des métriques liées à la confiance afin de détecter les nœuds malveillants. L'IDS fonctionne au sein d'une architecture IoV basée sur le SDN. Le SDN permet une flexibilité pour la gestion du processus de détection dans l'IoV tout en offrant une optimisation de l'architecture réseau. Le SDN peut permettre une allocation efficace des ressources nécessaires pour participer à la détection collaborative (par exemple énergie, bande passante). Cette gestion de ressources optimise les performances de l'IDS. De plus, le SDN permet une segmentation efficace du réseau ce qui permet une prise de décision locale pour la détection. En traitant les données et en prenant des décisions localement, l'IDS peut réduire le temps de latence de la détection, ce qui est crucial pour la détection des menaces en temps réel dans l'IoV. L'IDS s'appuie sur l'apprentissage fédéré pour encourager la détection des comportements malveillants invisibles pour un nœud particulier tout en tout en préservant la confidentialité des données traitées. L'apprentissage fédéré est considéré pour mettre en œuvre la collaboration entre les contrôleurs SDN locaux pour la détection des nœuds non dignes de confiance.

8.4.1.1 Architecture proposée

Le plan de données dans notre architecture basée sur le SDN comprend les véhicules intelligents, les RSUs, et les nœuds d'autorité. Le plan de contrôle consiste des contrôleurs SDN et d'un serveur Cloud. Les élements associés au plan de données sont surveillés par les contrôleurs SDN. Les nœuds d'autorité se chargent de la phase d'authentification des nœuds. En outre, ces nœuds d'autorité fournissent des informations sur la légitimité des nœuds du réseau qui peuvent être utilisées dans le processus de détection des nœuds malveillants. Les RSUs sont déployées pour fournir des information routiéres et environnementales. Les RSUs peuvent également contrôler les panneaux de signalisation en coordination avec les contrôleurs SDN afin d'optimiser le trafic routier. Chaque contrôleur SDN est déployé pour etre responsable de la gestion d'une zone du réseau IoV. Les contrôleurs SDN gèrent le flux de données entre les nœuds du réseau local. Nous supposons que les contrôleurs SDN sont entièrement fiables. Les contrôleurs SDN assurent des fonctions de détection des nœuds non dignes de confiance au niveau local. Ils analysent les données en temps réel provenant des nœuds dans leur zone de couverture afin d'identifier les anomalies dans les comportements. Le serveur cloud maintient une vue cohérente du réseau IoV en obtenant l'état du système de communication à partir des contrôleurs SDN. Le serveur Cloud sert pour une détection globale des nœuds non dignes de confiance.

8.4.1.2 Processus de détection

L'IDS comprend deux modules: (1) le module pour dérivation des attributs des nœuds, (2) le module de classification qui représente la partie décisionnelle de l'IDS. L'IDS commence par extraire les attributs utiles à partir des données collectées. Ces attributs sont ensuite trans-

mis au module de classification pour distinguer les comportements anormaux des nœuds. Les contrôleurs SDN et le serveur Cloud hébergent un algorithme d'apprentissage de confiance léger. En effet, l'apprentissage s'appuie sur des métriques liées aux propriétés du nœud reflètant la confiance des nœuds. Nous considérons les propriétés des nœuds suivantes:

• Nœud étranger: le nœud est étranger s'il n'est pas enregistré auprès du nœud d'autorité ou s'il n'a pas d'historique d'interactions avec les nœuds d'autorité. La métrique du nœud étranger évalue la fiabilité d'un nœud en fonction de son statut d'enregistrement au sein du réseau. Un nœud non enregistré est considéré comme un nœud non digne de confiance. Les contrôleurs SDN peuvent surveiller l'enregistrement des nœuds et utiliser cette métrique pour suspecter rapidement les nœuds non enregistrés. En outre, les nœuds qui interagissent avec des nœuds étrangers sont à considérer comme suspects. Les contrôleurs SDN peuvent déclencher une réponse en temps réel lorsque des nœuds étrangers sont détectés, comme l'isolation dans le réseau.

• Comportement de conduite en matière de vitesse: Cette métrique consiste à surveiller la vitesse d'un véhicule sur une route en calculant l'écart entre la vitesse actuelle du véhicule et la vitesse attendue ou moyenne sur cette route afin d'identifier un comportement de conduite susceptible. Ce calcul peut être basé sur des données historiques de vitesse pour la route considérée. Ainsi, lorsque des véhicules sont proches les uns des autres, nous nous attendons à ce que leurs vitesses soient relativement similaires afin de maintenir des distances de sécurité et d'éviter les changements de vitesse brusques. Si la vitesse d'un véhicule s'écarte considérablement de celle des véhicules voisins circulant sur la même voie ou à proximité, cela peut éveiller les soupçons.

• Comportement en matière de traitement des paquets: Le taux auquel un nœud réussit à délivrer des paquets peut être un indicateur de sa fiabilité. Un nœud qui envoie et reçoit régulièrement un nombre similaire de paquets est censé de se comporter d'une manière prévisible. Lorsque un nœud s'écarte de ce comportement attendu, soit en abandonnant des paquets de manière excessive ou en modifiant les attributs des paquets, il est considéré comme une anomalie. En tirant parti des fonctions de surveillance des flux dans le SDN, les contrôleurs SDN peuvent obtenir les statistiques de flux des paquets et déterminer le nombre de paquets abandonnés par un nœud afin de décider sur le statut de confiance du nœud.

Le processus de détection se déroule comme suit. Chaque contrôleur SDN distribué crée son modèle de détection local. Une fois l'apprentissage local terminé, chaque contrôleur SDN envoie son modèle local au serveur Cloud à l'aide du concept d'apprentissage fédéré. Le serveur Cloud agrège ces modèles locaux pour créer un modèle de détection global amélioré. Ce modèle permet une vision plus large sur la fiabilité du réseau. L'agrégation des modéles de détection locaux est pondérée afin de prendre en compte la performance des contrôleurs SDN et l'influence de chaque modèle local sur la création du modèle de détection global. Par example, un contrôleur SDN avec un taux d'erreur faible dans la détection

peut être favorisé. Le serveur Cloud améliore le modèle global en apprenant de nouveaux modèles. Le nouveau modèle global est distribué aux contrôleurs SDN. Après avoir reçu une copie de ce modèle global, les contrôleurs SDN chargent leurs ensembles de données mis à jour localement et entraînent le modèle pendant un nombre prédéfini d'époques. Les contrôleurs SDN renvoient ensuite ces modèles locaux au serveur Cloud. Ce processus itératif d'apprentissage, d'agrégation, d'amélioration du modèle et de distribution du modèle se poursuit sur plusieurs cycles, ce qui permet au modèle de détection partagé d'évoluer et de s'adapter à la nature dynamique du réseau IoV Lorsque une menace est confirmée, les contrôleurs SDN peuvent prendre des mesures locales pour atténuer la menace et appliquer des réponses de sécurité adaptatives. Ces réponses peuvent être adaptées au contexte spécifique de la menace (par exemple, isoler les nœuds malveillants du réseau, renforcer la surveillance, réacheminer le trafic du réseau ou alerter les véhicules à proximité pour qu'ils prennent des mesures de précaution).

En ce qui concerne l'algorithme d'apprentissage, nous avons utilisé Random Forest, CNN à une dimension et RNN à une dimension afin comprendre l'impact de la performance de ces algorithmes dans la structure d'apprentissage fédéré. L'utilisation de modèles d'apprentissage légers avait pour but d'éviter les opérations de calcul complexes. Nous avons comparé les performances de détection en fonction du rappel, de la précision et du score F1. Dans un premier temps, l'IDS proposé nous a conduit à un taux de rappel global atteignant 99,04 %, un taux de précision de ±99,30 % et un score F1 de ±99,17 % . Le CNN et la Random Forest ont montré des performances très proches, surpassant légèrement la RNN. L'IDS proposé a tiré profit de la visibilité offerte par chaque contrôleur SDN pour une détection locale des nœuds non dignes de confiance. Dans un deuxième temps, nous visons à améliorer l'IDS en mettant l'accent sur l'aspect de qualité de service.

8.4.2 Système de détection d'intrusion basé sur la confiance, le clustering, et le SDN

Nous suggérons d'améliorer l'IDS présenté ci-dessus en permettant une détection plus locale des nœuds non dignes confiance. Nous combinons le SDN, l'apprentissage fédéré, le clustering, et la détection basée sur des métriques liées à la confiance afin de tirer profit des avantages de ces techniques. Les zones locales des contrôleurs SDN seront subdivisées en clusters dans lesquels les chefs de clusters opèrent avec le contrôleur SDN local pour détecter les nœuds non fiables. Cela permet d'obtenir une détection plus granulaire et d'améliorer la réactivité de l'IDS. En outre, nous adoptons le concept de multi-tête de cluster et appliquons le critère de confiance pour la sélection des têtes de cluster. Le cadre de l'IDS proposé est étendu sur la base des points suivants:

-Nous définissons une nouvelle topologie de réseau basée sur le clustering des nœuds dans la zone locale du contrôleur SDN. Les nœuds de chaque zone locale d'un contrôleur SDN sont regroupés pour obtenir un réseau plus facile à gérer.

-Nous proposons un processus de détection plus localisé qui sera appliqué aux nœuds par clustering. Nous conservons les métriques liées à la confiance dans le processus de détection. Nous reconsidérons l'IDS collaboratif à deux niveaux qui utilise l'apprentissage fédéré. Nous définissons des têtes de clusters fiables chargés de la détection locale des nœuds malveillants. Le modèle de détection global est hébergé sur le contrôleur SDN locale.

-Nous appliquons le concept de multi-tête de cluster pour une détection plus granulaire. Nous optimisons la sélection des têtes de cluster en utilisant le concept d'algorithme d'optimisation par essaim de dauphins. La tête de cluster définie agit comme un dauphin et observe les nœuds proches pour sélectionner autre tête de cluster fiable.

8.4.2.1 Architecture proposée

La plateforme de l'IDS se compose de trois modules principaux: le plan de contrôle (cœur du réseau SDN), la couche de cluster et la couche des utilisateurs finaux de l'IoV. La couche "cluster" se compose de véhicules "edge" qui agissent comme des véhicules disposant de suffisamment de ressources et d'une puissance de calcul suffisantes par rapport aux véhicules ordinaires. véhicules ordinaires. Le plan de contrôle est défini par les contrôleurs SDN. Le serveur Cloud dans l'architecture sert à la perspective globale de l'ensemble du réseau IoV. Les rôles des composants sont comme suit: (i) le véhicule ordinaire permet la détection de l'environnement, la communication et le traitement des données locales de base, (ii) le rôle des nœuds d'autorité est principalement lié à la délivrance de certificats et à l'authentification des nœuds du réseau, le nœud d'autorité connaît l'identité de tous les nœuds appartenant à sa zone, (iii) les RSU fournissent des données environnementales optimisant le trafic routier, (iv) les véhicules "edge" peuvent être demandés aux véhicules ordinaires pour gérer certaines tâches de traitement et de calcul pour l'IoV. Les véhicules "edge" sont principalement chargés d'assumer le rôle de chef de groupe. Le processus de sélection permet d'identifier les véhicules "edge" ayant les capacités nécessaires pour effectuer une détection localisée des anomalies. En tant que chef de cluster, le véhicule "edge" facilite la communication entre les nœuds du cluster et le contrôleur SDN.

8.4.2.2 Processus de clustering et de détection

Nous associons un schéma de regroupement à notre IDS proposé afin de renforcer la sécurité du réseau et de mieux prendre en compte la qualité de service. Nous organisons les nœuds en clusters pour aider le contrôleur SDN à gérer l'IDS. Le contrôleur SDN est déployé pour gérer les éléments de réseau d'une zone particulière. On considère que les contrôleurs SDN sont entièrement fiables. Supposons que chaque zone géographique, contrôlée par un contrôleur SDN est composée de u clusters ($C_1, C_2, ..., C_u$). Chaque cluster C_u comprend un ensemble de nœuds IoV : véhicules ordinaires, véhicules "edge", RSU regroupés sur la base de la distance et de la similarité de la vitesse, et elle peut avoir un ou plusieurs chefs de

cluster. Tous les chefs de cluster communiquent avec le contrôleur SDN correspondant. Le nœud d'autorité lance une demande périodique de formation d'un cluster et de sélection de son chef. Les têtes de cluster sont élues parmi les véhicules "edge" sur la base de mesures de confiance et de ressources. Le nœud d'autorité joue le rôle de super chef de cluster dans le cluster C_u. Les nœuds de l'interface utilisateur doivent être authentifiés pour rejoindre les clusters. Le processus de sélection permet d'identifier les véhicules "edge" ayant les capacités nécessaires pour jouer le rôle de chef de cluster. Les véhicules "edge" sont évalués pour devenir chefs de cluster dans le cluster C_u sur la base de leur confiance et de leur indicateur de ressources. Le véhicule "edge peut être élu à la tête du cluster si il répond aux critères de fiabilité en termes de connaissances, de communication et d'énergie.

La décision relative à la sélection du chef de cluster suit la règle du processus de hiérarchie analytique floue (Fuzzy AHP). Cette approche permet d'attribuer des poids aux critères utilisés pour sélectionner un chef de cluster. Nous utilisons le concept de l'essaim de dauphins pour optimiser la sélection des têtes de cluster. Le véhicule "edge" sélectionné par le nœud d'autorité doit agir comme un dauphin pour optimiser la sélection des autres chefs de son cluster. Les critères de sélection d'une autre tête de cluster sont basés sur deux facteurs: (i) le véhicule "edge" proche du chef de cluster actuel, (ii) le niveau de confiance maximal parmi les véhicules "edge" proches. Le véhicule "edge" le plus proche ayant un niveau de confiance élevé est désigné comme nouveau chef de cluster. La confiance de ce nouveau chef de cluster sera envoyée au contrôleur SDN. La transmission de la confiance des chefs de cluster élus au contrôleur SDN est utile pour l'agrégation pondérée des modèles de détection locaux. L'IDS collaboratif utilise des caractéristiques liées à la confiance comme données d'entrée pour la formation. Nous reconsidérons les métriques utilisées dans notre première IDS. Les métriques sont liées au comportements en matière de vitesse et d'envoi de paquets. La tête de cluster mesure ces métriques pour ses membres. Étant donné que chaque gcluster possède certaines propriétés, notamment un profil de vitesse similaire, les membres qui s'écartent de ce profil seront signalés par le chef de cluster. Les modèles de détection locaux sont agrégés dans le contrôleur SDN correspondant. Le contrôleur SDN local crée un modèle de détection global basé sur les informations obtenues à partir des clusters locaux formés. Au départ, un modèle de détection global est hébergé sur le contrôleur SDN local afin d'être partagé et amélioré au cours de plusieurs cycles d'apprentissage. Les chefs de cluster obtiennent le modèle de détection global au début d'un cycle de communication. Chaque chef de cluster crée son propre modèle local pour un nombre prédéfini d'époques. Ces modèles apprennent à partir des données de leur zone de clusters respective. Une fois l'apprentissage local terminé, les chefs de cluster envoient les modèles locaux au contrôleur SDN. Le contrôleur SDN combine ces modèles pour créer un modèle global amélioré. L'agrégation des modèles de détection locaux utilise des poids décidés sur la base de la confiance des chefs de cluster afin de donner plus de poids aux modèles globaux. L'évaluation des performances de notre IDS s'est basée sur les performances de détection, les propriétés d'exécution et l'analyse du schéma de clustering.

8.4.3 Conclusion

Nous présentons un IDS collaboratif qui utilise l'apprentissage fédéré et des mesures liées à la confiance dans le cadre de l'architecture SDN-IoV. En combinant ces techniques, l'IoV peut bénéficier d'une détection efficace des anomalies. Le SDN a adapté l'IoV pour permettre une détection collaborative des nœuds malhonnêtes et une adaptation dynamique avec une meilleure confidentialité des données sous le paradigme de l'apprentissage fédéré. Dans un premier lieu, les contrôleurs SDN ont été conçus comme des nœuds IDS locaux qui partagent leurs modèles de détection avec une couche supérieure représentée par le Cloud. Le processus de détection utilise des mesures de confiance liées aux propriétés du nœud (nœud étranger, paquets abandonnés, profil de vitesse). Ensuite, nous avons intégré le clustering pour une prise de décision plus localisée et une meilleure réactivité de l'IDS. Les chefs de clusters ont joué le rôle de nœuds IDS locaux et le contrôleur SDN correspondant a joué le rôle d'agrégateur de modèles de détection.. Nous avons utilisé le concept de tête de cluster multiple pour renforcer la détection localisée et la stabilité des clusters. En outre, nous avons intégré le facteur de confiance pour l'IDS local, où la sélection des têtes de clusters repose sur les aspects de confiance et d'énergie. Cette sélection et a été optimisée en suivant la méthode de l'essaim de dauphins. Le SDN ainsi que le concept de tête de cluster multiple permet à l'IDS de satisfaire aux exigences de sécurité adaptative, de résilience et de disponibilité. Les résultats de l'évaluation ont démontré que l'intégration de têtes de cluster dignes de confiance en tant que nœuds locaux de l'IDS améliore les performances de détection tout en fournissant un profil de qualité de service satisfaisant en termes de fiabilité et d'évolutivité.

8.5 Routage assisté par drone et basé sur la confiance pour l'IoV

8.5.1 Architecture proposée

Nous abordons la sécurité de l'IoV, et la gestion de la confiance du point de vue du routage.Nous nous intéresserons à la sélection de chemins fiables pour la transmission des données. Nous présentons notre protocole de routage assisté par drone pour l'IoV. pour l'IoV. La solution de routage globale comprend les points suivants:

-Produire une architecture hiérarchique pour fusionner l'IoV avec le drone et le Fog computing afin de faciliter le routage de l'IoV assisté par le drone. Dans la solution proposée, les drones agissent comme des nœuds de Fog.

-Gestion du déploiement des drones afin d'exploiter le placement des drones. La distribution des drones est privilégiée suite à une assignation aux zones qui prend en compte la

couverture et les capacités d'apprentissage.

-Produire une stratégie d'acheminement à la demande assistée par un drone pour l'IoV en cas d'instabilité de l'acheminement au sol. Les drones sont impliqués en tant que relais, formant les chemins de données.

-La sélection d'un relais de drone tient compte de l'agrégation de mesures composites de qualité de service et de confiance pour établir le meilleur relais de drone.

Nous distinguons quatre niveaux dans l'architecture : les nœuds IoV au sol, les nœuds locaux/semi-centraux UAV_Fog, la couche centrale de UAV_Fog et le serveur 5G Cloud. Le partitionnement des drones en nœuds de Fog (c.-à-d. nœuds de Fog légers) et en nœuds de Fog centraux est utilisé pour obtenir un meilleur rendement de la couche de Fog, ce qui permet de réduire les coûts. Fog central est utilisé pour améliorer la qualité du service Fog. Les drones sont déployés pour être pleinement exploités. Les nœuds centraux UAV-Fog sont situés entre le nuage et les drones locaux pour mettre en œuvre les fonctions de calcul et de stockage. Le nœud central UAV_Fog sera en mesure de consulter les données historiques du réseau IoV. Les nœuds du nœuds du UAV_Fog central contrôlent et surveillent les drones locaux. En outre, le nœud UAV_Fog central est chargé d'établir le routage multicast des événements particuliers détectés. Les drones locaux sont directement connectés à l'UAV_Fog central. Les drones locaux sont considérés comme le premier point d'accès choisi par les nœuds IoV au sol pour communiquer avec les autres nœuds IoV.

Chaque groupe de drones locaux correspond à un Fog de drones central. Le drone central est pleinement conscient des données provenant des drones locaux. Chaque drone (c'est-à-dire semi-central ou central) dispose d'une base de données locale contenant des échantillons de données contenant des informations sur chaque lien, telles que la fiabilité et la connectivité (par ex. la densité du trafic, la ligne de vue, la largeur de bande disponible, la latence). Chaque drone peut utiliser sa base de données pour définir rapidement des itinéraires optimaux. Nous supposons que chaque drone survolant une zone donnée peut couvrir plusieurs nœuds au sol et établir une communication réussie. Par conséquent, les drones doivent être correctement répartis de manière à opérer des relais de communication à long terme qui relient intelligemment les nœuds terrestres déconnectés. L'affectation propre des drones à une zone permet une stratégie de couverture efficace, préserve la connectivité pendant la période de vol et empêche la rupture fréquente des liens de communication. L'affectation des zones de drones utilise l'information sur la capacité de couverture et la qualité de la détection en fonction de l'apprentissage de la malveillance. Un drone doté d'une couverture et de capacités de détection intelligentes permet une observation approfondie et de meilleurs taux de transmission des données.

8.5.2 Processus de routing

La stratégie de routage comprend (1) l'inondation adaptative à l'aide de paquets de route, (2) la sélection du meilleur relais sur la base d'un score à l'aide d'une sur la base d'un score utilisant l'optimisation multi-objectif, (3) le maintien du chemin en cas d'instabilité du chemin découvert, et (4) la sélection du meilleur relais sur la base d'un score instabilité du chemin découvert, et (4) la phase de post-routage. Nous pensons que ces normes contribuent à un routage efficace assisté par drone. En effet, ke paquet de route est transmis intelligemment au drone qui forme le chemin d'acheminement sélectionné. Il est suggéré que l'utilisation d'un relais extra/next-hop soit incluse dans l'étape de développement du chemin. L'inondation sélective avec une recherche optimisée est appliquée pour sélectionner le premier relais de drone, ainsi que le relais suivant. L'inondation sélective est appliquée pour trouver le premier relais de drone qui calculera le meilleur chemin pour acheminer les données vers la destination. La formation du reste du chemin alterne entre le routage direct et le routage relayé (utilisation d'un relais supplémentaire/next-hop). La stratégie de routage préconise de construire progressivement le chemin, segment de zone par segment de zone, sous la forme d'une succession de drones vers la destination. La décision concernant le chemin est prise sur la base de critères combinant la confiance (abandon des paquets, capacité d'apprentissage du drone, changement de position), et la qualité de service en termes de connectivité, délai et bande passante. La décision présente une optimisation multi-objectifs maximisant la confiance et la qualité de service du relais pour l'acheminement des données. En outre, la stratégie de routage vise à tracer les chemins possibles du drone vers la destination. Lorsque le chemin optimal se déconnecte, nous pouvons converger vers un chemin qui est très proche de l'optimum sans ré-initier le processus de découverte.

8.5.3 Métriques pour la sélection du meilleur route

La sélection d'un drone relais parmi un ensemble de candidats disponibles peut être représentée par des variables de décision discrètes. Étant donné que les drones locaux sont affectés à des zones spécifiques, chaque candidat peut être traité comme une variable binaire indiquant s'il est choisi comme relais de cette zone particulière ou non. Nous considérons une fonction qui maximise la qualité de service, représentée par une équation non linéaire de la bande passante et de la latence, tout en maximisant la réputation, de confiance non linéaire qui prend en compte le taux de chute des paquets, la capacité d'apprentissage et le changement de position. Cette formulation peut intégrer l'algorithme génétique de tri non dominé II (NSGA-II) pour fournir des solutions post-optimales. La maximisation de la réputation et de la qualité de service est soumise aux contraintes suivantes. **C1** Contrainte de confiance. Le relais sélectionné doit satisfaire à un seuil minimal de réputation. La réputation est une mesure composite intégrant le taux de chute des paquets, la capacité d'apprentissage et le changement de position qui sont susceptibles d'avoir des dépendances

non linéaires sur la variable de décision. **C2** Contrainte de qualité de service. Il convient de spécifier les exigences minimales en matière de LoS, de SINR, de latence et de largeur de bande pour le relais de drone sélectionné. **C2-1**. Contrainte de connectivité garantissant que l'indice de connectivité (dérivé de LOS et de SINR) atteint le seuil souhaité. **C2-2**. Contrainte de largeur de bande. La largeur de bande doit satisfaire le débit de données demandé au relais et le rapport SINR. **C2-3**. Contrainte de latence. En utilisant la fonction logarithmique, la contrainte de latence garantit que la largeur de bande du relais sélectionné est suffisante pour satisfaire la condition de latence maximale acceptable. Nous avons utilisé les paramètres suivants pour l'évaluation des performances de notre technique de routage: (i) ratio de livraison de paquets (PDR), (ii) délai de bout en bout (EED), (iii) nombre moyen de sauts, (iiii) ratio d'overhead.

8.5.4 Conclusion

Nous avons traité le routage dans l'IoV. Nous avons introduit une architecture à quatre niveaux IoV-UAV_Fog afin de permettre une configuration favorable au routage assisté par drone. Les drones ont été stratégiquement associés à des zones géographiques. Une couche semi-centrale de nœuds UAV_Fog et une couche centrale de nœuds UAV_Fog ont été mises en place. La couche centrale de l'UAV_Fog ont été les acteurs du routage assisté par drone. La stratégie de routage a permis de choisir le(s) nœud(s) relais de drone optimal(s) formant le meilleur chemin vers la destination en utilisant la qualité de service et la confiance. Le maintien du chemin a également été défini à ce stade, permettant des chemins alternatifs. L'acheminement assisté par drone utilise des concepts d'inondation sélective, d'avidité et d'optimisation. Les résultats de la simulation ont prouvé l'efficacité de notre modèle de routage hétérogène en fonction du PDR, du délai EDD, du nombre de sauts et de l'overhead.

8.6 Conclusion

8.6.1 Résumé des contributions

Ces dernières années, les travaux de recherche ont accordé une attention considérable aux technologies IoV. Cependant, le système de communication IoV comporte des risques de sécurité. Cette thèse aborde les obstacles à la fourniture d'un cadre IoV sécurisé qui maintient la qualité de service tout en atténuant les menaces qui pèsent sur le système de communication. Les solutions proposées s'appuient sur la gestion de la confiance ainsi que sur les technologies émergentes telles que la Blockchain, le SDN, et le UAV-Fog computing en raison de leur propriétés intéressantes pour soutenir la sécurité et la qualité de service dans un réseau critique comme l'IoV.

Nous avons passé en revue les approches existantes basées sur la confiance dans le

contexte des réseaux véhiculaires. Nous avons fourni une taxonomie des travaux examinés sur la base des outils utilisés. Ensuite, nous avons proposé de nous appuyer de confiance avec Blockchain, clustering et SDN pour nos deux premières solutions de sécurité.

En premier, nous avons élaboré une approche basée sur la confiance et la blockchain pour l'IoV. Notre objectif était d'aborder la sécurité en mettant l'accent sur l'intégrité des données de confiance tout en respectant la qualité de service. Dans un premier temps, nous avons proposé de nous appuyer sur la réputation et la proximité de l'emplacement pour établir la confiance au sein du réseau. Nous avons appliqué notre système de confiance à une Blockchain à deux niveaux pour préserver les données de confiance. La blockchain se compose d'une blockchain locale pour stocker la confiance locale et d'une blockchain globale pour exposer les informations de confiance globales. Aprés, nous étendons notre solution en appliquant le clustering pour mieux prendre en compte la qualité de service en termes de continuité et d'évolutivité. Notre proposition a été validée en termes de performances de détection et de qualité de service.

Dans un deuxième temps, nous avons proposé un IDS qui utilise la combinaison de mesures de confiance, l'apprentissage fédéré et la structure SDN pour détecter les nœuds malhonnêtes dans le réseau IoV. Nous avons cherché à étudier l'apprentissage collaboratif de la confiance des nœuds dans un SDN-IoV. Dans un premier temps, les contrôleurs SDN étaient chargés de la détection locale des nœuds malhonnêtes. Ces détections étaient basées sur des métriques de confiance liées aux propriétés des nœuds. Par la suite, nos travaux ont porté sur une détection plus localisée et plus réactive en plaçant les détections locales sur des têtes de clusters dignes de confiance, tout en hébergeant le modèle de détection global sur le contrôleur SDN. En soulignant que la formation de têtes de cluster dignes de confiance a intégré le concept de têtes de cluster multiples afin de mieux prendre en compte les performances en matière de stabilité. Les résultats ont montré que notre proposition offrait de bonnes performances en matière de détection et de qualité de service.

Enfin, nous avons abordé la sécurité de l'IoV du point de vue du routage. Nous avons présenté une solution de routage assistée par drone qui applique la métrique de confiance pour une architecture de réseau fiable de l'IoV. L'idée fondamentale était de trouver le chemin optimal vers la destination en sélectionnant le drone optimal en tant que nœud de relais avec l'objectif d'une confiance et d'une qualité de service maximales. Notre solution de routage s'est avérée fiable d'après les résultats obtenus concernant le PDR, l'EED et le nombre de sauts.

Amal HBAIEB

Doctorat : Systèmes SocioTechniques

Année 2024

Gestion de la confiance dans les réseaux véhiculaires

L'Internet des Véhicules (IoV) expose des exigences en matière de sécurité et de provisionnement de la confiance. Cette thèse porte sur la conception d'une plateforme de sécurité pour l'IoV qui prend en considération l'aspect de la performance. Nous proposons tout d'abord une plateforme de sécurité basée sur la gestion de la confiance et la Blockchain et qui s'appuie sur des métriques de réputation et de localisation pour valider la fiabilité du système de communication. La *Blockchain* est utilisée pour protéger les informations de confiance contre la falsification. En outre, nous étendons la plateforme proposée pour mieux prendre en compte l'aspect de la performance tout en renforçant la sécurité. Nous utilisons le regroupement pour construire la plateforme de sécurité étendue. Ensuite, nous proposons un système de détection des intrusions (IDS) léger qui s'appuie sur l'apprentissage fédéré et les métriques de confiance. Nous adoptons une infrastructure IoV basée sur le *Software-Defined Networking* (SDN) pour construire l'IDS collaboratif. De plus, nous améliorons l'IDS proposé. Nous appliquons le regroupement pour renforcer la détection et les performances globales du réseau. Enfin, nous proposons une solution de routage réactif assisté par drone qui offre un compromis entre la sécurité et la performance. Nous produisons une plateforme IoV-UAV_Fog pour le routage assisté par drone. Le routage proposé consiste à sélectionner le drone optimal maximisant à la fois la confiance et la performance pour servir comme relais.

Mots clés : réseaux ad hoc de véhicules – confiance numérique – systèmes de détection d'intrusion (informatique) – routage (informatique).

Trust Based Management for V2X

The Internet of Vehicles (IoV) opens up new requirements regarding security, privacy, and trust provisioning. This raises the research question of how trust may be concurrently considered during the design of security solution for the IoV. This thesis addresses this question by designing an IoV security framework that maintains the trust between involved actors. We first propose a trust and Blockchain based framework that relies on reputation and location metrics to validate trustworthiness of the communication system. The Blockchain is leveraged to protect derived trust information from tamper. Next, we extend the proposed framework to better take into account the QoS while enforcing security. We use a clustering scheme to construct the extended framework. Besides, we propose to detect untrustworthy nodes through a lightweight federated learning-based Intrusion Detection System (IDS). We adopt the Software-Defined Networking (SDN)-IoV infrastructure to the network to build the collaborative IDS. The trust features-based detection is proposed along with the SDN to enable a QoS-aware IDS. After that, we proceed to improve the trust-based IDS. We apply multi cluster head concept to boost local detection and overall network performances. Finally, we suggest a reactive Unmanned Aerial Vehicle (UAV)-aided routing solution with security and QoS trade-off. We produce an IoV-UAV_Fog framework for the UAV-aided IoV routing. The proposed routing consists of selecting the optimal UAV relay that maximizes both trust and QoS.

Keywords: vehicular ad hoc networks (computer networks) – trust (digital) – intrusion detection systems (computer security) – routing (computer network management).

Thèse réalisée en partenariat entre :

Financé par l'Union européenne

Université de Sfax

ENET'COM – ECOLE NATIONALE D'ELECTRONIQUE ET DES TÉLÉCOMMUNICATIONS DE SFAX

utt – UNIVERSITÉ DE TECHNOLOGIE TROYES

Ecole Doctorale "Sciences pour l'Ingénieur"