V2I Communication Scheme with SDN and Vehicle Clustering

Ran Duo

A Thesis Submitted in

Partial Fulfillment of the Requirements

For the Degree of

DOCTOR OF PHILOSOPHY

Graduate School of Informatics and Engineering

THE UNIVERSITY OF ELECTRO-COMMUNICATIONS

2020 September

V2I Communication Scheme with SDN and Vehicle Clustering

by

Ran Duo

Supervisory Committee:

Prof. Tsutomu Yoshinaga

Prof. Tadashi Ohmori

Assoc. Prof. Satoshi Ohzahata

Assoc. Prof. Suhua Tang

Assoc. Prof. Celimuge Wu

© Copyright 2020

by

Ran Duo

All Rights Reserved

SDN と車両クラスタリングを活用した V2I 通信方式

和文概要

次世代スマートモビリティ社会を実現するためには, 現在の自動運転技術で使用されている単一車両での制御だけではなく, 複数車両を協調させる制御が必要であり, そのために車両間･路車間通信及び協調制御が必要となる. また, 車両の移動特性, 有限の無線リソース, アプリケーションが要求する品質などが多様化しており, それを考慮した最適な通信方式を明らかにする必要がある.

本研究では，路車間通信（V2I）及び車車間（V2V）通信プロトコルに関する研究を行っている．移動性が高い，無線リソースが制限される路車間・車車間通信において，大容量，低遅延といった通信品質を保証することは困難な課題である．まず，高い移動性を持つ車と路側機（RSU）や基地局の間における頻繁なハンドオーバーが通信に影響を及ぼす．また，様々な車両ネットワーク応用を実現するためには, 車両の高密度分布やアプリケーションの多様性を考慮した通信手法の設計が必要となる.

そこで，本論文では,ア）ハンドオーバーの際の通信効率の改善，イ）多様な応用を考慮した無線リソースの効率的な利用，といった 2 点に焦点を当て，下記 2 つの手法を提案している．1 つ目は，セルラー/IEEE 802.11p ハイブリッド車両ネットワークにおける Software Defined Networking（SDN）のグローバル視点を利用したハンドオーバー方式を提案している. またモバイルエッジコンピューティング技術（MEC）を採用し，ハンドオーバー手順においてデータを事前にキャッシュすることにより, ハンドオーバーの際の通信を効率化し, 通信品質の向上を図っている．ネットワークシミュレータ OMNeT++，車両モビリティシミュレータ SUMO を組み合わせて現実的な車両ネットワークシナリオを構築し，提案手法の性能評価を行っている. 既存のハンドオーバー手法と比較しながら，提案手法の優位性を明らかにしている.

2 つ目は，コンテキストアウェア車両クラスタリング手法を検討し，アプリケーション要求に応じたクラスタリング基準を定めている. 具体的は，車両

ネットワークアプリケーションをトラヒックインテンシブ型（スループット重視），計算インテンシブ型（計算量重視），遅延センシティブ型（遅延重視）の 3 種類に分類し，それぞれの特徴に合わせたクラスタリング手法を提案している．また SDN フレームワークを利用しコンテキストの認識を強化している．ネットワークシミュレーションを用いて, 既存の方式と比較することで提案手法の優位性を十分示している．

上記のように，本論文は，路車間・車車間通信において，新たな SDN を活用したハンドオーバー手法とアプリケーションの多様性を考慮した車両クラスタリング手法を提案して, 現実的なネットワークシミュレーションを用いて既存手法と比較しながら，提案手法の有効性を確認している．

Table of Contents

List of Figure

List of Table

Abstract

In order to realize the next-generation smart mobility society, it is necessary to consider not only a single vehicle used in current automated driving technology but also multiple vehicles to realize the coordinated system in wider situations. Therefore, communications among cars and cooperative control mechanisms are required for Vehicle-to-Vehicle communication (V2V) and Vehicle-to-Infrastructure communication (V2I). Besides, considering vehicle mobility characteristics, limited radio resources, it is necessary to clarify the optimal communication method for different requirements of various applications.

In this thesis, the research is dedicated to studying V2I communication protocol. Vehicles possess high mobility characteristic and limited wireless resources which make them difficult to guarantee network communication quality. In road-to-vehicle communications, the network always accompanies with frequent topology changes due to the highly mobile feature of vehicles. Therefore, the frequent handovers happen between the highly mobile vehicle and the roadside unit (RSU) or base station, which affect network communication performance during the movement. In addition, to save limited wireless resources and prevent network congestion, clustered vehicle structure is an effective method. With the development of wireless technology and increasing user demand, various vehicle applications have emerged. Therefore, it is necessary to design a clustering algorithm that considers the high-density distribution of vehicles and the variety of applications.

In this paper, the two studies are proposed, focusing on two points: a) the improvement of handover efficiency, and b) efficient clustering algorithm considering various applications. The first one proposes a handover method in a cellular / IEEE 802.11p hybrid vehicle network using the global perspective of Software Defined Networking (SDN), so that the stable communication connections are kept during the handover. It also employs mobile edge computing technology (MEC) and supports data communication in the handover process by caching data in advance.

Secondly, this thesis discusses the context-aware vehicle clustering method and determines the clustering algorithm according to the application requirements. Specifically, applications are classified into three types: traffic-intensive (throughput-oriented), computation-intensive (computation-oriented), and delay-sensitive (delay-oriented), and the corresponding clustering methods are proposed respectively. The context recognition is enhanced by using the SDN framework. In order to evaluate the proposed method, realistic vehicle network scenarios are constructed by combining the network simulator OMNeT ++ and the vehicle mobility simulator SUMO. The superiority of the proposed method is sufficiently shown by comparing it with the existing method.

Acknowledgment

My research was completed with the help and support of many people. I feel honored to express my gratitude to everyone who helped me throughout the writing process.

Firstly, I must thank to my supervisor Associate Professor Celimuge Wu, who has profound professional knowledge and rigorous academic attitude. During my study life, he made instructive opinions for my research and gave me careful guidance and encouragement when I encounter difficulties and doubts. Besides, he also gives me a lot of care for my life in Japan. I would like to express my sincere gratitude for everything he has taught and helped me during my research.

Secondly, I would like to thank the supervisory committee of my thesis defense, Professor Tsutomu Yoshinaga, Professor Tadashi Ohmori, Associate Professor Satoshi Ohzahata, and Associate Professor Suhua Tang for their valuable comments and guidance about this paper.

Thirdly, I am so grateful to Hirose Foundation and Japan Student Services Organization (JASSO) for their financial support during my research life. Besides, I would also like to thank the International Student Office of the university for their kindly support and consultancy for my study life.

Finally, I want to thank my family who always encourages me and never stops my way forward. Without their love and support, I was not able to go this far. Thanks to everyone who cares and helps me.

Chapter 1

1 Introduction

At the beginning of the thesis, this chapter introduces the research about the V2I (Vehicle to Infrastructure) communication scheme with SDN (Software-Defined Networking) and vehicle clustering. The structure of this chapter is as follows. In section 1.1, the background of the research as well as vehicular network technologies are introduced. section 1.2 explains the motivation and research problems that inspire the research about the V2I communication and clustering algorithm using SDN. section 1.3 lists the contribution of this thesis. section 1.4 explains the overview of the thesis.

1.1 Research background

In modern society, increasing traffic congestion and rising traffic accidents are major challenges. The main reason for the problems is that the detection of surrounding information and the decision of car action mainly depend on the driver's physiological perception ability. However, and the driver's perception ability is usually very limited. Therefore, vehicles have to possess an ability that they can communicate with each other in real-time to form a network and build an Intelligent Transportation System (ITS) [1] [2]to alleviate traffic congestion and improve traffic safety.

With the rapid development of the technology, Vehicular Ad Hoc Network (VANET) [3] [4] [5] [6] [7] [8] technology has become the one solution that is able to support vehicle communication and provide a wide variety of vehicle services. As shown in Figure 1, vehicles are regarded as the network nodes which can build up the large network and realize vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication in VANET.

For V2V communication, industry and academia have proposed many solutions to support high-volume communication data between vehicles. Among them, the most commonly used technology is the Institute of Electrical and Electronics Engineers (IEEE) 802.11p which is originated from the Dedicated Short-Range Communication (DSRC) [9]. IEEE 802.11p-based vehicular networks utilize unlicensed bands at the frequency of 5.9GHz with 7 channels and provide around 300m short-range communication.

In V2I, the vehicle gets network services or internet access by communicating with the base station or Road Side Unit (RSU). With the use of Hotspot and longwave wireless technologies, communication can be set up by Cellular, Wi-Fi, or Wi-Max access [10]. In VANETs, vehicle easily to cause highly dynamic changes for the network topology, because of its natural high-speed feature.

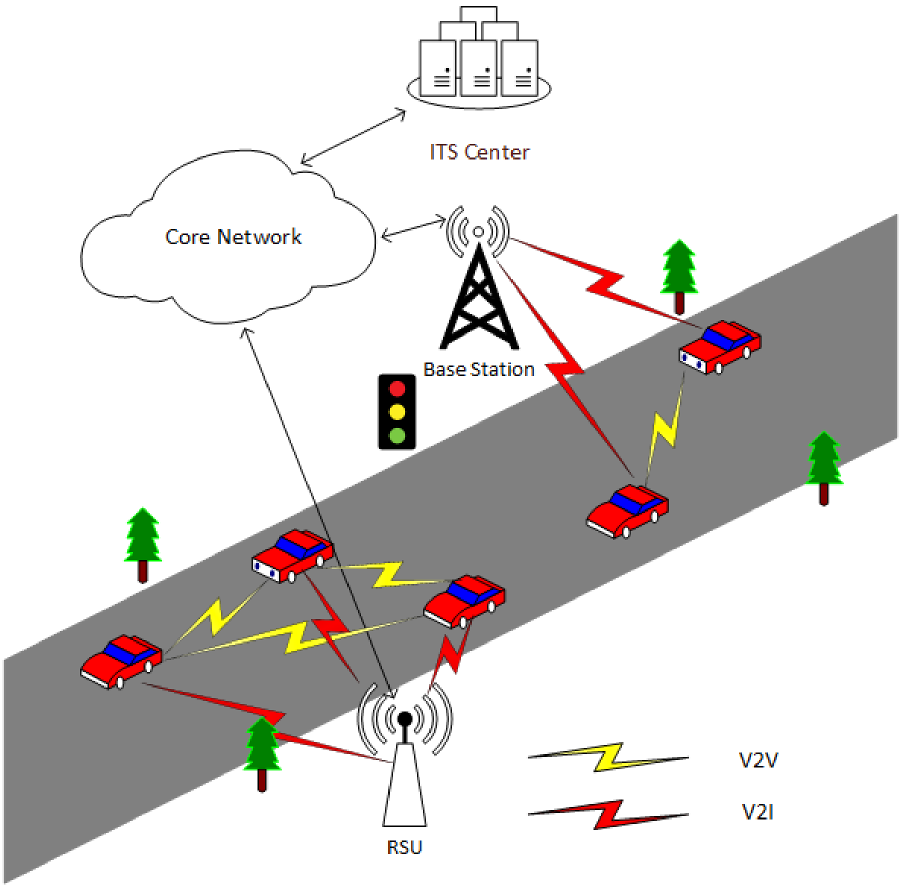

Figure 1 General VANET architecture.

The development trend of vehicular network technology presents a higher requirement for network performance. In order to provide vehicles high reliability and low-latency communications, the Fifth-Generation (5G) [11] [12] [13] cellular network is supposed to meet the network performance requirements of ITS applications [14]. However, the growing density of vehicles that connect with the base station or RSU probably causes a lack of scarce spectrum resources, traffic congestion, and signal overhead. A cluster-based VANET can properly solve these problems.

In recent years, Software-Defined Networking (SDN) technology is regarded as one of the promising technology in improving traditional network architecture by making flexible management and forming programmable networks [15]. At the beginning of SDN development, it is mainly deployed in the data centers and small-scale experiment network. However, nowadays, the SDN concept with its widely used northbound OpenFlow [16] technology has begun to involve in the next-generation wireless network deployment [17] [18].

The ONRC OpenRoads project set up by the Stanford is evaluated in [19], and it concludes that on a large-scale testing platform, SDN could help users with live video application go across different wireless infrastructures [20]. Also, a software-defined WLAN management architecture called cloud-medium access control (MAC) [21] is proposed to support the control of wireless access points from the remote data center which makes the management easier. Furthermore, VANET architecture basing on SDN technology is proposed in [22] and its advantages and the services that are able to implement are discussed. On the other hand, they conclude that the routing method based on the SDN brings higher packet delivery efficiency when compared with the traditional one.

1.2 Motivation

With the increasing number of smart vehicles, the fundamental network challenges are brought about to VANET technology. Most of the challenge is because of the dynamic network topology feature of the VANETs, caused by the continuous movement of vehicles. The high mobility feature of vehicles results in frequent handover between different wireless access and radio channels. The real-time information collection that is required to support ITS real-time applications, may be obstructed by the frequent handover. Furthermore, the growth of network performance and utilization are hindered by the erratic network connection.

Therefore, future VANET are facing with numbers of challenges like the following description:

1) Low latency and real-time application: for real-time applications such as automatic driving and safety services, it is the most basic requirement to support low latency packet transmission in the future VANETs. Future VANETs should support real-time applications, like instruction messages for manage driving behavior, must be transported by very low latency.

2) High bandwidth: for future VANETs, entertainment applications such as high definition video services will require large data traffic. Moreover, some navigation applications who need frequent updates to ensure traffic information synchronization also require the network to support large data traffic.

3) Connectivity: in order to support higher communication quality, VANETs should realize a stable connection for the vehicle in communication. Connectivity between the vehicle and the remote correspondent node should be guaranteed for some safety applications to ensure continuous and highly reliable communication, thereby reducing the possibility of transmission failure during the communication process.

In order to face the challenges described above, it is necessary to propose an efficient communication scheme to support the vehicle's mobility. There are several kinds of research involved in the mobility problem, as the discussion in chapter 2, but they always inevitably faced with the problem of triangular routing, large signaling overhead, and scalable issues.

SDN is a promising future network architectural technology that can be introduced both in the wired and wireless networks. In SDN network structures, the centralized control logic can manage the network function and performance more flexibly and easily, by cooperating with the programmable entity. Since programmable SDN architecture is able to provide flexible control in routing decisions, triangular routing may be eliminated. With standard Southbound Interface, SDN provides better scalability for network function deployments.

Another VANET controlling technology called clustering makes network communication less sensitive to the frequent topology changes as well as saves scarce spectrum resources. There have been various researches about the vehicle's clustering method as discussed in section 2.3. These researches mostly just focus on solving the cluster mobility issue to improve the cluster's mobility, but do not considers from the application side. Few studies explain their aim of supporting specific application requirements and propose corresponding cluster algorithms. However, they just consider the single possible scenario in network communication, do not consider the difference of applications in the requested Quality-of Services (QoS). Therefore, a clustering method that targets different kinds of applications is needed to be proposed. Benefiting from the controller's global view of the network and SDN centralized control provides the ability to execute clustering only when needed. Executed by the SDN programmable device, the SDN controller makes it easier to schedule the handover scheme.

1.3 Contribution

As mentioned in the previous part, there are challenges and opportunities in the VANET where SDN technology is emerged as a promising solution to meet the future network demand. Therefore, SDN-based VANET is applied throughout this thesis considering its programmable feature and global perspective supply. The main challenges in the VANET are based on its highly dynamic network feature, which is meted, in this thesis, by proposing a handover scheme and also, clustering algorithms.

This thesis aims to design an SDN based communication scheme to support large traffic, low latency, and diverse application Vehicular network. Firstly, an SDN based VANET architecture is introduced to provide a global view and programmable network for further research. In order to provide a stable connection for V2I communication, it is necessary to build up an efficient handover approach. However, even though the handover is successfully executed, inevitable data loss exists during the handover. Therefore, the data caching algorithm for the handover period should be deployed.

On the other hand, to save scarce spectrum resources and reduce signaling overhead, the clustered vehicular network is introduced in this thesis. The existing clustering methods which support a different kind of situation, are always mentioned alone without considering the possibility of implement the other kinds of applications. Therefore, propose a clustering method to support a different kind of application requirement is extremely important.

According to the reason mentioned above, this thesis proposes an SDN-based handover scheme and context aware clustering algorithms to support VANET.

For the SDN-based handover scheme, the contributions are as follow:

- An SDN-based VANET architecture is proposed, in which vehicles are grouped into clusters and the network is managed by the two-level controller. Level 1 is responsible for the continuous monitoring of the network status and provides a global view of the network. Otherwise, the level 2 controller is responsible for managing vehicle within its control range, as well as reporting network status to the level 1 controller.

- A Mobile Edge Computing (MEC) server is introduced and deployed on the roadside infrastructure to achieve a caching scheme during the handover and restrict data loss. In order to guarantee the network service smoothly, the MEC server is able to cache the data being received or transmitted by the vehicle that is about to happen handover from other roadside infrastructure.

- Two kinds of handover scenarios are discussed: handover between different cluster head communication range and handover between different V2I communications, to ensure handover integrity.
- The rerouting work after the handover between different V2I communication ranges is explained from the two aspects: the same controller handover and different controller handover.

For the context-aware clustering algorithms, the contributions are as follow:

- For various applications that claim for the different requirements for the network performance, applications are divided into three types, which are delay-sensitive, traffic-intensive, and computation-intensive applications.
- According to context obtained from the cluster, a series of calculations are executed by the SDN controller, so that the essential parameters can be put forward to support further clustering algorithm.
- Different clustering algorithms are proposed, based on the application types to respond to the different requirements of applications. The clustering algorithms related to the above-mentioned parameters that are managed and calculated by the SDN controller.

1.4 Organization of the thesis

The dissertation provides research about solving the problem caused by the vehicle mobility feature by introducing SDN architecture. A seamless handover scheme is proposed to reduce the handover frequency and vehicle's communication connection stability. Besides, context aware clustering algorithm is put forward to support diverse application requirement. In this thesis, these researches are proposed by five chapters that following the description:

- Chapter 1 gives the background and explains the motivation of the research.

Furthermore, the contribution and thesis outline are also described.

- Chapter 2 presents the introduction of SDN-based VANET and explains the benefits that SDN brings about. Besides, the related technology about handover and clustering algorithms are also discussed.

- In Chapter 3, a handover scheme on SDN-based VANET architecture is discussed. The connection stability problem in the high mobility network is discussed by comparing different mobility solutions. Basing on the architecture, the handover scheme is classified into different situations that decided by the vehicle's position in the network and accompanied by a caching scheme at the mobile edge. The simulations are conducted to show the benefits of the proposed handover scheme from several aspects.

- In Chapter 4, a context-based clustering algorithm is put forward and allows clusters possible to meet the requirement of different applications. By discussing the different network scenarios, the applications are divided into different types, according to their requirements for the network performance. Then, different clustering algorithms are proposed and executed via the SDN controller. To evaluate the proposal, network simulation is conducted for different network scenarios, and the results show the superior of the proposal.

- Chapter 5 concludes the thesis and discusses future work for future research.

Chapter 2

2 SDN based vehicular network and related technology

This chapter introduces SDN technology and the research about SDN-based VANET technologies as well as explains the benefits that SDN brings about. Besides, the related technology about handover and clustering algorithms are also discussed.

In section 2.1, SDN technology is introduced and its development path is presented. Not only that, but VANET which is supported by the SDN is also descript in the subsection and the benefits that are provided by the involvement of SDN are discussed. In section 2.2, the related research about the handover technologies are listed from the aspect of both normal wireless networks and vehicular networks. For section 2.3, related research about clustering algorithms is explained from both Mobile Ad hoc NETwork (MANET) and VANET architecture.

2.1 Software-defined networking

Software-Defined Networking (SDN) [23] [24] [25] is considered as a new type of next-generation network architecture aiming to extract the control process (control plane) and packet forwarding (data plane) from the network. At the control plane, formed by several logical entities, the controller is responsible for the monitoring network and collecting information, to improve the management of network resources. At the data plane, data forwarding behaviors are executed on the network devices with wired or wireless communication, according to the instruction of the controller. The communication interface between the control plane and data plane are standardized to support various vendors. The most widely used standardized interface between two planes is OpenFlow [26] protocol which achieves programmable network and flexible control of network traffic. Except for the OpenFlow protocol, many companies developed their southbound API to meet their special needs, such as OpFlex protocol [27] proposed by Cisco and Nicira's NETCONF [28]. The application plane is the third plane of SDN architecture, which deploys various third-party network service applications. The applications at the application layer communicate with the

SDN controller by a northbound interface and indicate their network requirements about security, network resources, and QoS. Consequently, SDN architecture provides a programmable and flexible network and optimizes the utilization of network resources what brings ease in network innovation.

The concept of SDN is firstly introduced by the Clean State research group of Stanford University in the United States in 2006 [29]. They aim to change the existing network infrastructure which is slightly out of date and difficult to evolve and invent a new network architecture. In 2008, the concept of OpenFlow is introduced in detail for the first time, as well as the explanations of several application scenarios by the same group of people [26]. In 2011, the Development and Deployment Initiative (NDDI) partnership, consist of Internet2, Indiana University, and Stanford University, creates a testbed to a new nationwide network platform that allows researchers to apply new network protocols, as well as global cooperation cross different campus [30]. For industrial deployment, Google achieves fully operational SDN architecture at its backbone network indicating the SDN is not just a research model, but a mature applicable product [31]. In recent years, the SDN-based mobile network [32] [33] has emerged as a hot research topic that enables the protocol features to implemented as applications and maximizes the usage of both core network and access network. With the arrival of commercial deployment of 5G, SDN technology is seamed as one of the available solutions for lower power and high efficiency in resource management to support 5G requirements [34] [35].

2.1.1 SDN-based VANET

The SDN natural feature of providing flexible, scalable, and programmable network can help VANETs to overcome most of the challenges [36] [37]. In recent years, SDN-based VANET has become a promising approach to simplify network management and potentiate programmable network which attracts great attention from scholars and researchers. The introduction of the SDN concept in decoupling of control and data plane provides VANET:

1) Virtualization of underlying network infrastructure for VANET application by controllers standardized southbound interface.

2) A centralized network control with the global view of the network monitored continuously by the SDN controller.

As SDN may bring diverse possibilities to the future network, SDN based VANET has become a passionate topic for researchers. For example, to achieve a programmable and flexible network, the research [22] applies the SDN concept into VANET and confirms improvement in mobile device management and resource allocation that SDN brings to the network by comparing SDN-based routing protocol with the traditional VANET routing protocol.

In Figure 2, the general architecture of SDN-based VANET is shown with its three-plane architecture and major components. According to the SDN concept, the architecture is composed of a data plane, a control plane, and an application plane.

1) The data plane consists of physical infrastructures and network devices such as roadside stations, RSU, vehicles, and other network devices in the core network. Vehicles in the data plane create their data information such as speed, position, data processing capability, and historical path from several onboard sensors equipped on the vehicle. The infrastructures like RSU and base stations are able to provide low latency V2I services for vehicles to support local network with local vehicle information.

2) In the control plane, the controller continuously collects and monitors the vehicles' current information by the SouthBound Interface (SBI) and maintains the historical information, thereby forming a global view of the network. By using this information, controllers are able to use flow rules to apply various functions, such as routing, information collecting, resource allocating, and security service to data plane abide by the instruction that release by the application layer entities through NorthBound Interface (NBI).

3) In the application plane, various SDN or VANET applications are deployed as software modules written by the developers/administrators to implement diverse network functions, such as network traffic management, network visualization, and road traffic information management. These software modules on the application layer are able to cooperate with the SDN controller through NBI and use global information of the network to implement network policy on the data plane devices. In this way, with the open NBI interface, the application layer provides great advantages in optimizing network decisions and updating network technologies.

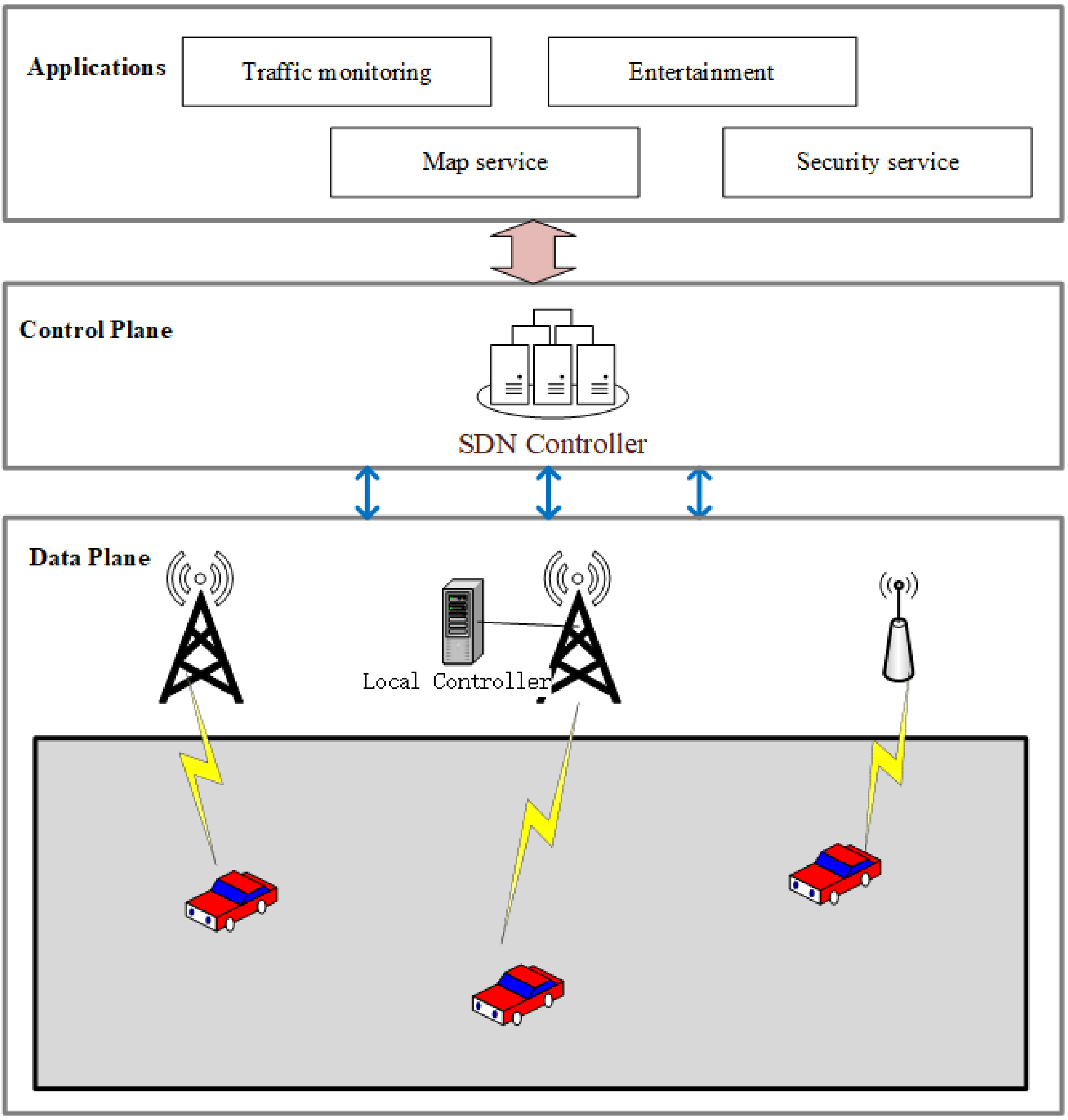

Figure 2 SDN-based VANET architecture.

2.1.2 Related research about SDN-based VANET

SDN provides collaborative work and improves network performance in VANET to solve the challenges facing by VANET. There are several studies on deploying SDN in the VANET, and their works are discussed from different aspects.

In order to achieve low latency and high-quality communication in the network, a new SDN-based VANET is proposed in [38] with the deployment of the Mobile Edge Computing (MEC) cloud server which is used to integrate different access technologies. The result in different kinds of services shows the benefits of providing low latency and high-reliability networks. Moreover, another network architecture is proposed in [39] that not only utilizes the SDN concept but also introduces fog computing technology that migrates cloud computing to the edge of the network to realize security services, QoS management, and location-aware services.

The security and privacy problem is the key challenges for SDN-based VANET. For autonomous technology for driver-less cars, security issue brings about catastrophic effects on traffic safety. Therefore, the author in [40], presents the security problem in VANET and proposes a security policy by combining artificial neural networks and SDN. Considering the responsibility of the controller to provide centralized information management and function of sharing global information, [41] discusses the safety issue in the SDN based VANET and proposes a control scheme to ensure the security of NBI under dynamic access.

In order to facilitate network management, [42] studies a VANET architecture combined with edge computing technology, which uses edge computing technology to distribute common information required by vehicles. This architecture also allows the SDN controller to utilizes beacon information between V2Vs to establish a global view of the VANET. Furthermore, the study [43] is devoted to studying a full SDN network architecture with layered controllers. This architecture provides network devices with flow-based services, unified routing decisions, and handover strategies. Moreover, the study also discusses the programmable handover functions that happen between different network devices.

In order to maximize the network resource utilization, the research [44] proposes a multi-level VANET architecture which regards idle vehicles parked on the roadside as for computing infrastructures and uses SDN to provide rational use of fog computing resources.

Considering the communication scenarios of vehicles in heterogeneous networks, authors in [45] introduce SDN technology in the VANET to enable the control plane to provide centralized scheduling and management of data plane network resources. In this way, SDN provides the best network interface for the mobile nodes under communication and achieves low communication overhead for heterogeneous vehicle communication.

As a promising technology in supporting future 5G networks, SDN is supposed to meet the trend of network demands, such as better scalability, flexibility, resource management, and energy-saving [46] [47]. Different from the typical two-plane (control plane and data plane) structure of SDN, an SDN based three-plane 5G architecture is proposed, where the power plane is introduced additionally so that SDN can be used to monitor and schedule energy consumption in the energy plane.

A three-plane software-defined 5G architecture that additionally introduces energy plane to the network, is first presented in [48] to control energy consumption by using SDN to monitor data flows. A 5G-based software-defined vehicular network architecture is proposed in [49]. The study [50] investigates the security and privacy issue using blockchain in the transportation system and the vehicular IoT environment in SDN-enabled 5G-VANETs.

2.1.3 Benefits of SDN

Joining with SDN, VANET could use a global view of the network to perform effective resource usage in the data plane, which greatly simplifies network management. Besides, SDN applications enable the abstraction of data plane that allows various requirements for the data plane requested by the application, so as to realize the flexible management of network services. In detail, the introduction of SDN technology brings the following advantages to the VANET:

- **Support for heterogeneity:** Network devices and vehicles provided by the different vendors are able to cooperate and communicate with each other due to the standardized SBI of the SDN controller to support heterogeneity.

- **Global view:** The controller continuously monitors underlying network devices and collects vehicles' mobility information, to build up and maintain the global view of the vehicular network. The global view can be shared by the virtualization of networks basing on the application.

- **Improve resource utilization:** Providing a global view of the network, the SDN controller can improve network resource utilization in VANET. For example, with the global knowledge in wireless resources, the SDN controller could help to decide the occasion of changing in the access network and select a suitable wireless interface for better communication quality.

- **Improved network security:** SDN enabled switches distributed in the network are able to obtain traffic information through network traffic analysis and various network anomaly detection tools. At the same time, the SDN controller collects the information from the SDN switches through the standard SBI and combines the data plane interaction information to analyze the network security and maintain the global view of the network. In this way, the SDN controller can issue new security policies to network entities to avoid network security threats.

2.2 Handover technology

In VANET, the most pending challenges are related to the vehicle's mobility feature. Mobility of vehicle accompanies by the handover between different wireless access such as roadside base station and RSU which results in the connectivity problem. To meet VANET requirements for lower latency, high capability, and reliability network, the connectivity problem is an important issue to be solved, in other words, an efficient handover scheme is urgent needs to support vehicle's mobility.

2.2.1 Description of handover

Handover of vehicle communication means, the vehicle getting services through communication with one point of attachment (PoA) changes its communication method by accessing another PoA. According to the access types of vehicles to PoA, the handover scheme can be classified into two types.

1) Horizontal handover: happens between two PoA using the same type of network and wireless access technology, which happens most often in the VANET.

2) Vertical handover: happens between two PoA with the different wireless interfaces using different wireless access technology, which infrequently happens in VANET, because of additional cost to the vehicle.

The handover scheme is triggered by special decisions according to different network quality parameters, such as Received Signal Strength, Received Signal Quality, Vehicles Speed, traffic type, and Available Bandwidth [51]. According to the reason of triggering handover, there are two types of handover scheme [52]:

1) Link layer handover: A handover between different PoA in the same network, that triggered by the reduction of received signal strength or received signal quality or differentiation of vehicle speed.

2) Network layer handover: A handover to a different PoA of the new network, that implements the vehicle's new addresses, triggered by the traffic type and available bandwidth.

2.2.2 Traditional handover

2.2.2.1 Handover in the network layer

To support the stable connection in the mobility environment, there are many traditional handover managements proposed at the network layer. The earliest and most well-known study [53] stipulates the Mobile IP protocol aiming to keep network connection during the communication and make the communication peer always find the Mobile Node. To achieve these goals, MN is equipped by Home Address (HoA) which indicates the permanent address owned by the mobile node to identify the individual MN and Care-of Address (CoA) that indicates the current network position of MN for routing decision. In order to track MN's position, an entity called Home Agent (HA) is introduced to store the mapping relationship of MN between HoA and CoA, and another entity called Foreign Agent (FA) is introduced to provide routing services between MN and HA and located in the current network of MN. When the MN handover between the different network, it registers its new mapping relationship to HA between HoA and the new CoA to its HA by directly sending to HA or by new FA. After the registration, the packet with the destination to the MN is firstly sent to the HA directed by the HoA of MN, then tunneled from HA and FA, according to the registered mapping relationship, and finally, detunneled and delivered to the MN. Besides, MN can send packets directly to the correspondent node.

Comparing with Mobile IPv4, Mobile IPv6 [54] brings lots of improvement in the handover scheme. By defining a new IPv6 protocol function and introducing an additional option field to support mobility in the IPv6 network, the mobile node could directly receive the packages through its CoA instead of forwarding through HA. When handover happens, MN registers its mapping relationship between CoA and HoA to the correspondent node directly. The correspondent node receives packets with CoA as the source address, directly from the MN, and then replaces the source address with the HoA before passing to the upper layer. Different from Mobile IPv4, the correspondent node can send packets to MN without tunneling between the HA and FA, to solve the triangular routing problem.

In order to support handover in a heterogeneous network with IPv4/IPV6 hybrid environment, [55] discusses the different handover procedure according to the different hybrid network scenarios. These handover schemes are in the most used to support network mobility, but not the best choice in the VANET, for they are always accompanied by the triangular routing problem, which is not suitable for the low latency, high resource utilization requirements of VANET.

2.2.2.2 Handover support in the transport layer

Except for the network layer's solution, the transport layer's multi-path protocols are able to support keeping the same connection for the handover of the mobile node. The extension researches are based on the most well-known transport layer protocol Transmission Control Protocol (TCP) [56] and Stream Control Transmission Protocol (SCTP) [57].

In SCTP, the connection between the communicating nodes is called association that naturally supports multiple source-destination address pairs. In the basic SCTP transmission scenario, there are one source-destination address pairs in an association is considered as the main path, otherwise, the remaining path as a backup to prevent network failure. Encouraged by its multi-homed advantage, Mobile SCTP (mSTCP) [58] is proposed to support mobility in the transport layer. When MN happens handover to another network, mSCTP informs the correspondent node of a new address assigned by the new network and adds the address to its current association as soon as the new address is detected. After achieving the handover, MN can communicate with the correspondent node directly by new source-destination pairs interruption of the association.

For the TCP is the most widely used transmission protocol, there are several solutions in supporting handover. An extension protocol MPTCP (Multi-Path TCP) [59] [60] is the most active TCP based protocol in supporting a stable connection with the address change. Different from basic TCP that just allow one source-destination address pair in one connection, MPTCP allows one connection to build up and include multiple Sub-flow, where each Sub-flow represents different address pairs. The Sub-flow in the connection can be added and deleted according to the requirement. When the handover happens, a new Sub-flow represents the new path between MN in the new network and correspondent node is added in the current connection, so as to keep the stable connection in the mobile network.

2.2.2.3 Vertical handover

Besides, there are many studies on supporting vertical handover in different network interfaces. The author in [61] discussed handover decision between cellular network and Wireless Local Area Network (WLAN) in a heterogeneous wireless network where Kalman filter and fuzzy logic are used. For a solution in other aspects, split signaling packets from the backhaul channel in a wireless mesh network may reduce the handover latency [62]. The study [63] uses channel average nonfading duration to make handover decisions and maintains reliable links by applying a preemptive handover strategy and achieve reuse of available links. These studies mentioned above can effectively support the handover caused by the large-scale movement in global mobile networks but are not appropriate ideas for frequent handover scenarios in smaller-scale networks.

2.2.3 VANET handover

In order to support the vehicular network with highly dynamic topology, there are a large number of handover approaches have been proposed in the VANET.

2.2.3.1 Pre-handover

For common cases, the delay caused by the handover scheme is always a serious problem in improving network performance. These handover schemes include wireless resource allocation, connection establishment, authentication, and forwarding path creation takes time to achieve successful handover.

To solve the handover latency problem caused by the Mobile IP handover scheme and achieve improvement in user experience on VANET applications, a pre-handover method [64] is proposed. In this method, the vehicle can get the IP address from the other vehicles and predict handover possibility with the help of the vehicle moving forward so that the handover scheme can be executed in advance. This pre-handover scheme is designed to be applied for vehicles moving on the straightway in the same direction and is proved to effectively work in real-bus and virtual-bus scenarios. For another example, the idea of combining NEtwork MObility (NEMO) and VANETs is proposed to support seamless handover in different wireless access [65]. In order to minimize handover latency, neighboring nodes can provide information about available new access point for the vehicle in advance, so that the vehicle can immediately connect with the new access point when the handover happens.

Before performing a handover scheme, the vehicle can get the available information of new access points in advance through assistance from other vehicles, so that the new access point can be configured immediately according to the need of a vehicle, reducing handover latency.

2.2.3.2 VANET cluster handover

To satisfy the VANETs needs of scalability and reliability and high utilization of network resources, some studies divide VANETs into different clusters to support the handover issue. In this case, the handover may happen in diverse network scenarios such as the cluster's handover between different wireless access and vehicle handover between different cluster heads [66]. In the study [67], a second cluster head is selected as a backup to ensure the vehicle's handover between different cluster heads by prediction about cluster status using vehicle movement information including speed and location. but it does not discuss the scenario that the possibility of cluster's handover between the base stations. Otherwise, clustered architecture can reduce handover frequency in the VANET by the appropriate clustering algorithm, but causes extra overhead signals, in order to maintain a stable cluster.

The research [68] is dedicated to maximizing the connection time between vehicles. It considers the road topology and the possibility of vehicle movement to measure the failure time of the communication link, to reduce the handover frequency. To ensure communication security, the access point needs to authenticate the identity of the new vehicle that happens handover from other access. For this reason, [69] proposes a group authentication scheme in 5G environment to reduce the cost of authentication calculation overhead when handover occurs, and proposes a clustering algorithm considering relative speed, position, and signal strength to reduce the handover frequency.

Considering the handover between radio spectrums, [70] proposes a way to proactively predict channel handover by estimating the available channel using the hidden Markov model. In addition, it can send the most lightweight important information extracted from the data to the destination in advance, so that the user can be less sensitive to the handover.

As the main standard for vehicular communications, IEEE 802.11p faces inevitable challenges such as network scalability, improvement in capacity and intermittent connectivity. To meet these challenges, Long Term Evolution (LTE) mobile communication technology is introduced in clustered VANET architecture [71]. To support this hybrid architecture, a novel vertical handover protocol is proposed in the study [72] to support handover between IEEE 802.11p and LTE wireless access. Although these proposals do improve the handover performance but have not noticed the influence of handover at the transport layer.

There exist a study [73] that takes the effect of the transport layer into account. it proposes a beacons control data dissemination protocol to allow the vehicle's TCP connection to be interrupted and reestablished actively during the handover scheme, thereby reducing the network performance degradation caused by TCP connection interruption. Even though the effect of the transport layer is discussed in the above research, but the connection disruption has not been eliminated.

2.3 Clustering technologies

There is a challenging network environment that hinders the deployment of VANET. It is especially delicate to the hidden node problem and is easy to be influenced by the on-road or roadside obstruction, as well as limited by spectrum resources. So the concept of the cluster is proposed forming a distributed architecture via a specific clustering algorithm [74].

The concept of clustering is to logically gather vehicles with close geographic locations into a group through some algorithms. The size of the cluster is related to the wireless radio transmission range of network devices in the cluster and different from one to another. According to the clustering algorithm, at least one vehicle is selected as a cluster head and the rest of the vehicles are performed as cluster members. Under normal circumstances, each vehicle in the cluster may be selected as the cluster head. However, according to different clustering algorithms, the appropriate cluster head has to be selected to meet different network requirements. The role of cluster head in the cluster varies because of the different clustering algorithms that may be responsible for the routing function as a relay node or selection of cluster members.

The general procedure for the clustering algorithm can be classified as follows:

1) Cluster initialization: At the beginning of the clustering algorithm, every vehicle that is willing to join in the VANET is initialized as an only member in the cluster. Then,

the vehicle begins to announce its existence periodically by broadcast message to its neighbors and collects the surrounding information by announcements from neighborhood vehicles.

2) Cluster head selection: After the collection of information about its surroundings, vehicles in the cluster elect a suitable node to be their cluster head. Every vehicle in the cluster tries to establish a cluster relationship with a selected cluster head or the vehicle selected as a cluster head announces its roles to the neighborhood vehicles to form a cluster.

3) Cluster maintenance: After the establishment of the cluster, the vehicle periodically queries the status of the cluster member and surrounding information to maintain the cluster. The cluster may change its status according to the corresponding algorithm. For example, a vehicle that requests for the VANET service can join into the cluster or a vehicle moving out of the cluster coverage can be deleted from the cluster or when some relatively small clusters are close enough, they are allowed to merge into a big cluster.

The processes of the clustering algorithm are commonly the same as the above description in both MANET and VANET environment. The main difference between the two kinds of networks is that the nodes in the MANET can move freely in the field but on contrary, nodes in the VANET are limited to a certain pattern [75].

2.3.1 MANET clustering

As VANETs is a subset of MANET, at present, there are many studies about clustering algorithms in VANET are originated from the MANET.

The earliest notable clustering research is started from the study [76] for MANET to form a distributed network. The algorithm introduces a site which is assumed to know the identities of all network nodes, and then choose one node as a cluster head that has to ask other nodes to build up a cluster. This process is carried out several rounds, so in this way, all nodes in the network are separated into clusters. However, because only one cluster can be generated per round, the clustering for all nodes takes a too long time.

The simple clustering algorithm is proposed as popular Lowest ID [77] and Highest

Degree [78] algorithms. These two methods are very simple, only based on single information in clustering. For the Lowest ID, each node is pre-assigned a unique ID within the whole network. Through the periodic broadcast of nodes, select the node with the smallest ID as the cluster head and added its neighborhood nodes as cluster members. The maximum connectivity algorithm selects cluster heads based on the connectivity of nodes. Each node calculates its connection degree according to the interactive control information and broadcasts it to the network. The algorithm selects the node with the highest connection degree as the cluster head, and the neighboring nodes as its cluster members.

Mobility Clustering (MOBIC) [79] is one of the most frequently mentioned MANET clustering algorithms which is based on the Lowest ID algorithm and optimized by introducing signal power levels and mobility metrics. The nodes in the network broadcast the hello message to each other, then measure the signal power levels and calculate the mobility metrics according to the received hello message. The cluster head can be selected by the value of the lowest mobility metric and its neighboring nodes can join the cluster as cluster members.

2.3.2 VANET clustering

Different from the MANET scenario with distributed location and random movement in the network, VANET clustering possesses superior conditions in designing clustering algorithm, because of the vehicles' regular movement and established road infrastructure. Many types of research have proceeded on clustering technology for VANETs.

Table 1 shows an overview of the clustering algorithm which includes both the MANET clustering algorithm and the VANET clustering algorithm. For the VANET is a special form of MANET, the clustering algorithms for MANET are also able to use in the VANET.

Table 1 Clustering algorithm overview

Research name	**Category**	**Metrics**
Ephremides et al. [76]	Non	ID
Lowest ID [77]	Non	ID
Highest Degree [78]	Connectivity degree based	connectivity
MOBIC [78]	Mobility based	Signal power
Rawashdeh et al. [81]	Mobility based	moving directions, relative speed

CDS-SVB [83]	Mobility based	speed, moving direction, relative location
APROVE [84]	Mobility based	affinity metric
MPBC [85]	Mobility based	relative speed
DBC [86]	Traffic based	density of connection, road traffic, link quality
Maglaras et al. [87]	Traffic based	possibility of traffic flow, connectivity ability, average distance, average velocity
Ucar et al. [88]	Mobility based	average relative speed
HCA [89]	Connectivity degree based	connectivity status
PMC [90]	Connectivity degree based	connection number, neighbors number
Duan et al. [49]	SDN based	signal strength, velocity
Qi et al. [91].	SDN based	social attributes, inter-vehicle, distance, relative speed

2.3.2.1 Mobility aware clustering

For the existing clustering algorithm, the most popular goal in forming a cluster is to keep a stable clustered network.

A stability based clustering algorithm is proposed in [81], which forms stable clusters moving on the highway. The algorithm takes the vehicles' moving directions and relative speed between the different vehicles into consideration for forming stable clusters. The vehicles are divided into groups according to the speed, and the clustering process is started by the slowest vehicle in the group. The vehicle is selected as the cluster head, whose position is close to the center and having speed close to the cluster average speed.

Togou et al. [83] propose the Connected Dominating Set-Stable Virtual Backbone (CDS-SVB). They consider vehicles' speed, moving direction, and relative location to be used as the metrics of the clustering algorithm so that the virtual backbone structure can be formed in high stability and low transmission delay.

The Affinity PROpagation for VEhiclar networks (APROVE) [84] is a unique clustering algorithm that introduces the affinity metric to measure the stability between the vehicle and its neighbors. The affinity metric is defined by two parameters: responsibility and availability shared by the neighbors. The algorithm aims to efficiently minimize the relative mobility and distance between the cluster head and cluster members, and finally builds up the distributed stable clustered network.

In research [85], a Mobility Prediction-Based Clustering (MPBC) scheme is proposed where relative speed is used as the main metric to predict the vehicle's mobility in the clustering algorithm. The node periodically sends Hello to its neighbors and estimates the relative speed based on the Doppler shift of the received signal. The cluster is initialized with the cluster head being selected to perform the minimum relative speed. In the maintenance stage, mobility prediction is performed to support the issue caused by the relative mobility.

2.3.2.2 Traffic based clustering

The research [86] proposed A Density-Based Clustering (DBC) scheme, utilizing the density of connection, link quality, and road traffic conditions to form a stable network. Considering the crossroads map in the urban scenario, the research [87], proposes a lane-based clustering algorithm to form a stable cluster. They assume that the vehicle may move towards an intersection that has three possible traffic flow (forward, left, right) to join in. Every vehicle has to measure its connectivity ability, average distance, and average velocity to calculate the cluster head level. The vehicle with the highest cluster head level will be selected as a cluster head, thus making the lifetime of the cluster longer.

This kind of clustering is like Mobility aware clustering aiming to build geographically stable clusters. These clustering methods are conducive to large-volume data transmission but are not appropriate for the applications that require computation capability. And some algorithms even use a multi-hop mechanism, making them difficult to apply in low-latency scenarios.

2.3.2.3 Connectivity degree based clustering

A multi-hop cluster-based IEEE 802.11p-LTE hybrid architecture is proposed in the research [88] for the first time aiming to decrease the number of clusters and improve network stability. The research uses a matric of the average relative speed between the vehicle and its neighbors in deciding the cluster head to prevent the stability of the network. The vehicle can join in a cluster by directly communicating with the neighbors with the role of cluster head or cluster members instead of multi-hope communication with cluster head to minimize signaling overhead.

A Hierarchical Clustering Algorithm (HCA) is proposed in [89] aiming for forming clusters as fast as possible. Different from other clustering algorithms, it constructs clusters according to the connectivity status from the neighborhood vehicles, instead of the vehicle's location pattern. Otherwise, it deals with the channel access and manage the data flow transmission into the cluster to avoid traffic collision and achieve reliable communications.

In recent researches, [90] proposes a passive multi-hop clustering algorithm (PMC) to construct a large scale, high stability cluster. It introduces a priority-based neighbor-following strategy to carry out CH selection which is calculated by the node's connection number and neighbor's number. The vehicle selects its neighbor with the highest priority into the same cluster, in this way, improves the stability and reliability of the cluster.

2.3.2.4 SDN-based clustering

There are few clustering algorithms based on SDN architecture are proposed by the researchers. The natural function of SDN in information collection and centralized control can help to perform the clustering scheme. For example, the research [49] uses SDN to achieve adaptive vehicle clustering and designs a dual CH as a back-up CH used in the cases that emergency occurs or CH leaves to ensure the network robustness. Considering the growing number of vehicles, a social-aware cluster algorithm is proposed in [91] that the SDN's centralized function is utilized. This algorithm records the road segment that vehicles have traversed and form a discrete time-homogeneous semi-Markov model to predict the vehicle's future routes. As a result, the vehicles potentially possess similar future mobility features are grouped into the same cluster to improve cluster stability.

2.4 Conclusion

In this chapter, a brief overview of SDN architecture, handover technology, and cluster scheme is introduced. SDN-based VANET typical architecture is described and the recent researches about the deployment of SDN into VANET are discussed. This chapter also introduced the handover scheme and handover type from different aspects, as well as discussed some typical handover scheme in both low mobility and VANET network environment. Furthermore, this chapter introduced the common clustering processes and presents some current clustering algorithms from MANET to VANET for different purposes.

Although there are numbers of researches that have been proposed in the handover scheme and clustering algorithm to support VANET dynamic network topology changes, the related research still needs to be improved to keep up with the increasing requirement for the network quality. In order to achieve network quality requirements, the SDN-based handover scheme of V2I communication and cluster algorithm is proposed, aiming to increase network throughput and reduce transmission delay.

Chapter 3

3 SDN-based VANET handover

In this chapter, the SDN-based VANET handover scheme is presented to support V2I communication. With the introduction of SDN technology, a view of the global VANET network can help to schedule the handover scheme actively only when it is needed, instead of provoked by periodically detection or the vehicle requirement for the handover.

In section 3.1, three types of typical solutions in helping to solve handover mobility issues are discussed and their shortcomings in supporting the VANET environment are presented. Besides, the advantages of introducing SDN to the handover scheme are discussed.

In section 3.2, SDN-based clustered VANET architecture is shown to support the handover scheme. IEEE 802.11p/Cellular hybrid network topology is used to save scarce spectrum resources and reduce signaling overhead.

In section 3.3, MEC technology is introduced in the handover scheme to support seamless transmission. In the proposed architecture, the MEC server works as an intermediate processor that can cache data previously from the network, so that the data loss during the handover can be significantly reduced.

In section 3.4, the detailed handover scheme is described in two aspects. The one is the vehicle’s handover between two different clusters and the other is the cluster’s handover between different roadside equipment.

In section 3.5, the simulation works are designed to evaluate how the handover proposal affects network performance. Finally, section 3.6 summarizes this chapter.

3.1 Problem discussion

In VANET, the high mobility characteristic of the vehicle makes the wireless network topology highly variable. Under the normal wireless network conditions, vehicles can obtain network services by accessing roadside nodes. However, for V2I communication, due to the high-speed movement of the vehicle, the vehicle always faces frequent handovers between different roadside infrastructure. As a result, it is difficult for the vehicle to maintain a stable connection, especially in the transport layer.

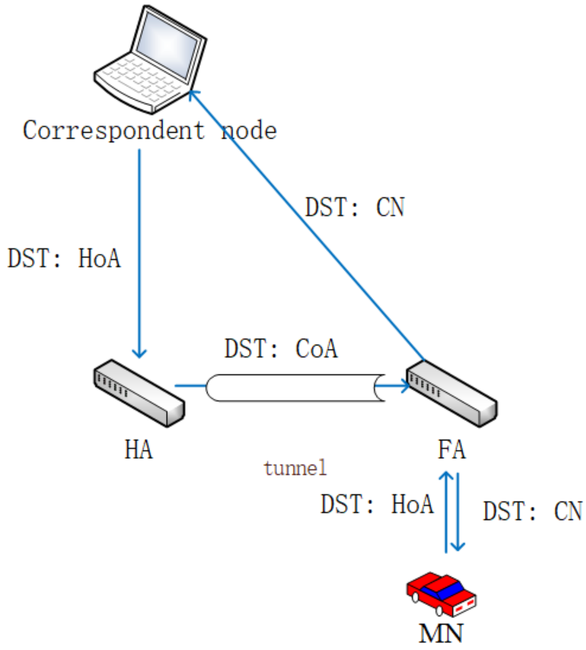

Figure 3 Triangular routing.

Researchers are committed to finding handover methods that can maintain a stable connection to the wireless network with highly dynamic topology. There are roughly three types of these solutions. The first one introduces a new address that represents the location wireless node and communicates with the peer through the tunnel. Mobile IP [53] [54] is one of the earliest and most well-known protocols. Mobile IP is an Internet Engineering Task Force (IETF) standardized protocol. As shown in Figure 3, Mobile IP provides two address to identify a mobile node (MN) where HoA represents the unique identity of the MN and a CoA indicates its network location. In order to manage the relationship between the two addresses, Mobile IP introduces two agents HA and FA. When handover happens, Mobile IP allows the MN to obtain a new CoA from the new FA without changing the HA, and create a tunnel between the new FA and HA. In this way, the MN can achieve handover without the change of address. When the MN leaves its home, all data packets sent from the server to the MN must bypass the HA to reach the destination, which creates a problem of triangular routing. Therefore, large signaling cost and heavy burden on HA is caused by the same reason.

The second type of solution, such as Host Identity Protocol (HIP) [92] or Identifier-Locator Network Protocol (ILNP) [93], does not exist triangular routing problems. For these solutions, when the MN switches, its new location information is notified to all Correspondent Nodes (CNs), so the CN can directly communicate with the MN using the optimal path. These protocols possess similar ideas for mobile handovers. When the MN moves to a new network, its new location information is notified to the correspondent node, so the correspondent node can directly communicate with the MN using the optimal path. However, because the network changes always accompanied by huge routing updates, it may cause large signaling overhead (especially in a frequent handover) and routing update latency during each handover. They are more suitable for global mobility changes but have drawbacks in supporting micro-mobility scenarios.

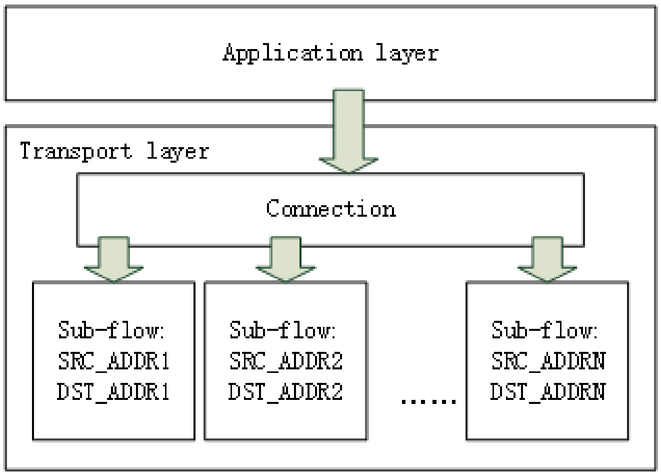

Figure 4 Transport layer solution with multiple flows.

The third type of solution is to bind several sub-flows for one connection by using multiple addresses to the MN in the transport layer as shown in Figure 4. SCTP and MPTCP are the most well-known transport layer solution to supporting multi-address. SCTP protocol establishes a connection called Association for a pair of MN and corresponding node, where the multiple sources can be bind into the one connection. MPTCP is an extension for TCP protocol to support multi-home and multipath transmission. MN can establish a connection to the corresponding node with numbers of Sub-flow, where each sub-flow represents a different source-destination address. In this way, MN can keep the connection unchanged, even though the address of MN has changed because of the handover. However, they require extensive modification of transport layer protocols, it is difficult to modify the standard protocol that is in used and the control messages between MN and corresponding node, which are used to execute the network address binding scheme, brings about signal overhead and additional latency when the handover happens.

SDN architecture can help to solve problems in keeping the connection in high mobility: firstly, programmable network devices in SDN with its network global view can alleviate the bad effect brought by the triangular routing or even eliminate triangular routing. It is because the routing decision for an MN can be flexibly placed on a router that needs to schedule a new routing rule instead of a set of the address registration process. Secondly, centralized control and standardized NBI in SDN help to reduce complexity in network management. Thirdly, the SDN controller schedules the faster handover initiatively without MN address reconfiguration and provides less signaling overhead especially on wireless links.

3.2 SDN-based VANET architecture

To overcome the aforementioned shortcomings, vehicular networks are managed with two-level SDN controller architecture as Figure 5 shows. Level 1 is composed of SDN central controller connecting with the core network. The controller in level 1 maintains a global view of the network and is responsible for the global mobility in handover between different level2 controller domains. Besides, level 2 is formed by base stations that are equipped with SDN controllers and MEC servers which is used to proactively cache data in MEC servers to reduce packet loss during the handover. The SDN controller in level 2 manages the part of the network that is in charge of its domain. The controller is equipped with routing management ability and is able to possess detailed information on the vehicle under its jurisdiction. for the inter-domain handover, Level 2 controllers are able to manage the mobility of the network by itself, where the vehicle handover to a new base station or RSU which is under the same controller's management.

As Figure 5 shows, distributed clusters are built by geographically adjacent vehicles, and every network nodes that are shown in the architecture is the SDN enabled device. Both IEEE 802.11p interfaces and cellular interfaces are equipped on the vehicles for different purposes. Therefore, the wireless network is built based on two kinds of wireless communication technologies. In order to support large amounts of data transmission in a highly dynamic mobile environment, wireless protocol IEEE 802.11p is used for short-range communications among vehicles. The IEEE 802.11p-based VANETs provides vehicle-to-everything communication by utilizing unlicensed bands. The communication parameters between vehicles are set based on the IEEE 802.11p specification, providing about 300m communication distance and working at the spectrum of 75MHz at 5.9GHz with 7 channels with the ideal data transmission rate in the range of 6Mbps to 27Mbps.

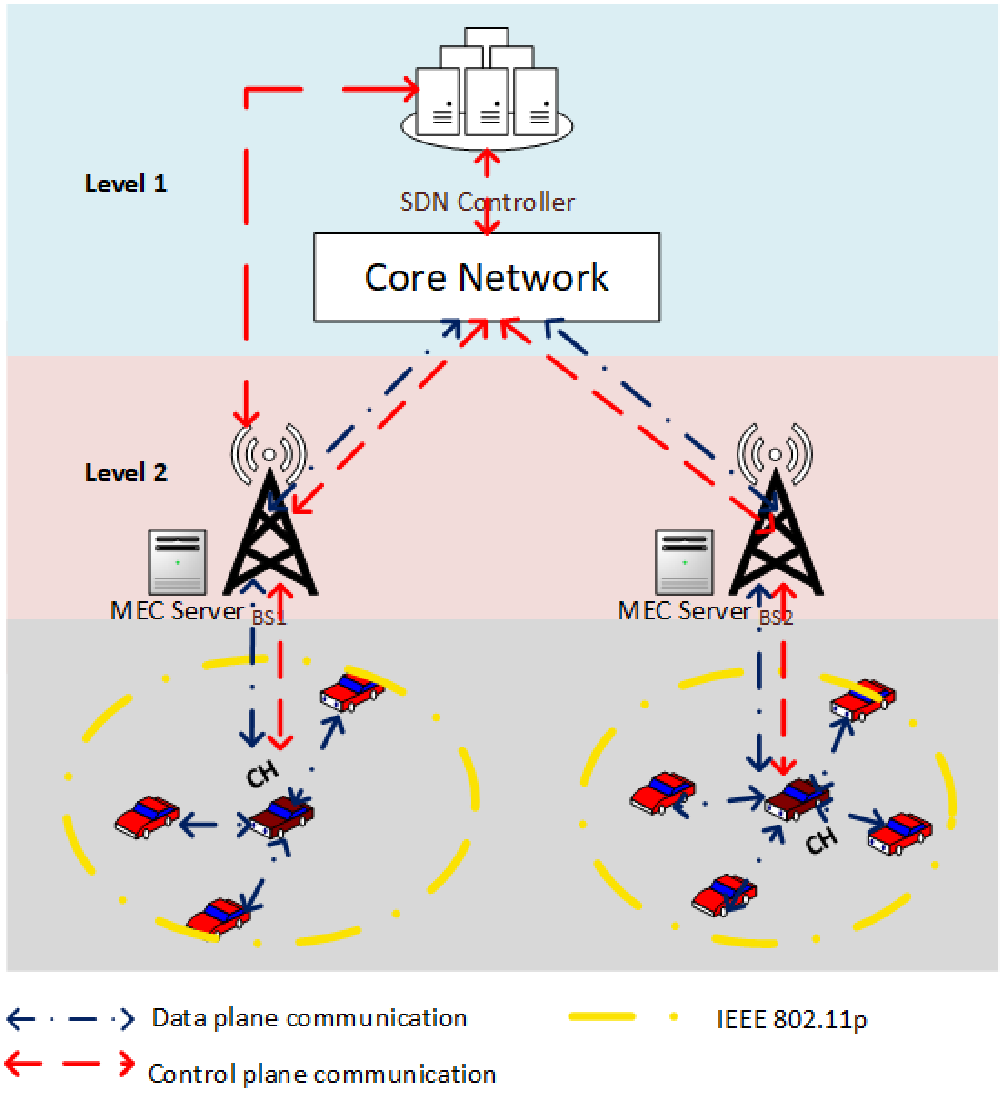

Figure 5 SDN-based clustered VANET.

Besides, cellular network (such as LTE and 5G network) is used to support long-range wireless connection providing communication between vehicle and base station [94]. On one hand, the vehicle can connect to the Internet through communicating with the base station or RSU infrastructure by the cellular interface. On the other hand, a CH with the cellular connection to the RSU or base station always works as a gateway for the vehicle in the same cluster. A vehicle that acts as a cluster member, is able to get Internet services through CH by IEEE 802.11p to save scarce spectrum resources and reduce signal overhead during the handover scheme.

The architecture introduces SDN technology to achieve information sharing and collaborative control among network devices. SDN separates the network control functions from the infrastructure to form an individual control plane, to achieve various network management functions, including traffic control, network monitoring, and cluster management, by implementing applications on the SDN controller [95]. In this architecture, the control plane is formed by controllers both in level 1 and level2. The controller in level 1 provides a global view of the network and is responsible for global mobility issues, such as a vehicle moving to another network area after a long trip. The controller in level 2 provides local network decisions to provide quick response and less control overhead.

The Data plane consists of various SDN enabled devices, including vehicles willing to get network services, infrastructure such as base station and RSU, and other devices in the wired network. In the data plane, SDN enabled devices process packets according to the specific established policies which are distributed or modified by the SDN controller.

3.3 Caching scheme during handover

In the general handover scheme, the data is transmitted through the old base station may not be completely received by the mobility node. In this case, the data are lost but are not recognized by either vehicle or correspondent server in the core network. Therefore, the server keeps sending new data without noticing the data loss, and simultaneously, the vehicle requests for the lost data to get a complete service. The server will not respond to the vehicle the right data until it receives the same request three times or happens a timeout according to the congestion control in the transport layer. As a result, the vehicle costs a long time before getting its desired data. Therefore, in order to reduce the losing data, MEC [96] technology is introduced in the handover scheme.

3.3.1 MEC introduction

A new concept Mobile Edge Computing (MEC) has emerged as a solution of future network with the growth of data that are generated by the mobile devices, a new concept Mobile Edge Computing (MEC) emerges. It is firstly introduced by the European Telecommunications Standard Institute (ETSI) aiming to build up a new platform that is deployed close to mobile nodes and provide IT and Cloud computing functions within the Radio Access Network (RAN) [97]. On the contrary, compared with computing, which is more centralized, the MEC server is deployed at the edge of the mobile network, which means closer to the mobile node. It brings the computation ability and storage resources to the mobile edge instead of the core network. Since the edge is localized, it can immediately respond to the changes in the surrounding environment to support the need for low latency and location-aware services, so the MEC is used as an effective way of offloading. MEC architecture is considered to support several application cases, such as computational offloading, content delivery, video caching, and mobile big data analytics. Deployment of MEC at the base station enhances computation and avoids bottlenecks and system failure [98].

Nowadays several studies are attracted by the MEC advantages, such as, [99] considers the limited resources of vehicles and introduces MEC technology to mitigate the burden of vehicles, such as traffic and computation which saves vehicle energy consumption and reduces service delay. Considering the influence of storage cost and transmission latency, the study [100] enhances the quality of caching experience to better support MEC in traffic offload. In this thesis, MEC is employed as a caching server at the base station that caches the data proactively to support handover. To optimize the offload policy on the mobile edge considers the random mobility feature of the vehicles and their handover possibility, and proposes a mobility-aware computation offloading design. The design makes the offloading decision by comparing and calculating the average total costs of both local computations at the vehicle and remote computations on the MEC server.

3.3.2 Caching scheme

In the proposed architecture, a MEC server is deployed at each roadside infrastructure between the wireless access network and the core network. To ensure high-quality data transmission, the MEC server is used as a caching server during the handover scheme. The transmitting data can be cached by the MEC from both the core network and vehicle under the handover scheme, so as to reduce the time cost of restoring normal transmission and achieve stable network performance when the handover happens. Figures 6 and 7 show the caching process provided by the MEC server when the vehicle happen handover between eNB1 and eNB2.

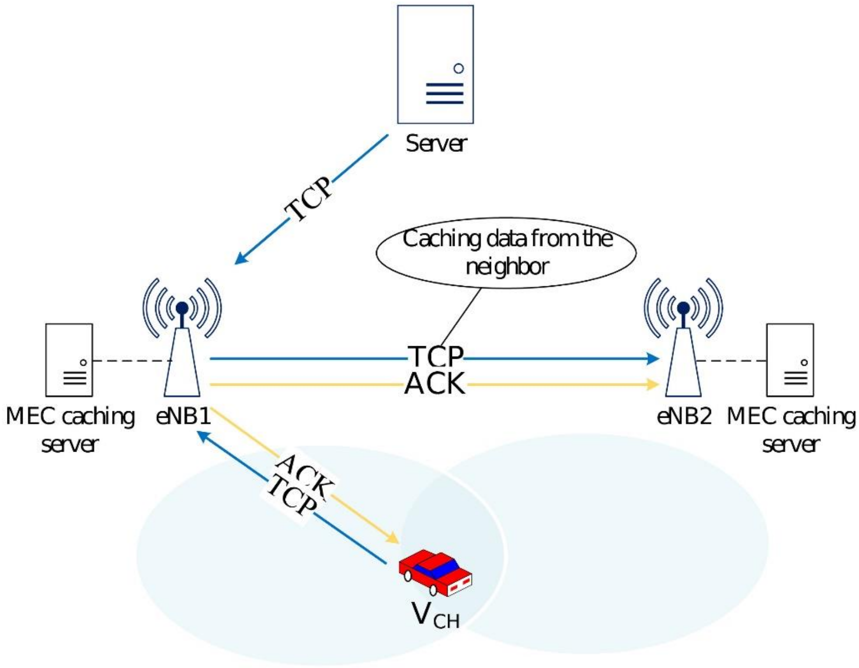

Figure 6 Caching scheme in the handover scheme (before handover).

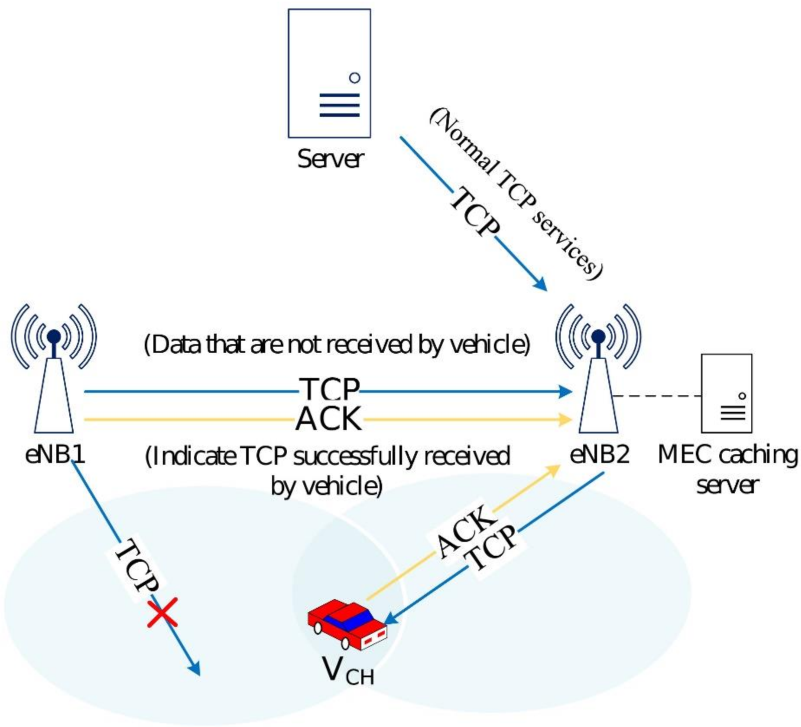

Figure 7 Caching scheme in the handover scheme (after handover).

As shown in Figure 6, when a vehicle is detected to possibly happen handover from eNB1's communication area to eNB2's, the access network starts the handover scheme and eNB2 starts a caching action. At this stage, the handover of the vehicle has not yet completed, so the vehicle communicates with the server still through eNB1. The caching scheme includes two actions includes storing and releasing. On one hand, the MEC server copies the communicating data of the server or vehicle and stores them into the caching queue during the handover. On the other hand, the MEC server collects acknowledge information carried by the communicating data and manages the caching queue by releasing the unnecessary data from the cache according to the acknowledge information.

Figure 7 shows that the vehicle starts to communicate with the server through eNB2 after the successful handover. The data that has already been cached by the eNB2 MEC server will be transmitted to the communicating nodes. Considering the possibility of data loss that happens to the data on the way, in the beginning, eNB2 tries to send a segment to the communicating peers to confirm the data that they request for, or when receives a new acknowledgment message, eNB2 responds to the communicating peers with the data they need.

3.3.3 Caching algorithm

To support the caching scheme, a caching queue is maintained for each service on the base station cache. The MEC module takes part in the handover scheme by caching the unreceived data and drops the received data by analyzing the information carried by the data and keeping the queue at a proper size. Thus, the data are enabled to be transmitted more correctly without extra unnecessary transmissions and minimize the data loss during the handover.

In order to support the management of the cache, five additional variables per service are introduced at a MEC module:

- *first_cached_seq* – stores the sequence number of first data in the caching queue.
- *seq_acked* – represents the highest sequence number whose corresponding data have been received by the vehicle.
- *last_cached_seq* – stores the sequence number of data at the tail of the caching queue.
- seq_i – represents the sequent number of the ith segment.
- ack_i – represents the ack number for the ith received segment.

Algorithm 1 Caching algorithm at MEC server

```
Initialize: first_cached_seq=0, seq_acked=0, last_cached_seq=0
Receive segment i
for Every packet i from wired network do
    if Destination address == vehicle's ID then
        if first_cached_seq=0 then
            Cache the package.
            first_cached_seq=seqi
        else if seq_i < first_cached_seq then
            first_cached_seq = seqi
            Cache the package.
        else
            Drop the package.
        end if
        if last_cached_seq = 0 then
            first_cached_seq = seqi
        else if last_cached_seq < seqi then
            last_cached_seq = seqi
            Cache the package.
        else
            Drop the package.
        end if
    end if
    if Source address == vehicle's ID then
        if seq_acked = 0 then
        else if seq_acked < acki then
            seq_acked = acki
        end if
        if first_cached_seq < seq_acked then
            Release the caching package from first_cached_seq to seq_acked-1
            first_cached_seq = seq_acked
        end if
    end if
end for
```

The caching algorithm is shown in Algorithm 1 deployed at the base station. The MEC server continuously collects available data from the network as soon as noticing the handover of the vehicle. When the data packet bounding for the handover vehicle is received and carrying valid data, the caching algorithm can decide whether to cache the data by comparing the variables in the caching queue and the sequent number of the received data and update the cache queue variables according to the decision. Besides, when the data packet is carrying a new data acknowledgment message, the caching algorithm drops the data that has been acknowledged from the caching queue according to the value of the ACK field on the data packet that denotes the next expected data of the sender and then updates the variables in caching queue. By caching available data using the MEC at the roadside during the handover scheme, the caching scheme effectively reduces the data loss during the handover scheme and even minimize the delay for resuming normal communication.

The proposed caching scheme is a bit like Snoop TCP protocol [101] [102], which deploys TCP cache on the base station to buffer the unacknowledged TCP packet and provide local retransmission. It aims to avoid the influence of the wireless channel from end-to-end TCP services. In contrast, the proposed caching scheme aims to provide the vehicle fast recovery from the wireless handover. Having a global view of the network by the SDN controller, different from the Snoop TCP, the proposed caching scheme can be executed previously, to ensure the transmission reliability. In terms of communication method, Snoop TCP caches the packet directly from the source, and the proposed scheme caches data from the neighbor base station during the handover scheme. In terms of delivering packets, Snoop TCP blocks the duplicate TCP ACK message to avoid fast retransmission but delivers TCP data messages from the TCP server that may cause retransmission time out at the TCP server sides. The proposed caching scheme allows the TCP ACK message to pass through, but avoid the same TCP data messages that have been cached to guarantee the wireless performance.

3.4 Handover scheme based on SDN

When happens handover between the different clusters, the vehicle has to change its network address to adapt to the change of the routing path [103]. Since a connection is identified by a pair of network addresses and port numbers, when the vehicle address changes, the transport layer connection will be released and reestablished. As the data transmission on the transport layer is limited by the congestion window, the network performance is likely to affect by the vehicle’s handover. In order to overcome the problems described above, this thesis proposes an SDN-based handover scheme to ensure the network performance during handover by keeping the transport layer connection unchanged.

3.4.1 Handover scheme

Since SDN architecture enables the network a way of both separated and centralized control, it has huge advantages in network scalability and transmission efficiency compared with the traditional network architecture [42]. The current network challenges can be overcome by these advantages, such as ensuring the mobility of the wireless network, supporting the dynamic characteristics, and managing large-scale networks. In order to maintain a stable connection during the handover, the SDN controller issues the instructions of mapping and reverse mapping actions to the cluster head for the newly accessed vehicle. In the cluster VANET, the cluster head is responsible for the network access of the vehicle and implements the mapping operation of the packet address to support the mobility of the vehicle.

For the proposed network, the vehicle is equipped by the identity address that represents the unique identifier of the vehicle and locational address that is used to indicate the location of the vehicle. The relationship of two kinds of address are maintained by the CH, on the contrary, the locational address is not recognized by the vehicle. As shown in Figure 8, before the handover happens, the vehicle with its identity address: ADR_V and locational address: ADR_1 communicates with the server through CH and eNB1. The server tries to deliver messages to the vehicle's identity address, however, because of the frequent change of vehicle in the network location, it is hard to track the new route to reach the vehicle. Therefore, the data packets towards the vehicle identity address should be redirected to the locational address to achieve transmission on the optimal route.

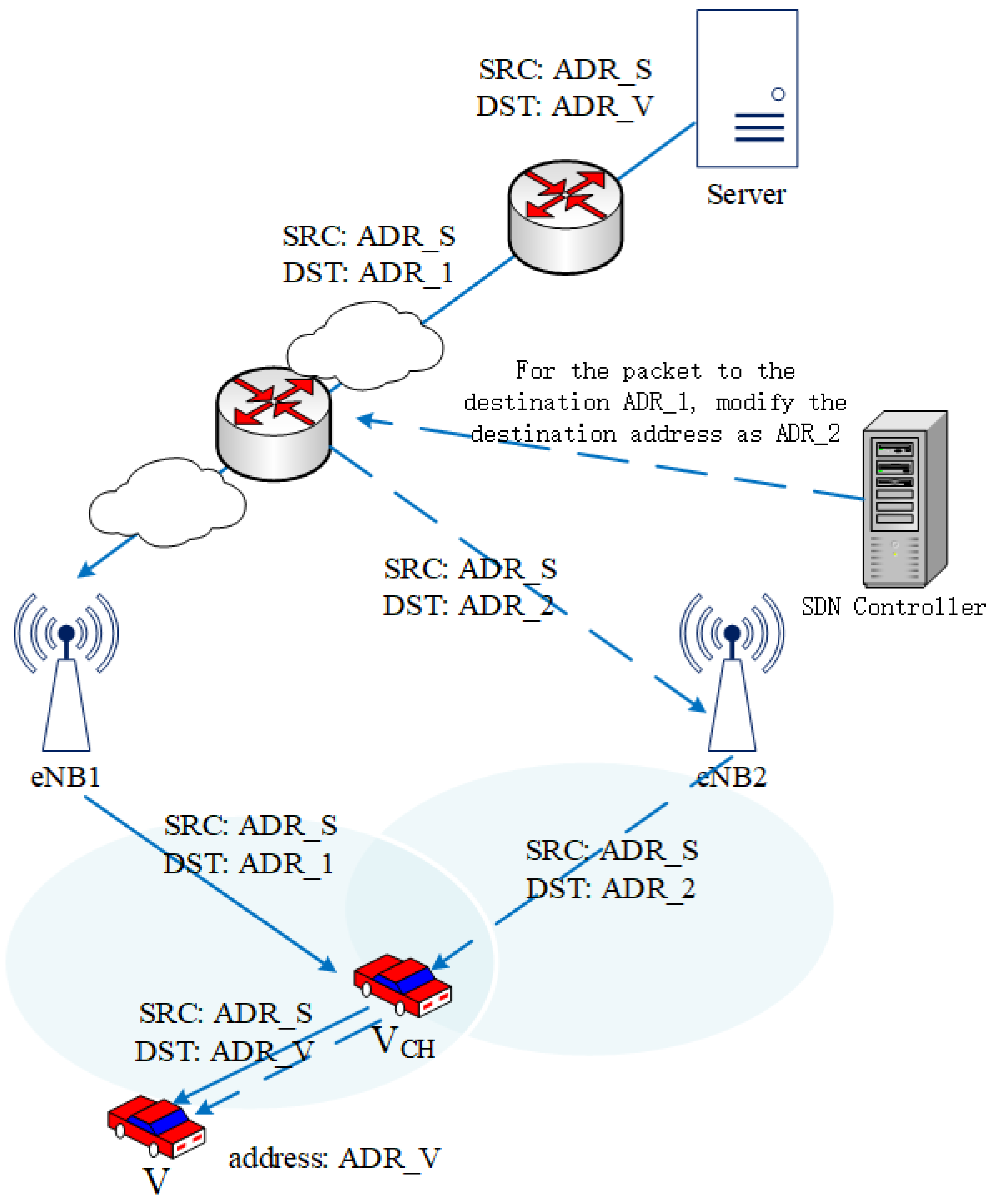

Figure 8 Route change in handover.

Assuming that the CH gives up access to eNB1 and moves into the communication range of eNB2. Simultaneously, detecting the CH's join in, eNB2 assigns a new address: ADR_2 to the vehicle and report to the SDN controller that the location address of the vehicle has changed to the ADR_2. After receiving acknowledgment and learning that the vehicle has just moved, the controller commences on accomplishing the handover by modifying the existing transmission route server-eNB1-CH-vehicle flow path towards the server-eNB2-CH-vehicle flow path. Take the scenario in Figure 8 as an example: since the controller is able to know suitable routing path from server to vehicle, it notices the new mapping relationship of vehicle's addresses to the involving router by issue a flow table that indicates "for all data packets with destination address ADR_1, redirects their destination addresses to ADR_2". Therefore, the new transmission route from service to vehicle will go through four pieces of the path: server to the router, router to eNB2, eNB2 to CH, and CH to vehicle. By using this kind of destination mapping process, the network only required two flow table downloading to the router where the new route and previous route converge, and to the CH indicate the new mapping relationship.

Considering clustered VANET architecture two kinds of cluster-based group mobility scenarios in VANET may occur:

1) Due to the mobility of vehicles, the cluster head may move out of the coverage area of its wireless access, need to perform handover to a new target RSU or roadside infrastructure. For handover does not process any action to the member vehicle, the handover of the cluster head is transparent to cluster members in the cluster.

2) Vehicles act as a cluster member of one cluster may perform handover to join in another cluster. This kind of handover can be triggered by the reason that reselection of the cluster head in the same cluster happens or the department of the vehicle from the cluster head communication range.

The detailed handover scheme is shown in Figure 9. It is assumed that vehicle leaves from the old cluster to a new cluster where the cluster head is CH_0 and CH_n respectively.

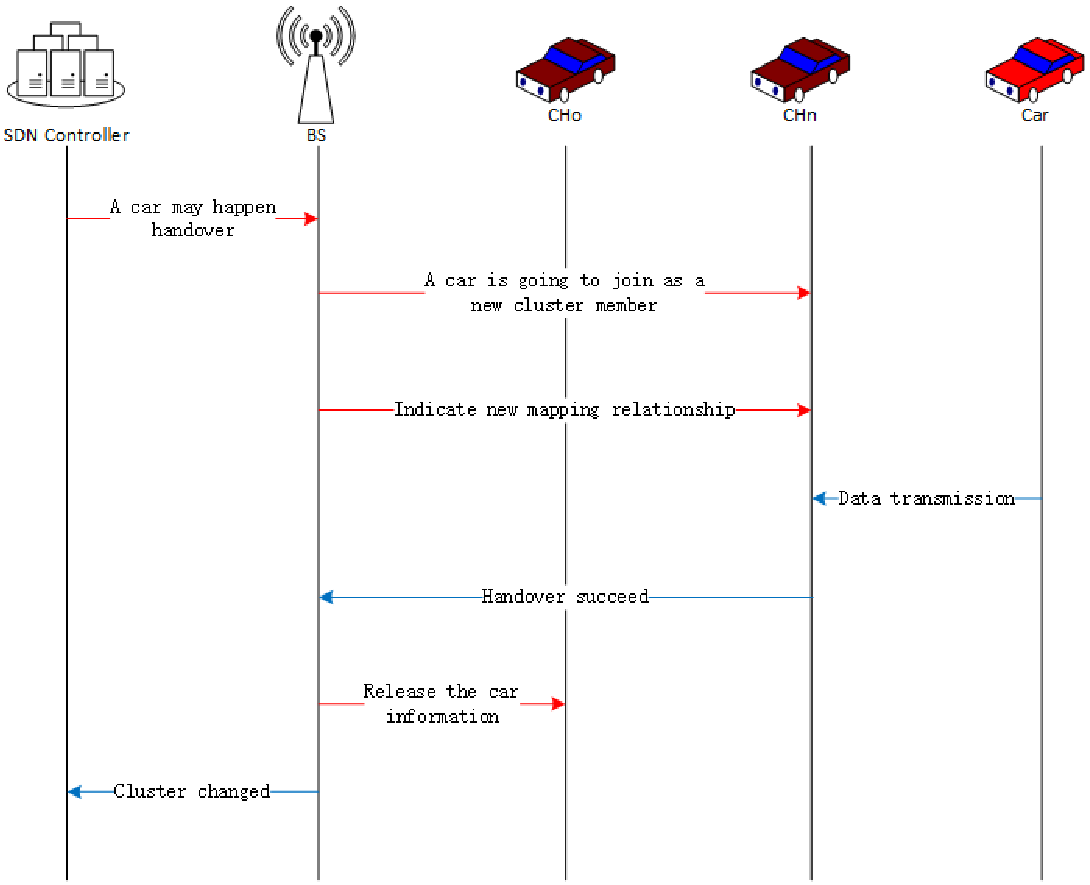

Figure 9 Handover process between different cluster heads.

- Level 2 SDN controller keeps monitoring the movement of vehicles and cluster information within its management range to achieve quick control of the network. When noticing a vehicle may happen handover to a new cluster, the level 2 controller will inform the preparation for the handover of the vehicle between CH_o and CH_n to the base station.

- After noticing the handover attention from the center controller, the controller on the base station reminds the CH_n that a vehicle is possible to join into the cluster, and issues an instruction in advance indicating the vehicle's new mapping relationship about its new locational address and identity address.

- After receiving the instruction from the controller, the CH_n sets the corresponding action with a timeout. This action indicates the new mapping relationship of a vehicle's address and its new location.

- If the vehicle does not join the new cluster after a while, flow table timeout will be executed and the action set on the CH_n will be deleted automatically.

- When the vehicle successfully moves into the management scope of CH_n, the base station is informed by CH_n of handover success and releases the vehicle's information including address mapping relationship from CH_o.

- After the vehicle happens handover successfully to the new cluster, the network services can be immediately got by the vehicle instead of a series of rerouting computations and occurrence of communication reconnection. The data packet forwarding to the vehicle will be directed to the new cluster guided by the vehicle's new locational address. The CH_n acts as a network gateway for the vehicle to Internet communication and is able to execute mapping action between the new locational address and identity address of the vehicle.

- The network topology will be updated by the SDN controller, after the handover scheme and monitored to prepare for the next handover.

In this way, when the vehicle happens handover to a new cluster, network communication can be build up immediately by setting up a new mapping relationship in advance saving the time for handover control information interaction and routing computation. During the handover scheme, the vehicle does not change its identity address but changes its locational address to support the new transmission route.

Except for the vehicle's handover between clusters, another kind of handover that may happen in the clustered VANET is the cluster's handover between different base stations. In this case, a series of the process will be executed as shown in Figure 10.

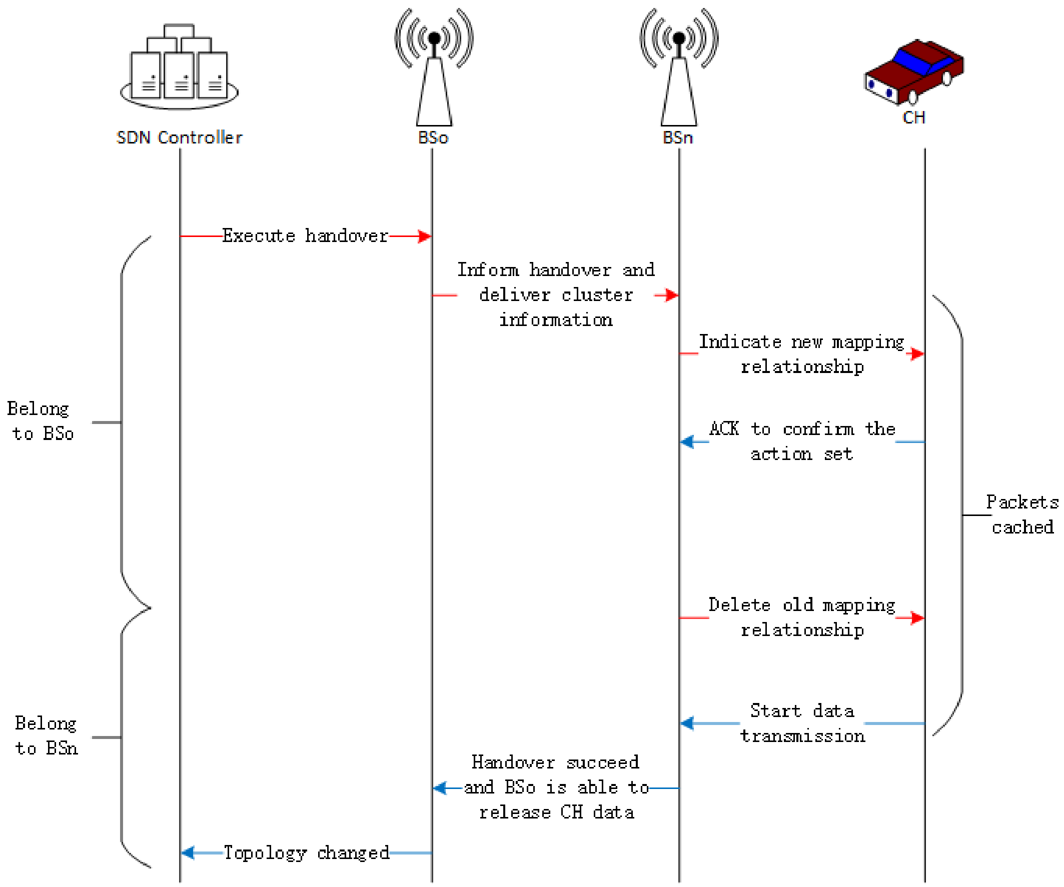

Figure 10 Handover process between different base stations.

- By monitoring the vehicle's movement and information of cluster, the level2 SDN controller notices a cluster is able to happen handover from the current base station (BS_o) to a new base station (BS_n). Then it will request the level1 controller to execute the handover process between BS_o to BS_n.

- The controller backup the cluster information owned by BS_o to the information owned by BS_n, and generates the new cluster mapping relationship according to the cluster's information when connecting with BS_o.

- The MEC server on the BS_n starts to cache the data related to the cluster and conducts related action for a new mapping relationship of the cluster to indicate the new access of cluster through BS_n.

- After the action has been successfully set, BS_n deletes the old mapping relationship from the cluster, so that the cluster can achieve data transmission by access to BS_n.

- After noticing the handover succeed, the level 2 SDN controller will release the remaining cluster information on the BS_o and informs the level 1 SDN controller to update topology changes.

In the way described above, a cluster is able to achieve handover from different base stations and starts to get network communication through BS_n. After moving into the new access, the data in the caching queue of BS_n will be delivered to the cluster as soon as possible and the data packet that forwards to the cluster will be redirected from BS_o to BS_n.

During the vehicle handover process, use many control messages are used by the SDN controller to manage the handover process. When the vehicle happens handover between different clusters, the SDN controller instructs three kinds of control messages to achieve a handover: (1) M_{vmap} to issue the new mapping relationship of the vehicle in CH_n; (2) M_{vstat} to confirm whether the mapping relationship is correctly set in the CH_n; (3) M_{vdel} to delete the expired mapping relationship in the CH_o after the handover succeed. Besides, the CH_o uses the M_{vreply} with set mapping relationship, to reply to the M_{vstat}. Therefore, the SDN control message overhead can be described as:

$$\frac{\text{Len}\left(M_{vmap} + M_{vstat} + M_{vreply} + M_{vdel}\right)}{AverageThroughput} \times f_v \tag{1}$$

where f_v means the handover frequency of the vehicle between different clusters.

When the cluster happens handover between the base stations, the SDN controller issues four kinds of messages to manage the handover of cluster: (1) M_{ccach} to instruct the data stream to back up to the BS_n; (2) M_{cmap} to issue the new route of vehicle in the cluster to the BS_n; (3) M_{cstat} to confirm whether the new route to the BS_n is successfully set in the cluster head; (4) M_{cdel} to delete the old route in the CH after the handover succeed. In addition, the CH uses the M_{creply} to reply to the M_{cstat} indicating whether the action is correctly set. Therefore, the SDN control message overhead can be described as:

$$\frac{\text{Len}\left(M_{ccach} + M_{cmap} + M_{cstat} + M_{creply} + M_{cdel}\right)}{AverageThroughput} \times f_c \tag{2}$$

Where f_c means handover frequency of cluster between different base stations.

3.4.2 Path selection in the handover

When the vehicle is going to move out of the same controller's control area, the controller in Level1 can help to support seamless handover. For the vehicle's handover accompanied by the data transmission route change, a new decision for the routing path has to be made. The routing path can be decided, according to the different handover situations.

3.4.2.1 Same controller handover

If the handover happens between the different access that in the charge of the same level1 SDN controller, it is called the same controller handover. For one controller is responsible to manage several base stations or RSUs, it is more common and frequent for moving vehicle that happens different controller handover. For frequent handover, it is inappropriate to download large amounts of flow tables to build up an optimized route path for every path change, considering the limited computing resource and large scale signaling overhead. Therefore, the new path is decided more simply that only one flow table is set on the router which is on the edge of the SDN controller management range and used to be part of the previous path.

Figure 11 shows the scenario of the same controller handover where the BS1, BS2, and Gateway Router are under the management of the same SDN controller. For the SDN controller that holds the full information about the network into its control, it is easy to find out the routing path between nodes within the management range of the same controller. Before handover happens, data transmission between vehicle and service can be achieved through the path Server-GatewayRouter-BS1-vehicle. To support the transmission route after the handover, the SDN controller selects a router that was acting as a gateway of network managed by the controller in the previous path to issue a flow table that indicates all data packet forwarding to the vehicle has to redirect to the BS2. After achieving handover to the BS2, the transmission path from the server to the vehicle has become Server-GatewayRouter-BS2-vehicle. In this way, the signaling overhead caused by the setting of the flow table and computing resource cost for scheduling transmission path can be minimized.

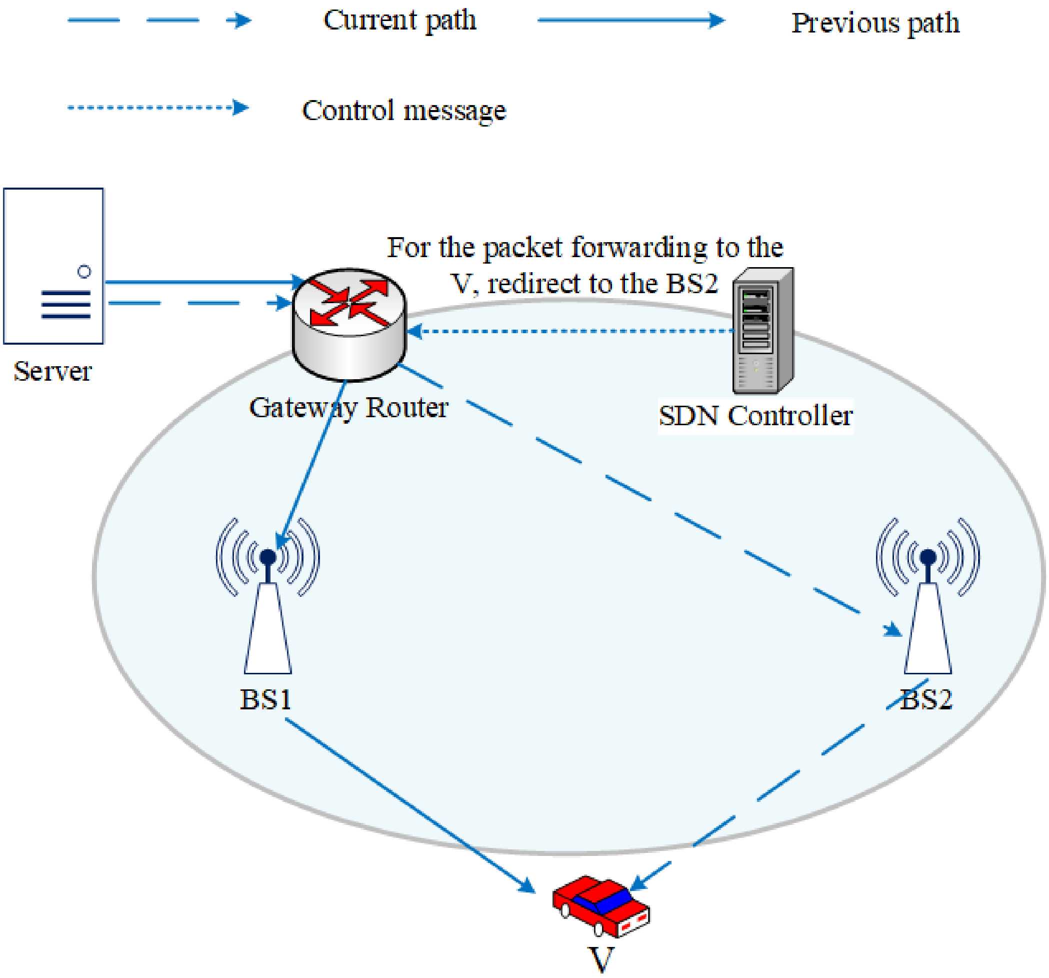

Figure 11 Same controller handover.

3.4.2.2 Different controller handover

If the vehicle moves for a long distance and happens handover from one controller's management to another, this is called different cluster handover. This kind of handover not frequently happens but unavoidable. In this case, reusing the path as much as possible is the basic principle, when deciding the new routing path. Because the overlapping path between the new transmission path and the previous one can be used by configuring a fork router located at the junction of two paths, so that signaling overhead for downloading flow rules from the controller can be reduced, and naturally rerouting process duration can also be shortened.

As shown in Figure 12, the vehicle used to get services from the Server by the Previous path: Server-BS3-Vehicle before the handover. When the vehicle happens handover from BS3 to BS4 belonging to the different controller management range, the level1 centralized controller has to involve in the routing path decision. The level1 controller holding with network global information can help to find out the appropriate routing path. A Fork Router is selected as a bifurcation point of the current path and previous path to support the rerouting process. When handover happens, the SDN controller issue a flow table to Fork Router that indicates all data packet forwarding to the vehicle have to redirect to the BS4, so that the current path from Server to Vehicle will become Server-ForkRouter-BS4-Vehicle after the handover.

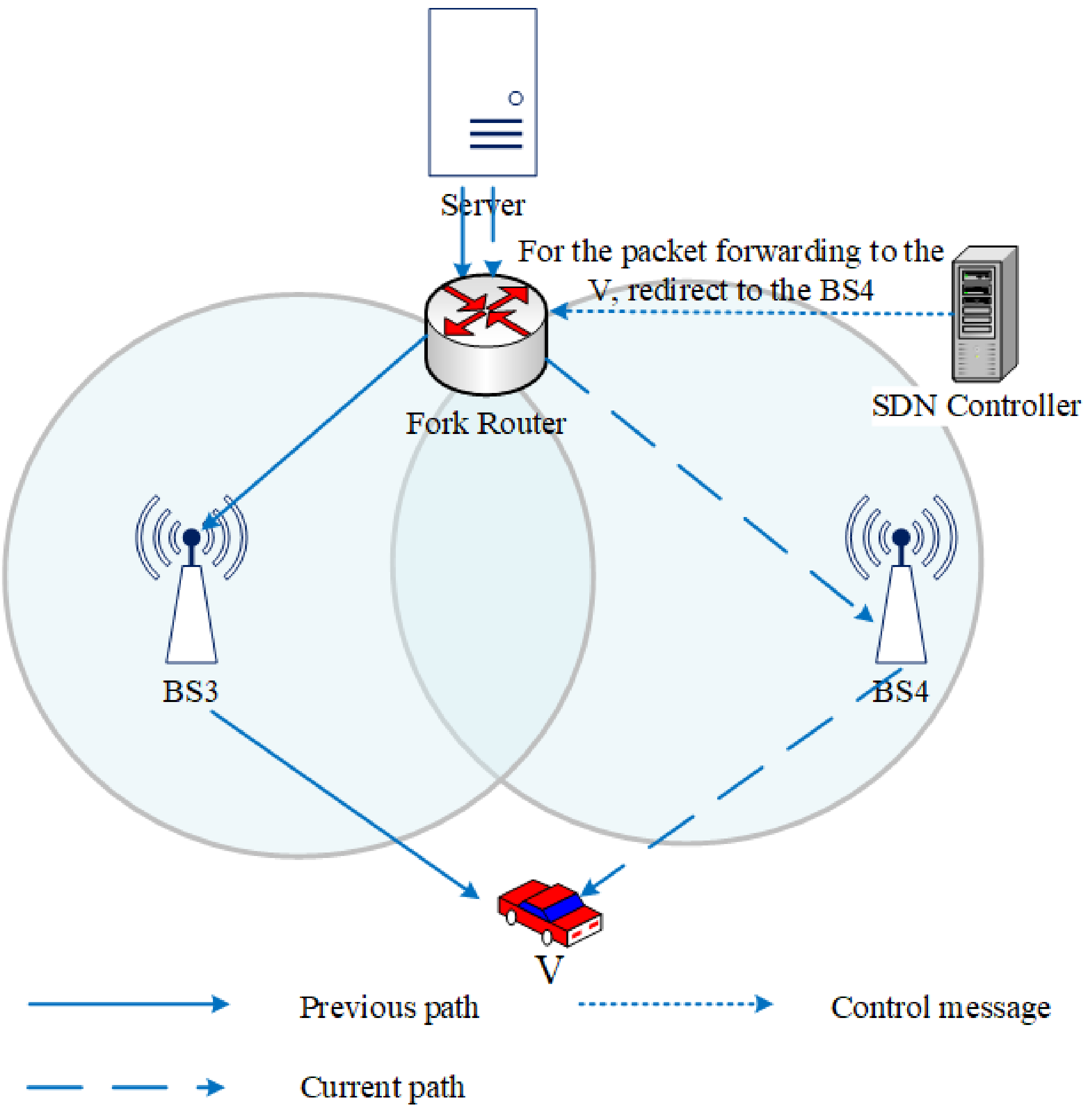

Figure 12 Different controller handover.

However, involving the existing path into a new path may not build up the most optimal route. Therefore, it is necessary to select an appropriate Fork Router, so that the new routing path can be closer to the best path. Theoretically, the Fork Router can be any routers on the previous path. However, inappropriate choice of Fork Router may lead to some network problems. For example, when choosing a Fork Router that is too near to the server, the new route may be established with the download of too many table flow. Furthermore, if BS3 is chosen as the Fork Router, the new path may form a triangular route as well as large numbers of forwarding hop.

As consideration mentioned above, Fork Router should satisfy the following two points as much as possible to support the handover scheme:

1) Short path: The Fork Router is chosen to form a path with fewer forwarding hop, to minimize the data transmission delay and avoid triangular routing.

2) Little flow-table change: The new path should be built to achieve less flow table download what means the forwarding hops from Fork Router to the vehicle above the new path should be as few as possible to reduce the control message signaling overhead.

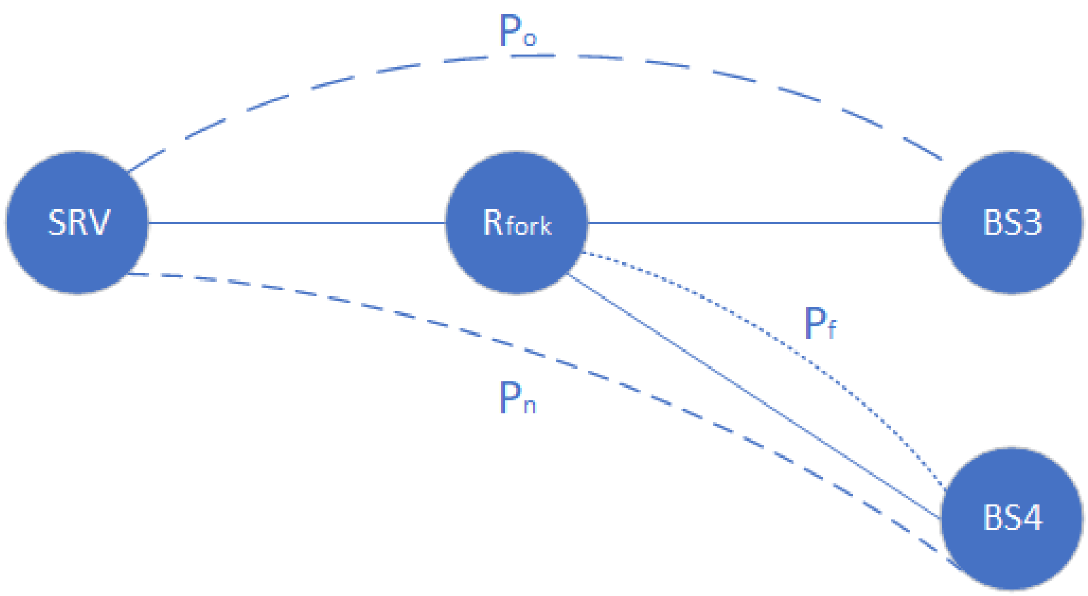

Figure 13 Route change with a fork node.

The transmission path change caused by the handover is depicted in Figure 13. The vehicle used to communicate with SRV by connecting with BS3 is moving towards BS4 and result in a handover. In order to reuse the previous path, a Fork Router, R_{fork} in Figure 13, is chosen from the routers that make up the old path. Data transmission on the old path is redirected at R_{fork} to the new destination BS4 by using the optimal path. In order to clarify the selection of R_{fork}, the paths are defined as the following:

- P_o - Indicate the old path before the handover from SRV to B3.
- P_n - Represent candidate for the new path after the handover from SRV to BS4 through R_{fork} that also belongs to P_o..
- P_f - An optimal path from R_{fork} to B4.

In order to minimize the route change and make an optimal transmission path, the R_{fork} is selected obeying the following conditions:

(1) The R_{fork} should be selected from the path P_o, so that the routing policies from SRV to R_{fork} do not have to change.

(2) The control signaling overhead caused by the flow table change only appears when establishing the path from R_{fork} to B4. Therefore, the shortest length from R_{fork} to B4 is the basis of selecting R_{fork}.

(3) Besides, R_{fork} is selected to achieve a short new path, so as to reduce transmission delay on P_n and avoid triangular routing. Therefore, it is necessary to choose R_{fork} that can make P_n shorter.

A parameter x_i is defined to be the length of the shortest path from R_i to BS4 and y_i is the total length of P_n where $\forall R_i \in P_o$ is a candidate for the fork router. It is assumed that the reduction in network performance can be measured and the influence of signal overhead and transmission latency are $\varphi(x_i)$ and $\omega(y_i)$, respectively. The reduction in network performance can be calculated:

$$\varepsilon(x_i, y_i) = \varphi(x_i) + \omega(y_i) \tag{3}$$

The transmission delay and signaling overhead are assumed to be proportional to the length of P_n and length of the path from R_i to BS4.

$$\varphi(x_i) = x_i \times \varphi(1) \tag{4}$$

$$\omega(y_i) = y_i \times \omega(1) \tag{5}$$

$$\Rightarrow \varepsilon(x_i, y_i) = x_i \times \varphi(1) + y_i \times \omega(1) \tag{6}$$

Since the aim is to find a R_{fork} to form a new path with high network performance, $\varepsilon(x_i, y_i)$ has to be limited in the minimum size, achieving:

$$\varepsilon(x_{fork}, y_{fork}) = \min[\varepsilon(x_i, y_i)] \tag{7}$$

In different network scenarios, $\varphi(1)$ and $\omega(1)$ can be measured by different kinds of methods. For the network scenario in this thesis, if it is assumed that $\varphi(1) = \omega(1)$, the reduction of network performance can be:

$$\varepsilon(x_i, y_i) \propto x_i + y_i \tag{8}$$

$$\Rightarrow \varepsilon(x_{fork}, y_{fork}) \propto \min(x_i + y_i) \tag{9}$$

Therefore, an appropriate R_{fork} can be selected by finding a R_i on the P_o with the minimum value of $x_i + y_i$. The detailed process in selecting R_{fork} is described in Algorithm 2.

Algorithm 2 Algorithm to find fork router and the new path
SL[x, y] is the shortest path length between x and y
PL[x, y] is a length of the path from x to y that forms a part of P_o
Initialization: $z = SL[SRV, BS4]$, $R_{fork} = SRV$
for each router $R_i \in P_o$ **do**
$x_i = SL[R_i, BS4]$
$y_i = PL[SRV, R_i] + x_i$
$z_i = x_i + y_i$
if $z_i \leq z$ **then**
$z = z_i$
$R_{fork} = R_i$
end if
end for
$P_n = P_o(SRV \text{ to } R_{fork}) + shortestPath(R_{fork} \text{ to } BS4)$

3.5 Simulation and result

In this section, the proposed handover scheme can be evaluated by comparing the traditional network handover with the unstable transport layer connection and the handover with a stable transport layer connection in different network scenarios. The results show the benefits of the proposed handover scheme in terms of network throughput and end-to-end delay.

3.5.1 Simulation set up

In order to evaluate the proposed handover scheme compared with the conventional method, the OMNET+5.0 simulator with INET 3.4.0 open-source model is used to perform network simulation. The simulation set up parameters are shown in Table 2. A mostly used routing protocol in MANET called AODV is used as a routing protocol in the network simulation. However, in the general setting, the AODV protocol takes too long to discover a new route, even if there is only one-hop routing, resulting in degradation of the network performance. For this reason, "Hello Interval" and "Allowed Hello Loss" in AODV protocol are set into a minimum size to support more accurate evaluation. The simulations are conducted on two different road types which are designed by the SUMO traffic simulator. The one is designed as a grid topology spanning an area of 1000 m × 600 m where the length of each road segment is 200 m. The other is a 4 lanes straight road with a total length of 2000m. In the simulation, the number of vehicles is set in the value varied from 180 to 540. Vehicles are set to get network services through the corresponding cluster heads, which are manually specified since the clustering algorithm has not been introduced. In order to evaluate the proposal in supporting TCP connection, a vehicle is selected to run continuous TCP service communicating with the TCP server in the core network during the simulation. Each simulation result is the average value provided by the vehicle after running on 10 different moving paths.

Table 2 Simulation parameters

Parameters	Values
Routing Protocol	AODV
Transport Layer	TCP(RENO)/UDP
Interface	IEEE 802.11p
Number of Vehicles	180, 360, 540
Average Velocity	40km, 60km, 80km, 100km
Data Rate	3Mbps, 6Mbps, 9Mbps, 12Mbps
Beacon Interval	1s, 0.5s, 0.1s
Simulation Topology	Grid and Straight road
Topology Size	1000m×600m, 2000m with 4 lanes

At the transport layer, most of the simulation scenarios are based on the TCP protocol with the RENO congestion control algorithm. TCP uses the congestion control algorithm to manage transmission flow where Congestion WiNDow (CWND) is one of the most important variables responsible for the control of the TCP transmission rate. When the TCP connection is established, CWND is initialized into minimum value. As the TCP data are successfully received, the value of CWND grows, according to the congestion control algorithm. When the TCP connection is reestablished, the CWND will be initialized and grow slowly through the congestion control algorithm.

The vehicle is set to happen handovers between the clusters and base stations. The proposal is evaluated by comparison with the "Conventional method" and "No cache" handover scenarios.

1) "Conventional method" represents the conventional wireless network where the vehicles happen handover between different base stations accompany address change and no cache scheme is conducted at the base station in supporting packet

loss during the handover.

2) “No cache” represents the SDN-based vehicular network with a global view where the handovers of vehicles are supported by the SDN controller so that the addresses are kept the same during the handover, but no caching scheme is conducted at the base station.

3) “Proposal” represents the proposed SDN-based VANET handover with MEC caching scheme deployment which not only keeps the connection unchanged but also minimizes the transmission delay and packet loss during the handover.

3.5.2 Simulation result

3.5.2.1 Effect of data rates

In the first scenario, the TCP application is deployed on the vehicle and request the continuous TCP services from a TCP server in the core network. The vehicle movement is written to drive on the grid map at the speed of 60Km/h, resulting in frequent handovers between different clusters and different base stations. In order to evaluate the effect of different data rates on network performance, the network simulation is set to different link rates, and the average throughputs of both the conventional method and the proposed method are recorded.

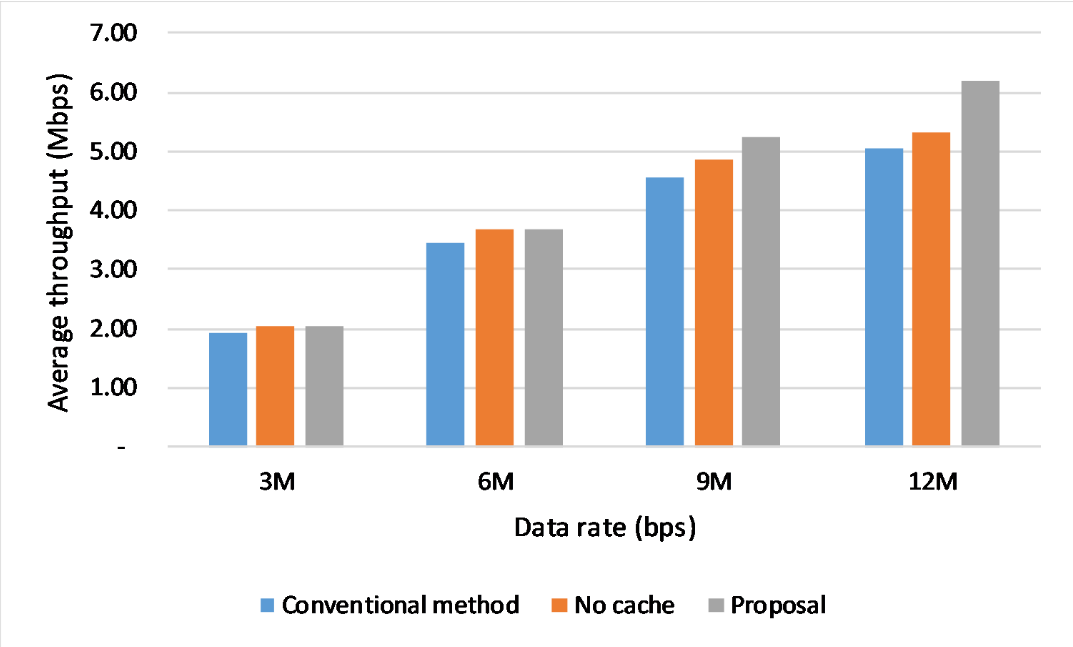

Figure 14 Throughput in different data rates.

Figure 14 shows a comparison of the average throughput for different link rates. The vertical axis presents the transmission average throughput. It is easy to see that seamless handover based on SDN shows higher average throughput compared to the conventional method, especially, the proposed method with the addition of the MEC caching scheme performs better network performance, according to Figure 14. As the link rate increases, the advantage of the proposal becomes increasingly obvious in terms of average throughput. It is because, in the conventional handover, a larger transmission rate means more data loss, while the proposed handover method can always maintain a relatively stable connection. Therefore, the proposed method can achieve higher performance for high-speed networks.

In order to further confirm the advantages brought by the proposed method, the detail information of result in 9Mbps link rate is selected to extract the real-time throughput and CWND changes for further discussion.

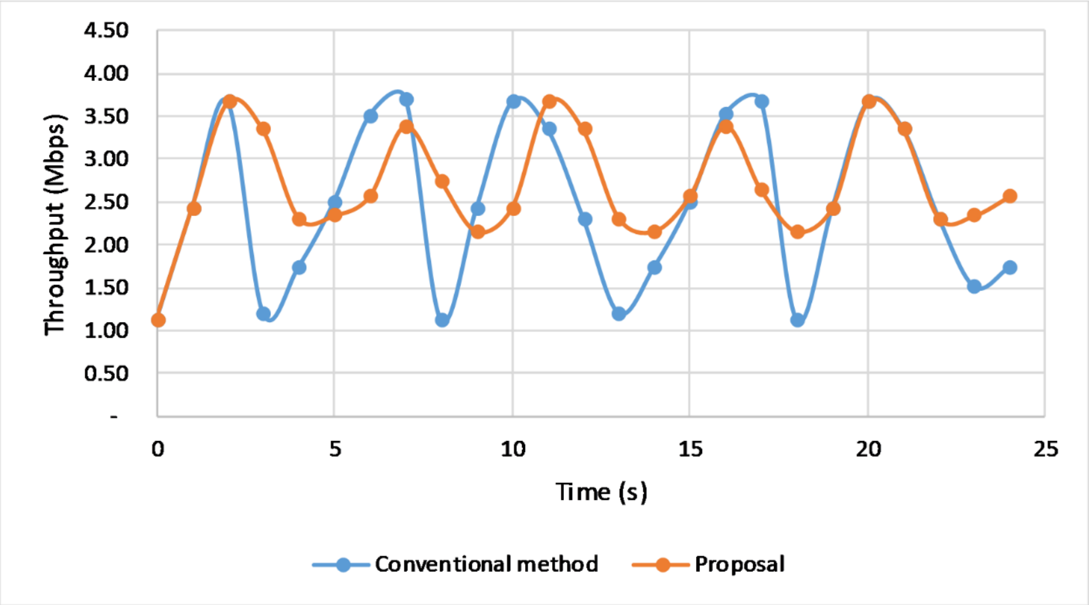

Figure 15 Real-time throughput.

Figure 15 shows the real-time throughput change of the vehicle with both the conventional method and proposal in which handovers have happened several times. The figure clearly shows the advantages of the proposal when handover occurs. During the handover, the traditional handover method may cause a sudden drop in throughput, but the proposed method can still maintain a relatively stable throughput and thus keeping a higher average throughput.

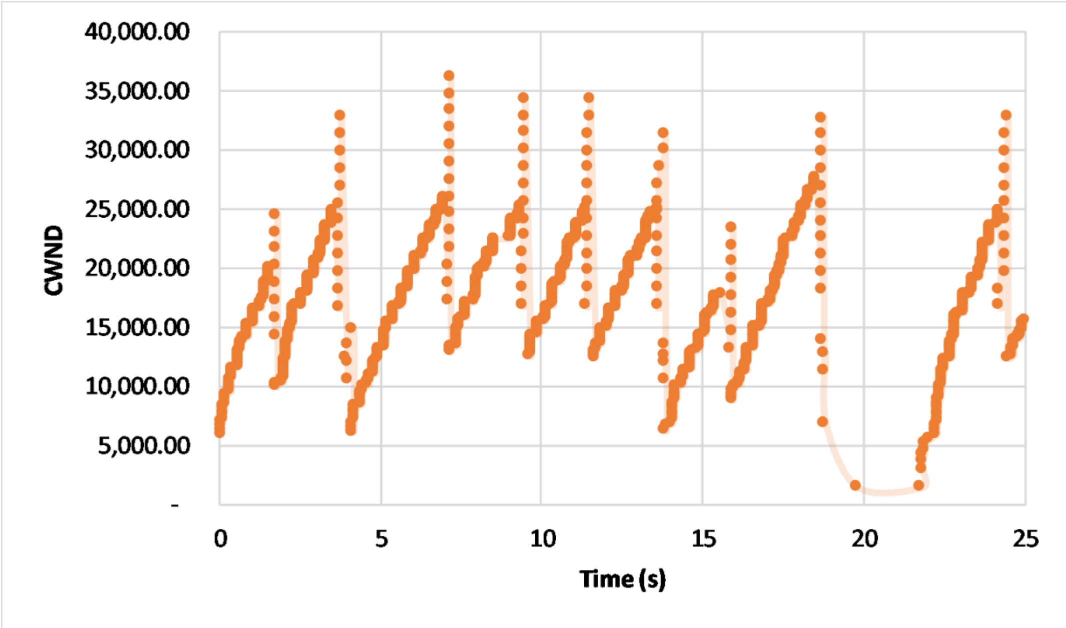

Figure 16 Real-time Congestion WiNDow (CWND) in the proposed scheme.

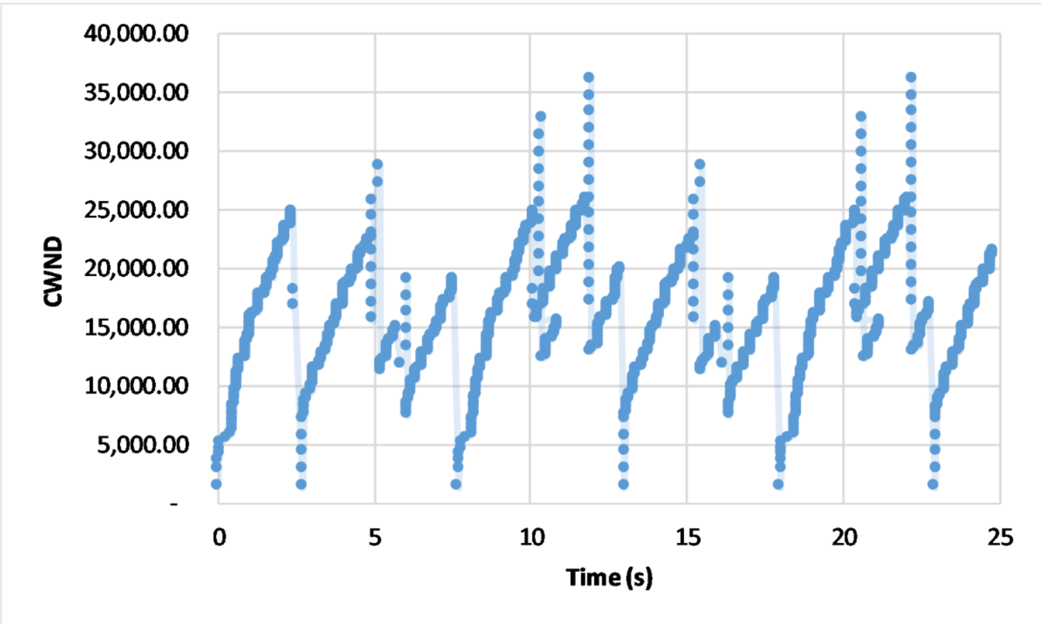

Figure 17 Real-time CWND in the conventional approach.

Figures 16 and 17 show the real-time changes of CWND. Figure 16 shows the real-time change of CWND when using the proposed scheme during the handover. It can be easily seen that the CWND value maintains a small range of jitter even if the handover occurs. On the contrary, as shown in Figure. 17, when the conventional switching method is used, the

CWND may suddenly plunge to the minimum initial value when the vehicle happens handover, and then increases slowly from the initial value. Since CWND is responsible for the main task of transport-layer congestion control and influences the transmission rate, initialization of CWND will accompany with throughput reduction, resulting in lower average throughput.

For the simulation in this chapter, the buffer hit rate is set to 100% without considering the influence of external factors to evaluate the benefits of the caching scheme introduction for handover performance. But in reality, the cache hit rate is affected by various factors. Such as cache capacity, caching data size, caching data replacement algorithm, and so on. The most common situation is that there are too many cluster heads under a base station control that the caching server can no longer be used by the upcoming handover. In such cases, the caching server cannot provide the data required by the vehicle, the retransmission from the sender will be executed, thereby reducing the transmission performance.

3.5.2.2 Effect of vehicle velocities

The second scenario compares the performance of the proposed method and the traditional switching method at different vehicle speeds. The vehicle runs the TCP application and sets the link rate to 6Mbps. The results are shown in Figure 18. The proposed scheme always performs superior to the traditional scheme in terms of average throughput, and the throughput gradually decreases as the vehicle speed becomes faster. In particular, the traditional handover scheme seems to be more susceptible to vehicle velocity changes. Since each simulation runs on the same map and distances between the roadside infrastructures keep the same, the velocity change has the greatest impact on the handover interval. That means faster speed results in infrequent handover. Therefore, the traditional scheme is more susceptible to vehicle speed when compared with the proposed scheme.

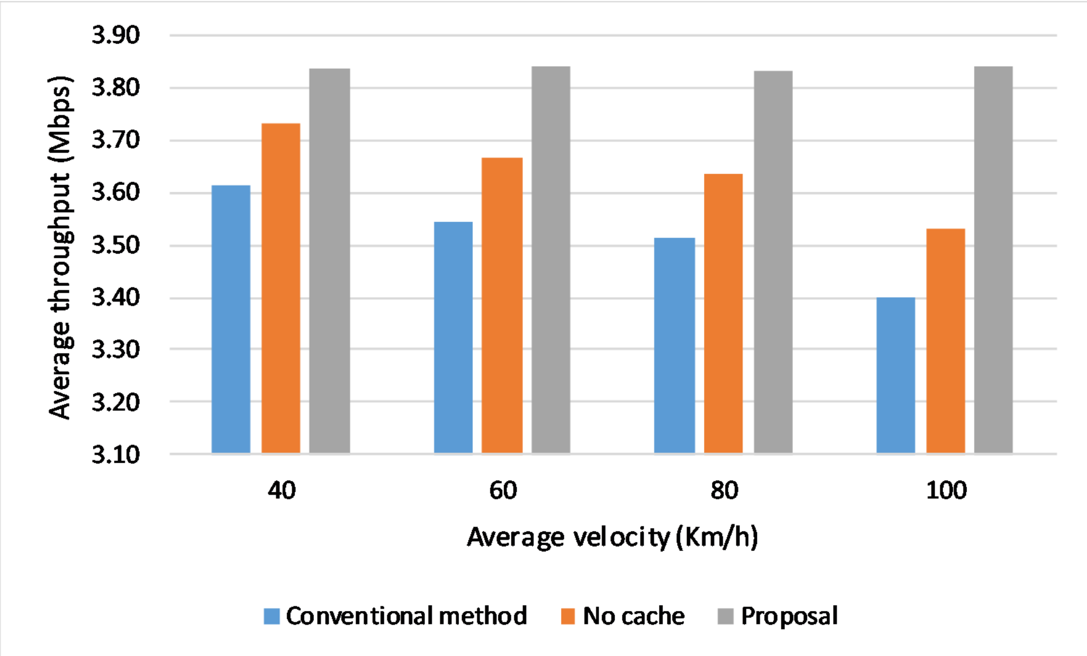

Figure 18 Throughput in different velocities.

To evaluate the effect of the proposed caching scheme on the transmission delay, the results of the transmission data delay of both the "No cache" method and the proposed scheme are extracted from the simulation described above. It can be concluded from Figure19 that the use of the proposed caching scheme can reduce the transmission delay when handover occurs. However, the transmission delay is not greatly affected by the velocity changes. This is because the result shown in the figure comes from the average delay of each data transmission. Therefore, the average end-to-end delay is not that affected by the small range of changes that happens during the handover.

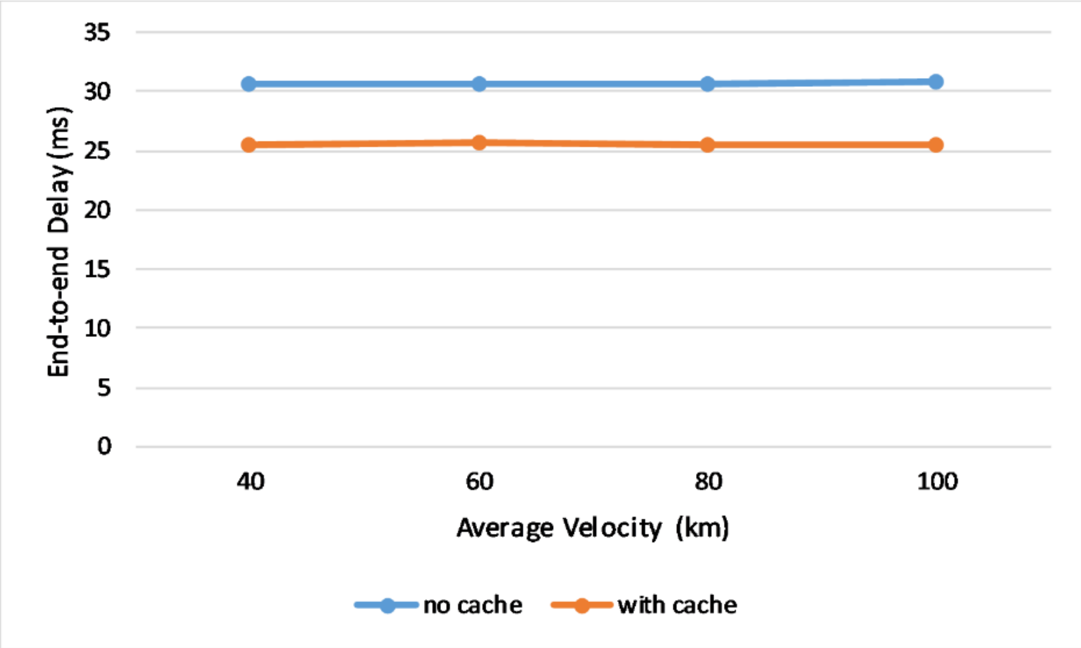

Figure 19 End-to-end delay for different velocities.

3.5.2.3 Effect of vehicle densities

In the third network scenario, the proposed scheme is evaluated at different vehicle densities. Figure 20 compares the average throughput of the conventional method and the proposed scheme at different vehicle densities. The results show that the increasing number of vehicles brings a bad effect on network performance. With the growing number of vehicles, the burden of roadside infrastructure will increase. At the same time, under the limited wireless environment, large amounts of wireless data will be more possible to cause data collision, thereby affecting the network performance. Nevertheless, the proposal shows advantages over conventional handover, especially when in the high density of the vehicle environment, the introduction of MEC caching has played its advantages.

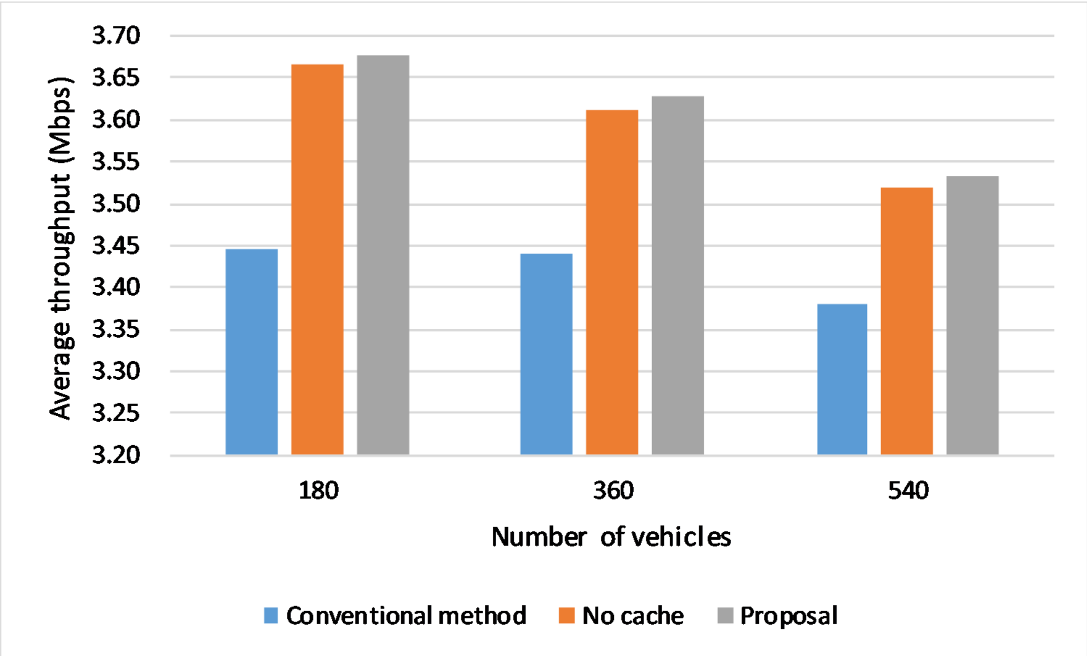

Figure 20 Throughput for different numbers of vehicles.

3.5.2.4 Effect of beacon intervals

The fourth scenario evaluates the network performance of the proposed method at different beacon intervals. In this simulation, the link speed is set to 6Mbps, and the average vehicle speed is set to 60km / s. For each simulation runs, different Beacon interval values are set to evaluate the influence of the background traffic on the vehicle's communication quality. Figure 21 depicts the result of the simulation, which shows that the proposed scheme maintains advantages at different Beacon intervals compared with the conventional one.

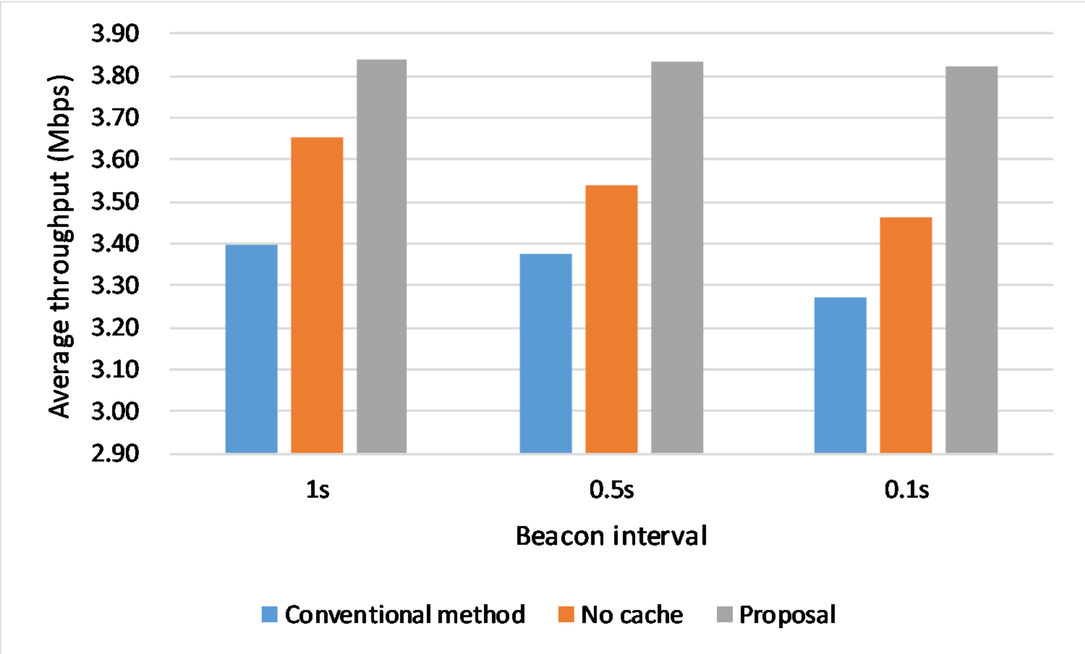

Figure 21 Throughput for different beacon intervals.

3.5.2.5 Effect of background noise levels

The performance of the proposed scheme under different noise backgrounds levels is also evaluated. The data transmission rate of the vehicle is kept at 6Mbps, and background noise values for each simulation are set differently at -110, -105, -100dBm. The results of the simulation are shown in Figure 22, showing that the proposed method performs the best in terms of average throughput for any background noise level. It is worth noting that the traditional handover scheme has not been greatly affected by the increase in background noise. High background noise is more likely to cause network congestion, thereby reducing CWND. When the vehicle happens handover using the conventional method, the changing trend of CWND is similar to the trend when there is severe traffic congestion. For the above reason, the conventional method is not affected as much as the proposed scheme shows.

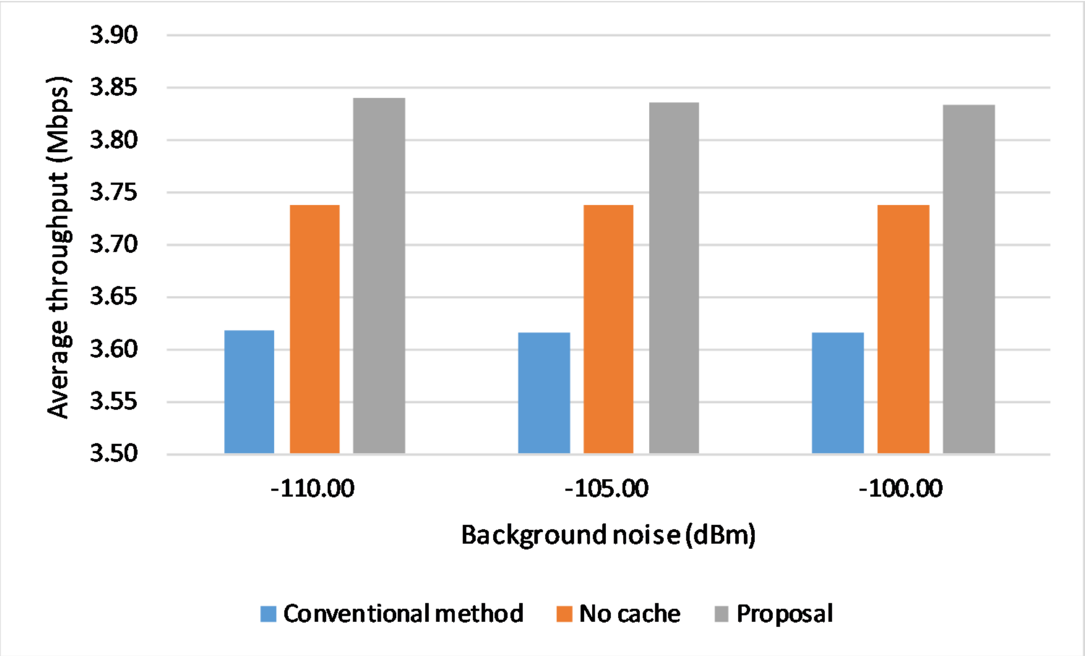

Figure 22 Throughput for different background noise levels.

3.5.2.6 Effect of distance between base stations

In this scenario, the proposed scheme is evaluated with different distances between neighboring base stations. Figure 23 shows the results of the simulation where the proposed scheme shows its advantages in all cases. The resulting graph can be discussed in two aspects. On the one hand, when the distance between the base stations is not too far, this distance affects the network performance by affecting the handover frequency. The longer the base station distance, the lower the handover frequency, so the better the network performance. On the other hand, when the distance between the base stations is too long, the network performance is affected not only by the handover frequency but also by the network quality and communication range, resulting in poor network performance.

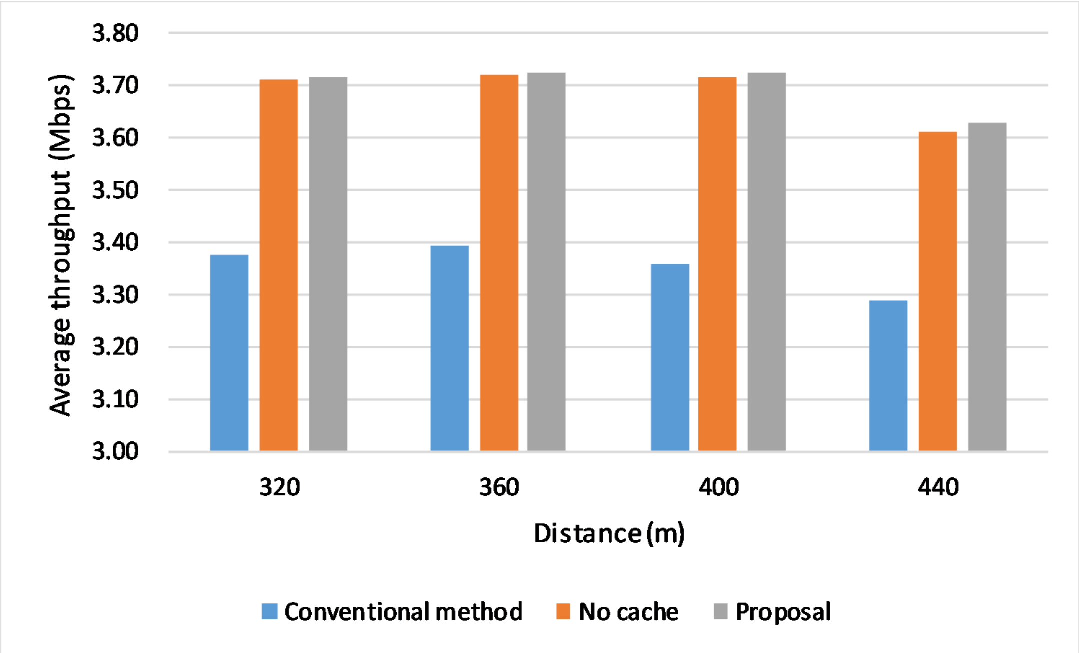

Figure 23 Throughput for different distances between base stations.

3.5.2.7 Effect on UDP application

The proposed handover scheme is also evaluated in the non-connection oriented transmission where the UDP application is used to communicate between vehicle and correspondent nodes in the core network. Figure 24 shows the result which compares the proposal with the conventional method through average throughput for different link rate. The proposed caching scheme is designed for the connection-oriented communication which provides acknowledgment to the received packet, so it does not participate in the comparison. It can be seen from the results that the proposal is superior to the conventional method in terms of average throughput. Moreover, the proposed scheme shows its greater advantages as the link rate increases. It is because UDP transmission does not consider the reliability of the link, the higher the transmission rate the more data will be lost during the handover latency.

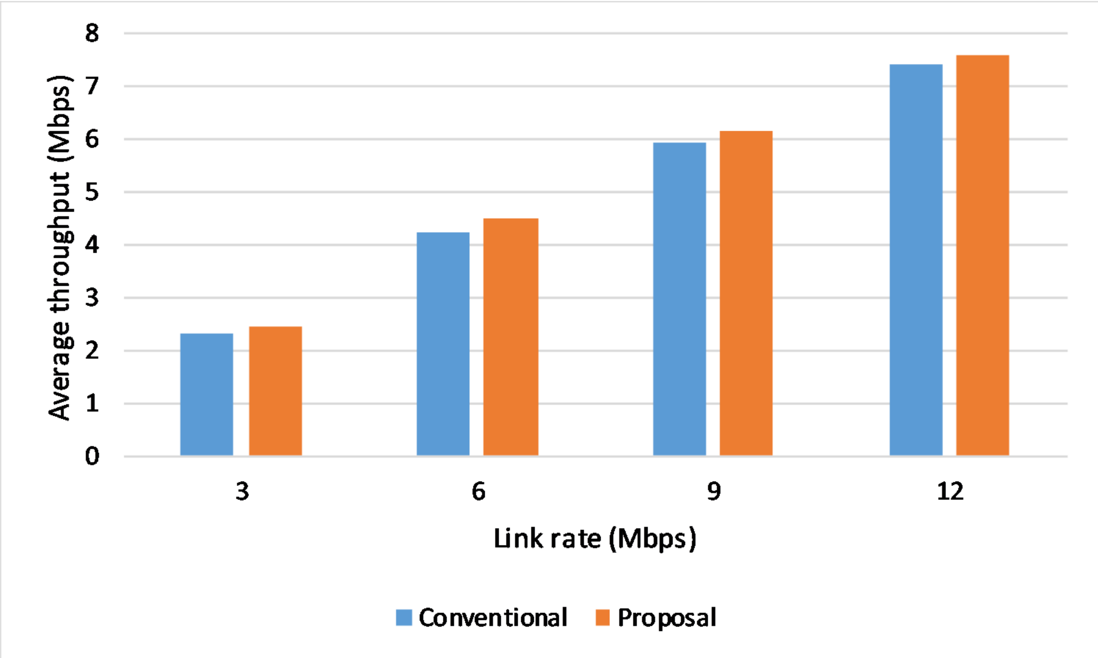

Figure 24 UDP application throughput for different link rates.

3.6 Conclusion

In this chapter, an SDN-based handover scheme was proposed to the VANET. In order to support seamless handover in a highly dynamic VANET scenario. Section 3.1 analyzed the existing research about keeping a stable connection during the handover and discussed the necessity of employing SDN technology into the VANET. Therefore, the SDN was introduced in the VANET architecture with a two-level controller structure supporting mobility issues in VANET and forming a clustered network.

To reduce the data loss caused by the route change in during the handover, MEC technology was introduced in section 3.3 to achieve caching at the roadside, so that the vehicle could resume the network communication as soon as possible after the handover. It was not only to cache the data preparation for the use but also to manage the caching data required by the network communications. So a caching queue management algorithm was proposed to support only cache useful data.

In section 3.4, the handover scheme was explained in detail according to the vehicle's role in the clustered network. SDN was used to decide and execute the mapping function realizing that the handover scheme performed transparently to the vehicle and correspondent node. In order to support the mobility issue, the data transmission rerouting problem after the handover is also discussed.

At last, the evaluation of the proposed handover scheme was conducted, using simulation tools in different network environments by changing the vehicle velocity, link rate, and wireless channel conditions. The results showed that the introduction of SDN into the handover performs obvious advantages compared with the traditional handover method especially in the network with frequent topology changes and high-quality transmission links. In addition, the employment of the MEC caching function showed its further improvement of the network performance both from transmission delay and average throughput.

Chapter 4

4 Context-aware clustering

In this chapter, a context aware clustering algorithm is proposed to support application requirements in different kinds. The global view of the SDN controller helps the clustering process more flexible and easier and schedules the clustering process only when it is needed instead of depending on the vehicle's periodical detection towards the neighbors.

In section 4.1, the shortcomings of considering a single network scenario are discussed, and the necessity of applying the clustering algorithm according to the application requirements is carried out.

In section 4.2, the motivation is discussed that using a single clustering algorithm is not appropriate to support various applications. The vehicle's applications are classified into three types: delay-sensitive application, traffic-intensive application, and computation-intensive application according to their different requirement to the network performance.

In section 4.3, the clustering algorithm is introduced with three-stage. In order to achieve a fast build-up for the cluster, vehicles are separated into clusters according to their geographic location at the initialization stage. The cluster head is selected for each cluster using the corresponding algorithm decided by the application type at the cluster head selection stage. This algorithm involves the stability of vehicles, signal quality, and computing ability in a cluster head selecting basis. In the cluster maintenance and adaption stage, the management of cluster after the cluster establishment is described.

In section 4.4, the simulation works are designed to evaluate the proposed clustering algorithm. Finally, section 4.5 summarizes this chapter.

4.1 Problem discussion

In most cases, current existing clustering algorithms are proposed according to the only one network scenario, but they are not an appropriate method when changed to another network scenario. Different vehicular applications exhibit different levels of QoS. For example, some vehicular applications for video download or sensor data collection need data transmission of large flow [104]. In contrast, some applications which provide V2V emergency message [105] need lower transmission delay for safety reason. Some applications, such as vehicle camera data analysis generate computational data [106] to conduct some intensive computing at the vehicles. For these reasons, it is necessary to put forward a method to support various network situations. A clustering method that generates clusters by different clustering algorithms according to context information has to be proposed. Different from the conventional transmission method where the data packets bounding for the same target would traverse the same path, in the proposed clustering algorithm, the data packets are forwarded by the cluster heads corresponding to the application types.

4.2 Cluster classification

There exist many kinds of research that are possible to classify different applications by extract the applications' transmission statistics [107] [108]. The introduction of SDN brings a global view to the controllers which allows the statistical feature of application traffic to be easily extracted from the network devices by scanning the flow tables, so as to realize application classification and feature selection [109] [110].

Context represents the surrounding environment of the object which is under discussion, sometimes it is also used to represent the circumstances in which a task is carried out. In order to classify the different kinds of clusters according to the application requirement, the SDN controller grasps the application context information, including source/destination address, source/destination port, forwarding bytes count, and low priority. According to the running application, the vehicle can be divided into three kinds of clusters: delay-sensitive, traffic-intensive, computation-intensive which is corresponding to three kinds of applications.

1) Delay-sensitive application- always carry the time-sensitive message that needs lower network delay. Some applications such as emergency warning and fleet management need to respond to emergencies and exchange real-time information are one of this kind.

2) Traffic-intensive application- needs large traffic flow from the network. In order to minimize packet loss, high signal quality is needed. Besides, long duration time in the cluster may decrease handover frequency, so that the packet loss and performance degradation can be reduced. Even though the 5G cellular network can

provide low-latency communications for vehicles, it is still impractical to use simple cellular communications in this case, due to the scarcity of spectrum and address resources.

3) Computation-intensive application- needs the computation ability of vehicles to achieve data processing. For example, in some autonomous driving situation, the surrounding information is collected not only from the driving vehicle but also from the other vehicle support to get wider information by calculation.

For the research attention is not on the classification, applications are simply classified by the context value. The IP address and port number pairs are used to identify an application. The applications with higher priority are considered to be the delay-sensitive applications, otherwise, with large amounts of forwarding bytes are defined as the traffic-intensive applications, and application traffic with few forwarding bytes at the roadside but having a unique port number is regarded as computation-intensive applications.

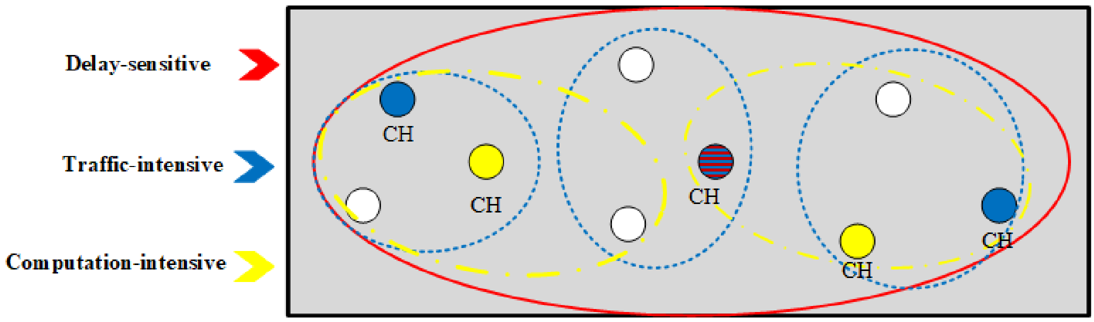

Figure 25 Different cluster classification.

As Figure 25 depicts, the proposed clustering algorithm divides vehicles into different types of clusters. The vehicle in the cluster is not only allowed to communicate with network nodes through a single cluster head but also able to get various vehicular services by correspondent cluster head. Multiple clusters are able to overlap each other and a vehicle that has been selected as a cluster head can be reused to support the transmission of other kinds.

For proposed different kinds of clustering algorithms for different applications, they can work independently and simultaneously. The SDN controller provides three kinds of clustering algorithms for vehicles at the same time to separately control the flow direction of the three kinds of applications. Vehicles run different kinds of applications which makes it plays a different role in different algorithms. For one vehicle, different applications can connect to the network through different cluster heads. For example, when the vehicle runs the delay-sensitive and traffic-intensive application at the same time, the delay-sensitive application can get the delay-sensitive network services through the cluster head that is selected by the clustering algorithm for a delay-sensitive application, otherwise, the traffic-intensive application transmits the traffic flow obeying the rules that instructed by the clustering algorithm for traffic-intensive application.

4.3 Context-aware clustering procedure

In this paper, the clustering algorithm is executed based on a set of parameters $\{D, S, Q\}$. These parameters are derived from the context as the table shows.

Table 3 Contexts using in the clustering algorithm

Primary context	**Processed parameters**
Vehicle velocity Vehicle location	Parameter D for estimating the duration time in the cluster.
Vehicle received signal quality	Parameter S for measuring the received signal quality of the vehicle.
Computing capability	Parameter Q for measuring the computational capability of the vehicle.

4.3.1 Context process

Parameter D in the cluster is calculated by the vehicle existing time in the cluster that used to measure the stability of the cluster. V_i is used to represents the vehicle i in the cluster and satisfies$\in (1, \mathrm{N})$, where the N is the number of vehicles in the current cluster. The variable k is used to represent the number of the vehicle in a cluster and satisfy $\mathrm{k} \in (1, \mathrm{N})$. Let D_k to indicate the stability of the cluster which use vehicle k to work as a cluster head and calculates:

$$\mathrm{D}_k = \sqrt{\frac{\sum_{i=1}^{N}\left(d_k^i\right)^2}{N}} \tag{10}$$

where d_k^i represents connection stability between V_k and vehicle V_i, where V_k is regarded as the cluster head and V_i as a cluster member. The value of d_k^i related to the location and speed of V_i and V_k. The location of V_i is set to $L_i = (x_i, y_i)$, $\mathrm{i} \in (1, \mathrm{N})$ and the velocity of V_i is set to v_i , $\mathrm{i} \in (1, \mathrm{N})$. So the distance between V_i and V_k is calculated as:

$$C_{ik} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \tag{11}$$

The mobility relationship between V_i and V_k is divided into four cases, according to there positions and relative velocities. In case one, V_i is moving behind the V_k and gradually approaching V_k, what means $x_k \leq x_i$ and $v_k \leq v_i$. The duration time of V_i to the V_k is predicted by:

$$d_k^i = \frac{\sqrt{(C_{ik})^2 - (y_i - y_k)^2} + \sqrt{d^2 - (y_i - y_k)^2}}{v_i - v_k} \tag{12}$$

where d indicates the communication scale of the V_k.

In case two, V_i is moving behind the V_k and gradually getting far away from V_k, what means $x_k \leq x_i$ and $v_i \leq v_k$. The duration time of V_i to the V_k is predicted by:

$$d_k^i = \frac{\sqrt{d^2 - (y_i - y_k)^2} - \sqrt{(C_{ik})^2 - (y_i - y_k)^2}}{v_k - v_i} \tag{13}$$

In case three, V_i is moving in front of the V_k and gradually approaching V_k, what means $x_i \leq x_k$ and $v_i \leq v_k$. The duration time of V_i to the V_k is predicted by:

$$d_k^i = \frac{\sqrt{d^2 - (y_i - y_k)^2} + \sqrt{(C_{ik})^2 - (y_i - y_k)^2}}{v_k - v_i} \tag{14}$$

In case four, V_i is moving in front of the V_k and gradually getting far away from V_k, what means $x_i \leq x_k$ and $v_k \leq v_i$. The duration time of V_i to the V_k is predicted by:

$$d_k^i = \frac{\sqrt{d^2 - (y_i - y_k)^2} - \sqrt{(C_{ik})^2 - (y_i - y_k)^2}}{v_i - v_k} \tag{15}$$

Therefore, a summary of the four cases can be:

$$d_k^i = \frac{\sqrt{d^2 - (y_i - y_k)^2}}{|v_i - v_k|} + \frac{\sqrt{(C_{ik})^2 - (y_i - y_k)^2}}{v_i - v_k} \times \frac{x_i - x_k}{|x_i - x_k|} \tag{16}$$

Combining calculation of D_k, the stability of the cluster that uses V_k as its cluster head can be calculated:

$$\mathrm{D}_k = \sqrt{\frac{\sum_{i=1}^{N}\left[\frac{\sqrt{d^2-\left(y_i-y_k\right)^2}}{|v_i-v_k|}+\frac{\sqrt{(C_{ik})^2-\left(y_i-y_k\right)^2}}{v_i-v_k}\times\frac{x_i-x_k}{|x_i-x_k|}\right]^2}{N}} \tag{17}$$

Received signal quality S is another important variable in the clustering algorithm. When assuming that V_k, $\mathrm{k} \in (1, \mathrm{N})$ is the cluster head in the cluster, the received signal quality can be calculated as:

$$S_k = \sum_{i=1}^{N} s_k^i \tag{18}$$

where s_k^i means, the received signal power of V_k from V_i.

For a computation-intensive application, Q is an important parameter to measure the vehicle's network performance. The controller holds the information about vehicles' CPU performance q and monitors the CPU usage δ continually. Computing ability is calculated:

$$Q_k = q_k(1-\delta) \tag{19}$$

where V_k, $\mathrm{k} \in (1, \mathrm{N})$ is the cluster head in the cluster and q_k represents the CPU performance of V_k.

4.3.2 Cluster initialization

SDN controller monitors vehicle' movement such as position and speed information and the vehicle reports network status to the BS periodically, hence the controller has enough knowledge of the vehicular network topology. Vehicles in the communication range of the same base station, are divided into clusters of different sizes. The scale of the vehicle group is defined as:

$$d \le \lambda \times R \tag{20}$$

where the *d* indicates the cluster scale and *R* denotes the value of the largest IEEE 802.11p communication range. λ is a coefficient to control the scale of clusters in different applications.

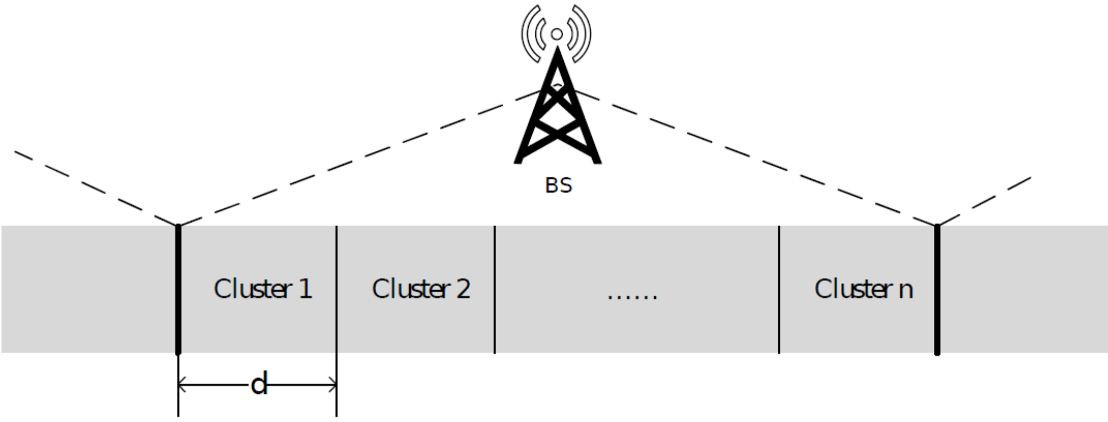

Figure 26 Cluster initialization in groups.

Figure 26 shows the initial grouping result. For the R is a constant value for the vehicle's communication, λ is the only one that influences the cluster scale. For different kinds of applications, λ is selected in different values. For example, the delay-sensitive application needs a large cluster scale to ensure fewer forwarding hop in case the emergency message transmission. The traffic-intensive application needs a small cluster to guarantee network stability and reduce infrequent handover. For the computing-intensive application, the smaller scale of the cluster is needed considering the vehicle's computing capability and cluster stability.

4.3.3 Cluster head selection

After initializing vehicles into clusters, a vehicle is selected as a cluster head for each cluster, and other vehicles in the cluster are regarded as cluster members. With the movement of the vehicles, the clusters cannot be kept the same as is originally created as well as the roles of vehicles in the clusters. The cluster head can be changed by using cluster head selection, according to the different situations. When the cluster accepts the join of a new vehicle, there is no need for the delay-sensitive cluster and traffic-intensive cluster to change the cluster head considering network stability and unnecessary packet loss for the change of cluster head. But for the computation-intensive cluster, the computational capability of cluster head may not be able to satisfy the requirement of application with the join of a new vehicle, so that the cluster head has to be changed by using cluster head selection algorithm. Otherwise, when the clusters are so close to each other that executes the merging process to gather the clusters into the one cluster, the cluster head can be reelected by using a cluster head selection algorithm.

Different cluster head selection methods are proposed for the three different types of applications respectively. To facilitate mathematical calculations and comparisons of vehicles' performance in the cluster, the Min-Max Normalization method is utilized to normalize $\{D, S, Q\}$ into the same range between (0,1).

$$D_k^n = \frac{D_k - min(D_i)}{\max(D_i) - \min(D_i)}, i \in (o, N) \tag{21}$$

$$S_k^n = \frac{S_k - min(S_i)}{\max(S_i) - \min(S_i)}, i \in (o, N) \tag{22}$$

$$Q_k^n = \frac{Q_k - min(Q_i)}{\max(Q_i) - \min(Q_i)}, i \in (o, N) \tag{23}$$

The array $\{D_k^n, S_k^n, Q_k^n\}$ is the result of normalization. By using these data, different clustering algorithms can be carried out to meet the requirement of the different types of applications.

4.3.3.1 Delay-sensitive applications

For delay-sensitive application, the cluster size should be relatively large to decrease the transmission delay which is influenced by multi-hop communications. The Algorithm 3 is the cluster head selection method, where N_i denotes the number of one-hop connections of vehicle i to other vehicles in the same cluster and D_i represents the lifetime of vehicle i belonging into the cluster. After deciding the cluster scale, the vehicle with a maximum number of connections is selected as the CH. However, if multiple vehicles show the same number of connections, the vehicle has the maximal duration time is selected as CH.

Two additional variables are introduced in the algorithm:

- *conn_max* - used to temporarily record the vehicle's maximum number of one-hop connections.
- *life_max* – used to temporarily record the max value of D_k^n for vehicle V_k.

Algorithm 3 CH selection algorithm for delay-sensitive applications

Initialize: *conn_max* = 0, *life_max*=0, *VN* = 0, *CL* = {}

for Each vehicle *i* in the cluster **do**

if *conn_max* < *Ni* **then**

conn_max = *Ni*

For the candidate list *CL*, delete all the members that have less than *conn_max* connections, and add vehicle *i* to the list.

end if

end for

for Each vehicle *j* in the *CL* **do**

if *lif e_max* <Dj^n **then**

lif e_max = Dj^n

VN = j

end if

end for

Select vehicle *VN* as the CH

4.3.3.2 Traffic-intensive applications

For traffic-intensive applications, CH is selected according to the connection duration with the cluster members, and the average received signal quality condition from the other cluster members, as Algorithm 4 shows. Here, S_i represents the average received signal quality of vehicle i. Besides, μ_1 and μ_2 are introduced as variable factors which are used to regulate the algorithm and satisfy:

$$\mu_1 + \mu_2 = 1 \tag{24}$$

The Algorithm 4 introduce additional variables to support cluster head selection:

- *para_max* - used to temporarily record the vehicle's maximum value of $para_i$.
- $para_i$ – Overall performance evaluation parameters related to stability and communication quality of clusters with the vehicle i as their cluster head.

Algorithm 4 CH selection algorithm for traffic-intensive application

Initialize: *para_max* = 0, *VN* = 0
for Every vehicle *i* in the cluster **do**
$para_i = \mu_1 Di^n + \mu_2 S_j^n$
if *para_max* <*parai* **then**
para_max = $para_i$
VN = i
end if
end for
Select vehicle *VN* as the CH

4.3.3.3 Computation-intensive applications

For computation-intensive application, the cluster head is selected according to Algorithm 5 where Q is the computational capability required by the applications and Q_i represents the computational capability of vehicle i. The selection of the cluster head is based on the computational capability of the vehicle and its stability in the cluster.

Algorithm 5 CH selection algorithm for computation-intensive application

Initialize: *life_max* = 0, *V N* = 0, *CL* = {}
for Each vehicle *i* in the cluster **do**
if $Q < Q_i^n$ **then**
i is candidate vehicle as CH
For the candidate list *CL*, delete all the members that cannot satisfy *Q*, and add *i* to the list.
end if
end for
for Each vehicle *j* in the *CL* **do**
if *life_max* < D_i^n **then**
life_max = D_i^n
VN = j
end if
end for
Select vehicle *VN* as the CH

4.3.4 Cluster maintenance and adaptation

Due to dynamic topology changes in VANET, the cluster will not always maintain the initial state, maintenance and adaptation should be employed for the clusters. The maintenance processes are executed in some cases, including vehicle join, the vehicle left, and the merge of clusters.

4.3.4.1 Join of vehicle

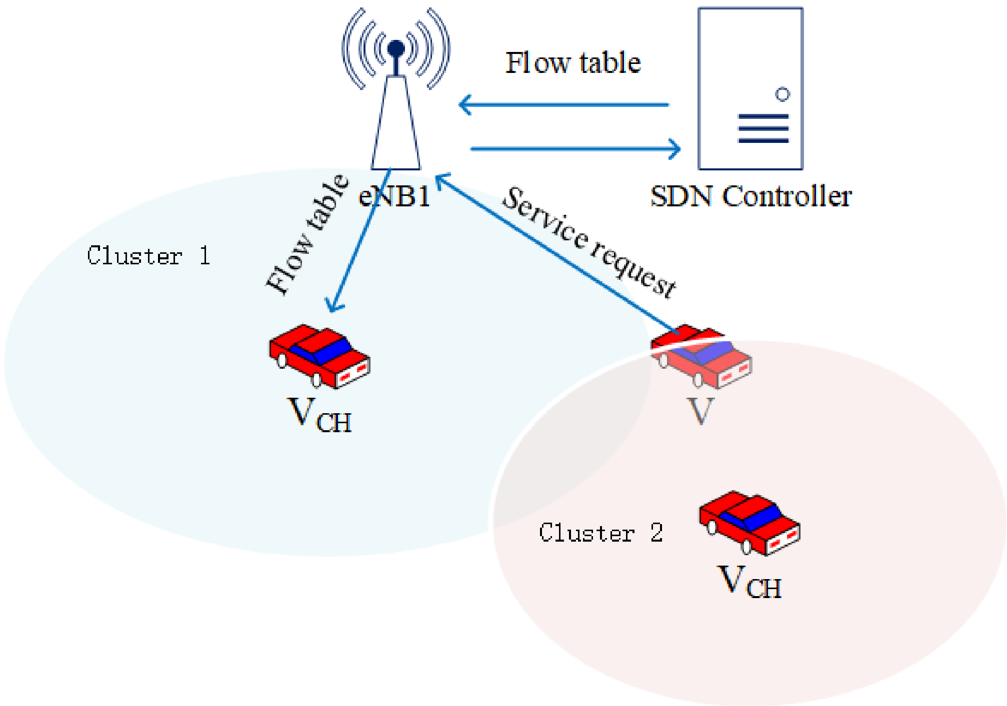

Figure 27 Vehicle request for the network service.

Figure 27 shows the situation where a vehicle is about to join a cluster. When a vehicle starts an application to get service from the network, it must send a request to the base station for the network services at first. After recognizing the vehicle's willingness to join in the network, the base station informs the SDN controller at level 2 of the request from the vehicle. SDN controller maintains the information about vehicles within its management range and responsible for the decision of the vehicle's communication method. The controller firstly distinguishes the type of application service requested by the vehicle, then introduce the vehicle to the appropriate cluster. If there are clusters around the vehicle that can provide the same types of application services, the SDN controller executes the following algorithms to determine the communication method of vehicles.

For delay-sensitive application, the vehicle is able to select a cluster to join by Algorithm 6 where N_i represents the number of vehicles in the cluster i, and d_v^i indicate the communication duration time between vehicle and cluster head of cluster i. Two additional variables are introduced in the algorithm to support cluster selection:

- *conn_para* – used to indicate the cluster's connection status influenced by the number of vehicles in cluster and stability of the cluster.
- *conn_para_max* – enables the max value of conn_para recorded temporarily.

Algorithm 6 Selection of delay-sensitive cluster for the new vehicle

Initialization: *conn_para = 0*, *conn_para_max = 0*, *CH = 0*
for every alternate cluster *i* can join in **do**
    $conn\_para = N_i \times d_v^i$
    if *conn_para_max < conn_para*
        conn_para_max = conn_para
        CH = i
    end if
end for
Select the cluster CH to join in.

For traffic-intensive applications, the vehicle is able to select a cluster to join by Algorithm 7. The algorithm tends to select the cluster that allows the connection between vehicle and cluster head staying longer. Two additional variables are introduced in the algorithm to support cluster selection:

- dur_para_i – used to indicate the duration that the vehicle can stay in the cluster i.
- *dur_para_max* – enables the max value of dur_para recorded temporarily.

Algorithm 7 Selection of traffic-intensive cluster for the new vehicle

Initialization: $dur\_para_i = 0, dur\_para\_max = 0, CH = 0$
for every alternate cluster *i* can join in **do**
        $dur\_para_i = d_v^i$
    if $dur\_para\_max < dur\_para_i$
        $dur\_para\_max = dur\_para_i$
        $CH = i$
    end if
end for
Select the cluster CH to join in.

Algorithm 8 shows the cluster selection rules for vehicle running computation-intensive applications. The algorithm calculates the amount of maximum computation ability that can be performed during the time as a cluster member, and select the cluster with the maximum value. The vehicle requires the cluster to possess computation capability larger than Q and use Q_i to represents the remaining computation ability of cluster i. Two additional variables are introduced in the algorithm to support cluster selection:

- *comp_para* - used to indicate the amount of maximum computation ability.
- *comp_para_max* – use to record the maximum comp_para value.

Algorithm 8 Selection of computation-intensive cluster for the new vehicle

Initialization: $comp\_para = 0, comp\_para\_max = 0, CH = 0$
for every alternate cluster *i* can join in **do**
    if $Q \leq Q_i$ **then**
        comp_para = $Q_i \times d_v^i$
        if $comp\_para\_max < comp\_para$
            $comp\_para\_max = comp\_para$
        $CH = i$
    end if
end for
Select the cluster CH to join in.

If there are no other clusters to join as a cluster member, the vehicle will act as both an independent wireless node waiting to join the new cluster and as a cluster head that using cellular to get network services. After a period of time, if moving into the communication range of the surrounding cluster, the vehicle will give up the role of cluster head and join in the new cluster as a cluster member.

4.3.4.2 Clusters merge

Figure 28 shows the merging scenario that may happen to the network where the communicating range of two clusters (cluster 1 and cluster 2) overlap as well as all vehicles in cluster 1 are located in the range of cluster 2 and similarly all vehicles in cluster 2 are located in the range of cluster 1. SDN controller tries to merge two clusters and forms a larger cluster. For the new cluster, the cluster head selection is executed the same as the section 4.3.3 proposed and the old cluster head of the smaller cluster may become a normal cluster member of the new cluster.

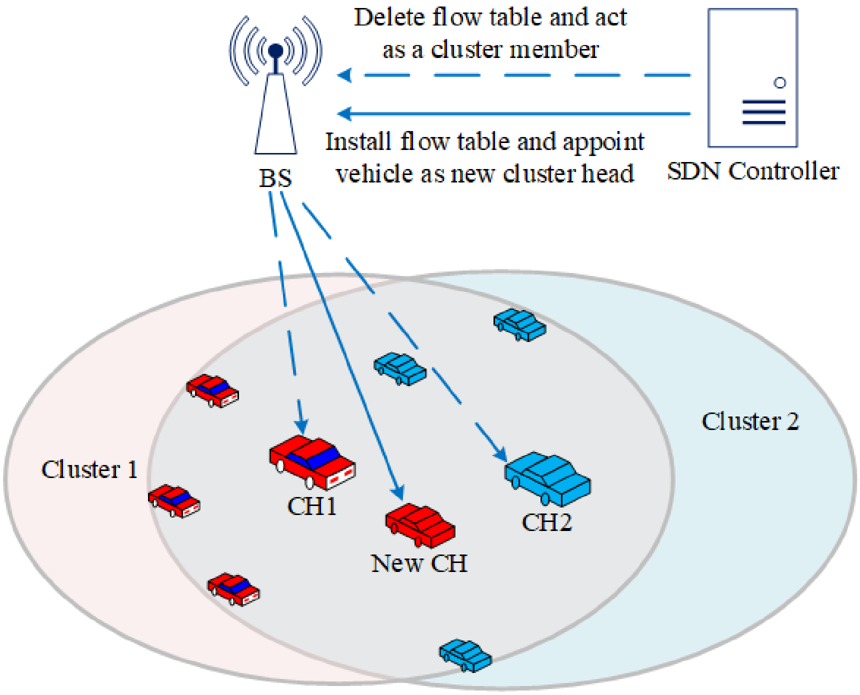

Figure 28 Overlapping cluster merge into one.

However, there is a special case for the computation-intensive cluster that may lead to a failure of merging clusters. It is possible that having no vehicle in the new cluster can achieve the computation capability required by the application. In these cases, the merging process is terminated and two clusters keep unchanged.

4.3.4.3 Vehicle leave

Due to the highly dynamic topology feature of VANET, the Vehicle may leave the current cluster. There are several situations for the vehicle leaving the current cluster.

- When the SDN controller find a vehicle has moved out of the range of the cluster scale (distance between vehicle and cluster head are longer than $\lambda \times R$ which are defined in the (18)) and moves into the range of other clusters, it tries to process handover for the vehicle from current cluster to the new one. In this case, the vehicle is regarded as a new node to join clusters, the SDN controller process the cluster selection algorithm as section 4.3.4.1 describes.

- When a vehicle leaves the range of cluster scale but doesn't find any new clusters around, it continues to perform as a cluster member in the current cluster until it moves out of the longest transmission range from the cluster head. The vehicle will act as both a cluster head with a cellular connection and an independent wireless node waiting to join a new cluster, after leaving the previous cluster.

- When the vehicle acting as a cluster head loses all of its cluster members, the SDN controller regards the cluster dead. The vehicle is still allowed to keep the cellular connection to the network and waiting to join the new cluster.

4.3.5 Clustering procedure summary

According to the description in section 4.3.1 to section 4.3.4, the clustering procedure can be summarized as Figure 29 shows.

At the beginning of the network construction, SDN initializes the network dividing the vehicles into different groups. Then the controller performs a different cluster head selection algorithm for different application requirements. After the successful establishment of the cluster, the controller starts a series of processes to maintain and adapt the cluster changes. The cluster maintenance includes three actions: join of vehicle, cluster merge, vehicle leave. When a vehicle joins the cluster, the controller executes the cluster selection algorithm and returns to the maintenance waiting status to wait for the other changes no matter whether the joining of the vehicle is successful. When finding the two clusters are able to merge, the controller executes the merging algorithm for the clusters, unless the clusters are computation-intensive clusters with no vehicle can satisfy the application requirement in the new cluster, then return to the maintenance waiting for status. When the controller finds a vehicle leaving the cluster, it executes the vehicle leave process, the vehicle that leaves the cluster can act as an independent wireless node and wait for the joining in the new cluster. On the other hand, if the cluster head is left alone without other cluster members, it either acting as a normal vehicle waiting to join the new cluster. After the finish of all maintenance process, the controller returns back to the maintenance waiting status and monitoring the next changes.

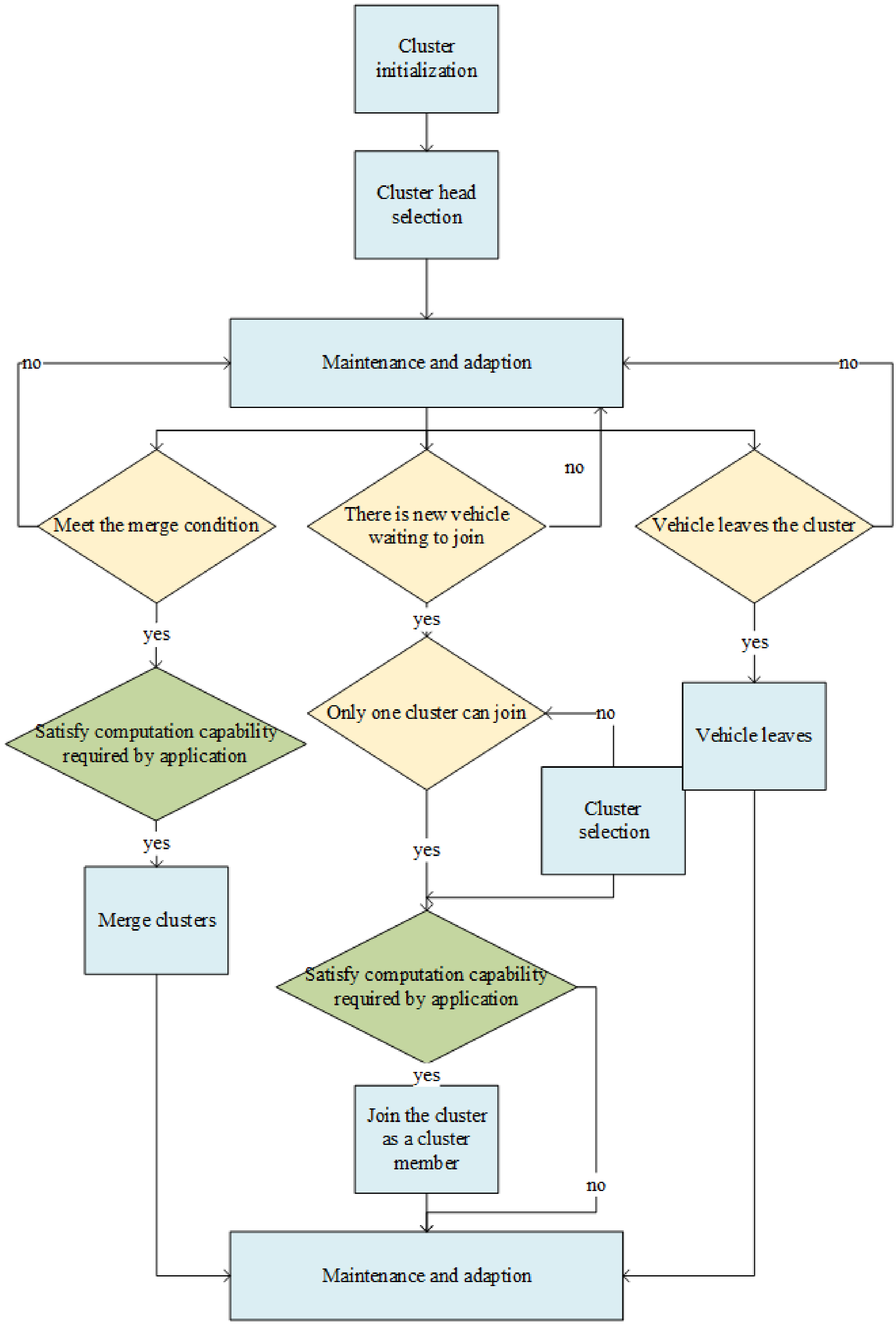

Figure 29 Context-aware clustering procedure.

4.4 Simulation and result

In this section, the improvements of the proposed algorithm on network performance are evaluated by different simulation scenarios. The results show the benefits of the proposed clustering algorithm in terms of network throughput, end-to-end delay, and computation size.

4.4.1 Simulation set up

The same simulators are used as the 3.5 section discussed to conduct network simulations. Table 4 shows the parameters used in the simulations. Simulations are conducted in a topology where numbers of vehicles are running on the 1000m straight road with 4 lanes. A vehicle equipped by both IEEE 802.11p and the cellular interface is set to run a continuous TCP service requirement, connecting to the server in the core network or run UDP applications broadcasting to the other vehicle on the road.

Table 4 Simulation parameter for the clustering scheme

Parameters	**Values**
Transport Layer	TCP(RENO)
Interface	IEEE 802.11p/cellular
Data Rate	6Mbps
Beacon Interval	0.1s
Simulation Topology	Straight road
Topology Size	2000m with 4 lanes

4.4.2 Simulation result

4.4.2.1 For delay-sensitive application

In order to confirm the advantage of the proposed scheme over possible baselines, the influence of different factors in forming clusters are evaluated. The result shown in Figure 30 is based on the simulation that makes the vehicle run UDP application to simulate a network alarm in case of an emergency. Two kinds of network situations, namely, pure IEEE 802.11p and cellular/IEEE 802.11p hybrid networks, are considered.

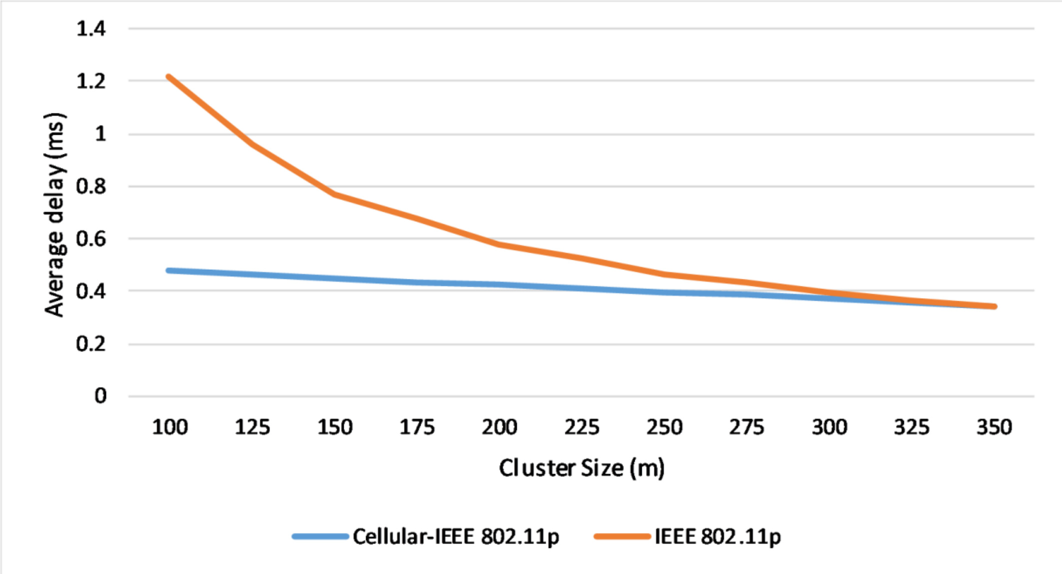

Figure 30 Delay for different cluster sizes.

The result depicts the data transmission delay of a random communication pair in a road with a length of 1000m for different cluster sizes. In this case, the vehicle equipped with only an IEEE 802.11p network interface, floods the emergency message to the network, results in a high transmission delay. On the contrary, the increase in the cellular network interface allows emergency messages to be propagated not only by two hops intra-cluster communication but also by connecting to the cellular network to vehicles in other clusters. For the transmission mechanism of cellular-IEEE 802.11p clustered network uses up to four hops of wireless propagation to reach all vehicles, the communication can get a relatively short transmission delay. Observing the transmission delay that varies with the cluster size, it is clear that large cluster size is better for delay-sensitive applications, which confirms the advantage of the proposed scheme.

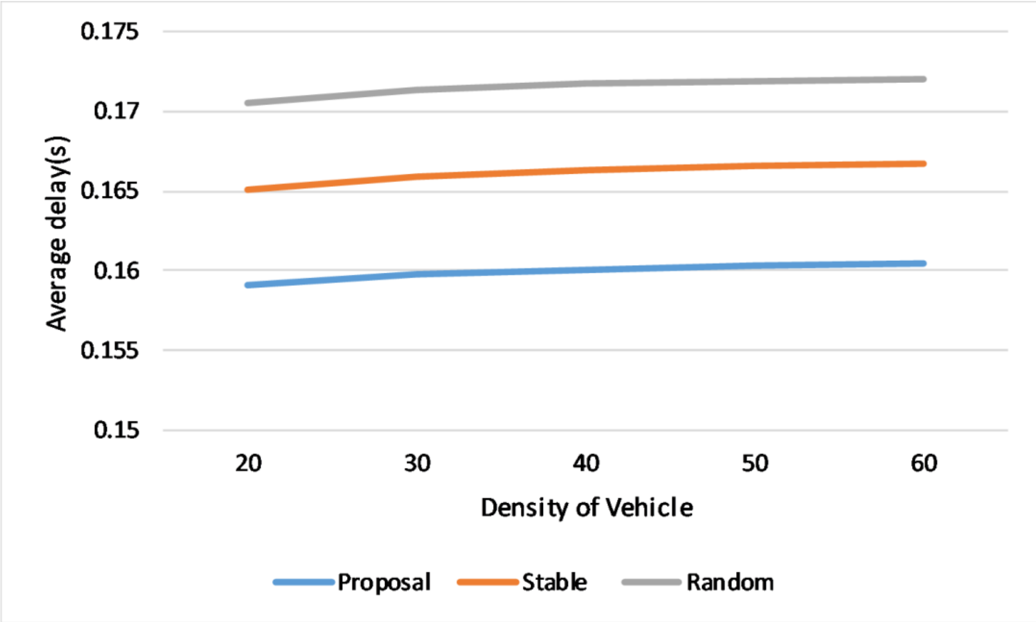

Figure 31 Average delay in different vehicle densities.

Figure 31 shows the influence of vehicle density on the transmission delay of different clustering methods, which are the proposal, stable and random. The stable clustering method selects the vehicle with the longest lifetime among the clusters as the cluster head and selects the cluster with the longest connection time for the vehicle joining the network. In addition, the random clustering method randomly selects cluster heads in the cluster, and randomly selects clusters for vehicles to get network services. In this result, the average delay is extracted from the vehicles which receive the same delay-sensitive message within the range of 1000m. With the growth of vehicle density, the transmission delay also gradually increases, and it is not difficult to see that the proposal considering the number of connections has a lower transmission delay.

4.4.2.2 For traffic-intensive application

To evaluate the influence of cluster lifetime on network performance, the vehicle is set to run TCP application continuously communicating with the server in the wired network. The simulation sets different lifetimes for clusters and compares their impact on average throughput.

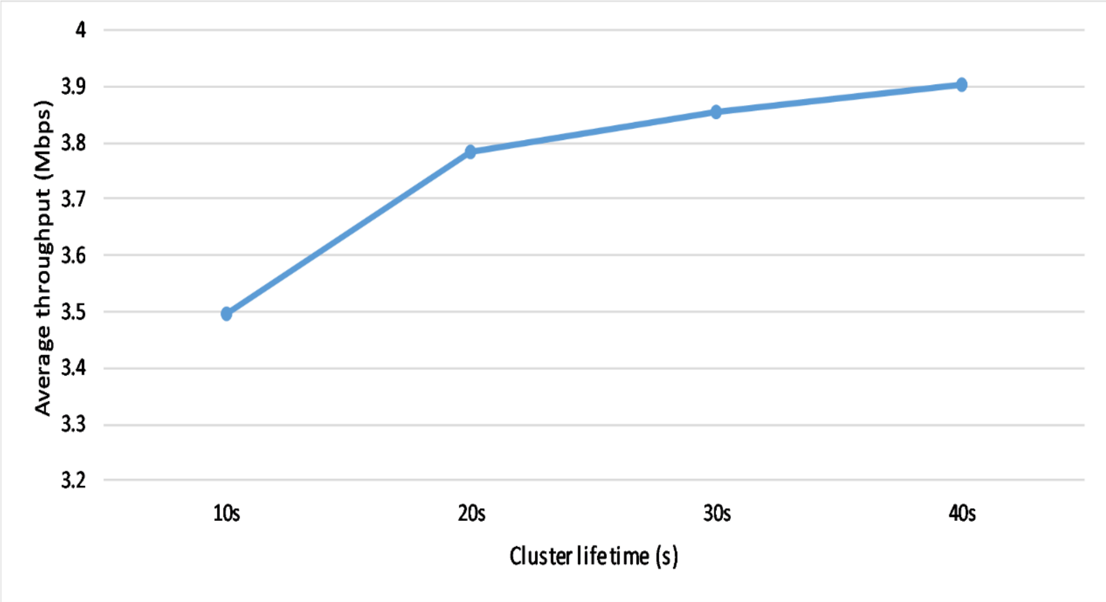

Figure 32 Throughput for different clusters lifetime.

The result shown in Figure 32 makes us consider the importance of connection stability. In cellular-IEEE 802.11p clustered architecture, vehicles are required to communicate via cluster head. When the connection to the cluster is interrupted, the handover process has to be executed, to influence the network performance. As the cluster lifetime increases, the cluster handover frequency gradually decreases, thereby increasing the average throughput. This result shows the importance of considering the connection duration of links between a cluster head and members.

In wireless communication, the received signal strength is an important factor affecting the communication quality. The influence of reception error rates for different signal strengths is evaluated. Figure 33 shows the result that as the received signal becomes stronger, the information loss rate will gradually increase. For traffic-intensive applications, transmission capability and reliability are the most important requirements. Therefore, it is very important to consider the received signal strength to improve the transmission reliability in the clustering algorithm.

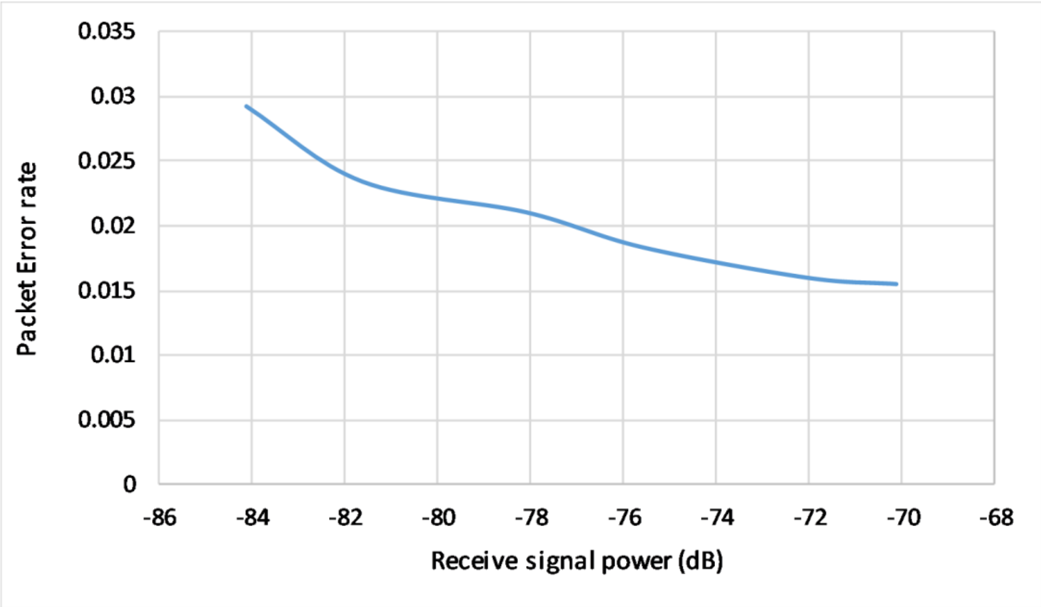

Figure 33 Error rate in different received signal strength.

4.4.2.3 For computation-intensive application

In order to evaluate the data computing ability of the cluster, different cluster sizes are set by defining the different number of cluster members. A network scenario is assumed that vehicles in the same cluster share their surrounding view to the cluster head so that a larger road map can be synthesized by cluster head after a series of computations and shared with the cluster members. To achieve this, vehicles are set to run TCP applications communicating with the cluster head continuously. Different lifetimes are set for clusters to compare their influence on computed data size.

Figure 34 Data size for computation in various cluster sizes.

Figure 34 shows the result of computing data size influenced by the number of cluster members and the cluster's lifetime. Because of the different computational complexity for computing applications, this network scenario only considers the consumption of cluster head computing capability caused by data transmission. As a result, the requirement for the computing capability is far less than the computing capability possessed by the cluster head. Even so, it can be still learned from the result that with the increasing number of cluster members, it becomes more important to consider the computational capability of the cluster head node as more requests are expected to happen. Meanwhile, the consideration of cluster stability ensures that the cluster head node could satisfy the requests from a large number of vehicles.

The computing speeds of data with different data sizes are also evaluated when processing different clustering algorithms with the cluster size is set as 200m. This scenario uses the TCP application to simulate the computation-intensive application which allows the vehicle to send the data to be computed on the cluster head and then returns to the vehicle after a series of calculations. The result in Figure 35 shows the total time required by the computing process.

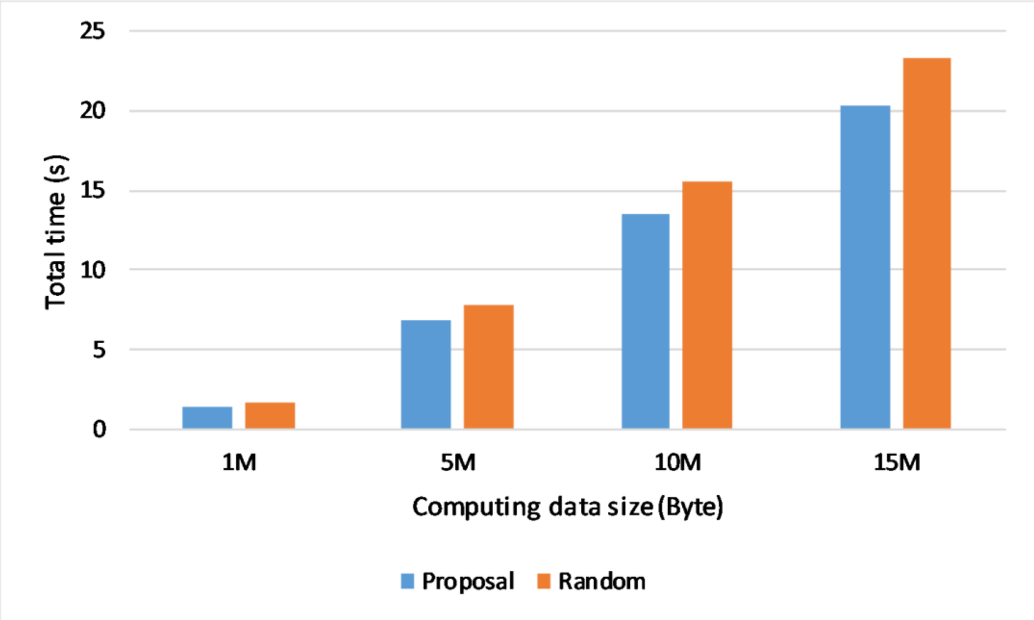

Figure 35 The total required time in computing different sizes of data.

The total time includes propagation delay, data transmission delay, and calculation delay. When the size of computing data becomes larger, the total time is mainly affected by data transmission delay and computation delay. When the data transmission capabilities are the same, the computing capability of the cluster head is the main factor that affects the change of total time. It can be seen from the results that the proposal that considers the computing capability of the cluster head can reduce the data computing time, which brings great advantages in the highly dynamic vehicular network.

4.5 Conclusion

In this chapter, an SDN-based context-aware clustering algorithm was proposed to the VANET. The introduction of the SDN providing a global view of the network made the clustering algorithm more flexible and easier to deploy. Scheduling the vehicle's communication using a single clustering algorithm was hard to satisfy all network scenarios.

Therefore, to meet different kinds of application requirements as much as possible, in section 4.2, the applications were classified into three types: delay-sensitive application, traffic-intensive application, and computation-intensive application according to their transmission behavior recorded by the SDN controller. The independent clusters could be established according to different application types.

The context aware clustering algorithm was explained in detail in section 4.3. The three most important parameters in the clustering algorithm were introduced, which were duration time of cluster, received signal quality, and computation capability. With the help of the controller, these parameters could be calculated according to the information of the vehicle. The procedures of the clustering algorithm included cluster initialization, cluster selection, and cluster maintenance and adaptation.

At last, in section 4.4, the evaluations of the proposed scheme were conducted to confirm the advantages in network performance, in terms of throughput, end-to-end delay, and computed data size. The results showed the influence of cluster size on the delay-sensitive application, as well as the importance of connection lifetime between cluster members and clusters for traffic-intensive applications. In addition, for the computation-intensive application, the computed data became more with the growing number of cluster members, so that the computation capability became more important in clustering algorithm, as too many cluster members may cause the exhaustion of computation capability. Therefore, it can be conclude that it is important to consider the different QoS requirements in the clustering algorithm.

Chapter 5

5 Conclusion and future work

This chapter summarizes the research on SDN-based clustered VANET. Section 5.1 concludes the research from two aspects: SDN-based VANET handover and context-aware clustering. Then, in section 5.2, discuss possible future works.

5.1 Conclusion

In VANET, the vehicle with its high mobility feature forms a dynamic network topology happens frequently handover. A clustered architecture of VANET makes the network benefit not only from saving scarce spectrum resources but also from reducing handover frequency. SDN technology is a promising technology providing centralized control to realize network programmability and flexibility and scalability. In this thesis, an SDN-based handover scheme and context aware cluster algorithm were proposed.

In chapter 3, SDN technology was introduced in the VANET to achieve seamless handover. Providing a global view of the network, SDN makes it easy to detect handover happens and allow execute handover scheme before it happens. In addition, the MEC servers are employed in the network to achieve caching function when handover happens. A caching queue management algorithm was also explained to help seamless handover. Considering the handover happens to the cluster, the handover procedure between both different clusters and different roadside infrastructures were explained. The handover scheme also tried to use SDN to achieve rerouting after the handover. Data traffic can be rerouted to the new path by downloading a flow table instructed by the SDN controller to a fork router. The fork router was selected by the algorithm that aims to achieve less communication delay and signaling overhead caused by the path change. Simulations were conducted to evaluate the proposed handover scheme. The results show that the handover scheme with SDN employment performs better when compared with the traditional network with connection interruption especially in the network with frequent topology changes and high communication quality. Moreover, the deployment of the MEC caching server, brought further improvement in network performance, in terms of the average throughput and end-to-end transmission delay.

In section 4, an SDN-based context-aware clustering algorithm was proposed. The introduction of SDN makes clustering easier to deploy and allows the clustering algorithm to execute only when it is needed instead of asking for the status periodically. Considering that a single clustering algorithm in forming clusters does not appropriate to support various applications, different clustering algorithms were proposed respectively to meet the requirements of different kinds of applications, including delay-sensitive application, traffic-intensive application, and computation-intensive application. The algorithms were based on the parameters which were the lifetime in the cluster, received signal quality, and computation capability that could be calculated from the network context. The procedures of the clustering algorithm were explained that include cluster initialization, cluster head selection, and cluster maintenance and adaption. The simulation was conducted to evaluate the proposed clustering algorithm. The results showed that large cluster size would perform a short end-to-end delay on delay-sensitive application. On the other hand, longer connection lifetime is preferred to the traffic-intensive application as shown in the results. Besides, for the computation-intensive application, the computation capability is the most important parameter, as too many cluster members divide limited resources may cause the exhaustion of computing capability in cluster head. Therefore, considering the different network requirements is important in executing cluster algorithms.

5.2 Future work

This thesis has introduced the SDN technology to the VANET, proposed a seamless handover scheme to support V2I communication and context aware clustering algorithm to satisfy the various requirements from applications. To further improve the SDN-based V2I communication in supporting the mobility of vehicles, there are still many unsolved problems that should be considered.

Simulation improvements: Considering the possibility of three applications being carried out at the same time, evaluation of the mutual influence of the three applications when they are carried out simultaneously will be the next work. In chapter 4, the simulation is conducted for the different kinds of applications respectively. In this paper, in order to verify the proposed technology, the two proposals were evaluated separately. In the future, the two proposals can be combined for further research.

The interaction between level1 and level2 controller: In the SDN-based network, the controller at the edge of the network needs the cooperation of the center controller to solve some problem. For example, in this thesis, when the handover happens between the different level 2 controller's domain, level1 controller needs to involve in the handover scheme and determine the new data traffic path by its global view which brings the large cost to the network.

Machine learning usage in clustering: In the proposed context-aware clustering algorithm, the applications were classified into different types by simple context extract from the network devices. But this classification method is not very accurate in some situations. Machine learning technology with its high-precision and fast processing ability is considered to use in the classified work. Moreover, as VANET can be regarded as a social network and its movement performs particular rules, machine learning can be regarded as a method to predict the vehicle's movement in helping the clustering algorithm.

Reference

[1] G. Dimitrakopoulos and P. Demestichas, "Intelligent Transportation Systems," in *IEEE Vehicular Technology Magazine*, vol. 5, no. 1, pp. 77-84, March 2010

[2] B. Chang and J. Chiou, "Cloud Computing-Based Analyses to Predict Vehicle Driving Shockwave for Active Safe Driving in Intelligent Transportation System," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, pp. 852-866, Feb. 2020.

[3] F. Goudarzi, H. Asgari and H. S. Al-Raweshidy, "Traffic-Aware VANET Routing for City Environments—A Protocol Based on Ant Colony Optimization," in *IEEE Systems Journal*, vol. 13, no. 1, pp. 571-581, March 2019.

[4] B. Sharma, M. S. P. Sharma and R. S. Tomar, "A survey: Issues and challenges of vehicular ad hoc networks (VANETs)", *International Conference on Sustainable Computing in Science, Technology & Management (SUSCOM)*, Jaipur, India, 2019. Available: https://ssrn.com/abstract=3363555.

[5] L. Nkenyereye, L. Nkenyereye, S. M. R. Islam, C. A. Kerrache, M. Abdullah-Al-Wadud and A. Alamri, "Software Defined Network-Based Multi-Access Edge Framework for Vehicular Networks," in *IEEE Access*, vol. 8, pp. 4220-4234, 2020.

[6] Y. Liu, "Intelligent Processing in Vehicular Ad hoc Networks: a Survey," *Networking and Internet Architecture (cs.NI)*, pp. 1-11, Mar. 2019.

[7] C. Wu, Z. Liu, D. Zhang, T. Yoshinaga and Y. Ji, "Spatial Intelligence toward Trustworthy Vehicular IoT," in *IEEE Communications Magazine*, vol. 56, no. 10, pp. 22-27, OCTOBER 2018.

[8] C. Wu, S. Ohzahata and T. Kato, "Flexible, Portable, and Practicable Solution for Routing in VANETs: A Fuzzy Constraint Q-Learning Approach," in *IEEE Transactions on Vehicular Technology*, vol. 62, no. 9, pp. 4251-4263, Nov. 2013.

[9] IEEE Standards Association, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 6: Wireless Access in Vehicular Environments, " IEEE Standard for Information technology, 802.11p, 2010.

[10] A. Gupta, R. Singh, D. Ather and R. S. Shukla, "Comparison of various routing algorithms for VANETS," *2016 International Conference System Modeling & Advancement in Research Trends (SMART)*, Moradabad, 2016, pp. 153-157.

[11] F. Z. Yousaf, M. Bredel, S. Schaller, and F. Schneider, “NFV and SDN—Key Technology Enablers for 5G Networks,” in *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2468–2478, 2017.

[12] L. Chettri and R. Bera, "A Comprehensive Survey on Internet of Things (IoT) Toward 5G Wireless Systems," in *IEEE Internet of Things Journa*l, vol. 7, no. 1, pp. 16-32, Jan.

2020.

[13] P. Zhang, X. Yang, J. Chen and Y. Huang, "A survey of testing for 5G: Solutions, opportunities, and challenges," in *China Communications*, vol. 16, no. 1, pp. 69-85, Jan. 2019.

[14] E. T. Sağlam and Ş. Bahtiyar, "A Survey: Security and Privacy in 5G Vehicular Networks," *2019 4th International Conference on Computer Science and Engineering (UBMK)*, Samsun, Turkey, 2019, pp. 108-112.

[15] W. B. Jaballah, M. Conti and C. Lal, "A survey on software-defined VANETs: benefits, challenges, and future directions, " *arXiv:1904.04577*, 2019, [online] Available: https://arxiv.org/abs/1904.04577.

[16] OpenFlow Switch Specification. [online] Available: https://www. opennetworking.org/images/ stories/downloads/specification/ openflow-spec-v1.3.0.pdf.

[17] V. Nguyen, A. Brunstrom, K. Grinnemo and J. Taheri, "SDN/NFV-Based Mobile Packet Core Network Architectures: A Survey," in *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1567-1602, third quarter 2017.

[18] S. Bera, S. Misra, and A. V. Vasilakos, "Software-defined networking for internet of things: A survey," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1994–2008, 2017.

[19] K.-K. Yap, M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown, "Openroads: Empowering research in mobile networks," in *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 1, pp. 125–126, 2010.

[20] M. Kobayashi, S. Seetharaman, G. Parulkar, G. Appenzeller, J. Little, J. van Reijendam, P. Weissmann, and N. McKeown, "Maturing of openflow and software-defined networking through deployments," in *Computer Networks*, vol. 61, pp. 151 – 175, 2014.

[21] J. Vestin, P. Dely, A. Kassler, N. Bayer, H. Einsiedler, and C. Peylo, "Cloudmac: Towards software defined wlans," in *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 16, no. 4, pp. 42–45, 2013.

[22] I. Ku, Y. Lu, M. Gerla, F. Ongaro, R. L. Gomes, and E. Cerqueira, "Towards software-defined VANET: Architecture and services," *13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*, Piran, Slovenia 2014, pp. 103-110.

[23] M. N. Irshad, L. Du, I. A. Khoso, T. B. Javed and M. M. Aslam, "A Hybrid Solution of SDN Architecture for 5G Mobile Communication to Improve Data Rate Transmission," *28th Wireless and Optical Communications Conference (WOCC)*, Beijing, China, 2019, pp. 1-5.

[24] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," in *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14-76, Jan. 2015.

[25] T. Huang, F. R. Yu, C. Zhang, J. Liu, J. Zhang and Y. Liu, "A Survey on Large-Scale Software Defined Networking (SDN) Testbeds: Approaches and Challenges," in *IEEE*

Communications Surveys & Tutorials, vol. 19, no. 2, pp. 891-917, Second quarter 2017.

[26] N. McKeown, T. Anerson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner, "OpenFlow: enabling innovation in campus networks," In *ACM SIGCOMM Computer Communication Review*, vol.38, no.2, pp. 69-74, 2008.

[27] OpFlex Control Protocol OPFLEX, [online] Available: https://tools.ietf.org/html/draft-smith-opflex-03.

[28] Network configuration protocol (NETCONF), [online] Available: https://tools.ietf.org/html/rfc6241.

[29] M. Casado, T. Garfinkel, A. Akella, M. J. Freedman, D. Boneh, N. McKeown, S. Shenker, "SANE: A protection architecture for enterprise networks", *15th Conference USENIX Security Symposium*, vol. 15, pp. 137-151, 2006.

[30] D. Gunter, R. Kettimuthu, E. Kissel, M. Swany, J. Yi and J. Zurawski, "Exploiting Network Parallelism for Improving Data Transfer Performance," *SC Companion: High Performance Computing, Networking Storage and Analysis*, Salt Lake City, UT, USA, 2012, pp. 1600-1606.

[31] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, J. Zolla, "B4: Experience with a globally-deployed software defined WAN," *in ACM SIGCOMM Computer Communication Review*, vol. 43, no.4, pp. 3-14, 2013.

[32] K. Pentikousis, Y. Wang and W. Hu, "Mobileflow: Toward software-defined mobile networks," in *IEEE Communications Magazine*, vol. 51, no. 7, pp. 44-53, 2013.

[33] L. E. Li, Z. M. Mao, J. Rexford, "Toward software-defined cellular networks," *European Workshop on Software Defined Networking (EWSDN)*, Darmstadt, Germany, pp. 7-12, 2012.

[34] J. Lai, L. Jiang, M. Lei, A. Abdollahpouri, W. Fang, "Software-defined cellular networking: A practical path towards 5G," *International Journal of Communication Networks and Distributed Systems*, vol. 14, no. 1, pp. 89-105, 2015.

[35] F. Z. Yousaf, M. Bredel, S. Schaller and F. Schneider, "NFV and SDN—Key Technology Enablers for 5G Networks," in *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2468-2478, Nov. 2017.

[36] I. Yaqoob, I. Ahmad, E. Ahmed, A. Gani, M. Imran and N. Guizani, "Overcoming the Key Challenges to Establishing Vehicular Communication: Is SDN the Answer?," in *IEEE Communications Magazine*, vol. 55, no. 7, pp. 128-134, July 2017.

[37] C. Jiacheng, Z. Haibo, Z. Ning, Y. Peng, G. Lin and S. X. Sherman, "Software Defined Internet of Vehicles: Architecture, Challenges and Solutions," in *Journal of Communications and Information Networks*, vol. 1, no. 1, pp. 14-26, June 2016.

[38] J. Liu, J. Wan, B. Zeng, Q. Wang, H. Song, and M. Qiu, "A scalable and quick-response software defined vehicular network assisted by mobile edge computing, " in *IEEE*

Communications Magazine, vol. 55, no. 7, pp. 94–100, 2017.

[39] N. B. Truong, G. M. Lee and Y. Ghamri-Doudane, "Software defined networking-based vehicular Adhoc Network with Fog Computing," *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, Ottawa, ON, Canada, 2015, pp. 1202-1207.

[40] A. Ydenberg, N. Heir and B. Gill, "Security, SDN, and VANET technology of driver-less cars," *IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, 2018, pp. 313-316.

[41] J. Weng, J. Weng, Y. Zhang, W. Luo and W. Lan, "BENBI: Scalable and Dynamic Access Control on the Northbound Interface of SDN-Based VANET," in *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 822-831, Jan. 2019.

[42] M. Azizian, S. Cherkaoui, and A. S. Hafid, “Vehicle software updates distribution with sdn and cloud computing,” in *IEEE Communications Magazine*, vol. 55, no. 8, pp. 74–79, 2017.

[43] V. Yazıcı, U. C. Kozat and M. O. Sunay, "A new control plane for 5G network architecture with a case study on unified handoff, mobility, and routing management," in *IEEE Communications Magazine*, vol. 52, no. 11, pp. 76-85, Nov. 2014.

[44] A. Soua and S. Tohme, "Multi-level SDN with vehicles as fog computing infrastructures: A new integrated architecture for 5G-VANETs," *21st Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, Paris, France, 2018, pp. 1-8.

[45] Z. He, D. Zhang, and J. Liang, "Cost-efficient sensory data transmission in heterogeneous software-defined vehicular networks, " in *IEEE Sensors Journal*, vol. 16, no. 20, pp. 7342–7354, 2016.

[46] C. Bouras, A. Kollia and A. Papazois, "SDN & NFV in 5G: Advancements and challenges," *20th Conference on Innovations in Clouds, Internet and Networks (ICIN)*, Paris, France, 2017; pp.107–111.

[47] M. Ruffini, "Multidimensional Convergence in Future 5G Networks," in *Journal of Lightwave Technology*, vol. 35, no. 3, pp. 535-549, 2017.

[48] J. Mi, K. Wang, P. Li, S. Guo and Y. Sun, "Software-Defined Green 5G System for Big Data," in *IEEE Communications Magazine*, vol. 56, no. 11, pp. 116-123, 2018.

[49] X. Duan, X. Wang, Y. Liu and K. Zheng, "SDN Enabled Dual Cluster Head Selection and Adaptive Clustering in 5G-VANET," in *IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Montreal, QC, Canada, 2016, pp. 1-5.

[50] L. Xie, Y. Ding, H. Yang and X. Wang, "Blockchain-Based Secure and Trustworthy Internet of Things in SDN-Enabled 5G-VANETs," in *IEEE Access*, vol. 7, pp. 56656-56666, 2019.

[51] S. S. Salihin, L. A. Nissirat, R. M. Noor and I. Ahmedy, "Handover Schemes for Vehicular Ad-Hoc Networks Over Long Term Evolution Advanced: A Survey," *International Conference on Computational Approach in Smart Systems Design and*

Applications (ICASSDA), Kuching, Malaysia, 2018, pp. 1-7.

[52] B. Haider, S. Henna, A. Gul and F. Aadil, "A Survey on Mobility Management Techniques in VANETs," *IEEE International Conference on Computer and Information Technology (CIT)*, Nadi, Fiji, 2016, pp. 125-133.

[53] IP Mobility Support for IPv4, Available [online]: https://tools.ietf.org/rfc/rfc3344.txt

[54] Mobility support in IPv6, Available [online]: https://tools.ietf.org/rfc/rfc3775.txt

[55] J. Xie and U. Narayanan, "Performance Analysis of Mobility Support in IPv4/IPv6 Mixed Wireless Networks," in *IEEE Transactions on Vehicular Technology*, vol. 59, no. 2, pp. 962-973, 2010.

[56] Transmission Control Protocol, Available [online]: https://tools.ietf.org/rfc/rfc793.txt

[57] Stream Control Transmission Protocol, Available [online]: https://tools.ietf.org/rfc/rfc4960.txt, IETF, 2007

[58] Mobile SCTP for transport layer mobility, Available [online]: https://tools.ietf.org/id/draft-sjkoh-sctp-mobility-04.txt

[59] TCP Extensions for Multipath Operation with Multiple Addresses, Available [online]: https://tools.ietf.org/rfc/rfc6824.txt

[60] T. Zhang, S. Zhao and B. Cheng, "Multipath Routing and MPTCP-Based Data Delivery Over Manets," in *IEEE Access*, vol. 8, pp. 32652-32673, 2020.

[61] I. Kustiawan and K. Chi, "Handoff Decision Using a Kalman Filter and Fuzzy Logic in Heterogeneous Wireless Networks," in *IEEE Communications Letters*, vol. 19, no. 12, pp. 2258-2261, Dec. 2015.

[62] H. Li and J. Xie, "A Handoff Solution in Wireless Mesh Networks by Implementing Split Channels," *IEEE Global Telecommunications Conference (GLOBECOM)*, Miami, FL, USA, 2010, pp. 1-5.

[63] X. Chen, H. M. Jones and D. Jayalath, "Channel-Aware Routing in MANETs with Route Handoff," in *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 108-121, Jan. 2011.

[64] Y. S. Chen, C. S. Hsu and C. H. Cheng, "Network Mobility Protocol for Vehicular Ad Hoc Networks," in *International Journal of Communication Systems*, vol. 27, no. 11, pp. 3042-3063. 2014.

[65] A. Prakash, S. Tripathi, R. Verma, N. Tyagi, R. Tripathi, K. Naik, "Vehicle assisted cross-layer handover scheme in NEMO-based VANETs (VANEMO), " in *International Journal of Internet Protocol Technology*, vol. 6, no. 1-2, pp.83–95, 2011.

[66] S. Vodopivec, J. Bešter and A. Kos, "A survey on clustering algorithms for vehicular ad-hoc networks," *35th International Conference on Telecommunications and Signal Processing (TSP)*, Prague, 2012, pp. 52-56.

[67] K. A. Hafeez, L. Zhao, Z. Liao and B. N. Ma, "A fuzzy-logic-based cluster head selection algorithm in VANETs," *IEEE International Conference on Communications (ICC)*, Ottawa, ON, Canada, 2012, pp. 203-207.

[68] H. Ahmed, S. Pierre and A. Quintero, "A Cooperative Road Topology-Based Handoff Management Scheme," in *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3154-3162, 2019.

[69] S. Taha and X. Shen, "Lightweight Group Authentication with Dynamic Vehicle-Clustering for 5G-Based V2X Communications," *IEEE Global Communications Conference (GLOBECOM)*, Abu Dhabi, United Arab Emirates, 2018, pp. 1-6.

[70] M. Jalil Piran, N. H. Tran, D. Y. Suh, J. B. Song, C. S. Hong and Z. Han, "QoE-Driven Channel Allocation and Handoff Management for Seamless Multimedia in Cognitive 5G Cellular Networks," in *IEEE Transactions on Vehicular Technology*, vol. 66, no. 7, pp. 6569-6585, 2017.

[71] G. Araniti, C. Campolo, M. Condoluci, A. Iera and A. Molinaro, "LTE for vehicular networking: a survey," in *IEEE Communications Magazine*, vol. 51, no. 5, pp. 148-157, 2013.

[72] W. Du, Q. Liu, Z. Gao and G. Tan, "Seamless Vertical Handoff Protocol for LTE-802.11p Hybrid Vehicular Communications Over the Tactile Internet," *IEEE International Symposium on Haptic, Audio and Visual Environments and Games (HAVE)*, Dalian, China, 2018, pp. 1-5.

[73] S. K. Singh, "Performance evaluation of beacons control data dissemination protocol in handover scenario for VANET," *IEEE International Conference on Electrical, Instrumentation and Communication Engineering (ICEICE)*, Karur, India, 2017, pp. 1-6.

[74] C. Cooper, D. Franklin, M. Ros, F. Safaei and M. Abolhasan, "A Comparative Survey of VANET Clustering Techniques," in *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 657-681, First quarter 2017.

[75] S. M. AlMheiri and H. S. AlQamzi, "MANETs and VANETs clustering algorithms: A survey," 2015 *IEEE 8th GCC Conference & Exhibition*, Muscat, Oman, 2015, pp. 1-6.

[76] A. Ephremides, J. E. Wieselthier and D. J. Baker, "A design concept for reliable mobile radio networks with frequency hopping signaling," in *Proceedings of the IEEE*, vol. 75, no. 1, pp. 56-73, Jan. 1987.

[77] M. Gerla and J. T.-C. Tsai, “Multicluster, mobile, multimedia radio network,” in *Wireless Network*, vol. 1, no. 3, pp. 255–265, 1995.

[78] A. K. Parekh, "Selecting routers in ad-hoc wireless networks," in *Proceedings of the SBT/IEEE International Telecommunications Symposium*. Vol. 204. Rio de Janeiro, Brazil, 1994.

[79] P. Basu, N. Khan and T. D. C. Little, "A mobility based metric for clustering in mobile ad hoc networks," *21st International Conference on Distributed Computing Systems*

Workshops, Mesa, USA, 2001, pp. 413-418

[80] Y. Zhang and J. M. Ng, "A Distributed Group Mobility Adaptive Clustering Algorithm for Mobile Ad Hoc Networks," *IEEE International Conference on Communications*, Beijing, China, 2008, pp. 3161-3165.

[81] Z. Y. Rawashdeh and S. M. Mahmud, "A novel algorithm to form stable clusters in vehicular ad hoc networks on highways, " *Eurasip Journal on Wireless Communications and Networking*, pp. 1-13, 2012.

[82] Z. Ning, X. Wang, X. Kong and W. Hou, "A Social-Aware Group Formation Framework for Information Diffusion in Narrowband Internet of Things," in *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1527-1538, 2018.

[83] M. A. Togou, A. Hafid and P. K. Sahu, "A stable minimum velocity CDS-based virtual backbone for VANET in city environment," *39th Annual IEEE Conference on Local Computer Networks*, Edmonton, Canada, 2014, pp. 510-513.

[84] C. Shea, B. Hassanabadi and S. Valaee, "Mobility-Based Clustering in VANETs Using Affinity Propagation," *IEEE Global Telecommunications Conference (GlOBECOM)*, Honolulu, USA, 2009, pp. 1-6.

[85] M. Ni, Z. Zhong and D. Zhao, "MPBC: A Mobility Prediction-Based Clustering Scheme for Ad Hoc Networks," in *IEEE Transactions on Vehicular Technology*, vol. 60, no. 9, pp. 4549-4559, 2011.

[86] S. Kuklinski and G.Wolny, "Density based clustering algorithm for vehicular ad-hoc networks," in *International Journal of Internet Protocol Technology*, vol. 4, no. 3, pp. 149-157, 2009.

[87] L. A. Maglaras and D. Katsaros, "Distributed clustering in vehicular networks," *IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Barcelona, spain, 2012, pp. 593-599.

[88] S. Ucar, S. C. Ergen and O. Ozkasap, "Multihop-Cluster-Based IEEE 802.11p and LTE Hybrid Architecture for VANET Safety Message Dissemination," in *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2621-2636, 2016.

[89] E. Dror, C. Avin and Z. Lotker, "Fast randomized algorithm for hierarchical clustering in Vehicular Ad-Hoc Networks," the10th IFIP Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), Favignana Island, Sicily, 2011, pp. 1-8.

[90] D. Zhang, H. Ge, T. Zhang, Y. Cui, X. Liu and G. Mao, "New Multi-Hop Clustering Algorithm for Vehicular Ad Hoc Networks," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1517-1530, April 2019.

[91] W. Qi, Q. Song, X. Wang, L. Guo and Z. Ning, "SDN-Enabled Social-Aware Clustering in 5G-VANET Systems," in *IEEE Access*, vol. 6, pp. 28213-28224, 2018.

[92] Host Identity Protocol (HIP) Architecture, Available [online]: https://tools.ietf.org/rfc/rfc4423.txt, IETF, 2006.

[93] Identifier-Locator Network Protocol (ILNP) Architectural Description, Available [online]: https://tools.ietf.org/rfc/rfc6740.txt, IETF, 2012.

[94] G. Remy, S. Senouci, F. Jan and Y. Gourhant, "LTE4V2X: LTE for a Centralized VANET Organization," *IEEE Global Telecommunications Conference (GLOBECOM)*, Houston, TX, USA, 2011, pp. 1-6.

[95] B. A. A. Nunes, M. Mendonca, X. Nguyen, K. Obraczka and T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," in *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1617-1634, Third Quarter 2014.

[96] N. Abbas, Y. Zhang, A. Taherkordi and T. Skeie, "Mobile Edge Computing: A Survey," in *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450-465, 2018.

[97] M. Patel, B. Naughton, C. Chan, N. Sprecher, S. Abeta and A. Neal, "Mobile-edge computing introductory technical white paper," *White Paper, Mobile-edge Computing (MEC) industry initiative*, pp. 1089-7801, 2014.

[98] D. Satria, D. Park, and M. Jo, "Recovery for overloaded mobile edge computing," in *Future Generation Computer Systems*, vol. 70, pp. 138–147, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X16302096

[99] Y. Wang, K. Wang, H. Huang, T. Miyazaki and S. Guo, "Traffic and computation co-offloading with reinforcement learning in fog computing for industrial applications," in *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 976-986, 2019.

[100] X. He, K. Wang and W. Xu, "QoE-Driven Content-Centric Caching With Deep Reinforcement Learning in Edge-Enabled IoT," in *IEEE Computational Intelligence Magazine*, vol. 14, no. 4, pp. 12-20, 2019.

[101] H. Balakrishnan, S. Seshan, and R. Katz. "Improving reliable transport and handoff performance in cellular wireless networks", in *Wireless Networks*, vol. 1, no. 4, pp. 469-481, 1995.

[102] T. Nguyen, M. Park, Y. Youn and S. Jung, "An improvement of TCP performance over wireless networks," *Fifth International Conference on Ubiquitous and Future Networks (ICUFN)*, Da Nang, Vietnam, 2013, pp. 214-219.

[103] S. A. Bugti, X. Chunhe, L. Wie and E. Hussain, "Cluster based addressing scheme in VANET (CANVET stateful addressing approach)," *IEEE 3rd International Conference on Communication Software and Networks*, Xi'an, China, 2011, pp. 450-454.

[104] C. Konstantopoulos, D. Gavalas, and G. Pantziou, "Clustering in mobile ad hoc networks through neighborhood stability-based mobility prediction," in *Computer Networks*, vol. 52, no. 9, pp. 1797–1824, 2008.

[105] W. Dong, F. Lin, H. Zhang and Y. Yin, "A cluster-based recursive broadcast routing algorithm to propagate emergency messages in city VANETs," *IEEE 9th International Conference on Communication Software and Networks (ICCSN)*, Guangzhou, China,

2017, pp. 187-190.

[106] C. Huang, H. Wang, H. Zhou, S. Xu and D. Ren, "EVAC-AV: The Live Road Surveillance Control Scheme Using an Effective-Vision-Area-Based Clustering Algorithm with the Adaptive Video-Streaming Technique," in *IEEE Systems Journal*, vol. 11, no. 3, pp. 1228-1238, 2017.

[107] M. Finsterbusch, C. Richter, E. Rocha, J. Muller and K. Hanssgen, "A Survey of Payload-Based Traffic Classification Approaches", in *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 1135-1156, Second Quarter 2014.

[108] C. Lu, C. Huang, Y. Lin and Y. Lai, "High performance traffic classification based on message size sequence and distribution", in *Journal of Network and Computer Applications*, vol.76, pp. 60-74, 2016.

[109] P. Amaral, J. Dinis, P. Pinto, L. Bernardo, J. Tavares and H. S. Mamede, "Machine Learning in Software Defined Networks: Data collection and traffic classification," *2016 IEEE 24th International Conference on Network Protocols (ICNP)*, Singapore, 2016, pp. 1-5.

[110] J. Yan and J. Yuan, "A Survey of Traffic Classification in Software Defined Networks," *the 1st IEEE International Conference on Hot Information-Centric Networking (HotICN)*, Shenzhen, China, 2018, pp. 200-206.

List of Abbreviation

V2I	Vehicle-to-Infrastructure
SDN	Software-Defined Networking
ITS	Intelligent Transportation System
VANET	Vehicular Ad Hoc Network
V2V	Vehicle-to-Vehicle
IEEE	Institute of Electrical and Electronics Engineers
DSRC	Dedicated Short-Range Communication
RSU	Road Side Unit
5G	Fifth-Generation cellular network
MAC	Medium Access Control
MANET	Mobile Ad hoc NETwork
NDDI	Network Development and Deployment Initiative
SBI	SouthBound Interface
NBI	NorthBound Interface
MEC	Mobile Edge Computing
PoA	Point of Attachment
HoA	Home Address
CoA	Care-of Address
HA	Home Agent
FA	Foreign Agent
TCP	Transmission Control Protocol
SCTP	Stream Control Transmission Protocol
mSTCP	Mobile SCTP
MPTCP	Multi-Path TCP
WLAN	Wireless Local Area Network
NEMO	NEtwork MObility
LTE	Long Term Evolution
MOBIC	Mobility Clustering
CDS-SVB	Set-Stable Virtual Backbone
MPBC	Mobility Prediction-Based Clustering
DBC	Density-Based Clustering
HCA	Hierarchical Clustering Algorithm
PMC	Passive Multi-hop Clustering
IETF	Internet Engineering Task Force
HIP	Host Identity Protocol
ILNP	Identifier-Locator Network Protocol
CN	Correspondent Node

MN	Mobile Node
ETSI	European Telecommunications Standard Institute
RAN	Radio Access Network

Author Biography

Ran Duo received the BE and ME degree from the School of Electronic and Information Engineering, in Beijing Jiaotong University, China in 2010 and 2017.

Since 2017, she has been working towards her Ph.D. degree at the Department of Computer and Network Engineering, Graduate School of Informatics and Engineering, The University of Electro-Communications, Japan.

Her current research interests include software defined networking, vehicular ad hoc networks, and mobile edge computing.

List of Publication

Journal Paper

[1] Ran Duo, Celimuge Wu, Tsutomu Yoshinaga, Jiefang Zhang, Yusheng Ji, "SDN-based Handover Scheme in Cellular/IEEE 802.11p Hybrid Vehicular Networks", Sensors, vol.20, no.4, article number:1082, 17 pages, Feb. 2020.

（Related to the content of Chapter 3）

International Conference

[1] Ran Duo, Celimuge Wu, Tsutomu Yoshinaga, and Yusheng Ji, "SDN-based Handover Approach in IEEE 802.11p and LTE hybrid vehicular networks", in Proceedings of 4th IEEE International Conference on Cloud and Big Data Computing (CBDCom 2018), Guangzhou, pp.1870-1875, Oct. 2018.（Related to the content of Chapter 3）

[2] Ran Duo, Celimuge Wu, Tsutomu Yoshinaga, and Yusheng Ji, "Context-aware Clustering for SDN Enabled Network", in Proceeding of 28th IEEE International Conference on Network Protocols (ICNP) Workshop on Network Protocols Riding with AI towards Mission-Critical Communications and Computing at the Edge (ICNP AIMCOM2 2020), Madrid, Spain, Oct. 2020.（Related to the content of Chapter 4）

Other Conference

[1] Ran Duo, Celimuge Wu, Tsutomu Yoshinaga, and Yusheng Ji, "SDN-based Handover Approach for High-Mobility Vehicles in a Hybrid VANET", IEICE General Conference 2019, B-11-8, March 2019.

[2] Ran Duo, Celimuge Wu, Tsutomu Yoshinaga, and Yusheng Ji, "Evaluation on performance gain of an SDN-based handover approach in IEEE 802.11p and LTE hybrid vehicular networks", IEICE Technical Report, CQ2017-86, pp.5-9, Jan. 2018.

[3] Ran Duo, Celimuge Wu, Tsutomu Yoshinaga, and Yusheng Ji, "Enabling high performance handover in IEEE 802.11p and LTE hybrid vehicular networks with SDN and MEC", IEICE Technical Report, CQ2017-53, pp.19-23, Aug. 2017.