Multi-Dimensional Resource Orchestration in Vehicular Edge Networks

by

Jiayin Chen

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2021

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Cheng Li Professor, Memorial University of Newfoundland
Supervisor(s):	Xuemin (Sherman) Shen University Professor, University of Waterloo
Internal Member:	Fakhri Karray Professor, University of Waterloo
Internal Member:	Xiaodong Lin Adjunct Associate Professor, University of Waterloo
Internal-External Member:	Wei-Chau Xie Professor, University of Waterloo

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

In the era of autonomous vehicles, the advanced technologies of connected vehicle lead to the development of driving-related applications to meet the stringent safety requirements and the infotainment applications to improve passenger experience. Newly developed vehicular applications require high-volume data transmission, accurate sensing data collection, and reliable interaction, imposing substantial constrains on vehicular networks that solely rely on cellular networks to fetch data from the Internet and on-board processors to make driving decisions. To enhance multifarious vehicular applications, Heterogeneous Vehicular Networks (HVNets) have been proposed, in which edge nodes, including base stations and roadside units, can provide network connections, resulting in significantly reduced vehicular communication cost. In addition, caching servers are equipped at the edge nodes, to further alleviate the communication load for backhaul links and reduce data downloading delay. Hence, we aim to orchestrate the multi-dimensional resources, including communication, caching, and sensing resources, in the complex and dynamic vehicular environment to enhance vehicular edge network performance. The main technical issues are: 1) to accommodate the delivery services for both location-based and popular contents, the scheme of caching contents at edge servers should be devised, considering the cooperation of caching servers at different edge nodes, the mobility of vehicles, and the differential requirements of content downloading services; 2) to support the safety message exchange and collective perception services for vehicles, communication and sensing resources are jointly allocated, the decisions of which are coupled due to the resource sharing among different services and neighboring vehicles; and 3) for interaction-intensive service provisioning, e.g., trajectory design, the forwarding resources in core networks are allocated to achieve delay-sensitive packet transmissions between vehicles and management controllers, ensuring the high-quality interactivity. In this thesis, we design the multi-dimensional resource orchestration schemes in the edge assisted HVNets to address the three technical issues.

Firstly, we design a cooperative edge caching scheme to support various vehicular content downloading services, which allows vehicles to fetch one content from multiple caching servers cooperatively. In particular, we consider two types of vehicular content requests, i.e., location-based and popular contents, with different delay requirements. Both types of contents are encoded according to fountain code and cooperatively cached at multiple servers. The proposed scheme can be optimized by finding an optimal cooperative content placement that determines the placing locations and proportions for all contents. To this end, we analyze the upper bound proportion of content caching at a single server and provide the respective theoretical analysis of transmission delay and service cost (including content caching and transmission cost) for both types of contents. We then formulate an op-

timization problem of cooperative content placement to minimize the overall transmission delay and service cost. As the problem is a multi-objective multi-dimensional multi-choice knapsack one, which is proved to be NP-hard, we devise an ant colony optimization-based scheme to solve the problem and achieve a near-optimal solution. Simulation results are provided to validate the performance of the proposed scheme, including its convergence and optimality of caching, while guaranteeing low transmission delay and service cost.

Secondly, to support the vehicular safety message transmissions, we propose a two-level adaptive resource allocation (TARA) framework. In particular, three types of safety message are considered in urban vehicular networks, i.e., the event-triggered message for urgent condition warning, the periodic message for vehicular status notification, and the message for environmental perception. Roadside units are deployed for network management, and thus messages can be transmitted through either vehicle-to-infrastructure or vehicle-to-vehicle connections. To satisfy the requirements of different message transmissions, the proposed TARA framework consists of a group-level resource reservation module and a vehicle-level resource allocation module. Particularly, the resource reservation module is designed to allocate resources to support different types of message transmission for each vehicle group at the first level, and the group is formed by a set of neighboring vehicles. To learn the implicit relation between the resource demand and message transmission requests, a supervised learning model is devised in the resource reservation module, where to obtain the training data we further propose a sequential resource allocation (SRA) scheme. Based on historical network information, the SRA scheme offline optimizes the allocation of sensing resources (i.e., choosing vehicles to provide perception data) and communication resources. With the resource reservation result for each group, the vehicle-level resource allocation module is then devised to distribute specific resources for each vehicle to satisfy the differential requirements in real time. Extensive simulation results are provided to demonstrate the effectiveness of the proposed TARA framework in terms of the high successful reception ratio and low latency for message transmissions, and the high quality of collective environmental perception.

Thirdly, we investigate forwarding resource sharing scheme to support interaction intensive services in HVNets, especially for the delay-sensitive packet transmission between vehicles and management controllers. A learning-based proactive resource sharing scheme is proposed for core communication networks, where the available forwarding resources at a switch are proactively allocated to the traffic flows in order to maximize the efficiency of resource utilization with delay satisfaction. The resource sharing scheme consists of two joint modules: estimation of resource demands and allocation of available resources. For service provisioning, resource demand of each traffic flow is estimated based on the predicted packet arrival rate. Considering the distinct features of each traffic flow, a lin-

ear regression scheme is developed for resource demand estimation, utilizing the mapping relation between traffic flow status and required resources, upon which a network switch makes decision on allocating available resources for delay satisfaction and efficient resource utilization. To learn the implicit relation between the allocated resources and delay, a multi-armed bandit learning-based resource sharing scheme is proposed, which enables fast resource sharing adjustment to traffic arrival dynamics. The proposed scheme is proved to be asymptotically approaching the optimal strategy, with polynomial time complexity. Extensive simulation results are presented to demonstrate the effectiveness of the proposed resource sharing scheme in terms of delay satisfaction, traffic adaptiveness, and resource sharing gain.

In summary, we have investigated the cooperative caching placement for content downloading services, joint communication and sensing resource allocation for safety message transmissions, and forwarding resource sharing scheme in core networks for interaction intensive services. The schemes developed in the thesis should provide practical and efficient solutions to manage the multi-dimensional resources in vehicular networks.

Acknowledgements

First, I would like to express my sincere gratitude to my supervisor, Professor Xuemin Shen, for his continuous help, support, and guidance during my Ph.D. study in University of Waterloo. Professor Shen's valuable advice and encouragement inspired me to accumulate knowledge and academic skills. The weekly meetings coordinated by Professor Shen gave me a great opportunity to broaden my knowledge and improve my presentation skills. From his comments after each meeting, I learned from his comprehensive understanding and creative thoughts of the research. His conscientious attitude to the career and every students has also influenced me, which will be the precious treasure in my future life. During the four years' Ph.D. study, I deeply feel not only the inspirational research talents but also the personal charisma from my professor. I am so grateful and honored for being one of his students.

I would also like to express my gratitude to Professor Weihua Zhuang for her great support in my Ph.D. study, especially in the Huawei SONAC project. Her insightful thoughts and valuable guidance always inspired me to do in-depth thinking on my research. The project experience also provided me a chance to do research work in a group, developing my communication skills and teamwork spirit.

I would like to thank Professor Cheng Li, Professor Fakhri Karray, Professor Xiaodong Lin, Professor Wei-Chau Xie for serving my thesis examination committee. Their valuable suggestions, comments, and questions have helped me to improve the quality of my thesis.

In the past four years, I have also gained lots of help and precious friendship from my current and former colleagues in our BBCR group. I would like to express my sincere appreciation to all the BBCR members and my friends for their kind help, especially Dr. Nan Cheng, Dr. Jianbing Ni, Dr. Haibo Zhou, Dr. Ning Zhang, Dr. Shan Zhang, Dr. Wenchao Xu, Dr. Qiang Ye, Dr. Feng Lyu, Dr. Peng Yang, Dr. Wei Quan, Dr. Junling Li, Dr. Weisen Shi, Dr. Omar Alhussein, Dr. Phu Thinh Do, Dr. Kaige Qu, Si Yan, Dr. Haixia Peng, Huaqing Wu, Conghao Zhou, Mushu Li, Dr. Nan Chen, Dr. Dongxiao Liu, Dr. Wen Wu, Dr. Qihao Li, Dr. Dairu Han, Liang Xue, Yingying Pei, Amr Salah Matar, and many others.

Finally, I would like to greatly thank my parents for their deep love, endless support, and understanding during my Ph.D. study.

Jiayin Chen
April 12, 2021
Waterloo, Ontario, Canada

Dedication

This thesis is dedicated to my beloved parents, Xuezhi Mao and Wei Chen.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

5G	Fifth Generation
ACO	Ant Colony Optimization
AP	Access Point
AQM	Active Queue Management
BP	Back Propagation
CAM	Cooperative Awareness Message
CP	Collective Perception
CPM	Collective Perception Message
CV	Connected Vehicle
DENM	Decentralized Environmental Notification Message
DL	Downlink
DRR	Deficit Round Robin
E2E	End-to-End
EDD	Earliest Due Date
ETSI	European Telecommunications Standard Institute
HVNet	Heterogeneous Vehicular Network
LoS	Line-of-Sight
MAB	Multi-Armed Bandit
MBS	Macro Base Station
MMKP	Multi-dimensional Multi-choice Knapsack Problem
NLoS	Non-Line-of-Sight
PDF	Probability Distribution Function
QoS	Quality-of-Service
RB	Resource Block
RSU	Roadside Unit

SBS	Small Base Station
SCV	Sensing-Enabled Connected Vehicle
SDN	Software-Defined Networking
SRA	Sequential Adaptive Resource Allocation
TARA	Two-Level Adaptive Resource Allocation
TDMA	Time Division Multiple Address
TDS	Transmitting Dominant Sub-frame
UCB	Upper Confidence Bound
UCV	Connected Vehicle without Sensing Capability
UL	Uplink
V2B	Vehicle-to-MBS
V2I	Vehicle-to-Infrastructure
V2R	Vehicle-to-RSU
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
WFQ	Weighted Fair Queuing
WTP	Wait Time Priority

List of Publications

Journal Papers

1. **J. Chen**, P. Yang, Q. Ye, W. Zhuang, X. Shen, and X. Li, "Learning-Based Proactive Resource Allocation for Delay-Sensitive Packet Transmission," ***IEEE Transactions on Cognitive Communications and Networking*(TCCN)**, accepted, Sept. 2020.

2. **J. Chen**, H. Wu, P. Yang, F. Lyu, and X. Shen, "Cooperative Edge Caching with Location-based and Popular Contents for Vehicular Networks," ***IEEE Transactions on Vehicular Technology*(TVT)**, vol. 69, no. 9, pp. 10291 - 10305, Sept. 2020.

3. **J. Chen**, Q. Ye, W. Quan, S. Yan, P. T. Do, P. Yang, W. Zhuang, X. Shen, X. Li, and J. Rao, "SDATP: An SDN-Based Traffic-Adaptive and Service-Oriented Transmission Protocol," ***IEEE Transactions on Cognitive Communications and Networking*(TCCN)**, vol. 6, no. 2, pp. 756 - 770, Jun. 2020.

4. **J. Chen**, Q. Ye, W. Quan, S. Yan, P.T. Do, W. Zhuang, X. Shen, X. Li, and J. Rao, "SDATP: An SDN-Based Adaptive Transmission Protocol for Time-Critical Services," ***IEEE Network Magazine***, vol. 34, no. 3, pp. 154 - 162, Jun. 2020.

5. H. Wu, **J. Chen**, C. Zhou, W. Shi, N. Cheng, W. Xu, W. Zhuang, and X. Shen, "Resource Management in Space-Air-Ground Integrated Vehicular Networks: SDN Control and AI Algorithm Design," ***IEEE Wireless Communications***, vol. 27, no. 6, pp. 52 - 60, Dec. 2020.

6. H. Wu, **J. Chen**, W. Xu, N. Cheng, W. Shi, L. Wang, and X. Shen, "Delay-Minimized Edge Caching in Heterogeneous Vehicular Networks: A Matching-Based Approach," ***IEEE Transactions on Wireless Communications*(TWC)**, vol. 19, no. 10, pp. 6409 - 6424, Oct. 2020.

7. S. Zhang, **J. Chen**, F. Lyu, N. Cheng, W. Shi, and X. Shen, "Vehicular Communication Networks in Automated Driving Era," ***IEEE Communications Magazine***, vol. 56, no. 9, pp. 26-33, Sept. 2018.

8. H. Wu, F. Lyu, C. Zhou, **J. Chen**, L. Wang, and X. Shen, "Optimal UAV Caching and Trajectory in Aerial-Assisted Vehicular Networks: A Learning-Based Approach," ***IEEE Journal on Selected Areas in Communications*** **(JSAC)**, vol. 38, no. 12, pp. 2783 - 2797, Dec. 2020.

9. N. Cheng, F. Lyu, **J. Chen**, W. Xu, H. Zhou, S. Zhang, and X. Shen, "Big Data Driven Vehicular Networks," ***IEEE Network Magazine***, vol. 32, no. 6, pp. 160-167, Dec. 2018.

10. W. Xu, H. Zhou, N. Cheng, F. Lyu, W. Shi, **J. Chen**, and X. Shen, "Internet of Vehicles in Big Data Era," ***IEEE/CAA Journal of Automatica Sinica***, vol. 5, no. 1, pp. 19-35, Jan. 2018.

Conference Papers

1. **J. Chen**, H. Wu, F. Lyu, P. Yang, and X. Shen, "Multi-Dimensional Resource Allocation for Diverse Safety Message Transmissions in Vehicular Networks," ***IEEE ICC'21 — IEEE International Conference on Communications***, Canada, June 2021.

2. **J. Chen**, S. Yan, Q. Ye, W. Quan, P. T. Do, W. Zhuang, X. Shen, X. Li, and J. Rao, "An SDN-Based Transmission Protocol with In-Path Packet Caching and Retransmission," ***IEEE ICC'19 — IEEE International Conference on Communications***, China, May 2019.

3. **J. Chen**, W. Xu, N. Cheng, H. Wu, S. Zhang, and X. Shen, "Reinforcement Learning Policy for Adaptive Edge Caching in Heterogeneous Vehicular Network," ***IEEE GLOBECOM'18 — IEEE Global Communications Conference***, UAE, Dec. 2018.

4. H. Wu, **J. Chen**, C. Zhou, F. Lyu, N. Zhang, L. Wang, and X. Shen, "Load- and Mobility-Aware Cooperative Content Delivery in SAG Integrated Vehicular Networks," ***IEEE ICC'21 — IEEE International Conference on Communications***, Canada, June 2021.

5. H. Wu, **J. Chen**, F. Lyu, L. Wang, and X. Shen, "Joint Caching and Trajectory Design for Cache-Enabled UAV in Vehicular Networks," ***WCSP'19 — Wireless Communications and Signal Processing***, China, Oct. 2019.

6. W. Xu, **J. Chen**, H. Wu, W. Shi, H. Zhou, N. Cheng, and X. Shen, "ViFi: Vehicle-to-Vehicle Assisted Traffic Offloading via Roadside WiFi Networks," ***IEEE GLOBECOM'18 — IEEE Global Communications Conference***, UAE, Dec. 2018.

7. H. Wu, W. Xu, **J. Chen**, L. Wang, and X. Shen, "Matching-based Content Caching in Heterogeneous Vehicular Networks," ***IEEE GLOBECOM'18 — IEEE Global Communications Conference***, UAE, Dec. 2018.

Chapter 1

Introduction

1.1 Overview of Vehicular Networks

Recent technical development of vehicular applications has led to the age of automated vehicles. According to new survey results by the World Economic Forum, not only consumers are willing to travel in automated vehicles, nearly 60% of 5,500 respondents in ten countries around the world, but also most city authorities are looking forward to automated [1]. They believe that the public can benefit from automated, applications such as vehicle sharing will be a potential last-mile solution of daily urban transit. Thus, automated vehicles should be introduced by city planners and governments toward smart mobility cities such as Gothenburg and Singapore.

It is challenging to meet the stringent requirements of automated vehicles solely based on the on-board sensors. A large amount of external information exchange is also necessary[2]. If an emergency event is detected by a roadside infrastructure or another automated vehicle, e.g., a vulnerable road user or an accident, the automated vehicle should be notified of this event. To support automated, high-quality maps are necessary, including high definition and fine-grained contextual information, that can be collected and distributed by vehicles or connected infrastructures, such as cellular base station (BS) and roadside unit (RSU). In the case of that, the required information is unavailable locally, the vehicle can fetch this information from remote information server through access to connected infrastructures.

In addition to automated, passengers also have strong demand on vehicular applications based on connected vehicles [3]. Telematics applications and Vehicle-to-Vehicle (V2V)/Vehicle-to-Infrastructure (V2I) applications are developed to accommodate this need. Telematics applications combine telecommunications and informatics together to enable information exchange between vehicles and the world. Most of the popular applications are related to driving safety, such as automatic collision notification and remote diagnostic, where the data are collected locally to inform remotely-run algorithms. In terms of V2V/V2I applications, a broad group of applications are included, ranging from safety related vehicle tracking to driver experience enhancement, which are operated upon network consisting of connected vehicles and infrastructures. These applications are divided into four types, information services, safety services, individual motion control, and group motion control [3]. Various requirements are raised by these applications, such as low data processing and transmission latency for safety services and broader connectivity for group motion control.

In order to make automated and well designed connected vehicles applications possible, rapid data transmission and processing need to be guaranteed even for the "big data", which are not only of huge amount, but also of enormous types. Traditionally, vehicles are using cellular networks to fetch data from Internet and cloud computing server. However, through cellular network, the prohibitive cost and network capacity constraint of delivering such massive data have driven the exploration of other network spectrum (e.g., WiFi) to provide alternative V2I communication data pipes. In addition, the long transmission distance and unpredicted delay motivate the paradigm shift from mobile cloud computing to mobile edge computing.

1.2 Edge Assisted Vehicular Networks

1.2.1 Introduction of Edge Assisted Vehicular Networks

To overcome the "big data" challenge of vehicular network, infrastructures are introduced to enhance quality of service (QoS) for vehicles. Firstly, infrastructures can be used as relay nodes for connections between vehicles, which significantly improves the communication quality. Data transmission capacity is demonstrated to be improved with the increase of infrastructure network. In particular, RSUs can greatly reduce the data transmission time in sparse scenarios, such as highway vehicular networking [4]. One motivation of using RSUs as relay nodes is that it is easier for vehicles to receive/send packets from/to a stationary node, and therefore RSUs are proposed to act as gateway in vehicular networks [5].

Infrastructures are also used for providing Internet access to vehicles in [6], which is possible because there are only a few hops between vehicles and the infrastructure (e.g., WiFi, cellular). Instead of relying on single access network, Zheng *et al.* elaborated both the wide coverage and real-time services, which can be guaranteed via cooperation among the heterogeneous wireless access networks including V2V and V2I communications [7]. It is proved that in Heterogeneous Vehicular Networks (HVNets), the communication cost of vehicles can be greatly reduced while the network throughput can be improved [8].

In terms of selection of infrastructure, a widely adopted solution is to employ publicly available infrastructure, such as WiFi Access Points (APs) and cellular BSs. Traditionally, vehicles get connected to Internet via BSs. Due to the prohibitive cost and constrained capacity of cellular network, WiFi APs are utilized to provide additional V2I communication. It is measured that a roadside WiFi AP can provide significant UDP and TCP throughput for drive-through vehicles [9]. Cheng *et al.* utilized the opportunistic connection between vehicles and roadside hotspots to offload part of the vehicular traffic from cellular network to WiFi networks and showed that the communication cost can be greatly reduced [10]. For fast-moving vehicles, to meet the requirements of safe driving environment, an adaptive dynamic scheduling policy is designed for RSUs [11]. In addition, the practical feasibility is also verified, for instance, WiFi signals over the city are measured in [12] and the handover between different access networks (e.g., 4G cellular and WiFi) are studied in [13].

1.2.2 Testbeds and Practical Deployments

Some testbeds and practical deployments of edge assisted vehicular network are established around the world. The first smart highway in Europe is the Cooperative Intelligent Transport System Corridor, which enables communications between vehicles and infrastructures [14]. It introduces two applications, i.e., roadworks warning and improved traffic management. Roadworks warning aims at improving local road safety by applying a positioning/communication system on roadworks safety trailers. In contrast, traffic management service requires cooperation between RSUs and cellular networks, in which vehicles send information to RSU, and after preprocessing the data, it will be transmitted to highway operators through cellular network. Road safety and traffic management are also studied and tested in Japan. Measurements show that infrastructures can effectively reduce path vulnerability in critical areas [15]. In the US, Connected Vehicle Safety Pilot Model Deployment was launched in 2012, where both V2V and V2I connections are enabled. Based on the project, researchers can test connected vehicle operations for applications such as traffic efficiency, energy efficiency, and environmental benefits [16].

Recently, some new use cases and applications are introduced to test advanced Cellular-based Vehicle-to-Everything (V2X), including safety, traffic efficiency as well as comfort for the driver, where V2X communication is the transmission of information exchange between a vehicle and any entity that may affect the vehicle. The applications rely on low latency and high throughput. For instance, “see through” between two connected vehicles on road requires high-resolution visual data transmission [17]. In addition to connected vehicular applications, automated vehicles have been tested as well. A project called Connected Intelligent Transport Environment creates an environment, where various V2X technologies are applied on urban roads, dual-carriageways, and motorways. The project proves that these technologies can improve the on-road experience, reduce traffic congestion, and provide entertainment and safety services through better connectivity [18].

Furthermore, new service requirements are specified to promote the development of vehicular applications. The 3GPP standard group defines service requirements for Cellular-V2X in three areas: nonsafety V2X services, safety-related V2X services, and support for V2X services in multiple 3GPP radio access technologies [19]. To accommodate these applications, new levels of connectivity and intelligence are required. Since improved network performance of throughput, latency, reliability, connectivity, and mobility will be provided by fifth generation (5G), it becomes a potential access solution of vehicle networks.

1.3 Motivations and Contributions

1.3.1 Challenges of Edge-Assisted Vehicular Networks

Newly developed vehicular applications (e.g., augmented reality (AR), automated, HD map, trajectory management, etc.) require high-volume data transmission and reliable interactions among vehicles and traffic management controllers, which can only be achieved with the high-level capability of data communication and storage. Since conventional vehicular networks have difficulties in satisfying these requirements, edge nodes with communication and storage capabilities are introduced to assist vehicular networks. Connected infrastructures including BSs and RSUs can provide stable Internet access for vehicles. They are also equipped with edge caching capabilities to support data-intensive and interaction-intensive applications.

Although edge-assisted vehicular networks are introduced to support vehicular content downloading services (e.g., HD map and entertainment content downloading), safety message transmissions, and interaction-intensive applications (e.g., trajectory management), the service provisioning still faces following challenges:

1. *Large backhaul link overhead* – Considering the data-intensive applications, it requires high-volume data transmission, imposing substantial pressure on backhaul links if vehicles download data from the Internet;

2. *High mobility of vehicles* – Due to the high mobility of vehicles, both the connection among vehicles and the connection between vehicles and edge nodes are intermittent. For data-intensive applications, the vehicle needs to download content through varying access infrastructures. To support the safety message transmissions among vehicles, the reliable message exchange is required, which is challenging due to intermittent connections;

3. *Congestion in core networks* – For the interaction-intensive applications (i.e., interactions between vehicles and management controllers are required), packet transmissions with low latency and high reliability requirements are essential to ensure interactivity. To meet these requirements, the utilization of buffer spaces at each network switch needs to be properly controlled. Specifically, the dominant contributing factor for end-to-end (E2E) packet transmission delay is the delay for packet queuing at network switches, and the packet loss is mainly caused by buffer overflow during network congestion.

1.3.2 Approaches and Contributions

To overcome the challenges faced by vehicular service provisioning, we have investigated the cooperative caching placement for content downloading services, joint communication and sensing resource allocation for safety message transmissions, and forwarding resource sharing scheme in core networks for interaction-intensive applications. In specific, we focus on the following three research topics and address each topic in one chapter.

1. In chapter 3, a cooperative edge caching scheme is developed to support various vehicular content downloading services, which allows vehicles to fetch one content from multiple caching servers cooperatively. In specific, we consider two types of vehicular content requests, i.e., location-based and popular contents, with different delay requirements. Both types of contents are encoded according to fountain code and cooperatively cached at multiple servers. The proposed scheme can be optimized by finding an optimal cooperative content placement that determines the placing locations and proportions for all contents. To this end, we first analyze the upper bound proportion of content caching at a single server and provide the respective theoretical analysis of transmission delay and service cost (including content caching and transmission cost) for both types of contents. We then formulate an optimization problem of cooperative content placement to minimize the overall transmission delay and service cost. As the problem is a multi-objective multi-dimensional multi-choice knapsack problem, which is proved to be NP-hard, we devise an ant colony

optimization-based scheme to solve the problem and achieve a near-optimal solution. Extensive simulation results validate the performance of the proposed scheme, including its convergence and optimality of caching, while guaranteeing low transmission delay and service cost;

2. In chapter 4, a two-level adaptive resource allocation (TARA) framework is proposed to support vehicular safety message transmissions. In particular, three types of safety message are considered in urban vehicular networks, i.e., the event-triggered message for urgent condition warning, the periodic message for vehicular status notification, and the message for environmental perception. Roadside units are deployed for network management, and thus messages can be transmitted through either vehicle-to-infrastructure or vehicle-to-vehicle connections. To satisfy the requirements of different message transmissions, the proposed TARA framework consists of a group-level resource reservation module and a vehicle-level resource allocation module. Particularly, the resource reservation module is designed to allocate resources to support different types of message transmission for each vehicle group at the first level, and the group is formed by a set of neighboring vehicles. To learn the implicit relation between the resource demand and message transmission requests, a supervised learning model is devised in the resource reservation module, where to obtain the training data we further propose a sequential resource allocation (SRA) scheme. Based on historical network information, the SRA scheme offline optimizes the allocation of sensing resources (i.e., choosing vehicles to provide perception data) and communication resources. With the resource reservation result for each group, the vehicle-level resource allocation module is then devised to distribute specific resources for each vehicle to satisfy the differential requirements in real time. Extensive simulation results are provided to demonstrate the effectiveness of the proposed TARA framework in terms of the high successful reception ratio and low latency for message transmissions, and the high quality of collective environmental perception;

3. In chapter 5, a forwarding resource sharing scheme is devised to support interaction-intensive applications in HVNets, especially for delay-sensitive packet transmissions between vehicles and management controllers. A learning-based proactive resource sharing scheme is proposed for the core communication networks, where the available forwarding resources at a switch are proactively allocated to the traffic flows in order to maximize the efficiency of resource utilization with delay satisfaction. The resource sharing scheme consists of two joint modules, estimation of resource demands and allocation of available resources. For service provisioning, resource demand of each traffic flow is estimated based on the predicted packet arrival rate. Considering the distinct features of each traffic flow, a linear regression scheme is developed

for resource demand estimation, utilizing the mapping relation between traffic flow status and required resources, upon which a network switch makes decision on allocating available resources for delay satisfaction and efficient resource utilization. To learn the implicit relation between the allocated resources and delay, a multi-armed bandit learning-based resource allocation scheme is proposed, which enables fast resource allocation adjustment to traffic arrival dynamics. The proposed scheme is proved to be asymptotically approaching the optimal strategy, with polynomial time complexity. Extensive simulation results demonstrate the effectiveness of the proposed resource sharing scheme in terms of delay satisfaction, traffic adaptiveness, and resource allocation gain.

1.4 Thesis Outline

The remainder of the thesis is organized as follows: In Chapter 2, we present a comprehensive review of related works in terms of edge caching technology, resource allocation in HVNets, and resource sharing scheme in core networks. In Chapter 3, a cooperative edge caching scheme is proposed to support various vehicular content downloading services, in which vehicles are allowed to fetch one content from multiple caching servers cooperatively. In Chapter 4, we propose a two-level adaptive resource allocation framework to support the vehicular safety message transmissions, considering three types of safety messages. Chapter 5 presents a learning-based proactive resource sharing scheme for the core communication networks, aiming at supporting interaction-intensive services in HVNets. We conclude the thesis and give future research directions in Chapter 6.

Chapter 2

Literature Review

This chapter aims to present a review of related works, including edge caching technology, resource allocation in HVNets, and resource sharing scheme in core networks.

2.1 Edge Caching for HVNets

2.1.1 Cooperative Caching for Vehicular Network

Yao *et al.* predicted the probability of vehicles arriving at different hotspots, and selected the vehicles with longer sojourn time as caching nodes [20]. Then, the decision of content replacement is made, considering the file popularity. Zhao *et al.* proposed to cache content at RSUs instead of on the vehicles, and a mobility prediction based content prefetching mechanism was proposed to improve content hit rate and reduce content delivery latency [21]. The frequent disconnection between vehicles and edge nodes makes it difficult to download the complete file during one connection. Thus, coded caching strategies (e.g., maximum distance separable code and fountain code [22, 23]) are widely used, since they can improve the delivery reliability/flexibility and cache utilization by dividing the files into small fragments.

2.1.2 Cross-Tier Cooperation in Edge Caching

The cooperative caching in vehicular networks mainly focuses on the cooperation between vehicular cloud and RSU cloud [24], while the cooperative caching in mobile user networks takes advantage of multi-tier cellular networks (i.e., macro-cell and small-cell) [25, 26]. The caching strategy was proposed by leveraging the cross-tier cooperation, including the vertical cooperation between a macro base station (MBS) and a small base station (SBS), and the horizontal cooperation between SBSs [27]. Then, the mobile user can download its requested content from one or several BSs that have cached the content. To reduce the content provisioning cost and delivery delay, a cooperative caching strategy was designed, which considers content placement at the centralized MBSs and distributed SBSs [28].

2.1.3 Discussion

In spite of the aforementioned works, the following issues, which are essential for caching placement design in vehicular networks, are insufficiently studied in literatures.

1. Most existing works on edge caching in HVNets consider the cooperation between edge caching servers (e.g., RSUs or BSs) and vehicular caching nodes, instead of the cross-tier cooperation between multi-tier servers (e.g., between RSUs and BSs). The cooperation between MBSs and SBSs has been widely investigated for mobile users in low-speed scenarios, in which intermittent connection rarely happens within the course of downloading one content. However, for vehicular users, the high mobility poses challenges to the cooperation, considering the frequent handover between different caching servers. Thus, to guarantee delay requirements and save costs for vehicular content delivery service, we focus on placing contents on cross-tier caching servers cooperatively;

2. In existing works, caching schemes are designed to support the content delivery service with a single QoS metric, e.g., the delivery deadline or the downloading rate. In reality, the QoS metrics for downloading different types of contents are diverse, e.g., the low latency required by safety-related vehicular applications and the high throughput required by entertainment services. Hence, different QoS metrics for different content delivery services are considered in our work.

2.2 Resource Allocation for Vehicular Safety Message Dissemination

2.2.1 Reliable Safety Message Dissemination

To support the periodic safety message dissemination, a mobility-aware time division multiple address (TDMA) MAC protocol was proposed in [29] to avoid slot-assignment collisions, where vehicles on different lanes of road segments are assigned with disjoint time slot sets. With the assigned slot set, each vehicle selects its own slot in a distributed way, and the selection is updated if slot collision or lane change happens. Furthermore, a novel capture-aware MAC protocol was developed to improve the channel resource utilization efficiency by setting the optimal frame length [30]. In addition to periodic messages, such as the beacon message and the map data message [31], event-triggered safety messages are generated and transmitted by vehicles as well, which is essential for cooperative driving among vehicles. In [32], the beacon rate was optimized to meet the reliability requirements of event-triggered safety messages and maintain the accuracy of neighborhood information collected by beacons. Considering the urban intersection scenario, infrastructures (e.g., the RSUs) deployed at intersections have been widely leveraged to enhance the packet transmission [33, 34], where V2I connections are implemented to complement V2V connections among vehicles.

2.2.2 Adaptive Resource Allocation for Vehicular Networks

To support diversified vehicular applications, slicing schemes are developed to reserve resources for various services [35, 36]. In [37], a service-dependent resource slicing scheme was developed to satisfy the differentiated requirements of content downloading services, through making decisions on the RSU transmission rate and content caching. In [38], resource slicing was adopted to establish slices for two sets of applications, the rate constrained and the delay constrained applications, and resources allocated to each individual slice were adapted to the user population. An online network slicing strategy was developed in [39], aiming to maximize the long-term time-averaged system capacity while satisfying the strict requirements of vehicle communication links.

To further allocate the required resources to each vehicle, C-V2X mode-3 resource scheduling schemes are investigated for V2V connections [40]. In [41], vehicles were grouped into non-overlapping clusters, then the mode-3 resource scheduling was designed for each cluster to perform sub-channel assignment. Not only the prevention of allocation conflicts

Table 2.1: Comparison for three types of safety messages.

Message type	**Generation interval**	**Priority**	**Requirement**	**Transmission mode**
DENM	Event-triggered	High	High reliability, low latency	V2V / V2I
CAM	Periodic (fixed CAM period, e.g., 100 ms)	Medium	High reliability, low latency	V2V / V2I
CPM	Periodic (adjustable CPM period)	Low	Sensing coverage and accuracy	V2I

but also the fulfillment of QoS should be considered when allocating sub-channels to the vehicles [42]. To deal with the NLOS issue, an RSU relaying system named relay assisted enhanced V2V was proposed in [43], where the RSU and vehicular users operate on the same frequency band during the assigned separate time slots. Specifically, the distributed decisions on resource scheduling and the V2I/V2V resource separate ratio are made by vehicles and the RSU, respectively.

2.2.3 Discussion

In spite of the aforementioned works related to resource allocation in vehicular networks, the following issues are insufficiently studied.

1. In existing works, vehicular services are divided into different types, e.g., safety-related and non-safety services. However, the safety-related services can present different traffic patterns and service requirements, as given in Table 2.1 [44–46], which is rarely considered;

2. Resource allocation schemes with different levels, which were studied separately in most existing works, are jointly investigated in this work. Through joint optimization of the group-level resource reservation and the vehicle-level resource allocation, the overall performance can be further improved. For instance, the vehicle-level resource allocation can provide feedback about resource utilization and QoS satisfaction for potential resource reservation adjustment in the future;

3. Safety messages are transmitted via V2V or V2I connections on the same frequency band, which calls for a resource allocation scheme design considering V2I and V2V connections simultaneously. However, most of the existing resource reservation/slicing

schemes only consider V2I links, while most of the vehicle-level resource allocation schemes are developed for V2V links. To close the gap, the TARA framework is proposed in this chapter, considering both V2V and V2I transmissions and specific performance metrics of multifarious types of message.

2.3 Forwarding Resource Sharing for Delay-Sensitive Packet Transmissions

To support interaction-intensive vehicular applications, packet transmission delay in core networks should be minimized, in order to meet the delay requirement. The delivery delay of each service flow in core networks is determined by the routing path and the allocated forwarding resources at each switch on route. To satisfy the decomposed per-hop delay requirement with efficient resource utilization, each switch makes decision on resource sharing among traffic flows. Delay requirements for per-hop packet transmission are taken into account by wait time priority (WTP) [47] and earliest due date (EDD) [48] algorithms. In using the algorithms, the switch makes decision each time that a packet is forwarded. Considering the high forwarding rate of core network switches, flow-level resource sharing schemes with lower computational complexity are investigated for fairness and delay requirements, such as weighted fair queuing (WFQ) [49] and deficit round robin (DRR) [50, 51]. To adapt to varying traffic and network conditions, dynamic WFQ updates the allocation of resources at the beginning of each resource sharing interval [52], where the resource sharing decisions are made based on the estimation of average packet queuing delay in the upcoming interval. For supporting applications with stringent delay requirements, resource sharing is expected to be conducted for delay satisfaction of individual packets. Furthermore, if the resource sharing is conducted in line with packet arrival instants, the QoS can be degraded due to burst of traffic in the upcoming resource sharing interval. Hence, a proactive resource sharing and packet scheduling solution potentially can help timely QoS enhancement based on traffic prediction [53, 54].

To support delay-sensitive applications in a dynamic networking environment, learning-based scheduling algorithms are investigated [55, 56]. To be adaptive to traffic dynamics, the resource scheduler applies machine learning techniques to categorize traffic flows with different characteristics [57]. Based on instantaneous system states (e.g., number of arrived packets and resource occupancy), different decisions on allocating resources are made by the learning module, such as reinforcement learning (RL) [58], deep Q network (DQN) [59], and stacked auto-encoder deep learning [60], through which the implicit relation between the allocated resources and the achieved QoS satisfaction can be learned.

2.3.1 Discussion

In spite of the aforementioned resource sharing schemes developed for core networks, the following issues are insufficiently studied.

1. In existing flow-level resource sharing schemes, the switch estimates the average queuing delay for each flow, based on the packet arrival rate and queue length state. Taking into account the delay requirements of different flows, the decisions on allocating resources are made to guarantee the average delay requirements. However, for real-time applications, the packet will be dropped if its delay is larger than the required threshold. Thus, the allocation of resources should be determined based on the delay for each packet rather than the average delay;

2. According to existing resource sharing schemes, forwarding resources are always fully allocated to the traversing flows by the switch. It may lead to low resource utilization efficiency, especially when available resources at the switch are over-provisioning. Hence, to improve resource utilization efficiency, we allocate the resources to each flow that matching its resource demand, i.e., the resources required by one flow to satisfy its delay requirement.

Chapter 3

Cooperative Edge Caching for Location-based and Popular Content Delivery

In this chapter, we propose a cooperative edge caching scheme, which allows vehicles to fetch one content from multiple caching servers cooperatively. In particular, we consider two typical types of vehicular content requests, i.e., location-based and popular contents, with different delay requirements. The proposed scheme can be optimized by finding an optimal cooperative content placement that determines the placing locations and proportions for all contents. To this end, we first analyze the upper bound proportion of content caching at a single server. Then, the respective theoretical analysis of transmission delay and the service cost (including content caching and transmission cost) are provided. Based on the theoretical analysis, we then formulate an optimization problem of cooperative content placement to minimize the overall transmission delay and service cost. Since the problem is a multi-objective multi-dimensional multi-choice knapsack one, which is proved to be NP-hard, we finally devise an ant colony optimization-based scheme to solve the problem and achieve a near-optimal solution. Simulation results validate the performance of the proposed scheme, including its convergence and optimality of caching, where low transmission delay and service cost are guaranteed.

3.1 Background and Motivations

Recent development of vehicular technologies has led to the era of autonomous vehicles, and both consumers and government authorities are looking forward to autonomous vehicles. Yet, it is challenging to meet the stringent requirements of autonomous vehicles solely based on the on-board sensors in hazardous conditions, e.g., collision avoidance with poor visibility [1]. Information exchange with external entities is also necessary, such as hazard broadcasting and high-quality maps downloading [2]. In addition to driving-related applications, passengers also have strong demands for mobile applications in connected vehicles [61]. Newly developed vehicular applications (e.g., in-car entertainment and mobile advertising) require high-volume data transmission, imposing substantial pressure on vehicular networks that solely rely on cellular networks to fetch data from the remote server. To improve the network capacity, HVNets have been proposed, in which both BSs and RSUs can provide network connections, resulting in significantly reduced vehicular communication cost [5].

Edge caching is proposed to reduce both the backhaul traffic and the transmission time for high-volume data delivery. To facilitate content delivery, it is essential to develop a content placement scheme for edge caching servers [62], considering the intermittent connection of moving vehicles. In particular, to satisfy the service delay requirements of the vehicles driving through different edge nodes (e.g., RSUs), edge cooperation has been introduced and the cooperative content placing scheme has been proposed [63, 64]. However, as edge resources are constrained, to adapt to the high mobility of vehicles, caching on both vehicles and RSUs should be considered [65], [24]. Both the cooperation among RSUs and the cooperation between RSUs and vehicles have been investigated. However, as vehicular connections are intermittent due to the limited coverage range of edge servers, content downloading time can be unacceptable. Within the interlaced coverage of multi-tier edge caching servers (e.g., BSs and RSUs) in HVNets, the cross-tier cooperation among different servers can provide seamless connections to facilitate content downloading. Meanwhile, within the overlapping coverage, differential communication features and caching capacities of multiple servers are considered in content caching. Hence, compared with cooperation between RSUs, a more fine-grained content placement scheme is required, e.g., determining the proportions of cached contents on multi-tier edge servers. In addition, vehicle's high mobility renders the design further intractable, as it constrains the vehicular connection duration. A precise vehicle mobility model is thus required for the connection duration analysis.

Besides, most existing works consider vehicular downloading applications with a single QoS metric, such as the delivery deadline or the downloading rate. However, the QoS metrics for different applications are diverse [66]. For instance, the safety-related vehicular applications call for low latency, while entertainment services desire high throughput. Therefore, different QoS metrics for two content services need to be considered: 1) the location-based contents (e.g., HD map downloaded for driving assistance applications), which require a stringent downloading time, and 2) popular contents (e.g., multimedia data for infotainment applications), which require to be delivered before a less stringent service deadline. Moreover, through placing the contents on edge servers equipped with low-cost wireless resources (e.g., WiFi), the cost of data transmission can be saved as well. Due to the differential QoS metrics, the objective functions for diverse services should be derived separately, and both cost and QoS should be considered in the content placement problem formulation. This lead to the high complexity of problem formulation and difficulty of multi-objective solution.

In this chapter, based on the HVNets with multi-tier edge caching servers (i.e., BS and RSU), a cooperative edge caching scheme is proposed to accommodate the delivery services for both location-based and popular contents. In particular, one content can be cached at multiple servers cooperatively. Considering the cross-tier cooperation and vehicle mobility, the duration of vehicles driving through the coverage of RSU and BS is first analyzed, which determines the upper bound of vehicular downloading time from a single server. To meet differentiated QoS requirements, the content placement schemes of the two services are designed respectively. For a location-based content requiring minimal downloading time, the delay and service cost for each possible content placement scheme are derived. For popular content delivery with a service deadline, we analyze the delay guaranteed minimum content placing requirement for all possible caching modes, i.e., placing one content at both BS and RSU, only at BS, or only at RSU. Accordingly, the service cost is also derived for each mode.

Based on the theoretical analysis, a cooperative content placement problem is then formulated to jointly minimize the transmission delay of the location-based contents and the service cost of both types of contents. The formulated problem turns out to be an NP-hard multi-objective multi-dimensional multi-choice knapsack problem (MMKP), and we propose an ant colony optimization (ACO) based scheme to solve it. Then, the fine-grained cooperative content placement is obtained, including the caching proportions of each content at different caching servers, which can achieve a near-optimal performance in terms of delay and service cost.

The contributions in this chapter are as follows:

1. *Cooperative edge caching scheme* – We propose a cooperative caching scheme, in which a vehicle in HVNet can fetch a content from multiple edge servers while driving

along the road. This cooperation among edge servers can not only reduce transmission delay through seamless content delivery, but also save service cost by caching contents in low-cost servers.

2. *Theoretical analysis of delay and cost* – We analyze and derive the content transmission delay for two types of contents. For location-based contents, given a content placement scheme, we derive the delay under the vehicle mobility and communication models, which can be optimized by determining an optimal content placement from the available set. For popular contents with a specified delay requirement, we analyze the delay guaranteed minimum content placing requirement for all possible caching modes. Thus, the placement scheme of each popular content can be optimized by selecting an optimal caching mode. Likewise, the service cost is also derived.

3. *Cooperative content placement scheme design* – Based on the delay and cost analysis, we formulate an optimization problem of cooperative content placement to jointly minimize the delay and the cost, which turns out to be a multi-objective MMKP. To solve the optimization problem with low complexity, we devise an ACO-based scheme, which can achieve a fine-grained cooperative content placement with near-optimal performance.

The remainder of this chapter is organized as follows. We describe the system model in Section 3.2. The delay and cost of cooperative content caching are theoretically analyzed in Section 3.3, followed by the cooperative content placement problem formulation in Section 3.4. In Section 3.5, we devise the ACO-based scheme to solve the optimization problem. Simulation results are presented in Section 3.6 to demonstrate the performance of the proposed scheme. Finally, conclusions are drawn in Section 3.7. Important mathematical symbols of this chapter are listed in Table 3.1.

Table 3.1: Summary of mathematical symbols.

Symbols	**Definition**
a	Amount of speed variation during each speed update interval
f	File index
l	Size of a coded content packet
n	Possible content placement scheme index
q_0	Exploitation probability of caching mode selection in Algorithm 2
s	Skewness parameter of Zipf-like popularity distribution for contents
s_B^f ($s_{R_w}^f$)	Number of precached encoded packets of file f at BS (RSU W_w)

t_B^f ($t_{R_w}^f$, t_{BL}^f)	Transmission delay for each packet of file f from BS (RSU W_w, the remote server)
$v_i(t)$	Speed of VU_i at time t
$x_{f,n}$	Binary variable representing the selection of n-th possible content placement scheme for file f
$B_i(t)$	Allocated BS bandwidth for VU_i at time t
D_B (D_R)	Communication ranges of BS (RSU)
$\overline{D}^f(n)$ ($\overline{C}^f(n)$, $T_{R_w}^f(n)$)	Required service delay (cost, access resources) for file f under the n-th possible content placement scheme
F (M)	Total number of files (Number of popular content files)
N (W)	Number of vehicles (RSUs) within the coverage of the BS
N_C^f	Number of possible content placement schemes for file f
P_f (P_{V2I}^f)	Probability of file f being requested (downloaded through V2I connection)
$\overline{R_B^L}$ (R_W)	Average transmission rate from the BS (remote server)
$R_i(t)$	Distance between VU_i and BS at time t
S_{MBS} (S_{RSU}, S_{VU})	Storage capacity at each BS (RSU, vehicle)
S_f	Required number of encoded packets for file f recovery
TN_B^f ($TN_{R_w}^f$, TN_{BL}^f)	Number of packets downloaded from BS (RSU W_w, the remote server) of file f
VU_i (W_w)	The i-th vehicle (w-th RSU) of the vehicle (RSU) set
$V_{\min}$ ($V_{\max}$)	Minimal (Maximal) speed of vehicles
W_D (W_C)	Weight for delay (service cost) as performance metric
X_A (X_I)	Number of ants (iterations) in Algorithm 1
λ	Arrival rate of vehicles
σ^2	Additive Gaussian noise power density of link between vehicle and BS
$\tau_{\max}$ ($\tau_{\min}$)	Upper (Lower) bound of pheromone value in Algorithm 1
$\tau_{f,n}$ ($\eta_{f,n}$, $p_{f,n}$)	Pheromone value (Heuristic information, Probability) of choosing mode n for file f
$\mathcal{A}_{ji}$	Amount of data transmitted from VU_j to VU_i
$\mathbb{LF}^w$ ($\mathbb{PF}$)	Set of location-based content files under RSU W_w (popular content files)
ND	Non-dominated content placement scheme set in Algorithm 1
$\mathcal{T}_R$	Amount of downloading data provided by RSU

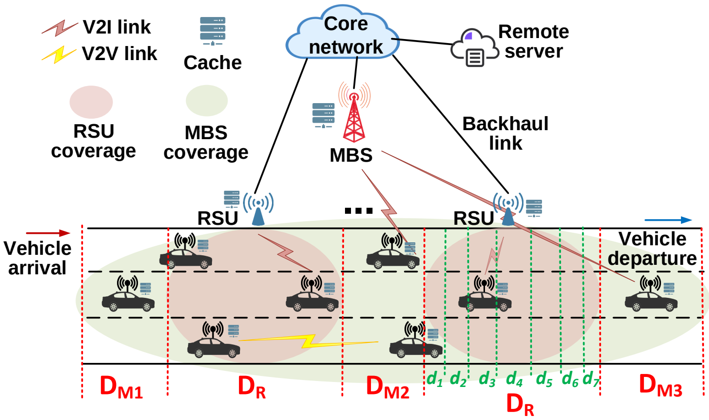

Figure 3.1: HVNet system model.

3.2 System Model

3.2.1 System Overview

As shown in Fig. 3.1, we consider a multi-tier HVNet, including vehicular, RSU, and cellular tiers. For cellular tier, it includes MBS and the backhaul link that connects to the core network and the remote server. To support the content delivery services for vehicles, we develop a cooperative edge caching scheme for the HVNets with multi-tier edge caching servers (i.e., at MBS and RSUs), in which one content can be cached at multiple servers cooperatively. If the requested content has been cached at the MBS and/or RSUs, it can be delivered to the vehicle by the MBS and/or RSUs, depending on the trajectory of the vehicle and the locations of cached contents. Compared with fetching content from the remote server, downloading content from edge servers can effectively reduce both delivery delay and cost. Due to the mobility of vehicles, cooperatively caching one content at multiple servers can further improve the caching efficiency, where vehicles can download content when they drive through the coverage area of edge nodes.

Without loss of generality, we focus on the coverage area of one MBS, within which W RSUs are deployed along the road using WiFi access technology. The coverage areas of different RSUs are assumed to be nonoverlapping, and the communication ranges of RSU and MBS are denoted by D_R and D_B, respectively. Let $\mathbb{W}$ be the set of RSUs, i.e.,

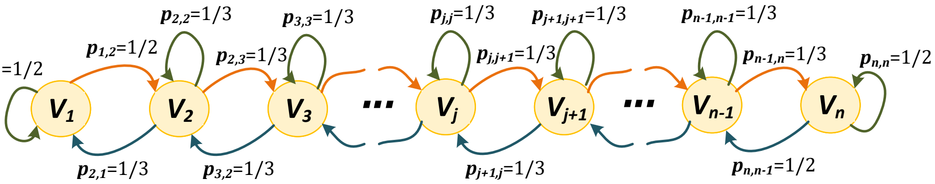

Figure 3.2: State transition model of vehicle's speed variations.

$\mathbb{W} = \{W_1, W_2, ..., W_W\}$. Let $\mathbb{V}$ be the set of N vehicles within the coverage of the MBS, i.e., $\mathbb{V} = \{VU_1, VU_2, ..., VU_N\}$. MBS, RSUs, and vehicles are equipped with caching capability, with the storage capacity denoted by S_{MBS}, S_{RSU}, and S_{VU}, respectively. A network management controller is deployed at MBS, which collects information from vehicles and edge servers and makes decisions on content caching. To proceed, the vehicle mobility model and the communication model of HVNet will be introduced, followed by the content request and caching mechanism.

3.2.2 Vehicle Mobility Model

We consider HVNet with N moving vehicles requesting file downloading services. We assume the arrival of vehicles follows Poisson distribution with arrival rate λ, and the arriving time interval between two adjacent vehicles, ΔT, follows an exponential distribution, i.e., $\Delta T \sim \exp(1/\lambda)$ [67].

In our model, N vehicles are moving towards the same direction. To characterize the real vehicular environment, free driving vehicles with speed constrained in $[V_{\min}, V_{\max}]$ are considered [68], and the speed is updated periodically. In particular, consider the time span is slotted with equal slot duration Δt, the speed is updated at the beginning of each slot, and remains invariant subsequently. The speed update of VU_i is

$$v_i(t + \Delta t) = v_i(t) + \alpha_i(t) \cdot a \cdot \Delta t, \tag{3.1}$$

where $v_i(t)$ is the speed of VU_i at time t, $\alpha_i(t) \in \{1, 0, -1\}$ is a uniformly distributed random parameter that represents the adjustment of acceleration or deceleration, and a is a constant representing the amount of speed variation [68].

This update process can be described as the state transition, using a Markov model shown in Fig. 3.2. All available speed levels of VU are extracted as a set of ordered states $\{V_1, V_2, ..., V_j, ..., V_n\}$, in which $V_1 = V_{min}$, $V_n = V_{max}$, and $V_{j+1} = V_j + a$. Since $\alpha_i(t)$ is uniformly distributed, the transfer probabilities for V_j to its next state are equalized. In particular, for $j = 2, 3, ..., n-1$, V_j may tranfer to V_j, V_{j+1} or V_{j-1} with the probability of $p_{j,j} = p_{j,j+1} = p_{j,j-1} = \frac{1}{3}$. However, for V_1 and V_n, they only have two available transition states according to the speed constraint, and the state transition probabilities are equalized, $p_{1,1} = p_{1,2} = p_{n,n-1} = p_{n,n} = \frac{1}{2}$. Thus, we have the state transition probability matrix $\mathbf{P} = \{p_{i,j}\}$

$$\mathbf{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & \cdots & 0 & 0 & 0 \\ & & & & & \cdots & & & \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}_{n\times n} \tag{3.2}$$

In addition, let $\pi_j = \lim_{t\to\infty} Pr\left\{v_i\left(t\right) = V_j\right\}$ be the steady state probability of the Markov chain, and $\pi = [\pi_1, \pi_2, ..., \pi_n]$ be the probability vector. Considering the transition probability matrix $\mathbf{P}$, π can be calculated by solving the following balance equations,

$$\begin{cases} \pi \cdot \mathbf{P} = \pi \\ \sum_{j=1}^{n} \pi_j = 1. \end{cases} \tag{3.3}$$

Then, we get the steady state probability matrix,

$$\begin{cases} \pi_j = \frac{3}{3n-2} & j = 2, 3, ..., n-1 \\ \pi_j = \frac{2}{3n-2} & j = 1, n. \end{cases} \tag{3.4}$$

To describe the statistical characteristics of vehicle mobility, we obtain the expectation $(E\left[v\right])$ and the variance $(Var\left[v\right])$ of $v_i\left(t\right)$ based on (3.4).

$$E\left[v\right] = \frac{V_{min} + V_{max}}{2} = V_{min} + \frac{n-1}{2}a. \tag{3.5}$$

$$Var\left[v\right] = E\left[v^2\right] - E\left[v\right]^2 = \sum_{j=1}^{n} V_j^2 \cdot \pi_j - \left(\frac{V_{min} + V_{max}}{2}\right)^2 = \frac{n^3 - 2n^2 + 3n - 2}{12n - 8}a^2. \tag{3.6}$$

Vehicular Headway Distance Model

We consider the headway distance from VU_i to VU_j as a directional variable $D\left(t\right)$, which is positive if VU_i behind VU_j in the moving direction. A G/G/1 queue model is built to predict headway distance [67], which is given in Fig. 3.3, in which the queue length represents the distance. The movement of vehicles determines the arrival and departure of queue element. Thus, $E\left[v_j\right]$ and $Var\left[v_j\right]$ represent the expectation and variance of

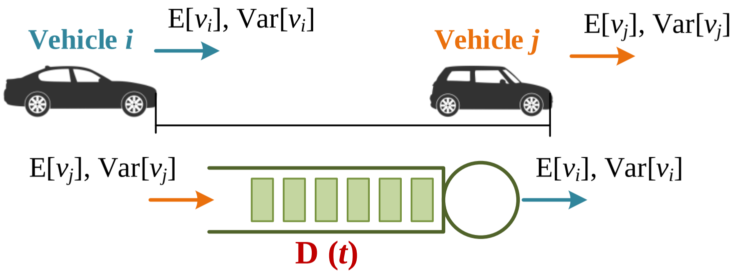

Figure 3.3: The queue model of vehicle's headway.

the queue arrival rate, while $E\left[v_i\right]$ and $Var\left[v_i\right]$ are used to describe the queue service rate. Applying G/G/1 queue model, the headway distance $D\left(t\right)$ can be modeled as a one-dimensional Wiener process under the simplified scenario, with the assumption that $E\left[v_i\right]$ and $Var\left[v_i\right]$ are stable over time. In addition, we assume all VUs follow the same mobility model, i.e. $E\left[v_i\right] = E\left[v\right]$ and $Var\left[v_i\right] = Var\left[v\right]$. Then, we get the drift $\mu = E\left[v_j\right] - E\left[v_i\right] = 0$ and variance $\sigma_D = Var\left[v_j\right] + Var\left[v_i\right] = 2Var\left[v\right]$ of $D\left(t\right)$. Given the initial headway distance, d_0, the probability density function of $D\left(t\right)$ at time t is

$$\begin{aligned} f_D\left(x; d_0, t\right) &= Pr\left\{x \leq D\left(t\right) \leq x + \Delta x | D\left(0\right) = d_0\right\} \\ &= \frac{1}{\sqrt{2\pi\sigma_D t}} \exp\left\{-\frac{\left(x - d_0 - \mu t\right)^2}{2\sigma_D t}\right\}. \end{aligned} \tag{3.7}$$

3.2.3 V2I Communication Model

V2B Communication Model

If VU_i is served by the MBS, the wireless transmission rate at t, denoted by $R_{B,i}(t)$, is

$$R_{B,i}(t) = B_i\left(t\right) \cdot \log_2\left(1 + \frac{P_T \cdot L\left(R_i\left(t\right)\right)}{\sigma^2}\right), \tag{3.8}$$

where $B_i\left(t\right)$ is the allocated bandwidth for VU_i at time t, P_T is the transmit power density of the MBS, σ^2 is the additive Gaussian noise power density, $R_i\left(t\right)$ is the distance between VU_i and MBS in kilometers at time t, and $L\left(R_i\left(t\right)\right)$ is the path loss between VU_i and MBS [69] given by

$$\begin{aligned} L\left(R_i\left(t\right)\right) &= 40(1 - 4 \cdot 10^{-3} \cdot Dhb) \log_{10}(R_i\left(t\right)) \\ &\quad -18 \log_{10}(Dhb) + 21 \log_{10}(f) + 80\text{dB} + X, \end{aligned} \tag{3.9}$$

Table 3.2: Adaptive transmission rate of WiFi access point.

Zone	1	2	3	4	5	6	7
d_k (m)	25	30	40	60	40	30	25
R_k $(Mbps)$	1	2	5.5	11	5.5	2	1

where f is the carrier frequency in MHz, Dhb is the antenna height of infrastructure in meters that measured from the average rooftop level, and X in dB represents shadowing channel fading, which follows Log-normal distribution. Note that, vehicles download content through orthogonal channels and the coverage area of different MBSs along the street are non-overlapping, so the interference is negligible in vehicle-to-MBS (V2B) communication model.

We assume the number of vehicles that enter and leave the MBS are equalized, so we use $N = \frac{D_B}{\frac{1}{\lambda} \cdot E[v]} \cdot N_L$ to denote the number of vehicles in MBS coverage, where D_B is the MBS coverage range and N_L is the number of lanes of the street. Considering the worst case that all the vehicles in the MBS coverage request MBS access, we can get the lower bound of $R_{B,i}(t)$ as $R_{B,i}^L(t)$, where $B_i(t) = B/N$ and B is the available bandwidth of MBS. We use this lower bound to estimate the average transmission rate of MBS as $\overline{R_B^L}$. Considering the fixed relative location of MBS and the midpoint of street, we replace $L(R_i(t))$ by $\check{L}(x)$, where x is the distance from vehicle to the midpoint, $x \in [0, D_B/2]$, it holds that

$$\overline{R_B^L} = \frac{2}{D_B} \int_0^{D_B/2} \frac{B}{N} \cdot \log_2 \left(1 + \frac{P_T \cdot \check{L}(x)}{\sigma^2} \right) dx. \tag{3.10}$$

If the vehicle is connected to the remote server through MBS and its wired backhaul link, the transmission rate is determined by the backhaul link R_W.

V2R Communication Model

The communication model between vehicles and RSUs with WiFi access technology is investigated in [70] as an adaptive vehicle to RSU (V2R) transmission model, in which the coverage area is divided into K zones as shown in Fig. 3.1, and the transmission rate achieved within the k-th zone is denoted as R_k. According to the IEEE 802.11b standard [71], $K = 7$ and the rates through the RSU coverage is symmetric, which is given in Table 3.2 with the range of each zone (d_k) [70]. To simplify the analysis, we consider a MAC protocol that the connection time for all the vehicles within the coverage is equally allocated, which means the transmission rate of the k-th zone is equally allocated to the associated vehicles. Therefore, the bit rate of VU_i at instant t is expressed as $R_{R,i}(t) = \frac{R_k}{N_k(t)}$, where VU_i is driving through the k-th zone at time t and $N_k(t)$ is the number of vehicles in the k-th zone at time t. Similar to MBS, the number of vehicles that enter and leave the k-th zone are considered to be equivalent, so we use N_k to replace $N_k(t)$ in the following analysis.

Each RSU provides download service to all the vehicles within its coverage, and the amount of data it can provide is

$$\mathcal{T}_R = \sum_{k=1}^{7} N_k \cdot \frac{d_k}{E\left[v\right]} \cdot \frac{R_k}{N_k} = \sum_{k=1}^{7} \frac{d_k \cdot R_k}{E\left[v\right]}, \tag{3.11}$$

where the duration of VU_i staying in the k-th zone is estimated by $d_k/E\left[v\right]$.

Assume that there is an admission control buffer with a size of $\mathcal{T}_R$ in each RSU, which stores the content that will be downloaded by vehicles in its coverage. If a new request of a vehicle is accepted, the requested content will be added to the buffer. Then, during the content transmission, downloaded data will be removed from the buffer. Thus, the admission decision of each vehicle to the RSU can be determined by this buffer. When a new request arrives, it will be accepted if there is enough buffer space for the requested content. A vehicle can always download the content within the RSU coverage as long as the buffer is not overflowed.

3.2.4 V2V Communication Model

When a vehicle requests for content download, it firstly estimates the probability of successful transmission through V2V communication, and then makes the decision on downloading the content either from other vehicles or from edge servers, e.g., the MBS or RSUs.

We consider the V2V communication on the DSRC spectrum from 5.850 to 5.925 GHz. In particular, the physical layer operation is specified by the IEEE 802.11p standard, while for the MAC layer, the IEEE 802.11b DCF scheme is applied for channel contention model. Given that the file size of requested content by VU_i is F_i, the data amount, $\mathcal{A}_{ji}$, transmitted from VU_j to VU_i during the time period of T_d ($T_d > 0$), can be evaluated following [67]. Then, the successful probability of transmitting at least F_i bits of data from VU_j to VU_i with the time constraint T_d is bounded by

$$Pr\left\{\mathcal{A}_{ji} \geq F_i\right\} \geq \frac{\left[E\left(\mathcal{A}_{ji}\right) - F_i\right]^2}{Var\left(\mathcal{A}_{ji}\right) + \left[E\left(\mathcal{A}_{ji}\right) - F_i\right]^2}, \tag{3.12}$$

where $E\left(\mathcal{A}_{ji}\right)$ is the mean of $\mathcal{A}_{ji}$, and $Var\left(\mathcal{A}_{ji}\right)$ is an upper bound of the variance of $\mathcal{A}_{ji}$. Both mean and variance are dependent on the headway distance in (3.7). This successful probability indicates whether the content can be downloaded from neighboring vehicles within T_d. For location-based content, T_d is set to be proportional to the file size, representing the average transmission delay through V2I connections. For popular content, T_d is set as the delivery delay requirement.

3.2.5 Content Request and Caching Model

Content Popularity Model

We consider two types of contents that are requested by vehicles: popular contents (e.g., news and music service) and location-based contents (e.g., HD map and local commercial information), denoted by PF and LF, respectively. The location-based contents provide local information, which is always needed by the vehicles around the location. For example, if an RSU is deployed near a shopping center, the advertisements are very likely to be requested by the vehicles driving into the RSU coverage. Thus, we assume a vehicle may request for location-based content when it enters the coverage of an RSU, which is dependent on the location. Within the coverage of RSU W_w, there are N_w location-based content files, and the sets of these files and the corresponding sizes are denoted by $\mathbb{LF}^w = \{LF_1^w, LF_2^w, ..., LF_{N_w}^w\}$ and $\mathbb{S}^{Lw} = \{S_1^{Lw}, S_2^{Lw}, ..., S_{N_w}^{Lw}\}$, respectively. However, the request for popular content may be generated within MBS coverage, and the sets of all the M popular files and the corresponding sizes are denoted by $\mathbb{PF} = \{PF_1, PF_2, ..., PF_M\}$ and $\mathbb{S}^{PF} = \{S_1^{PF}, S_2^{PF}, ..., S_M^{PF}\}$, respectively.

Due to the features of LF contents, the file set for a vehicle is dependent on location. If a vehicle sends a content request within the coverage of RSU W_w, an integrated file set $\mathbb{F}^w = \{1, ..., F_w\}$ is established as its overall content set, consisting of $\mathbb{LF}^w$ and $\mathbb{PF}$. The number of files in $\mathbb{F}^w$ is denoted by $F_w = M + N_w$. If a vehicle sends a request at a location not covered by any RSUs, which means the vehicle is not going to request for a location-based content, the file set $\mathbb{F}^0 = \mathbb{PF} = \{1, ..., F_0\}$ is established for the vehicle, where $F_0 = M$. All the file sets are constantly updated by adding new files. For $\mathbb{F}^j, j = 0, 1, ..., W$, the corresponding file request probability is $\mathbb{P}^j = \{P_{j,1}, ..., P_{j,F_j}\}$, which follows Zipf-like distribution [72]. The request probability of the k-th popular content file can be calculated as

$$P_{j,k} = Pr_j \cdot \frac{\frac{1}{k^s}}{\sum_{n=1}^{F_j} \frac{1}{n^s}}, \tag{3.13}$$

where Pr_j is the probability of vehicles sending the request within RSU W_j if $j = 1, ..., W$

or out of RSU if $j = 0$, F_j is the number of files, k is the rank of popularity, s is the parameter characterizing the skewness of the Zipf distribution.

Fountain Coded Caching

Due to the limited transmission range of each edge caching server, it is difficult for the vehicles to download the whole file within the coverage of one single server, especially when the vehicles are with high speed or the size of file is large. By employing coded caching, which divides one content into separated coded packets, it has a higher probability to successfully deliver a coded packet with smaller size within one contact duration between the vehicles and the edge servers. In our coded caching scheme, through random linear fountain coding [23], each content is encoded into independent packets with a size of l bits, which is fixed and equivalent for all contents. If a content has a size of Kl bits, it can be recovered from any set of K' encoded packets, where $K' = K \cdot \sum_{d=1}^{K} z\left(d\right)$ is no less than K, and $z\left(d\right)$ can be calculated as follows

$$z\left(d\right) = \begin{cases} \frac{1}{K} + \frac{S}{K \cdot d} & d = 1 \\ \frac{1}{d(d-1)} + \frac{S}{K \cdot d} & d = 2, 3, ..., (K/S) - 1 \\ \frac{1}{d(d-1)} + \frac{S}{K} \ln\left(\frac{S}{\delta}\right) & d = K/S \\ \frac{1}{d(d-1)} & d = (K/S) + 1, ..., K \end{cases} \tag{3.14}$$

where $S = c \cdot \ln\left(K/\delta\right) \cdot \sqrt{K}$, and δ is the bound on decoding failure probability after receiving K' packets. We set $c = 0.2$ and $\delta = 0.05$ for the fountain code scheme [23].

3.3 Delay and Cost Analysis for Cooperative Edge Caching

In this section, we perform the theoretical analysis of content delivery delay and service cost under the cooperative caching scheme. Particularly, we first elaborate on the workflow of cooperative content caching and content delivery. Then, we present the analysis of both location-based and popular contents, considering the differential delivery requirements.

3.3.1 Cooperative Content Caching

Denoted by $\mathbb{F} = \{1, 2, ..., f, ..., F\}$, the ground content set consists of $\mathbb{LF}^w$ and $\mathbb{PF}$. The size of $\mathbb{F}$ is given by $F = M + \sum_{w=1}^{W} N_w$. The data size of each file is represented by the required number of encoded packets for data recovery, $\mathbb{S} = \{S_1, S_2, ..., S_f, ..., S_F\}$. The caching capacities for each MBS, RSU, and vehicle, S_{MBS}, S_{RSU}, S_{VU}, are divided by packet size (l bits) to make *packet* as the unit. Similarly, the size of admission control buffer, $\mathcal{T}_R$, for RSU is modified to packets.

Let the MBS and RSU W_w precache $s_B^f (\leq S_f)$ and $s_{R_w}^f (\leq S_f)$ independent encoded packets of file f, respectively. Considering the limited caching capacities of the MBS and RSUs, the total number of packets cached by each server has to satisfy the constraint, i.e., $\sum_{f=1}^{F} s_B^f \leq S_{MBS}$ and $\sum_{f=1}^{F} s_{R_w}^f \leq S_{RSU}$, $w = 1, 2, ..., W$. In addition to capacity overhead, caching content also leads to a management cost. We define the price of caching one packet at MBS as CP_B, and CP_R for the RSU.

3.3.2 Content Delivery

If a target vehicle sends a content request, the request will be processed by the controller. Based on content placement and network access states, the controller makes the decision on association and content downloading for the vehicle. Then, the vehicle fetches the content following the instruction, including how many packets should be downloaded from other vehicles or edge caching servers.

PF content request can be raised by the vehicle at any locations within the coverage area of MBS. However, due to the dependency between location-based popularity of LF content and RSU coverage, the LF request can be raised by the vehicles entering the coverage of RSU W_w. The request is processed by the controller with the following steps:

1. *Availability of V2V transmission* – Check whether it is possible to transmit the content through V2V connections. Find the nearest vehicle holding the requested content and calculate the successful V2V transmission probability based on (3.12).

2. *Availability of edge cached content* – Obtain the list of MBS or RSUs that simultaneously cache the requested content and are available to the vehicle. Based on the moving speed and the duration of request, the edge caching servers (MBS and/or RSUs) that the vehicle will drive through can be determined. For an RSU, if serving the newly requested content makes its admission buffer overflow, the RSU should not be included in the list.

3. *Access to the remote server* – Make the decision on whether the target vehicle needs to fetch content from the remote server. Based on the total available cached content in the list, if the vehicle does not get sufficient packets for decoding by the transmission

deadline, it will download the content from the remote server regardless of which connection is being utilized currently.

Next, we analyze the average download delay and cost for the request, which are taken as performance metrics for content placement scheme. Note that, content delivery through V2V connection is not considered when designing the content placement scheme, because V2V transmission performance is not affected by the content placement.

We assume contents held in the vehicles follow the file popularity distribution. If the content is available from neighboring vehicles, the successful V2V transmission probability is given by the lower bound in (3.12). Accordingly, the probability that the vehicle downloads file f through V2I communication (P^f_{V2I}) can be calculated. If the lower bound is higher than a threshold ξ, the vehicle is arranged to download from the other vehicles, $P^f_{V2I} = 1 - Pr\left\{\mathcal{A}_{ji} \geq S_f\right\}$. Otherwise, the target vehicle needs to fetch content through V2I connections, i.e., $P^f_{V2I} = 1$.

3.3.3 Delay Analysis of Content Delivery

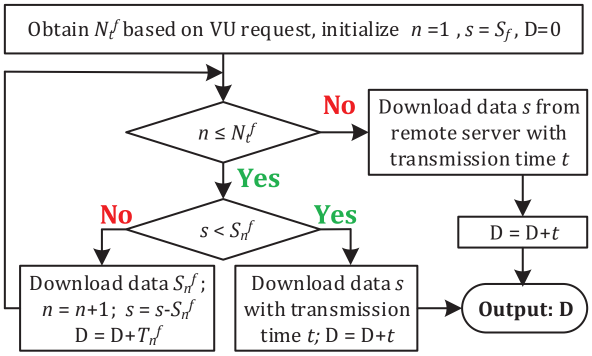
Obtain N_t^f based on VU request, initialize n =1 , s = S_f, D=0
$n \leq N_t^f$
No
Download data s from remote server with transmission time t
Yes
No
$s < S_n^f$
Yes
D = D+t
Download data S_n^f ; n = n+1; s = s-S_n^f D = D+T_n^f
Download data s with transmission time t; D = D+t
Output: D

Figure 3.4: Flowchart of downloading content through V2I connection.

For a vehicle served by V2I connections, the transmission process of file f can be divided into several segments depending on the handover of data providers. N^f_t denotes the number of segments that the vehicle can get connected, which is determined by the size and delay requirement of file f, and the list of edge servers caching the file f. In addition, the duration of each segment is defined as $T^f_n, n = 1, 2, ..., N^f_t$ and the downloaded data volume in packets during T^f_n is denoted as $S^f_n, n = 1, 2, ..., N^f_t$.

Based on the analysis of V2R communication, a vehicle can download the requested content from RSU W_w within the coverage, once it successfully accesses to the RSU. Thus, if the vehicle accesses to RSU W_w during the n-th segment, the amount of downloaded data of the n-th segment is $S_n^f = s_{R_w}^f$ and the transmission delay for each packet can be defined as $t_{R_w}^f = T_n^f / s_{R_w}^f$. If the vehicle accesses to MBS during the n-th segment, the transmission delay for each packet can be defined as $t_B^f = l/\overline{R_B^L}$, and the downloaded data volume $S_n^f = \left\lfloor T_n^f / t_B^f \right\rfloor$, where $\lfloor \cdot \rfloor$ is the floor function.

If the remaining required data, s (packets), is less than S_n^f, which means the transmission will terminate during segment T_n^f, the transmission delay for remaining s can be calculated as $s \cdot t_B^f$ or $s \cdot t_{R_w}^f$, depending on the n-th vehicle access server. Otherwise, if $s > 0$ after the vehicle goes through all N_t^f segments, the vehicle will download remaining data from the remote server, and the transmission delay is $T_{n+1}^f = s \cdot t_{BL}^f$, where $t_{BL}^f = l/R_W$ is the backhaul link transmission delay for each packet. The delay D can be derived by the process shown in Fig. 3.4. Considering the coverage ranges of RSU and MBS [68], we deploy two RSUs along the street as an example, but the analysis method and the content placement scheme design can be extended to scenarios with more RSUs.

For LF content downloading, the vehicle is expected to fetch the content as soon as possible, so we evaluate the average download delay and design the caching scheme to minimize the delay. In order to evaluate the delay performance, we calculate the mean of total content download delay for LF files ($\overline{D}$), the details are given in Appendix A.1. However, for PF content downloading, the vehicle always prefers to download the content within a latency requirement. We define three caching modes for PF content: *Mode 1* - only caching at MBS, *Mode 2* - caching at both MBS and RSUs, and *Mode 3* - no packet cached at MBS or RSUs. In Appendix A.2, for each mode, we evaluate the volume of downloaded data before the deadline of data delivery, and a content placement scheme is designed to ensure it is sufficient for data recovery.

3.3.4 Cost Analysis of Content Delivery

The cost of content service can be divided into two parts, the cost of caching the content and the cost of transmitting the data to vehicles.

In terms of caching cost, the prices for MBS, RSU W_w caching one packet are denoted as CP_B, CP_R, and $CP_B > CP_R$. Thus, the caching cost for file f, C_C^f, is given by

$$C_C^f = CP_B \cdot s_B^f + CP_R \cdot \sum_{w=1}^{W} s_{R_w}^f. \tag{3.15}$$

The total caching cost is denoted as $C_C = \sum_{f=1}^{F} C_C^f$, where F is the total number of files.

In terms of data transmission cost, the prices for MBS, RSU W_w and the remote server transmitting one packet are denoted as TP_B, TP_{R_w}, TP_{BL}, and $TP_{BL} > TP_B > TP_{R_w}$. In order to calculate the total transmission cost, numbers of packets downloaded by vehicle through these three methods need to be evaluated, which are denoted as $TN_B^f, TN_{R_w}^f, TN_{BL}^f, f = 1, 2, ..., F$. In addition, the number of vehicles requesting file f is $N \cdot P_f$, where P_f is the probability of file f requested by vehicle.

For the vehicle requesting file $f \in \mathbb{LF}^w$, it firstly downloads the cached packets, then from the remote server if necessary, thus we have $TN_B^f = s_B^f, TN_{R_w}^f = s_{R_w}^f, TN_{BL}^f = S_f - s_B^f - \sum_{w=1}^{W} s_{R_w}^f, f \in \mathbb{LF}^w$.

For vehicle requesting file $f \in \mathbb{PF}$, we first evaluate the average download duration of each method, $\overline{H_i^f}, i \in \{B, R_w, BL\}, w = 1, 2, \cdots, W$, which can be determined by D_R^f and handover duration sequences. Then, we can obtain its average number of downloading packets, $TN_i^f = \min\left(\left\lfloor \overline{H_i^f}/t_i^f \right\rfloor, s_i^f\right), i \in \{B, R_w, BL\}, w = 1, 2, \cdots, W$.

Thus, the average transmission cost for file f, C_T^f, is given by

$$C_T^f = N \cdot P_f \cdot \sum_{i \in \{B, R_w, BL\}} TP_i \cdot TN_i^f. \tag{3.16}$$

Then, the total transmission cost, C_T, is $C_T = \sum_{f=1}^{F} C_T^f$, where F is the total number of files.

Based on (3.15) and (3.16), the service cost, including both the caching and transmission cost, of all the files can be obtained. To reduce the total service cost, popular contents need to be cached at and transmitted from low-cost servers. In what follows, we achieve a low-cost content placement scheme via solving an optimization problem.

3.4 Cooperative Content Placement Problem

3.4.1 Multi-objective Cooperative Content Placement Problem

For each file, its content placement is denoted as $\mathbf{s}^f = \left(s_B^f, s_{R_w}^f\right), w = 1, 2, ..., W$. With the objective of jointly minimizing service cost and delay requirement of content service, the cooperative content placement problem can be formulated as **P1**. Since the objective of popular content service is downloading before the deadline, mode-based content placement schemes are designed for popular contents based on the delay analysis in Section 3.3.3. In **P1**, the solution set for each popular content consists of its mode-based placements,

$$(\mathbf{P1}) : \min_{\{\mathbf{s}^f\}} \left(\overline{D}, C_C + C_T\right)$$

$$\text{s.t.} \begin{cases} s_B^f \leq S_f, f = 1, 2, ..., F & (3.17a) \\ s_{R_w}^f \leq S_f, w = 1, ..., W, f = 1, 2, ..., F & (3.17b) \\ \sum_{f=1}^{F} s_B^f \leq S_{MBS}, & (3.17c) \\ \sum_{f=1}^{F} s_{R_w}^f \leq S_{RSU}, w = 1, ..., W & (3.17d) \\ \sum_{f=1}^{F} N \cdot P_f \cdot TN_{R_w}^f \leq \mathcal{T}_R, w = 1, ..., W & (3.17e) \end{cases}$$

which guarantee the delay requirements. For location-based contents, to achieve minimal downloading delay, we consider $\overline{D}$ obtained from delay analysis in the objective function. Meanwhile, the service costs for both types of contents are considered, and jointly minimized with the delay for location-based contents. In **P1**, (5.2a) and (5.2b) are set to avoid redundant content caching for the MBS and RSUs, (3.17c) and (3.17d) reflect the caching capacity constraints of the MBS and RSUs, respectively, while (3.17e) is based on the admission capacity discussed in (3.11).

In order to solve the problem, content placement design should consider the tradeoff between content diversity, service cost, and download delay. For the RSUs, if more packets for one file ($s_{R_w}^f$) are cached, the vehicle can download each packet with lower delay ($t_{R_w}^f$). Thus, larger $s_{R_w}^f$ contributes to faster download rate for the vehicle, but it reduces the content diversity of the RSU, resulting in a low cache hit rate. For the MBS, it guarantees a high hit rate by providing a large access coverage whereby vehicles can fetch the content anywhere, but the transmission cost of the MBS is higher than that of the RSU.

In addition, the intermittent connection during transmission should be considered. For RSU transmission, the caching resource would be wasted if excessive packets of one file are cached but only part of these packets can be downloaded for vehicles within the RSU's coverage.

To achieve the objective of PF and LF content delivery services, we design content placement for these two types of content in different ways. For LF content, the objective is a joint minimization of delay and service cost. To minimize the transmission cost, the LF file $f \in \mathbb{LF}^w, w = 1, 2, ..., W$ should be cached at RSU W_w, because RSU has a lower transmission cost price than the MBS. However, the transmission delay of RSU may be larger than that of the MBS for files with small S_f. For PF content, the objective is to minimize the service cost within the download latency constraint. Based on the analysis in Section 3.3, given a delay constraint, the required caching data placement is deterministic for each case. Then, the service cost can be determined accordingly. Thus, the content placement for PF content is simplified to a selection of caching mode.

Although the content placement principle is different for PF and LF content, they share both the caching capacity and access resources. Thus, a joint design of PF and LF content placement is a requisite. Another challenge of this problem is the cooperation, as the MBS and RSUs may cooperatively cache different packets of one file. Therefore, the placement problem has an unacceptable solution set, which causes the curse of dimensionality.

LF Content Cooperative Placement Subproblem

The objective of LF content caching is to jointly minimize the transmission delay and service cost, considering the limited caching capacity of both MBS and RSUs, and admission limitation of RSUs. For $f \in \mathbb{LF}^w, w = 1, 2, ..., W$, since the delay, service cost, required caching capacity, and access resources are determined by the content placement $\left(s_B^f, s_{R_w}^f\right), w = 1, 2, ..., W$, we build a matrix for each file to record the delay, service cost, and required access resources. Note that, for $f \in \mathbb{LF}^w, w = 2, ..., W$, content transmission start under RSU W_w, so $s_{R_w}^f = 0, w = 1, ..., w - 1$. To reduce the size of this matrix, the following principles are applied to build the matrix elements:

1. Avoid redundant content caching, $s_B^f + \sum_{w=1}^{W} s_{R_w}^f \leq S_f$;
2. Satisfy the caching capacity constraint, $s_{R_w}^f \leq S_{RSU}$ and $s_B^f \leq S_{MBS}$;
3. Satisfy the admission capacity, $N \cdot P_f \cdot s_{R_w}^f \leq \mathcal{T}_R$.

Based on the principles, we get the matrix for file $f \in \mathbb{LF}^w$, in which each column represents a possible placement scheme. We use N_C^f to denote the number of possible content placement schemes for file f. Then, we calculate the average delay, service cost, and access resources for each scheme. Given a placement scheme $\left(s_B^f(n), s_{R_w}^f(n)\right), w = 1, 2, ..., W, n = 1, 2, ..., N_C^f$, the corresponding average delay $\overline{D}^f(n)$ is calculated according to Section 3.3.3. The average service cost $\overline{C}^f(n) = C_C^f + C_T^f$, where C_C^f and C_T^f are discussed in Section 3.3.4. The required access resources $T_{R_w}^f(n) = N \cdot P_f \cdot TN_{R_w}^f, w = 1, 2, ..., W$. In addition, required caching resources can be determined by the content placement scheme. This matrix will be utilized as input information in the cooperative content placement scheme.

PF Content Cooperative Placement Subproblem

Different from LF content, the objective of caching PF content is minimizing the service cost with guaranteed delay. Based on previous discussions, the content placement for PF content is simplified to a selection of caching modes. We get the matrix for file $f \in \mathbb{PF}$, in which each column corresponds to one caching mode. Firstly, for each mode n, based on latency requirement, we determine its content placement scheme $\left(s_B^f\left(n\right), s_{R_w}^f\left(n\right)\right), w = 1, 2, ..., W, n = 1, 2, 3$ ($N_C^f = 3$). Then, the average service cost, $\overline{C}^f\left(n\right)$, and the access resources, $T_{R_w}^f\left(n\right)$, for each mode are calculated. In addition, $\overline{D}^f\left(n\right)$ is set to 0 in accordance with the LF matrix.

3.4.2 Multi-objective MMKP Formulation for Cooperative Content Placement

The cooperative content placement problem **P1** can be transferred to a multi-objective MMKP, as shown in **P2**, which is a variant of the knapsack problem. There are F groups of items, and the f-th group includes N_C^f items.

The objective function requires the joint minimization of delay and service cost, where $x_{f,n}$ is a binary variable representing the designed content placement scheme for file f, and $x_{f,n} = 1$ if scheme n is selected, and $x_{f,n} = 0$ otherwise. Due to the limited resources, $2W + 1$ constraints are considered in **P2**, including the caching capacity of MBS in (3.18b) and RSUs in (3.18c), and admission limitation for RSUs in (3.18d).

Dynamic programming (DP) is a widely used method to solve the knapsack problem. However, DP is inefficient for large scale problems due to the complex constraint calculation and considerable state storage requirements, which is known as the curse of dimensionality. In what follows, we propose a content placement scheme based on ACO to find near-optimal solutions to the multi-objective MMKP. Furthermore, to evaluate the gap between this near-optimal solution and the optimum, we obtain a lower bound of the objective value by relaxing **P2** to a linear programming (LP) problem.

$$(\mathbf{P2}): \min_{\{x_{f,n}\}} \left(\sum_{f=1}^{F} \sum_{n=1}^{N_C^f} \overline{D}^f(n) \cdot x_{f,n}, \sum_{f=1}^{F} \sum_{n=1}^{N_C^f} \overline{C}^f(n) \cdot x_{f,n} \right)$$

$$\text{s.t.} \begin{cases} \sum_{n=1}^{N_C^f} x_{f,n} = 1, x_{f,n} \in \{0,1\}, f = 1,2,...,F; & (3.18a) \\ \sum_{f=1}^{F} \sum_{n=1}^{N_C^f} s_B^f(n) \cdot x_{f,n} \le S_{MBS}; & (3.18b) \\ \sum_{f=1}^{F} \sum_{n=1}^{N_C^f} s_{R_w}^f(n) \cdot x_{f,n} \le S_{RSU}, w = 1,...,W; & (3.18c) \\ \sum_{f=1}^{F} \sum_{n=1}^{N_C^f} T_{R_w}^f(n) \cdot x_{f,n} \le \mathcal{T}_R, w = 1,...,W. & (3.18d) \end{cases}$$

3.5 ACO-based Scheme Design

The ACO was first proposed as an approximate method for solving complex optimization problems, inspired by how ant colonies find the path from food source to the nest[73]. At the beginning stage of foraging, the ants explore paths randomly and leave pheromone when they move on the ground. The quantity of pheromone is inversely proportional to the length of the path, which means a shorter path has more pheromone. Then, the ants can choose a path according to the pheromone, and they always prefer the path with stronger pheromone. This is how ants exchange information with each other through pheromone, and find the shortest path cooperatively. Note that, heuristic information, such as the potential gain of choosing a certain step along the path, will be also used by the ants in addition to pheromone. The problem formulated in **P2** is a multi-objective minimization problem, which can be solved by multi-objective evolutionary algorithms (MOEAs). The concept of dominance is widely used in MOEAs through establishing a non-dominated solution. We will first introduce the concept of dominance and non-dominated solutions, then present the dominance-based ACO scheme for cooperative content placement problem.

3.5.1 Non-dominated solution

Consider a multi-objective minimization problem with n objectives ($\mathbf{g} = \{g_1, g_2, ..., g_n\}$) and m decision variables ($\mathbf{x} = \{x_1, x_2, ..., x_m\}$). Thus, the solution can be denoted by $\mathbf{x}$ and its corresponding objective vector is $\mathbf{g}(\mathbf{x})$. Based on the definition in [74], if $\mathbf{x}_1$ is not worse than $\mathbf{x}_2$ in any objective and strictly better than $\mathbf{x}_2$ in at least one objective, the solution $\mathbf{x}_1$ is defined to dominate $\mathbf{x}_2$. If a solution is not dominated by any other solutions, it is defined as a non-dominated solution. The corresponding objective points of all non-dominated solutions form a front in the objective space, which is the Pareto optimal front [75]. Thus, the multi-objective minimization problem can be solved by finding the non-dominated solution set, which is also the set of Pareto optimal solutions.

3.5.2 Dominance-based ACO Scheme

In order to find the solution to **P2**, we propose a dominance-based ACO scheme, which optimizes multiple objectives by combining dominance with the ACO. Multiple objectives and dominance are incorporated in the following phases:

1. Pheromone update: The pheromone is updated every iteration, including general pheromone evaporation and additional pheromone that is incrementally deposited by the selected solutions from the non-dominated set;

2. Definition of pheromone and heuristic information: Pheromone and heuristic information are used by ants to make probabilistic decisions at each step. Due to the multiple objectives, pheromone and heuristic information can be stored in multiple matrices, each of which corresponds to one objective. Since the ant has to aggregate the pheromone/heuristic matrices when making the decisions, we calculate a weighted sum of multiple matrices to aggregate multiple objectives.

3.5.3 Dominance-based ACO Content Placement Scheme

Consider an ant colony with X_A ants, each ant has the capability of constructing a feasible content placement scheme. After all X_A ants constructing their schemes, the non-dominated content placement scheme set, **ND**, can be updated, in which the delay and cost for each newly established scheme are compared with the schemes in current **ND** to determine the updated non-dominated scheme. Then, based on **ND**, we can update the pheromone matrix to simulate the evaporation and accumulation of pheromone, and the updated pheromone will be used during the next iteration. The proposed scheme terminates after X_I iterations.

A set of pheromone vectors $\left[\tau_{f,1}, \tau_{f,2}, ..., \tau_{f,N_C^f}\right], f = 1, 2, .., F$ are built at the initialization stage, and each element is set to be $\tau_{\max}$, which is the upper bound of pheromone value. At the end of each iteration, the pheromone vectors are updated. First, in order to simulate the pheromone loss caused by evaporation, all elements are decreased by multiplying $(1-\rho)$, where $\rho \in [0, 1]$. Then, based on the updated non-dominated scheme set, all the pheromone values of elements (f, n) (selected by the scheme $\mathbf{x}_i, \mathbf{x}_i \in \mathbf{ND}$) are increased by multiplying $(1+\gamma_i)$. γ_i describes how good the performance of $\mathbf{x}_i$, i.e.,

$$\gamma_i = \frac{\mathcal{F}(\mathbf{x}_i)}{\sum_{\mathbf{x}_j \in \mathbf{ND}} \mathcal{F}(\mathbf{x}_j)}, \tag{3.19}$$

where $\mathcal{F}(\mathbf{x}_i) = 1/\left(W_D \cdot \overline{D}(\mathbf{x}_i) + W_C \cdot \overline{C}(\mathbf{x}_i)\right)$ is the performance evaluation function of the scheme (i.e., the reciprocal of weighted summation of delay and service cost after applying the scheme $\mathbf{x}_i$), and W_D and W_C are the weights for delay and service cost, respectively.

Therefore, the update of pheromone follows

$$\tau_{f,n}(t+1) = \tau_{f,n}(t) \cdot (1-\rho) \cdot \prod_{\mathbf{x}_i \in \mathbf{ND}_{f,n}} (1+\gamma_i), \tag{3.20}$$

where $\tau_{f,n}(t)$ is the pheromone value in the t-th iteration, and $\mathbf{ND}_{f,n} \subseteq \mathbf{ND}$ is the subset of non-dominated schemes that contains the choice of (f, n). Thus, the pheromone value $\tau_{f,n}$ represents how good the performance achieved by the element (f, n) in previous iterations, which can be used to guide the selection for the next iteration.

In order to balance the exploitation and exploration (i.e., selecting the known good-performance scheme and choosing the under-explored schemes), we set the upper and lower bound of pheromone value as $\tau_{\max}$ and $\tau_{\min}$, respectively. This can avoid the ACO algorithm entering a stagnation situation by bounding the level of exploration. A pseudo-random proportional rule is also used in content placement construction for balancing exploration and exploitation. With the exploitation probability q_0, the ant selects the content placement scheme with the best potential performance; otherwise, the decision is made probabilistically, where the probability of a scheme being selected is proportional to its potential performance.

In terms of heuristic information, we combine the two objectives (delay and service cost) together as a weighted sum. In addition to the objective functions, heuristic information

Algorithm 1 ACO-based Content Placement

```
Input:   Parameters of the system model, including vehicle mobility model, V2I and V2V communication models
Output:   Content placement decisions $\mathbf{x} = \{x_{f,n}\}$
Initialization:
Objectives: delay vectors $\left\{\overline{D}^f\right\}$; cost vectors $\left\{\overline{C}^f\right\}$
Required resources: caching $S^f_{R_w}$, $S^f_B$; admission $T^f_{R_w}$
ND $= \{\}$; Pheromone matrix $\tau = \tau_{\max}$
for $i = 1 : X_I$ do
    for $k = 1 : X_A$ do
        Ant $k$ constructs scheme $\mathbf{x}_k$ (following Algorithm 2)
    end for
    Update non-dominated content placement scheme in ND
    Update pheromone value following (3.20)
    if  Pheromone value $< \tau_{\min}$ or $> \tau_{\max}$ then
        Set pheromone value to $\tau_{\min}$ or $\tau_{\max}$
    end if
end for
Evaluate performance ($\mathcal{F}(\mathbf{x})$ in (3.19)) for all schemes in ND, and find the scheme $\mathbf{x}$ with the best performance
```

Algorithm 2 Content Placement Scheme Construction

```
Initialize remaining caching and admission resources to $S_{MBS}$, $S_{RSU}$, and $\mathcal{T}_R$; and randomly
order the files
for $f = 1 : F$ do
    for $n = 1 : N_C^f$ do
        if Remaining resources are sufficient for $(f, n)$ then
            Obtain selecting probability following (3.23)
        end if
    end for
    if $q < q_0$, where $q \sim U(0, 1)$ then
        Exploitation: mode with the highest probability
    else
        Exploration: probabilistic mode selection
    end if
    Update remaining resources
end for
Update $\mathbf{x}_k$ by $Local\ search(\mathbf{x}_k)$ (optional)
```

$\eta_{f,n}$ (representing the preference of mode n is chosen by the file f) also depends on the resources (including caching and admission) consumed by this choice. The ratio between the consumed resources and the remaining resources is defined as $RC_{f,n}$, which represents the tightness of selecting mode n on resource constraints.

$$RC_{f,n} = \frac{S_B^f(n)}{RS_{MBS}} + \sum_{w=1}^{W} \left(\frac{S_{R_w}^f(n)}{RS_{R_w}} + \frac{T_{R_w}^f(n)}{RT_{R_w}} \right). \tag{3.21}$$

Then, $\eta_{f,n}$ is calculated as

$$\eta_{f,n} = 1 \Big/ \left[\left(W_D \cdot \overline{D}^f(n) + W_C \cdot \overline{C}^f(n) \right) \cdot RC_{f,n} \right]. \tag{3.22}$$

From (3.22), if the mode achieves better performance (i.e., lower weighted summation of delay and cost) and consumes less resource (i.e., lower $RC_{f,n}$), then $\eta_{f,n}$ will be higher.

Probabilistic decisions are made by ants to construct a content placement scheme. For file f, the ant first finds out its feasible mode set $\mathcal{O}_f$, which are determined by resource constraints, (i.e., the remaining caching and admission resources are sufficient for mode $n \in \mathcal{O}_f$). After that, the probability of selecting feasible mode n is

$$p_{f,n} = \frac{\tau_{f,n}^{\alpha} \cdot \eta_{f,n}^{\beta}}{\sum_{l \in \mathcal{O}_f} \tau_{f,l}^{\alpha} \cdot \eta_{f,l}^{\beta}}, \tag{3.23}$$

Table 3.3: Parameters of system model.

V2B	Carrier Freq.	Bandwidth	Tx Power	Coverage
	2000 MHz	20 MHz	43 dBm	1000 m
Mobility	$V_{\min}$	$V_{\max}$	a	λ
	20 m/s	30 m/s	2 m/s^2	1/3 sec^{-1}

where $\tau_{f,n}$ and $\eta_{f,n}$ are the updated pheromone and heuristic information, respectively, and α and β determine the impact of pheromone and heuristic information on the decision.

It has been proved that, when applied to combinatorial optimization problems, ACO algorithms can achieve the best performance in conjunction with random local search method [76]. Starting from an initial solution, local search method tries to find an optimal solution within the predefined neighborhood of the initial point. Since ACO algorithms perform a rather coarse-grained search for the optimal solution, quality of the solutions can be enhanced with the aid of local search. The initial solution is the content placement scheme produced by each ACO iteration, then it can be locally optimized through local search. Since the pheromone is utilized by following ACO iterations, its update based on the locally optimized solutions will have a long-term impact on the performance.

The proposed ACO-based scheme for solving **P2** is described in Algorithm 1, and the ant constructs content placement scheme following Algorithm 2.

3.6 Numerical Results

In this section, we first evaluate the convergence performance of the proposed scheme. Then, we compare the caching performance, including the average delay and the service cost, of our scheme with the benchmark methods.

3.6.1 Simulation Setup

In our simulation scenario, 2 RSUs are deployed in the coverage of one MBS along the street. For V2R communication, the setting of RSU follows [68], and the transmission rates for each zone are shown in Table 3.2. The configurations of V2B and V2V communication follow the 3GPP standard and [67], respectively. The detailed parameters of V2B communication and vehicle mobility are given in Table 3.3.

Table 3.4: Parameters of files and caching capacities.

	Caching capacity		PF		LF		s
	MBS	RSU	Size	Num.	Size	Num.	
Case1	50	30	5	10	14	5	0.7
Case2	100	40	5	10	22	5	1
Case3	50	40	5	10	18	6	1

The files requested by the vehicles are classified as three sets, i.e., PF and $LF_w, w = 1, 2$. We consider the same data size for files belonging to one set and their packets are encoded by fountain code. To analyze the impact of different file size and caching capacity, we conduct simulation under several cases, as shown in Table 3.4, both file size and caching capacity are in the unit of packet and s is the parameter characterizing the skewness of Zipf-like popularity distribution. These cases represent varying proportions of PF and LF files (by changing the size of LF files) at different levels of caching capacity.

To evaluate the service cost of caching and transmission, we set the price of caching at MBS and RSU to be 0.3 and 0.5, respectively, and the price of transmission to be 0.8, 0.5, and 1.5, corresponding to the MBS, RSU, and backhaul link, respectively. In addition, we set the equal weights of delay and service cost in the objective function, i.e., $W_D = W_C = 0.5$. In our simulation results, we use cost to represent this weighted summation of delay and service cost.

We compare the proposed cooperative content placement scheme (denoted by coop) with two benchmark methods. In noncooperative caching scheme (denoted by noncoop), there is no cooperation between MBS and RSUs, which means a file can be cached at either MBS or RSUs. In greedy caching scheme (denoted by greedy), PF and LF files are greedily cached at MBS and corresponding RSUs based on file popularity, respectively. Then, we compare the caching performance of our scheme with the lower bound achieved by the LP method (denoted by LP). Comparing **P2** and its linear relaxation (**P3**), we can see that the feasible solution set of **P2** is a subset of the feasible solution set of **P3** [77]. Thus, the solution to **P3** can be used as the lower bound for the solution to **P2**.

Linear relaxation can be achieved by replacing (3.18a) with (3.24a). Multiple constraints are combined in (3.24c), where ω_B, ω_{RS}, and ω_{RT} are surrogate multipliers for MBS caching, RSU caching, and RSU access resources, respectively, which are positive real numbers. These multipliers can be obtained through the method proposed in [77], and they can be seen as the shadow price of its related constraint in the relaxation.

$$(\mathbf{P3}):\min_{\{x_{f,n}\}}\left(W_D\cdot\sum_{f=1}^{F}\sum_{n=1}^{N_C^f}\overline{D}^f(n)\cdot x_{f,n}+W_C\cdot\sum_{f=1}^{F}\sum_{n=1}^{N_C^f}\overline{C}^f(n)\cdot x_{f,n}\right)$$

$$\text{s.t.}\begin{cases}\sum_{n=1}^{N_C^f}x_{f,n}=1,x_{f,n_c}\in[0,1],f=1,2,...,F; & (3.24\text{a})\\ \sum_{f=1}^{F}\sum_{n=1}^{N_C^f}\left[\omega_B\cdot s_B^f(n)+\sum_{w=1}^{W}\left(\omega_{RS}\cdot s_{R_w}^f(n)+\omega_{RT}\cdot T_{R_w}^f(n)\right)\right]x_{f,n} & (3.24\text{b})\\ \leq\omega_B S_{MBS}+2\omega_{RS}S_{RSU}+2\omega_{RT}\mathcal{T}_R. & \end{cases}$$

3.6.2 Simulation Results

We first illustrate the convergence of the proposed method. Then, we evaluate the performance of the proposed scheme via both numerical results and Monte Carlo simulation results. Furthermore, the impacts of caching capacity resource allocation, weight parameters, and file set on the performance are analyzed. Caching performance is evaluated by the value of the objective function in **P3**, a lower value reflects a better performance.

Convergence Performance

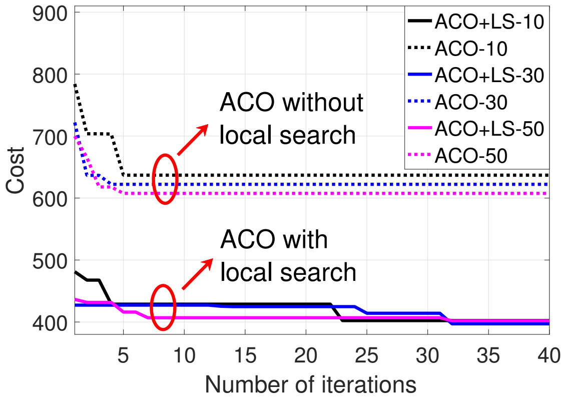

Figure 3.5: Convergence speed.

For the ACO-based cooperative content placement scheme design, the number of ants has an impact on the convergence. We can observe from Fig. 3.5 that how the convergence speed changes with the number of ants, i.e., 10, 30, and 50. Both ACO with and without local search are simulated with file assumption following Case1 in Table 3.4. Since the convergence of ACO is mainly dependent on the number of solutions generated, i.e., $number\ of\ ants * number\ of\ iterations$, more iterations are required if we use fewer ants, as shown in Fig. 3.5. In addition, we can also observe that the ACO algorithm with local search (denoted by ACO+LS) achieves a much better result than ACO without local search, due to the effectiveness of local optimization. In our following simulation, we set the number of ants to be 30 and run the ACO algorithm with local search for 35 iterations, which is sufficient to achieve the convergence.

Impact of Caching Capacity

The total caching capacity is defined as the ratio of total capacity, i.e., the summation of caching capacities at MBS and RSUs, to the total size of PF and LF files. To study how caching capacity affects the performance, we consider both the proportion of different edge caching servers' caching capacity and the amount of total caching capacity. We use the setting of Case2 in Table 3.4.

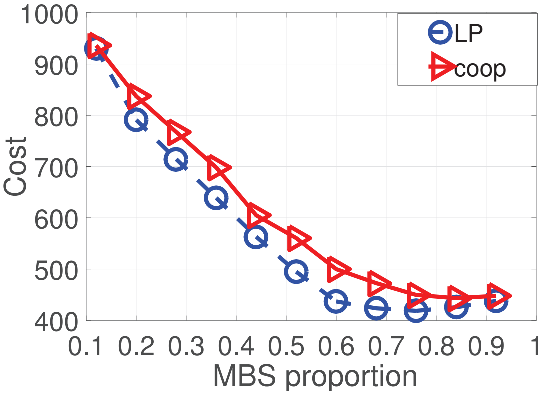

Figure 3.6: Performance changing with MBS caching capacity.

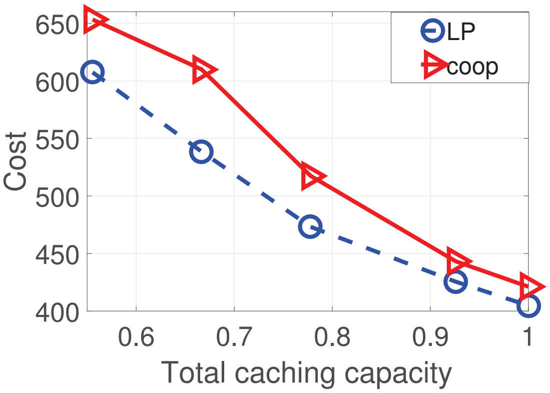

Figure 3.7: Performance changing with total caching capacity.

In Fig. 3.6, the total capacity is fixed at 0.78, and the proportion of MBS varies from 0.05 to 0.9. With an increased proportion of MBS caching capacity, the performance keeps improving until the MBS proportion reaches over 0.7. Due to the limited admission resources of RSU, the bottleneck of RSU caching performance is admission resources when the MBS proportion is under 0.7 (i.e., RSU proportion is over 0.3). In this region, more caching resources should be allocated to the MBS to improve the caching resource utilization efficiency. However, if the MBS proportion is over 0.7, increasing the MBS proportion will cause a slight performance degradation, which is caused by the higher transmission cost of MBS compared with RSU. Thus, the optimal performance can always be achieved at the point where caching and admission resources for one RSU are matched.

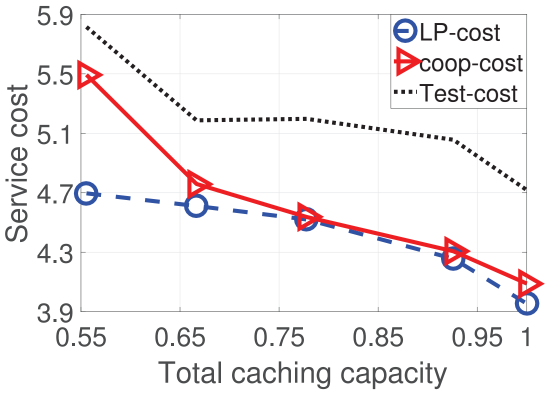

Figure 3.8: Service cost performance changing with caching capacity.

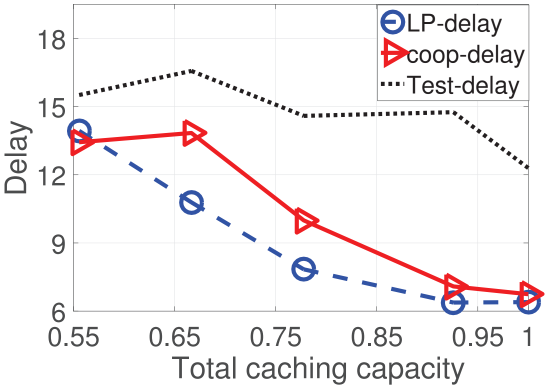

Figure 3.9: Delay performance changing with caching capacity.

We show the impact of total caching capacity in Fig. 3.7. For each point, the performance is the best value that can be achieved by the specific total capacity. When the total capacity grows from 0.56 to 1, the performance improves by 33%. The average service cost (i.e., $(C_C + C_T)$ divided by the data volume of transmitted PF and LF contents) and delay (i.e., $\overline{D}$ divided by the data volume of transmitted LF contents) for the vehicle downloading through V2I are evaluated in Fig. 3.8 and Fig. 3.9 respectively, where we compare the numerical results (coop, LP) and the Monte Carlo simulation results (Test). In both subfigures, the numerical results of the proposed scheme are slightly higher than the lower bound from the LP method. With the increase of caching resources, more files can be precached at MBS and RSUs, hence less data is transmitted through backhaul link, saving both service cost and downloading time. In addition, the Monte Carlo simulation results show similar trends to the numerical results, so we can design the caching scheme according to the numerical analysis.

Impact of Weight Parameters

Since the total cost is a weighted sum of the service cost and delivery delay, the weights influence the preference between different files and caching servers. In Fig. 3.10, we compare how the performance of different methods change with delay weight, keeping $W_C = 0.5$. We use the setting of files and caching capacity as Case3 in Table 3.4. Without

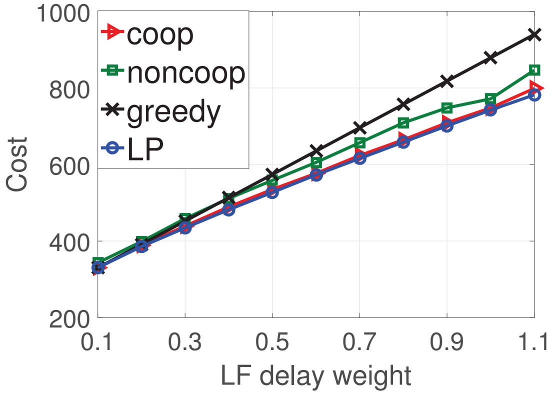

Figure 3.10: Performance changing with delay weight.

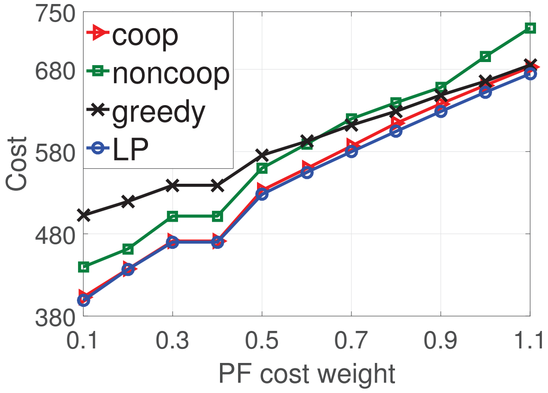

Figure 3.11: Performance changing with cost weight.

cooperative caching between MBS and RSUs for LF files, it leads to a higher cost than that of the cooperative caching method. Since the greedy method prefers to cache the PF files at MBS, it has the largest cost with high delay weight. This is because vehicles have to download the low-popularity LF files, that are not cached by both the RSUs and MBS, through the backhaul link. The service cost weight of PF file also affects the performance, as shown in Fig. 3.11. Since the greedy method has a fixed priority of caching, its performance is degraded when weight parameters change. However, the proposed method can adapt to variable weights, which keeps the small gap with performance lower bound obtained by the LP method.

Impact of File Parameters

In order to analyze the impact of file popularity on performance, in Fig. 3.12, we plot the performance as a function of parameter s of popularity distribution in (3.13), using the setting of files and caching capacity as Case2 in Table 3.4. We observe that the performance of caching improves as s increases. This is because popularity distribution becomes more skewed towards higher popularity contents, and these contents are given more priority during the resource allocation. If the distribution becomes more skewed, the resources allocated to high popular contents will be utilized more efficiently, which can improve the performance. From Fig. 3.12, we also obtain that the performance achieved by the proposed ACO-based scheme and the lower bound from the LP method is very close, indicating that the ACO-based scheme can achieve the near-optimal solution.

To evaluate the scalability of the content placement schemes, we design the content

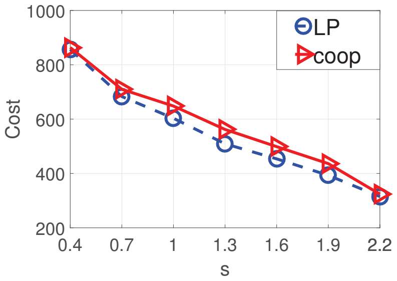

Figure 3.12: Performance changing with file popularity skewness s.

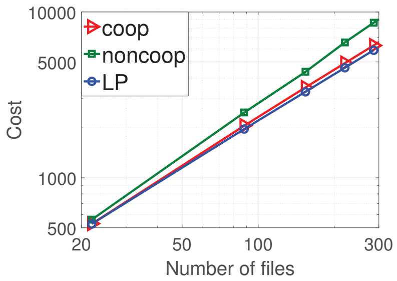

Figure 3.13: Cost changing with number of files.

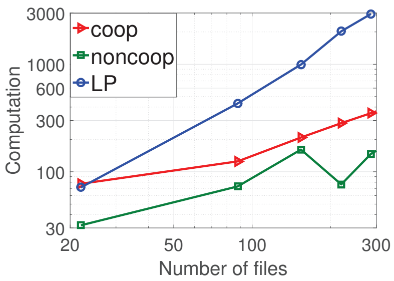

Figure 3.14: Computation complexity changing with number of files.

placement scheme for an increased number of files. On the basis of Case3 in Table 3.4, we scale up the number of files and caching capacities, and compare different content placement schemes in Fig. 3.13. Compared with the LP method, the proposed cooperative scheme always achieves near-optimal performance, while the performance gap for the noncooperative method is increased with larger file set. This means the noncooperative method suffers performance degradation for large file sets. Without cooperation between the MBS and RSUs, there are less feasible content placement schemes for the noncooperative method, which leads to lower computation complexity than the cooperative method, as shown in Fig. 3.14. For the cooperative scheme, due to the polynomial relationship between computation complexity and the number of files, its efficiency and scalability are verified.

3.7 Summary

In this chapter, based on a multi-tier HVNet, we have proposed a cooperative caching scheme to improve content delivery services for connected vehicles. Considering the differential delivery requirements for location-based and popular contents, a cooperative content placement scheme has been devised to reduce both the transmission delay and the service cost. We have provided the theoretical analysis of the content transmission delay, which can be generalized to content delivery services with different QoS requirements or the network with different access point deployment. We have evaluated the convergence speed of the proposed scheme, and demonstrated that it can effectively improve the overall performance. Besides, the robustness of the scheme has been validated under various parameter settings.

Chapter 4

Two-Level Adaptive Resource Allocation for Diverse Safety Message Transmissions in Vehicular Networks

In this chapter, we propose a two-level adaptive resource allocation (TARA) framework to support vehicular safety message transmissions. In particular, three types of safety message are considered in urban vehicular networks, i.e., the event-triggered message for urgent condition warning, the periodic message for vehicular status notification, and the message for environmental perception. Roadside units are deployed for network management, and thus messages can be transmitted through either vehicle-to-infrastructure or vehicle-to-vehicle connections. To satisfy the requirements of different message transmissions, the proposed TARA framework consists of a group-level resource reservation module and a vehicle-level resource allocation module. Particularly, the resource reservation module is designed to allocate resources to support different types of message transmission for each vehicle group at the first level, and the group is formed by a set of neighboring vehicles. To learn the implicit relation between the resource demand and message transmission requests, a supervised learning model is devised in the resource reservation module, where to obtain the training data we further propose a sequential resource allocation (SRA) scheme. Based on historical network information, the SRA scheme offline optimizes the allocation of sensing resources (i.e., choosing vehicles to provide perception data) and communication resources. With the resource reservation result for each group, the vehicle-level resource allocation module is then devised to distribute specific resources for each vehicle to satisfy the differential requirements in real time. Extensive simulation results are provided to demonstrate the effectiveness of the proposed TARA framework in terms of the high successful reception ratio and low latency for message transmissions, and the high quality of collective environmental perception.

4.1 Background and Motivations

To enhance the driving safety of connected vehicles, cooperative awareness messages (CAMs) are required to be periodically exchanged among vehicles to report their real-time driving status, e.g., ID, position, speed, and direction [45]. To further improve the intelligence of autonomous vehicles, the collective perception (CP) service has been proposed by the ETSI group [46]. By provisioning CP services, vehicles can generate, send, and receive collective perception messages (CPMs) to share the perceived environmental information. Besides, if a vehicle detects an accident, event-triggered messages (i.e., decentralized environmental notification messages (DENMs)) should be generated and sent out to warn its neighboring vehicles about the accident [44].

We consider the dissemination of DENMs, CAMs, and CPMs among connected vehicles in an urban scenario. One promising technique to enable vehicular networking is the C-V2X communication, which can use the existing cellular infrastructure to relay the packets and achieve a centralized control [78]. Particularly, via V2V connections, messages can be exchanged among the vehicles within the V2V communication range. However, due to the antenna height limitation, V2V connections can be easily blocked by obstacles, such as buildings, trucks, and buses. V2I technologies are leveraged to deal with the NLOS link condition, where RSUs relay messages for the blocked vehicles [79, 80]. Note that, to support CP services, the generated CPMs need to be processed by the RSU before being broadcasted to vehicles. Hence, CPMs should be transmitted via V2I connections, while DENMs and CAMs can be transmitted through V2V or V2I connections.

To improve the quality of message transmission, wireless resources should be properly allocated to avoid transmission collisions, considering uplink (UL) and downlink (DL) of V2I transmissions, and V2V transmissions. Moreover, to provision CP services, not only wireless resources but also sensing resources are required. The sensing resources are allocated by choosing a subset of sensing-enabled vehicles as sensing data providers, named CP seed vehicles. The sensing coverage of the chosen vehicles can be overlapped, leading to information redundancy among the generated CPMs. Therefore, the RSU aggregates original CPMs into one integrated CPM, and then broadcasts out. To evaluate the QoS for CPM transmissions, the integrated sensing coverage and accuracy are measured. In contrast, the reliability, i.e., successful packet reception, and delay are critical to DENM and CAM transmissions. As the resources are shared by multiple services, the differentiated requirements and priorities should be considered simultaneously when making decisions on the resource allocation.

In this chapter, the C-V2X based urban vehicular network is considered, where RSUs with cellular technology serve as network controllers, making resource allocation decisions for the vehicles. To guarantee the reliability and minimize the latency of event-triggered DENM transmissions and periodic CAM transmissions, wireless resource allocation decisions are made, including the V2I/V2V transmission mode selection and resource block (RB) allocation. Besides, to support CP services, both sensing and wireless resources are required for CPM generation and transmission, in which the selection of CP seed vehicles is constrained by the availability of wireless resources. Thus, the quality of CP service can be optimized via joint CP seed vehicle selection and wireless resource allocation. It is challenging to solve this multi-dimensional resource allocation problem, i.e., making the optimal decisions on the V2I/V2V transmission mode selection, RB allocation, and CP seed vehicle selection, since the decisions are inter-coupled in three aspects: 1) the allocation of the sensing and wireless resources for CP services; 2) the allocation of the wireless resources for transmitting DENMs, CAMs, and CPMs; and 3) the V2V/V2I mode selection for the DENM and CAM transmissions.

Considering the vehicular dynamics in terms of network topology and message transmission requests, an adaptive resource allocation scheme is necessary. To address the aforementioned challenges, we propose a TARA framework for safety message transmissions, consisting of a group-level resource reservation module and a vehicle-level resource allocation module, as shown in Fig. 4.1. The amount of resources that required by each type of message transmission is determined by the resource reservation module. It also guarantees that the overloaded transmission requests from one type of message will not affect the others. The reservation strategy should be designed to optimize the overall QoS satisfaction for all the types of message transmission. Based on the reserved resources for different types of message, the resource allocation module is devised to satisfy each vehicle's individual QoS requirements. The main contributions are summarized as follows:

1. *Multi-dimensional resource management* – We study the joint management of multi-dimensional resources to support safety message transmissions, which is of significant importance for future vehicular networks. By analyzing the impact of message transmission priorities and network conditions (e.g., resource availability and link reliability), wireless resources and sensing resources can be jointly allocated by the proposed TARA framework to satisfy the differential QoS requirements, the process of which is shown in Fig. 4.1;

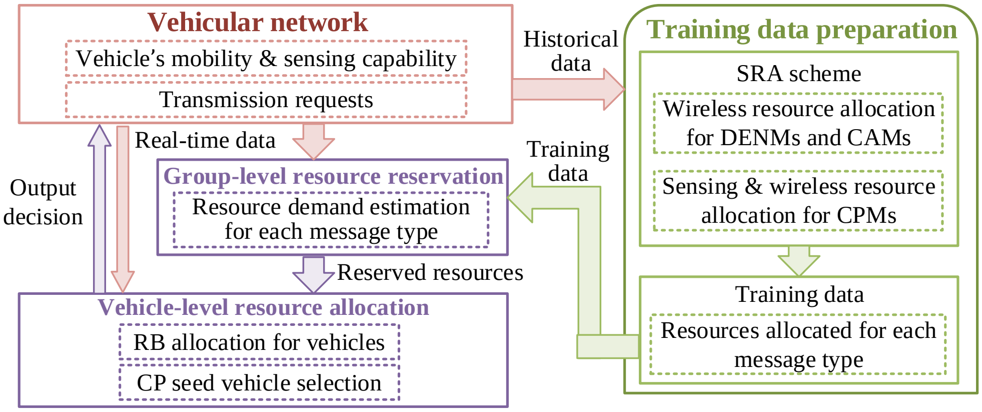
Vehicular network
Vehicle's mobility & sensing capability
Transmission requests
Historical data
Training data preparation
SRA scheme
Wireless resource allocation for DENMs and CAMs
Sensing & wireless resource allocation for CPMs
Real-time data
Training data
Output decision
Group-level resource reservation
Resource demand estimation for each message type
Reserved resources
Training data
Resources allocated for each message type
Vehicle-level resource allocation
RB allocation for vehicles
CP seed vehicle selection

Figure 4.1: An illustration of the TARA framework.

2. *SRA scheme* – Based on the historical information, the allocation decisions of wireless and sensing resources can be made by the SRA scheme according to the transmission priority of different messages. Particularly, for DENM and CAM transmissions, vehicles are sorted by their potential gains obtained by V2I transmission, to guide the selection of V2I/V2V mode and RB allocation. For CPM transmissions, the CP seed vehicles are selected iteratively, based on vehicles' sensing performance;

3. *Enabling efficient decisions in real time* – Since the SRA scheme needs to sequentially make decisions based on message transmission priorities, it is unable to keep pace with the dynamic vehicular environment. Thus, we propose the TARA framework to achieve real-time decision-making, consisting of a group-level resource reservation module and a vehicle-level resource allocation module. The resource reservation module reserves the resources to support different types of message transmission for each vehicle group, i.e., a set of neighboring vehicles, which allows the resource allocation module to perform in parallel for different groups and different message types. The supervised learning-based technique is applied in the resource reservation module to learn the implicit relation between the resource demand and message transmission requests, which is offline trained based on the data provided by the SRA scheme. With the reservation result, resources are allocated to each vehicle by the vehicle-level resource allocation module, which can be achieved in real time with a low time complexity.

The remainder of this chapter is organized as follows. We describe the system model in Section 4.2. We give an overview of the developed TARA framework in Section 4.3. The

SRA scheme is devised in Section 4.4. To further improve the time efficiency of resource allocation, a group-level resource reservation module and a vehicle-level resource allocation module are designed in Section 4.5. Simulation results are carried out in Section 4.6 to demonstrate the performance of the proposed schemes. Finally, conclusions are drawn in Section 4.7.

Table 4.1: Summary of mathematical symbols.

Symbols	**Definition**
$A_{i,j,m}$	Number of unoccupied RBs in $S_{i,m}$ available for CPM transmitted by $V_{i,j}$
$B_{i,m}$	Number of groups that contend $S_{i,m}$ for CPM transmissions
C	Threshold of sensing coverage
$C_{i,j}^{P}$ (C_{R}^{P})	Number of supported CPM upload transmissions for group $V_{i,j}$ (the RSU)
D_a (D_r)	Number of RBs available (required) for V2I DL transmissions
d	Distance between the sensing block center and the SCV
d_b	Critical distance of NLOS model
d_t (d_r)	Distance between the transmitting (receiving) vehicle and the intersection-center
d_w	Distance between the transmitting vehicle and the road-side
e (e_0)	SCV's sensing error for a given block (Penalty error for not sensed blocks)
G_i	Number of vehicle groups allocated with time segment T_i
$G_{i,j}^{m}$	Potential packet loss under the V2V mode of $V_{i,j}^{m}$
G_r	Receiver's antenna gain (dB)
$I_{i,j}$ ($D_{i,j}$)	Number of required TDSs (requests transmitted in TDSs) for $V_{i,j}$
$M_{i,j}^{m}$ ($S_{i,j}^{m}$)	Number of vehicles in the target (successful) receiver set of $V_{i,j}^{m}$
$N_{i,j}^{D}$ ($N_{i,j}^{A}$)	Number of V2I UL RBs allocated to the vehicles requesting for DENM (CAM)
n_{NLOS}	NLOS path loss exponent
PL_L (PL_N)	Path loss of the LOS (NLOS) case (dB)
P_{RX} (P_{TX})	The receiving (transmitting) signal power (dBm)
R	Sensing range of vehicular sensor
$R_{i,j}$ ($R_{i,j}^{P}$)	Overall number of DENM and CAM (Number of CPM) requests raised by $V_{i,j}$
S^{D} (S^{A}, S^{PU}, S^{PD})	The numbers of required RBs for DENM (CAM, CPM upload, CPM download)
$S_{i,m}$	The m-th sub-frame in time segment T_i
T_i (T_i^{n})	The i-th time segment (The i-th time segment of n-th CAM period)

T^P ($T^P_{i,j}$, T^P_R)	CPM period (Required CPM period for $V_{i,j}$ or for the RSU)
t_i	Number of sub-frames in time segment T_i
$V_{i,j}$ ($V^m_{i,j}$)	The j-th vehicle group with time segment T_i (The m-th vehicle in $V_{i,j}$)
$V^U_{i,j}$	Set of transmissions not assigned with V2I mode in $V_{i,j}$
V^D (V^{D+}, V^{D0})	Set of vehicles requesting DENM transmissions (Subset of V^D with positive or zero $G^m_{i,j}$)
V^A (V^{A+}, V^{A0})	Set of vehicles requesting CAM transmissions (Subset of V^A with positive or zero $G^m_{i,j}$)
V_S (V^+_S)	Sorted set of vehicles requesting message transmissions (Subset of V^S with positive $G^m_{i,j}$)
w_d (w_t)	Unit sensing error caused by distance (latency of CPM update)
w_r	Width of the receiving street
X	Number of V2I packets can be transmitted in one TDS
x	Distance between receiving and transmitting vehicles
λ	wavelength of transmission frequency
ρ_i	The frequency split ratio for time segment T_i

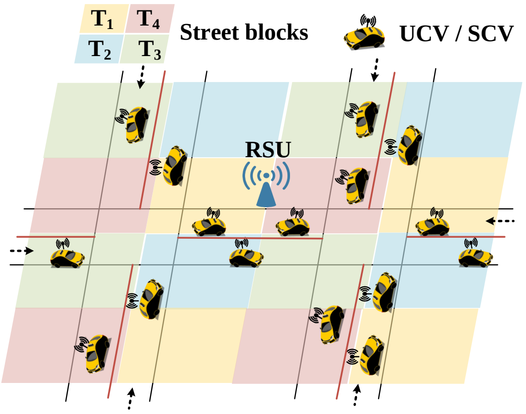

Figure 4.2: An illustration of vehicular network.

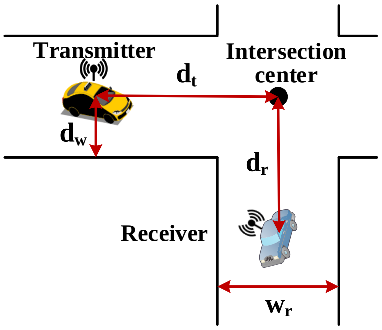

Figure 4.3: An illustration of NLOS case.

4.2 System Model

4.2.1 Network Model

As shown in Fig. 4.2, we consider a vehicular network in the urban scenario, including connected vehicles (CVs) and RSUs. According to the capability of sensing, the vehicles are classified into two types, namely the sensing-enabled CVs (SCVs) and CVs without sensing capability (UCVs). With vehicular sensors, an SCV can observe the environment within its sensing range and generate the sensing report, which includes the information of detected objects. The RSU can not only relay the packets, but also make resource allocation decisions for vehicles. A network management controller is deployed at the RSU, to collect the information (e.g., the vehicle's location, request, and sensing reports) and make decisions accordingly. Without loss of generality, we focus on the coverage area of one RSU. In this chapter, vehicles are clustered as non-overlapped groups. As shown in Fig. 4.2, vehicles in one square block constitute a group. For the sake of simplicity, we assume the block's side length equals the V2V communication range, hence there is no V2V connection between vehicles within any two nonadjacent blocks. Important mathematical symbols are listed in Table 4.1. In this chapter, the subscripts i, j, m, and R represent the i-th time segment, the j-th vehicle group, the m-th sub-frame, and the RSU, respectively. The superscripts A, D, P, PU, PD, and m represent the CAM, DENM, CPM, CPM upload, CPM download, and the m-th vehicle, respectively.

Communication Model

The path loss between two vehicles are modeled separately in line-of-sight (LOS) [81] and NLOS [33] cases, respectively. For LOS cases, the path loss is

$$PL_L\left(x\right) = 20\log_{10}(\frac{4\pi x}{\lambda}), \tag{4.1}$$

where x is the distance between receiving and transmitting vehicles and λ is wavelength of transmission frequency. As shown in Fig. 4.3, the path loss for NLOS cases is

$$PL_N\left(d_t, d_r, d_w\right) = 3.75 + \begin{cases} 10\log_{10}\left(\left(\frac{{d_t}^{0.957}}{(d_w w_r)^{0.81}}\frac{4\pi d_r}{\lambda}\right)^{n_{NLOS}}\right), & \text{if } d_r \leq d_b, \\ 10\log_{10}\left(\left(\frac{{d_t}^{0.957}}{(d_w w_r)^{0.81}}\frac{4\pi {d_r}^2}{\lambda d_b}\right)^{n_{NLOS}}\right), & \text{if } d_r > d_b, \end{cases} \tag{4.2}$$

where d_t (d_r) is the distance between the transmitting (receiving) vehicle and the intersection-center, d_w is the distance between the transmitting vehicle and the road-side, w_r is the width of the receiving street, d_b is a reference parameter called critical distance, and n_{NLOS} is the NLOS path loss exponent. Note that, the path loss model for NLOS cases can be applied to intersections with different corner angles, in addition to the 90-degree corner shown in Fig. 4.3 [82, 83].

The receiving signal power P_{RX} (dBm) is obtained based on the transmit power P_{TX} (dBm), path loss, and receiver's antenna gain G_r (dB),

$$P_{RX} = P_{TX} + G_r - \begin{cases} PL_L\left(x\right), & \text{LOS}, \\ PL_N\left(d_t, d_r, d_w\right), & \text{NLOS}. \end{cases} \tag{4.3}$$

The message can be successfully decoded when P_{RX} is larger than the threshold of decoding received signal strength.

Half-Duplex Problem of Connected Vehicles

The target receiver set for each transmitting vehicle consists of vehicles within the targeting message dissemination range, which is set as the V2V communication range in this chapter. Taking into account the half-duplex feature of CVs, they are not capable of receiving and transmitting packets simultaneously. In this chapter, time is partitioned into sub-frames with constant duration, which is applied as the unit of time in resource allocation. If a sub-frame is allocated for a source vehicle to transmit its packets through V2V connections, this sub-frame should not be reused by the target receivers for packet transmitting (V2V or V2I UL transmission). Hence, as shown in Fig. 4.2, to deal with the half-duplex issue, the adjacent four vehicle groups are allocated with isolated time segments (T_i, $i = 1, ..., 4$). Although the vehicles in nonadjacent blocks are allocated with the same time segment, there is no interference among them due to spatial separation. Each time segment T_i consists of t_i sub-frames.

4.2.2 Wireless Resource Pool

In this chapter, the V2V, V2I UL, and V2I DL transmissions are assumed to share a frequency band of 10 MHz (from 5.89 to 5.9 GHz), which is divided into 50 sub-channels. For the time domain, one sub-frame is defined as the duration of 1 ms [84]. As shown in Fig. 4.4, the wireless resource pool of T_i segment is represented by the combination of $50 \cdot t_i$ RBs, each of which represents the usage of 200 kHz bandwidth for 1 ms. Hence, the wireless resource allocation can be performed via allocating RBs to vehicles for packet transmissions. The numbers of required RBs for DENM, CAM, CPM upload (packet transmission from

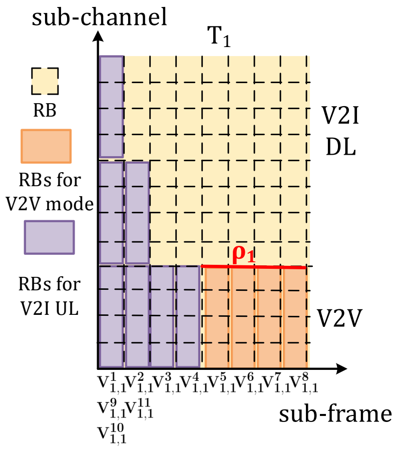
sub-channel
T$_1$
RB
RBs for V2V mode
RBs for V2I UL
V2I DL
ρ$_1$
V2V
$V_{1,1}^1$ $V_{1,1}^2$ $V_{1,1}^3$ $V_{1,1}^4$ $V_{1,1}^5$ $V_{1,1}^6$ $V_{1,1}^7$ $V_{1,1}^8$
$V_{1,1}^9$ $V_{1,1}^{11}$
$V_{1,1}^{10}$
sub-frame

Figure 4.4: An illustration of resource allocation within one vehicle group.

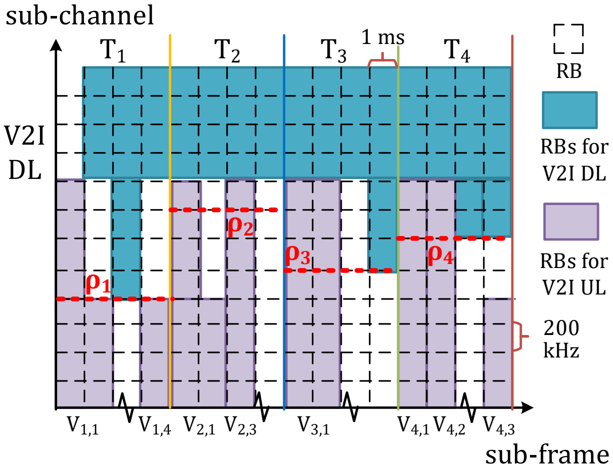

Figure 4.5: An illustration of resource allocation for V2I connections within the RSU.

SCVs to the RSU), and CPM download (the integrated packet transmission from the RSU to vehicles) are denoted by S^D, S^A, S^{PU}, and S^{PD}, respectively. As shown in Fig. 4.4, in order to prevent the collisions for packet reception, a ratio is defined to partition the frequency band between V2I DL and V2V transmissions, denoted by ρ_i $(i = 1, ..., 4)$ for the vehicle group with time segment T_i. Specifically, if one sub-frame is allocated for V2V transmission in segment T_i, the first $50 \cdot \rho_i$ sub-channels are available for V2V, while the remaining sub-channels are available for V2I DL transmissions. Otherwise, the 50 sub-channels are available for V2I DL transmissions as long as they are not occupied by V2I UL.

There are G_i vehicle groups being allocated with the same time segment, T_i, and the j-th group of which is denoted by $V_{i,j}$, consisting of $K_{i,j}$ vehicles, as shown in Fig. 4.4. The m-th vehicle in $V_{i,j}$ is denoted by $V_{i,j}^m$, $m = 1, ..., K_{i,j}$. To deal with the half-duplex problem and avoid the packet transmission collisions, the following principles should be followed by the wireless resource allocation for vehicles:

1. *V2V and V2I UL for each vehicle group* – Considering the half-duplex feature, separate sub-frames are allocated to vehicles transmitting packets via V2V connections. The number of sub-frames required by one vehicle can be calculated based on ρ_i and the size of the packet. In this chapter, ρ_i is determined to ensure that each V2V transmission can be accomplished in one sub-frame. For a vehicle group, in addition

to the sub-frames assigned for V2V transmissions, the unoccupied sub-frames can be allocated for V2I UL transmissions. In Fig. 4.4, the wireless resource allocation for both V2V and V2I UL transmissions in the vehicle group $V_{1,1}$ is illustrated as an example, where multiple vehicles can use the distinct RBs in the same sub-frame for V2I UL transmissions;

2. *V2I UL for vehicle groups with the same time segment* – Within the coverage of one RSU, all the vehicles under V2I mode transmit their packets to the RSU. Thus, V2I UL collision may happen among the G_i vehicle groups with the same time segment, T_i. To avoid the collision, these G_i groups should be considered simultaneously when making resource allocation decisions on V2I UL transmissions. As shown in Fig. 4.5, for the vehicles from $V_{i,j}$, $j = 1, 2, ..., G_i$, the V2I RBs allocated to them shall not overlap with each other;

3. *V2I DL for vehicles under the RSU* – V2I DL RBs are required for relaying the packets transmitted via V2I mode, such as the DENM, CAM, and integrated CPM packets. Since the V2I DL and V2V transmissions are assigned with separate frequency bands, there is no interference among them. When making V2I DL RB allocation decisions, only the V2I UL transmissions are considered to avoid potential collisions. In each time segment, the RBs beyond the V2V frequency bound are available for V2I DL transmission, as long as they are not allocated for V2I UL transmissions. The V2I DL resource pool consists of the available RBs from the four time segments, which is shared by all the vehicles within the RSU coverage, as shown in Fig. 4.5. For one V2I mode packet, its assigned sub-frames for DL transmission have to be after that for UL.

4.2.3 QoS for Safety Message Transmissions

To evaluate the performance of message transmissions, we consider differentiated metrics for multifarious messages:

1. *DENM and CAM* – Both latency and reliability are critical to DENM and CAM transmissions. The transmission delay of one packet is measured as the time elapsed between it being transmitted and received. For the reliability, the successful reception proportion is measured;

2. *CPM* – The performance of CPM transmission is evaluated by the quality of CP services. For one SCV, its sensing coverage area is a circular disk, the radius of which is the sensing range R. Within the sensing range, objects can be detected, but

the sensing accuracy degrades with the distance between the object and the SCV. Given the set of CP seed vehicles, the integrated sensing accuracy can be calculated by the RSU according to the contents, transmission latency, and update frequency of CPMs. On the other hand, the coverage rate, i.e., the proportion of the perceived street area to the overall street area, is also evaluated as one performance metric of CP services. Given a threshold of the coverage rate, the selection of CP seed vehicles and wireless resource allocation are performed to guarantee the coverage rate and achieve the optimal integrated sensing accuracy.

4.3 Overview of the TARA framework

In this chapter, the requests of the DENM and/or CAM transmission are sent by vehicles, and then the RSU allocates a specific bunch of RBs accordingly. To support CP services, the decisions on both sensing and wireless resource allocation are also made by the RSU. We propose the TARA framework, including a group-level resource reservation module and a vehicle-level resource allocation module, to make resource allocation decisions. In addition, data preparation is also needed to train the learning-based resource reservation model offline, as shown in Fig. 4.1.

4.3.1 Training Data Preparation

To prepare training data for the learning-based resource reservation model, we propose the SRA scheme to achieve optimal resource allocation decisions, based on the historical information on vehicle's mobility, e.g., location and speed, sensing capability, and transmission requests. To support CP services and transmissions of DENMs and CAMs, decisions are made for sensing and wireless resources, including the RBs allocated for vehicles and the selected CP seed vehicle set. With the proposed SRA scheme, the number of RBs allocated to each message type can be obtained as training data, which indicates the underlying mapping relationship between the allocated resources and message transmission requests.

The procedure of the SRA scheme is shown in Fig. 4.6. Considering the differential priorities of message transmissions, the SRA scheme is devised to firstly allocate the resources for the DENM and CAM, following by the CPM resource allocation procedures. Recall that V2I DL resource pool is determined by the V2I UL resource allocation results and the frequency partition ratio ρ_i. Therefore, the V2I DL RBs are allocated after performing V2I UL RB allocation procedures. Considering the wireless resources required to transmit one packet, both V2I UL and DL resources should be allocated for the V2I mode, which is

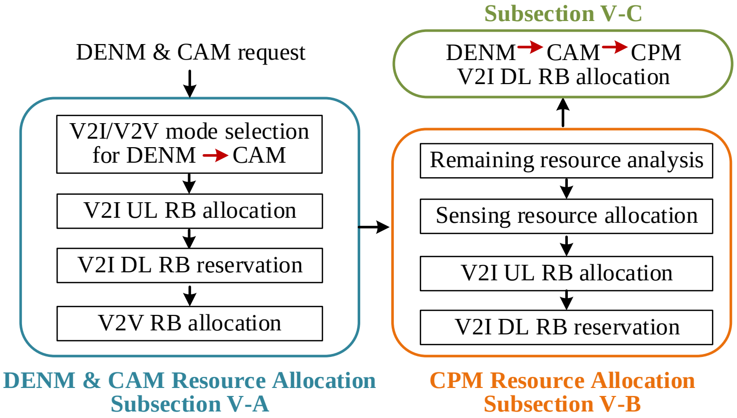

Figure 4.6: Sequential resource allocation procedures.

double of the resources required by the V2V mode. Therefore, to make efficient resource allocation decisions, the V2I mode is selected for one transmission only when the V2V mode cannot meet its requirement. In terms of the V2I mode selection for DENM and CAM transmissions, the vehicle with a stronger demand on the V2I resources has a higher priority. Meanwhile, to support CP services, wireless and sensing resources are allocated through: 1) analyzing the CPM upload capacity available for each vehicle group based on the RB allocation results of DENM and CAM; and 2) greedily selecting the CP seed vehicles based on their sensing gains. Details of the SRA scheme are given in Section 4.4.

4.3.2 Resource Reservation and Resource Allocation Modules

Based on the decisions made by the SRA scheme, we can obtain the number of RBs allocated for message transmissions. In light of the relationship between the number of allocated RBs and message transmission requests, we leverage the two-level resource management strategy to enable a parallel decision-making for vehicles requesting different messages. To make real-time decisions for different types of requests, we design the group-level resource reservation module and the vehicle-level resource allocation module, as shown in Fig. 4.1 and elaborated in Section 4.5. The learning-based resource reservation model is trained based on the wireless resource allocation results achieved by the SRA scheme. Based on the network information, the resource reservation module distributes resources to support the transmission of different types of message for each group. With the reserved resources, RBs are allocated to each vehicle by the resource allocation module. In addition, the selection of CP seed vehicles and required RBs for CPM transmissions are also determined by the resource allocation module.

4.4 Design of SRA

In this section, we present the designed SRA scheme to achieve the optimal resource allocation result. In this chapter, to adapt to environmental dynamics in terms of the vehicle mobility and event-triggered DENM requests, the resource allocation decisions are updated every CAM period, i.e., 100 ms [45]. One CAM period is divided into four time segments, each of which consists of 25 sub-frames, i.e., $t_i = 25$ for $i = 1, ..., 4$.

4.4.1 Resource Allocation for DENM and CAM Transmissions

At each CAM period, the RSU allocates wireless resources for vehicles requesting DENM and CAM transmissions, which are denoted by two sets V^D and V^A, respectively. Firstly, the RSU collects the vehicles' location information and obtains the target receiver set for each vehicle $V_{i,j}^m$. Considering the potential NLOS link conditions between $V_{i,j}^m$ and its $M_{i,j}^m$ target receivers, the number of vehicles that can successfully receive the packet via V2V connections is denoted by $S_{i,j}^m$, where $S_{i,j}^m \leq M_{i,j}^m$. For each transmission, the V2I mode is selected only when any of the following events happen:

1. *Sub-frame insufficiency* – Within one vehicle group $V_{i,j}$, the V2V transmissions request separate sub-frames to avoid the collision. In the case of pure V2V mode, i.e., all vehicles transmit packets via V2V connections, at least $R_{i,j}$ sub-frames are required, where $R_{i,j}$ denotes the overall number of DENM and CAM requests raised by $V_{i,j}$. If $R_{i,j} > t_i$, the sub-frames are insufficient for $V_{i,j}$ in the pure V2V mode. Recall that multiple packets can be transmitted in one sub-frame under the V2I mode. The sub-frame insufficiency issue can be mitigated through selecting the V2I mode for specific packet transmissions;

2. *NLOS condition* – If the NLOS link condition encounters between transmitting and receiving vehicles, the V2V connection based packet transmission may be blocked. To improve the reliability, the NLOS affected packets should be relayed by the RSU via V2I connections. For each transmitting vehicle $V_{i,j}^m$, its potential packet loss under the V2V mode can be estimated by the RSU, denoted by $G_{i,j}^m = M_{i,j}^m - S_{i,j}^m$. As the packet loss can be avoided via selecting the V2I mode, $G_{i,j}^m$ is defined as the potential V2I selecting gain for $V_{i,j}^m$, a larger value of which indicates a higher priority of utilizing V2I resources.

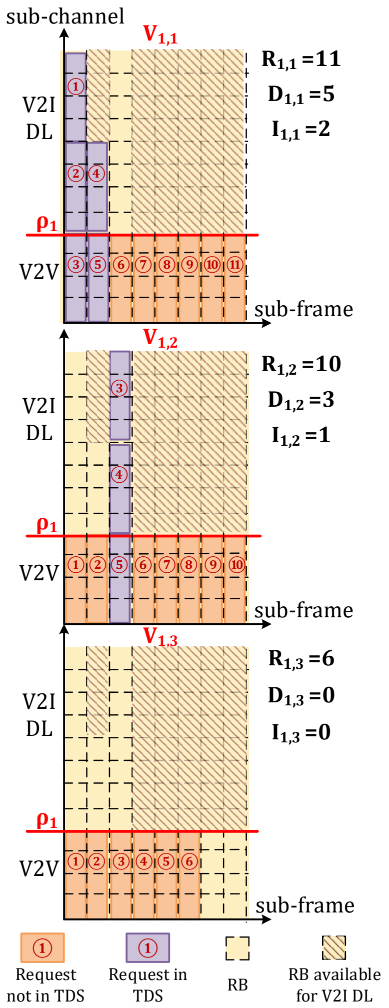

Figure 4.7: An illustration of resource allocation for insufficient sub-frame case.

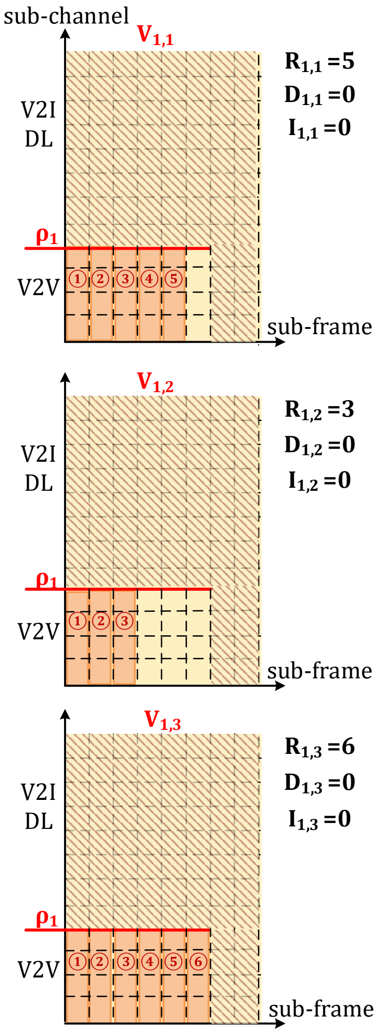

Figure 4.8: An illustration of resource allocation for sufficient sub-frame case.

The resource allocation can be operated by following steps:

1. *Determine vehicles' priorities* – Considering the street layout and the receiving signal power calculated based on Eq. (4.3), the value of $G_{i,j}^{m}$ can be obtained for each vehicle in V^{D} and V^{A}. The set of vehicles requesting DENM (CAM) transmissions with positive $G_{i,j}^{m}$, i.e., packet loss will occur if the vehicle is assigned with the V2V mode, is denoted by V^{D+} (V^{A+}), in which the vehicles are sorted in descending order by $G_{i,j}^{m}$. Likewise, the sets of vehicles with $G_{i,j}^{m} = 0$ are denoted by V^{D0} and V^{A0}, respectively. Due to the higher priority of DENM requests, the sorted set is recombined as $V_S = \left\{V^{D+}, V^{A+}, V^{D0}, V^{A0}\right\}$, and its subset consisting of vehicles with positive gain is defined by $V_S^{+} = \left\{V^{D+}, V^{A+}\right\}$. When assigning the V2I mode for vehicles, the vehicles' priorities are indicated by their orders in V_S or V_S^{+};

2. *Calculate the required number of transmitting dominant sub-frames* – For each sub-frame, the RBs can be allocated for V2V, V2I UL, and V2I DL transmissions, in which the RBs for V2V and V2I UL transmissions always occupy lower frequency band as shown in Fig. 4.4. To reserve more RBs for V2I DL transmissions, the V2I UL should occupy the RBs within the V2V frequency band, unless the group $V_{i,j}$ suffers from the sub-frame insufficiency. If the proportion of transmitting RBs is larger than ρ_i, the sub-frame is defined as a transmitting dominant sub-frame (TDS). An example of resource allocation for T_1 segment is illustrated in Fig. 4.7, where $G_1 = 3$ and $t_1 = 8$. Since $R_{1,1}$ $(R_{1,2}) > t_1$ in the case given in Fig. 4.7, the sub-frames are insufficient for $V_{1,1}$ and $V_{1,2}$ under the pure V2V mode. Thus, 5 and 3 requests are required to be transmitted in TDSs for $V_{1,1}$ and $V_{1,2}$, respectively. For vehicle group $V_{i,j}$, since one TDS can be shared by at most X $(X > 0)$ V2I UL transmissions, the required number of TDSs ($I_{i,j}$) and the number of requests transmitted in TDSs ($D_{i,j}$) can be calculated by

$$I_{i,j} = \lceil \max\left(R_{i,j} - t_i, 0\right)/(X-1) \rceil\,, \tag{4.4}$$

$$D_{i,j} = \max\left(R_{i,j} - t_i + I_{i,j}, 0\right). \tag{4.5}$$

   Hence, the RSU assigns the V2I mode for $D_{i,j}$ transmission requests in $V_{i,j}$, according to their priorities, i.e., the orders in V_S. Then, the set of transmissions not assigned with V2I mode is denoted by $V_{i,j}^{U}$;

3. *Assign sub-frames for packet transmission* – In segment T_i, the first $I_{i,1}$ sub-frames are allocated to $V_{i,1}$ as its TDSs, the following $I_{i,2}$ sub-frames are allocated as the TDSs for $V_{i,2}$, and so forth. In $V_{i,j}$, these $I_{i,j}$ TDSs are assigned for the $D_{i,j}$ transmissions in the V2I mode. For the remaining transmissions in $V_{i,j}^{U}$, each is assigned with

one specific sub-frame, always starting by occupying the first available sub-frame in T_i. Two examples of sub-frame assignment for the three groups in segment T_1 are illustrated in Fig. 4.7 and Fig. 4.8, considering the cases of sufficient and insufficient sub-frame, respectively;

4. *Determine the available RB pool for V2I DL transmissions* – Based on the sub-frame assignment, the number of available RBs for V2I DL transmission is determined, denoted by D_a. The available V2I DL RBs for T_1 segment are displayed in Fig. 4.8. Notice that, the RBs below the V2V frequency band can be used by V2I DL transmission only in the sub-frames unoccupied by any groups in the segment, as the last two sub-frames shown in Fig. 4.8. Meanwhile, for the transmissions already assigned with V2I modes, the overall number of required V2I DL RBs can be obtained as well, denoted by D_r;

5. *Select V2I modes for NLOS condition* – Since the V2I UL RBs in segment T_i are shared by G_i groups, the requests in $V_{i,j}^U$, $j = 1, ..., G_i$, with positive $G_{i,j}^m$ are jointly considered and assigned with V2I modes based on their orders in V_S^+ until the resources are depleted, i.e., $D_r \geq D_a$. In each iteration, the required V2I UL RBs are allocated to the selected transmission in its assigned sub-frame, if the RBs are not allocated for V2I UL transmission yet. Otherwise, this transmission will be assigned with a new sub-frame unoccupied by V2I UL. At the end of each iteration, the $V_{i,j}^U$, D_a, and D_r are updated accordingly;

6. *Allocate V2V RBs* – The V2V mode is selected for each transmission in $V_{i,j}^U$, and the required V2V RBs are allocated in its assigned sub-frame.

4.4.2 Resource Allocation for CPM Transmissions

Wireless Resource Allocation

Given the resource allocation results for DENM and CAM transmissions during each CAM period, the number of supported CPM upload transmissions for group $V_{i,j}$ and the RSU, denoted by $C_{i,j}^P$ and C_R^P, respectively, can be obtained as follows. Recall that CPM packets need to be uploaded via V2I connections.

1. *Contending degree of sub-frames* – For each group $V_{i,j}$, the sub-frames unallocated for V2V transmission can be used for CPM upload. Hence, the unoccupied RBs in a non-V2V sub-frame can be potentially allocated for CPM upload, which may be contended by multiple vehicle groups with the same segment. For the m-th sub-frame

in T_i segment, $S_{i,m}, m = 1, ..., t_i$, the number of groups that contend $S_{i,m}$ for CPM upload is defined as the contending degree of sub-frame $S_{i,m}$, denoted by $B_{i,m}$;

2. *Supported CPM upload transmissions for each group* – For group $V_{i,j}$, if sub-frame $S_{i,m}$ is not allocated for the V2V transmission, the unoccupied RBs in $S_{i,m}$ are available for CPM upload, the number of which is denoted by $A_{i,j,m}$. The $C^P_{i,j}$ is calculated as the weighted summation of $A_{i,j,m}$, i.e., $C^P_{i,j} = \left\lfloor \left(\sum_{m=1}^{t_i} A_{i,j,m}/B_{i,m} \right) / S^{PU} \right\rfloor$;

3. *Supported CPM upload transmissions for RSU* – Besides the limitation of $C^P_{i,j}$, the RB reservation for V2I DL transmissions should be considered as well. Thus, the overall available number of RBs for CPM upload transmission under the RSU coverage is determined by the gap between the available and required numbers of V2I DL RBs, i.e., $C^P_R = \left\lfloor \left(D_a - D_r - S^{PD} \right) / S^{PU} \right\rfloor$. Notice that, D_r represents the required number of V2I DL RBs for the DENMs and CAMs, while S^{PD} represents that for the integrated CPM packet.

Given the number of CPM requests for each group $V_{i,j}$, denoted by $R^P_{i,j}$, the required CPM upload interval (named by CPM period) is designed as an integral multiple of the CAM period. The integral multiplier, T^P, is calculated by

$$T^P = \max(T^P_{i,j}, T^P_R) = \max(\left\lceil R^P_{i,j}/C^P_{i,j} \right\rceil, \left\lceil \left(\sum_i \sum_j R^P_{i,j} \right) / C^P_R \right\rceil), \tag{4.6}$$

where the required CPM period for each group and RSU are denoted by $T^P_{i,j}$ and T^P_R, respectively. Since a longer CPM upload delay results in a larger sensing error, T^P should be minimized via improving the C^P_R and $C^P_{i,j}$. However, the C^P_R can be improved only through modifying the selected V2I mode to V2V mode, which may lead to the packet loss of DENMs and CAMs. On the contrary, $C^P_{i,j}$ can be improved via modifying the selected V2V mode to V2I mode, which can achieve a larger $C^P_{i,j}$ without impairing DENM and CAM transmissions. To be specific, for $V_{i,j}$, if modifying the transmission in $S_{i,m}$ from V2V mode to V2I mode, the impacts on following aspects are considered:

1. $C^P_{i,j}$ – Owing to the modification, $S_{i,m}$ can be applied by $V_{i,j}$ for CPM transmission, and the $B_{i,m}$ is updated correspondingly. Hence, the $C^P_{i,j}$ is updated with the added $(A_{i,j,m}/B_{i,m})$ RBs;

2. $C^P_{i,k}, k \neq j$ – For other groups belonged to T_i segment, if $S_{i,m}$ is available for CPM transmissions, $C^P_{i,k}$ is reduced due to the larger $B_{i,m}$. Otherwise, $C^P_{i,k}$ is not changed;

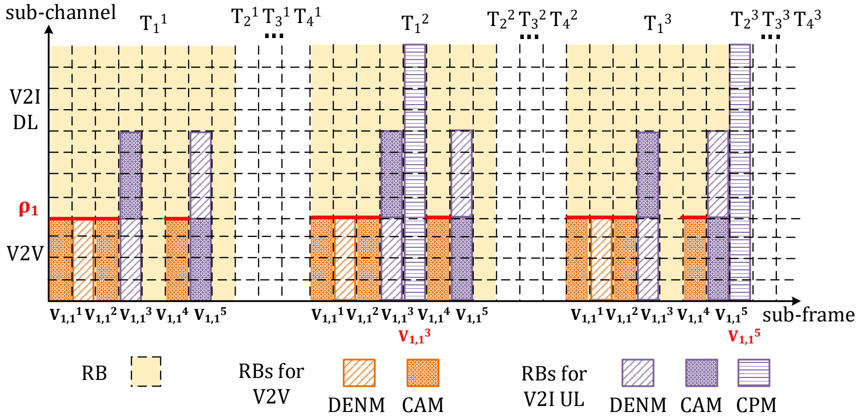

Figure 4.9: An illustration of resource allocation for DENM, CAM, and CPM.

3. C_R^P – Since one more transmission is selected for V2I mode, more RBs are required for V2I DL transmissions. Thus, the C_R^P is reduced with the larger D_r.

If $T_{i,j}^P$ is larger than T_R^P, T^P can be reduced through adjusting the selected V2V mode to V2I mode for the group with the highest $T_{i,j}^P$. Although the initial RB allocation results for DENM and CAM transmissions are achieved in Subsection 4.4.1, we still can adjust the RB allocation to minimize the CPM upload interval, as given in Algorithm 3. The procedures of determining which vehicle groups require to be adjusted are given from Line 3 to Line 4, and the detailed procedures of RB allocation adjustment are given from Line 5 to Line 22. The output of Algorithm 3 includes the updated RB allocation for DENM and CAM transmissions, the CPM period, $C_{i,j}^P$, and C_R^P, which will then be used to determine the CP seed vehicle selection.

In Fig. 4.9, an example of DENM, CAM, and CPM RB allocation for group $V_{1,1}$ is illustrated, where $R_{1,1}^P = 2$ and $T^P = 3$. To allocate the V2I UL RBs for CPM transmissions, the resource pool is extended to one CPM period (i.e., three CAM periods), and T_i^n denotes the T_i segment in the n-th CAM period. The RB allocation for DENM and CAM transmissions is repeated for each CAM period, obtained by running Algorithm 3. The RB allocation for CPM transmissions is designed based on the extended resource pool, such as the CP seed vehicles $V_{1,1}^3$ and $V_{1,1}^5$ shall transmit their CPM packets in T_1^2 and T_1^3 segments, respectively.

Sensing Resource Allocation

The CP seed vehicles are selected to reach the coverage rate threshold and optimize the integrated sensing accuracy, represented by the overall sensing error within the area. To

Algorithm 3 RB Allocation Adjustment to minimize T^P

```
1: Input: Initial RB allocation results for DENM and CAM transmissions; number of CPM transmission requests, $R^P_{i,j}$
2: Output: Modified RB allocation for DENM and CAM transmissions; required CPM period, $T^P$; $C^P_{i,j}$ and $C^P_R$
3: Calculate $C^P_{i,j}$, $C^P_R$, $T^P_{i,j}$, and $T^P_R$
4: The candidate group for adjustment, $\mathbb{C} = \left\{ V_{i,j} | T^P_{i,j} > T^P_R \right\}$
5: while $\mathbb{C}$ is not empty do
6:     Find the group with the largest $T^P_{i,j}$ in $\mathbb{C}$, $V_{i,j}$
7:     for The sub-frame assigned for V2V transmission in $V_{i,j}$, $S_{i,m}$ do
8:         Assuming $S_{i,m}$ is modified to V2I mode for $V_{i,j}$, calculate the potential $C^P_{i,j}$, $j = 1, ..., G_i$, and $C^P_R$
9:         if Existing $C^P_{i,j}$ or $C^P_R$ becomes 0 then
10:            Do not modify the transmission mode of $S_{i,m}$
11:        else
12:            Modify the transmission mode of $S_{i,m}$ to V2I
13:            Update $B_{i,m}$, $D_r$, $C^P_{i,j}$, and $C^P_R$
14:            Update $T^P_{i,j}$, $T^P_R$, and $\mathbb{C}$
15:        end if
16:        if $V_{i,j}$ is not the group with the largest $T^P_{i,j}$ in $\mathbb{C}$ then
17:            Terminate the modification for $V_{i,j}$
18:            Turn to Line 6
19:        end if
20:    end for
21:    Remove $V_{i,j}$ from $\mathbb{C}$
22: end while
23: Calculate the required CPM period, $T^P = \max(T^P_{i,j}, T^P_R)$
```

evaluate the overall sensing error, we divide the street area into small sensing blocks and calculate the sensing error of each block. Given an SCV, we can calculate its sensing error for the blocks, which is dependent on the sensing capability of the SCV. In addition, since a longer latency of sensing information update results in a larger sensing error, the CPM period T^P is also considered for sensing error calculation. For an SCV, its sensing error for a given block is

$$e = \begin{cases} w_d \cdot d + w_t \cdot T^P, & \text{if sensed,} \\ e_0, & \text{if not sensed,} \end{cases} \tag{4.7}$$

where w_d (w_t) is the unit sensing error caused by distance (the latency of CPM update), d is the distance between the block center and the SCV, and e_0 is the penalty error for not sensed blocks. Thus, for each SCV, its sensing quality map is built to represent its sensing performance, which is a matrix of e calculated for all blocks with $T^P = 0$.

The overall sensing error is obtained by accumulating the sensing error for all blocks. Based on the sensing quality maps of CP seed vehicles, the sensing error of each block is evaluated as the minimal e achieved by the selected SCVs. We propose an iterated method to select the CP seed vehicles and determine the required CPM period, as given in Algorithm 4. In each iteration, one SCV is added to the CP seed vehicle set, which is greedily selected based on the sensing gain. The sensing gain of one SCV is defined as the decrement of overall sensing error achieved by adding this SCV to the CP seed vehicle set. In **Phase1** of Algorithm 4, the RSU identifies the vehicle groups with adequate wireless resources for CPM upload. Within the qualified groups, the SCVs are firstly selected to reach the sensing coverage threshold C, then are selected to improve the overall sensing accuracy, which are elaborated in **Phase2** and **Phase3**, respectively. To minimize the sensing error caused by CPM update latency, the wireless resource allocation is adjusted (Algorithm 3) to minimize T^P. Hence, the selection of CP seed vehicles and optimization of wireless resource allocation are iteratively performed.

4.4.3 V2I DL Resource allocation

For the DENM and CAM transmitted under the V2I mode, S^D and S^A RBs are required for V2I DL transmissions, respectively. For CP services, the RSU broadcasts an integrated CPM packet in each CAM period, which requires S^{PD} RBs for V2I DL transmission. As shown in Fig. 4.6, for a packet transmitted via V2I mode, the RBs for DL transmission are allocated after the UL RB allocation. To minimize the packet transmission delay, the DL RBs are allocated to achieve the minimal time gap away from the allocated UL RBs. Considering the priority of different message types, the DL RB allocation performs following the allocation order of DENM, CAM, and CPM. In addition, considering the diversified receiver group size $M_{i,j}^m$, the transmitting vehicle with a larger $M_{i,j}^m$ has a higher priority on DL RB allocation. More specifically, the DL RBs are firstly allocated for the vehicles with DENM requests, following the descending orders by $M_{i,j}^m$. Then, the DL RBs are allocated for CAM transmissions in a similar way, followed by that for the CPM DL.

4.4.4 Resource Allocation Analysis for DENM and CAM

To analyze the resource allocation for DENM and CAM transmissions, we simulate the proposed SRA scheme based on the taxi GPS trace data set [85], including the vehicle ID,

Algorithm 4 Resource Allocation for CPM

```
1:  **Input:** Initial RB allocation results for DENM and CAM transmissions; sensing quality maps
    of SCVs
2:  **Output:** CP seed vehicle set, $\mathtt{V}^s$; required CPM period, $T^P$; V2I UL RB allocation for $\mathtt{V}^s$
3:  **Initialization**:
4:  CP seed vehicle set, $\mathtt{V}^s = \{\}$
5:  Sensing error for each block, $e_0$; the overall error, $E_0$
6:  **Phase1**: Identify the groups supporting the CPM upload:
7:  With $R^P_{i,j} = 1$, run Algorithm 3 and obtain $C^P_{i,j}$
8:  Establish candidate CP seed vehicle set $\mathtt{V}^c$, i.e., the SCVs in the groups with positive $C^P_{i,j}$
9:  **Phase2**: Select SCVs to reach the coverage rate threshold:
10: **while** Coverage rate $< C$ **do**
11:     Find the SCV in $\mathtt{V}^c$ with the highest sensing gain
12:     Add the selected SCV to $\mathtt{V}^s$, and remove it from $\mathtt{V}^c$
13:     Update sensing errors for the newly sensed blocks
14:     Calculate the coverage rate and overall sensing error
15: **end while**
16: $R^P_{i,j} \leftarrow$ the number of CPM requests in each group $V_{i,j}$
17: Run Algorithm 3 and obtain the required CPM period $T^P$
18: Update the overall error as $E_{T^P}$ by considering the additional error caused by the CPM
    update latency
19: **Phase3**: Select SCVs to improve the sensing quality:
20: Let $E_{T^P-1} = E_0$
21: **while** $E_{T^P} < E_{T^P-1}$ **do**
22:     **while** Required CPM period $= T^P$ **do**
23:         Repeat the steps from Line 11 to Line 14
24:         Update $R^P_{i,j}$
25:         Run Algorithm 3 and obtain the required CPM period
26:     **end while**
27:     $E_{T^P} \leftarrow$ the achieved minimal overall error with current CPM period $T^P$
28:     $T^P = T^P + 1$
29: **end while**
30: Find the optimal $T^P$ with the minimal overall error and the corresponding $\mathtt{V}^s$
31: **Phase4**: V2I UL RB allocation:
32: Allocate V2I UL RBs to each vehicle in $\mathtt{V}^s$, according to the remaining wireless resources in
    its belonging group
```

vehicle position, and the corresponding timestamp. We focus on a 800 m × 800 m square area, which covers two intersections. The DENM requests are randomly raised by vehicles, and a constant number of packets will be transmitted for each request. The numbers of required RBs for DENM, CAM, and CPM upload/download transmissions are given in Table 4.2. Moreover, the parameters of the vehicular network and the communication model [86] are given in Table 4.2.

Table 4.2: Simulation parameters.

Packet sizes (RB)			
DENM	15	CAM	15
CPM UL	50	CPM DL	250
Vehicular network parameters			
Number of sub-channels	50	Number of time segments	4
Number of sub-frames per segment	25	Number of DENMs per request	10
Communication model parameters			
RSU communication range (m)	500	V2V communication range (m)	200
Decoding threshold (dBm)	-75	Path-loss exponent, n_{NLOS}	2.69
Receiver's antenna gain, G_r (dB)	3	Transmission power per RB, P_{TX} (dBm)	23
Carrier frequency (GHz)	5.89	Critical distance, d_b (m)	100

Since the traffic density varies during peak and off-peak hours, the number of message transmission requests raised within each vehicle group changes accordingly. For one vehicle group, the numbers of RBs allocated for DENM and CAM transmissions are evaluated, with respect to the number of corresponding transmission requests. Both V2V and V2I modes are considered, and the sum of V2V and V2I UL RBs allocated for each message type is equal to its required number of RBs. Since the NLOS condition is severe at the intersection, V2I transmissions are required by most of the vehicles in the area, and thus the number of allocated V2I UL RBs, which is nearly proportional to the number of requests in Fig. 4.10. As each time segment consists of 25 sub-frames, the insufficient sub-frame condition happens when the overall number of requests becomes larger than 25, i.e., more V2I UL RBs are required at the last point in Fig. 4.10. If the group is close to the intersection, fewer vehicles are impacted by NLOS conditions, leading to smaller V2I resource requirements, as shown in Fig. 4.11. Considering the limited V2I UL resources, the number of V2I UL RBs becomes smaller slightly in Fig. 4.11 during the traffic peak hours, due to more V2I UL resource demands of the groups shown in Fig. 4.10. However, for the groups far from the intersection, most vehicles are assigned with V2V modes due to the good link quality, and hence the numbers of V2I RBs allocated for DENM and CAM transmissions become zero in Fig. 4.12. In addition to NLOS conditions, the differentiated priorities are considered for V2I resource allocation. Thus, when the V2I resources are scarce, more V2I RBs are allocated to the DENM transmission with a higher priority, as shown in Fig. 4.10.

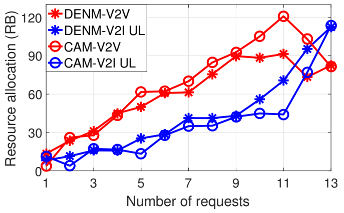

Figure 4.10: Resource allocation results for vehicle groups at intersection.

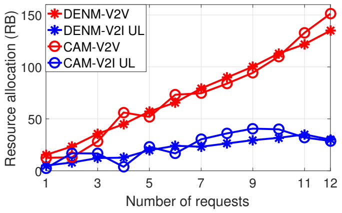

Figure 4.11: Resource allocation results for vehicle groups close to the intersection.

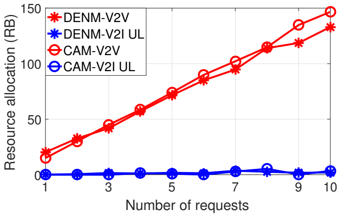

Figure 4.12: Resource allocation results for vehicle groups away from the intersection.

4.5 Resource Reservation and Resource Allocation Modules

As the SRA scheme needs to sequentially make decisions for vehicles requesting different messages, the TARA framework is proposed to enable parallel decision-making to improve the time efficiency. To make adaptive resource allocation decisions, a group-level resource reservation module and a vehicle-level resource allocation module are designed, as shown in Fig. 4.13. The resource reservation module distributes resources for different message types to satisfy their distinct QoS requirements. With the reserved resources, RBs are allocated to each vehicle by the resource allocation module.

4.5.1 Group-Level Resource Reservation Module

In the proposed resource allocation scheme for DENM and CAM transmissions introduced in Subsection 4.4.1, the decision on NLOS-drive V2I mode selection is made iteratively for all the vehicles, which requires a large number of iterations, especially for the peak hours with high traffic density. To improve the efficiency of the resource allocation scheme, the numbers of V2I UL RBs allocated for DENM and CAM transmissions in each group are estimated by the resource reservation module, enabling the decoupling of the V2I mode selection among different groups.

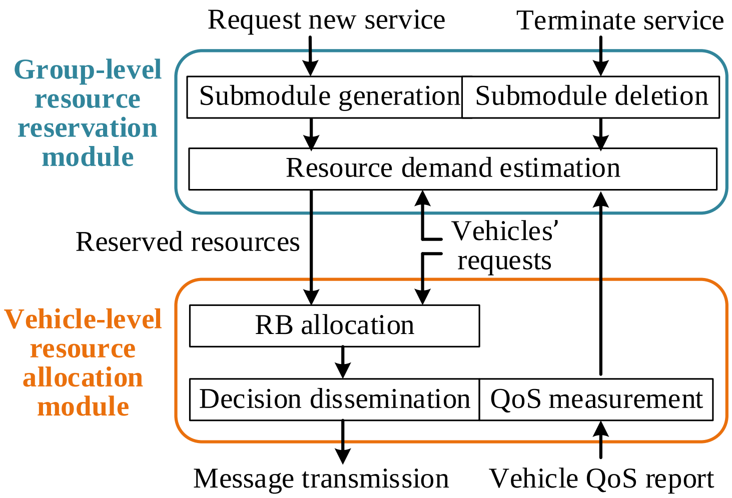

Figure 4.13: An illustration of the resource reservation module and the resource allocation module.

As shown in Fig. 4.13, resource reservation module consists of three functionalities, two related to submodule formation (i.e., generation and deletion) and one for determining the amount of reserved resources for each message type (i.e., resource demand estimation). Since CAMs and CPMs are periodically broadcasted, the corresponding submodules are established as default setting for each vehicle group by the resource reservation module. However, the transmission of DENMs is event-triggered, so the DENM submodule is established and deleted according to vehicles' requests. For each established submodule, the resource demand estimation function specifies the amount of reserved resources, based on the information of vehicles' requests. To be specific, for vehicle group $V_{i,j}$, the numbers of V2I UL RBs allocated to vehicles requesting for DENMs and CAMs are specified, denoted by $N_{i,j}^D$ and $N_{i,j}^A$, respectively.

The update is triggered by the detection of an unsatisfied submodule or a new submodule, as shown in Fig. 4.13. Here the unsatisfied message transmission is identified by QoS measurement results from the vehicle-level resource allocation module. If one submodule with insufficient resources is detected, the resource reservation module will first estimate the resources required by this submodule based on real-time requests. If the remaining resources are sufficient, the resource reservation result is only updated for the under-provisioning submodule. Otherwise, all the submodules require an update, including the redundant resource release and resource supplement for the over-provisioning and under-provisioning submodules, respectively.

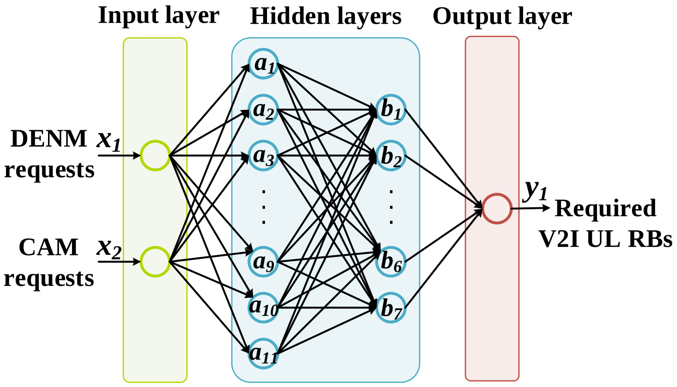

Figure 4.14: BP network based resource demand estimation.

Resource Demand Estimation

In light of the analysis in Subsection 4.4.4, the amount of allocated V2I UL RBs for DENM and CAM transmissions is closely correlated with the number of requests. Due to the priorities of different message types, the mapping relationships between requests and resources are diversified. On the other hand, the location of each vehicle group also determines its required amount of V2I UL resources, e.g., the vehicles approaching intersections are more likely to suffer from NLOS conditions and require V2I resources. In order to estimate the V2I UL RB demand, we model the mapping relationship for resource demand estimation. Without a precise mathematical model, the learning-based method, e.g. back propagation (BP) neural network [87], is a promising solution to this regression problem. Due to the multi-layer structure, the BP network has the capability to mimic complex nonlinear mapping relationship [88]. In addition, for unexpected inputs, the mechanism of error back propagation enables weight adjustment to minimize the estimation error. Hence, the BP network based resource demand estimation is trained for each vehicle group to estimate the required V2I RBs for DENM and CAM transmissions.

As shown in Fig. 4.14, for group $V_{i,j}$, a BP network is trained to learn the relationship between the input and the output, consisting of the input, hidden, and output layers. The numbers of received DENM and CAM requests are utilized as the input, and the output of the BP network is the allocated V2I UL RBs for DENM or CAM transmissions. To train the learning module, we run the SRA scheme and record its V2I resource allocation results, to obtain the training dataset. One sample of input and output can be achieved during each CAM period. An advisable configuration is essential for the BP network to accurately estimate resource demand. We configure the neural network with different transfer functions and different numbers of nodes, and find the optimal configuration based on the experimental results. In doing so, the BP network is set with two hidden layers of the node number of 11 and 7 with the transfer functions of tansig and linear, respectively.

4.5.2 Vehicle-Level Resource Allocation Module

Based on the resource reservation result, the RB allocation function determines CP seed vehicle set and the V2V, V2I UL, and V2I DL RBs allocated for each request. Then, vehicles receive the RB allocation decision and transmit the generated packets accordingly. At the end of each interval, the QoS measurement function measures the performance for message transmissions, according to the collected reports from vehicles. Comparing with the sequential decision-making in SRA scheme, the RB allocation decisions for DENM and CAM transmissions can be made in parallel in terms of groups and message types, as given in Algorithm 5. In each vehicle group, vehicles with higher $G_{i,j}^{m}$ are allocated with V2I RBs until the amount of V2I RBs reserved for this group is used up, as described by **Phase1**. Hence, the mode selection for DENMs and CAMs in different groups can be operated in parallel, which is more efficient than the method introduced in Subsection 4.4.1. After assigning transmission modes for all vehicles, the number of required TDSs and requests transmitted in TDSs is calculated in **Phase2**. With the configuration of transmission mode and TDSs, the corresponding V2I or V2V RBs are allocated to each vehicle in **Phase3**. Based on the RB allocation result of DENMs and CAMs, the resource allocation for CPMs and the V2I DL RB allocation follow the procedures introduced in Subsection 4.4.2 and Subsection 4.4.3, respectively.

Algorithm 5 RB Allocation for the DENM and CAM

```
Input: Vehicles' locations; DENM and CAM requests; reservation decisions, $N_{i,j}^{D}$ and $N_{i,j}^{A}$
Output: V2V and V2I UL RB allocation decision
Estimate the potential V2I gain for each vehicle, $G_{i,j}^{m}$
Phase1: V2I mode selection:
for Each group belonging to the RSU, $V_{i,j}$ do
    $V^{D}(V^{A}) \leftarrow$ the vehicles with DENM (CAM) requests
    for k=1:$\left\lfloor N_{i,j}^{D}/S^{D} \right\rfloor$ do
        while $V^{D}$ is not empty do
            $v \leftarrow$ the vehicle with the highest $G_{i,j}^{m}$ in $V^{D}$
            Select the V2I mode for $v$, remove $v$ from $V^{D}$
        end while
    end for
    for k=1:$\left\lfloor N_{i,j}^{A}/S^{A} \right\rfloor$ do
```

```
14:         while $V^A$ is not empty do
15:             $v \leftarrow$ the vehicle with the highest $G^m_{i,j}$ in $V^A$
16:             Select the V2I mode for $v$, remove $v$ from $V^A$
17:         end while
18:     end for
19: end for
20: $V^D_I(V^A_I) \leftarrow$ the vehicles requesting the DENM (CAM) transmission are assigned with V2I mode
21: Select V2V mode for the requests without V2I mode
22: Phase2: TDS identification:
23: for Each time segment, $T_i$ do
24:     Calculate the required number of TDSs, $I_i$
25:     for Each group belonging to $T_i$, $V_{i,j}$ do
26:         Calculate $D_{i,j}$ and $I_{i,j}$
27:     end for
28:     if $I_i > \sum_{j=1}^{G_i} I_{i,j}$ then
29:         Increase $I_{i,j}$ to satisfy $I_i = \sum_{j=1}^{G_i} I_{i,j}$ and modify the corresponding $D_{i,j}$
30:     end if
31: end for
32: $D_a(D_r) \leftarrow$ available (required) V2I DL RB number
33: while $D_r > D_a$ do
34:     while $V^A_I$ is not empty do
35:         $v \leftarrow$ the vehicle with the lowest $G^m_{i,j}$ in $V^A_I$
36:         Modify the transmission mode to V2V for $v$
37:         Update $V^A_I$, $D_a$, and $D_r$ (Line 23 to Line 32)
38:     end while
39:     while $V^D_I$ is not empty do
40:         $v \leftarrow$ the vehicle with the lowest $G^m_{i,j}$ in $V^D_I$
41:         Modify the transmission mode to V2V for $v$
42:         Update $V^D_I$, $D_a$, and $D_r$ (Line 23 to Line 32);
43:     end while
44: end while
45: Phase3: RB allocation:
46: for Each time segment, $T_i$ do
47:     Let $t_I = 1$, the first sub-frame available for V2I
48:     for Each group belonging to $T_i$, $V_{i,j}$ do
49:         Assign the sub-frames $(t_I, t_I + I_{i,j} - 1)$ as TDSs
50:         Determine the sequential sub-frames, $(t^s_I, t^e_I)$, allocated for V2I transmission, where $t^s_I = t_I$
51:         for The requests in $V_{i,j}$ do
```

```
                if The V2I mode is selected for the request then
                    if There are unoccupied RBs in TDSs then
                        Allocate the V2I RBs in TDSs
                    else
                        Allocate the V2I RBs in the first unoccupied sub-frame within $(t_I + I_{i,j}, t_I^e)$
                    end if
                else
                    Allocate the V2V RBs in the first unoccupied sub-frame out of $(t_I^s, t_I^e)$
                end if
            end for
            $t_I \leftarrow t_I^e + 1$
        end for
    end for
```

4.6 Numerical results

In this section, we perform a trace-driven simulation of both offline and online stages of the proposed TARA framework to evaluate its performance, under the scenario introduced in Subsection 4.4.4. Considering the impact of traffic, we perform the simulation during the hours of low (7:10 a.m.), medium (2:30 p.m.), and high (5:30 p.m.) traffic densities. We assume all the vehicles are equipped with sensors, configured according to the parameters in Table 4.3 [89]. To show the flexibility of the proposed method, we evaluate the performance with varying probabilities of DENM requests.

Table 4.3: Sensor parameters.

Coverage range, R (m)	186
Sensing error per distance, w_d	0.05
Sensing evaluation block size (m)	3
Sensing coverage threshold, C	0.9
Additional sensing error per period, w_t (m)	1.4
Penalty sensing error, e_0 (m)	50

4.6.1 Performance of DENM and CAM Transmissions

Both the reliability and delay are measured for DENM and CAM transmissions. Here, the reliability of each packet transmission is evaluated as the packet delivery ratio (PDR). As

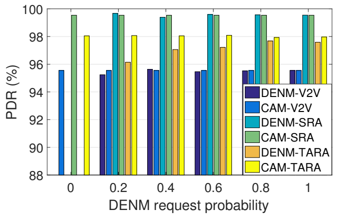

Figure 4.15: Successful reception during low traffic hours.

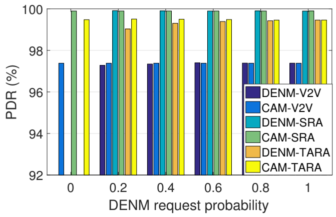

Figure 4.16: Successful reception during medium traffic hours.

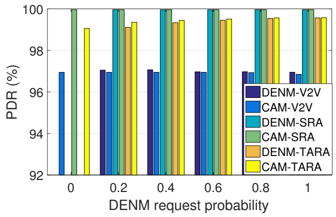

Figure 4.17: Successful reception during high traffic hours.

shown in Fig. 4.15, Fig. 4.16, and Fig. 4.17, if all the transmissions are supported by V2V connections, nearly 96% of packets are successfully received during the low traffic hours, while a higher reliability is achieved during the medium and high traffic hours, owing to the decreased distance among vehicles. Since the packets can be relayed at the RSU, the packet loss resulting from NLOS conditions can be alleviated, and nearly 100% reliability is achieved by the SRA scheme. Different from the SRA, the resource allocation of TARA is based on the results of its resource reservation module. The reliability performance of 98% and 99% are respectively achieved for the low and high traffic scenarios. Notice that, for the TARA results shown in Fig. 4.15, Fig. 4.16, and Fig. 4.17, the reliability of DENM transmission degrades with a lower DENM request probability. This is because the resource reservation module tends to neglect the small number of DENM requests, and allocate the majority of V2I resources to CAM transmissions in this case.

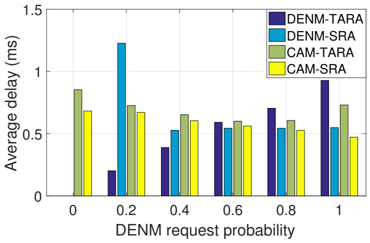

Figure 4.18: Average transmission delay during low traffic hours.

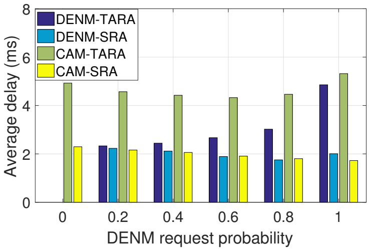

Figure 4.19: Average transmission delay during medium traffic hours.

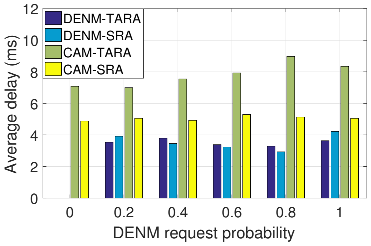

Figure 4.20: Average transmission delay during high traffic hours.

To evaluate the delay of packet transmission, we measure the sub-frame gap between the allocated UL and DL RBs for each V2I mode transmission. Due to the simultaneous transmission and reception, the delay of V2V transmission is negligible. The average delay of packet transmissions with varying vehicle densities and DENM request probabilities is shown in Fig. 4.18, Fig. 4.19, and Fig. 4.20. In the medium and high traffic scenarios, according to the priority order considered by the V2I DL RB allocation, a lower delay is achieved for the DENM transmission, comparing with that of the CAM. In the low traffic scenario, packets are mainly transmitted via V2V connections, and the high priority of DENM is not reflected by transmission delay results. Since the V2I DL RB allocation is implemented after the V2I UL RB allocation, if more V2I RBs are allocated for the UL transmission, the sub-frame gap between DL and UL can become larger due to the less options for DL RB allocation. Thus, the transmission delay increases with the vehicle density and DENM request probability. Although the average delay achieved by the TARA is longer than that of the SRA, it still manages to be lower than 1.5 ms, 6 ms, and 8.5 ms during low, medium, and high traffic hours.

4.6.2 Performance of CPM Transmissions

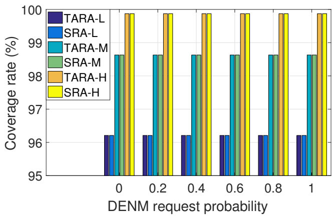

Figure 4.21: Cooperative perception coverage rate.

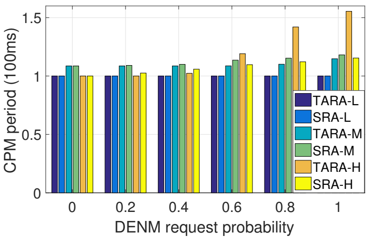

Figure 4.22: CPM period.

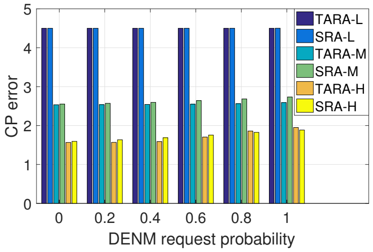

Figure 4.23: Average cooperative perception error.

For CP services, both the coverage rate (CP coverage) and the average sensing error (CP error) are evaluated as QoS metrics, under different traffic conditions, i.e., low (L), medium (M), and high (H) traffic densities. Since the area of interest is divided into blocks, the sensing error is measured for each block. In Fig. 4.21, the coverage refers to the proportion of sensed blocks to the amount of blocks within the area of interest, which is determined by the distribution of vehicles. Thus, with more vehicles on the street, the coverage rate becomes higher. The optimal CPM period is impacted by the vehicle density and the DENM probability. With a higher density of vehicles and a higher DENM request probability, more wireless resources are required to support DENM and CAM transmissions in each CAM period, which calls for a longer CPM period as shown in Fig. 4.22. On the other hand, with a higher density of vehicles, the sensing gain of selecting one more SCV decreases due to the more compact distribution of SCVs. Thus, for the condition with

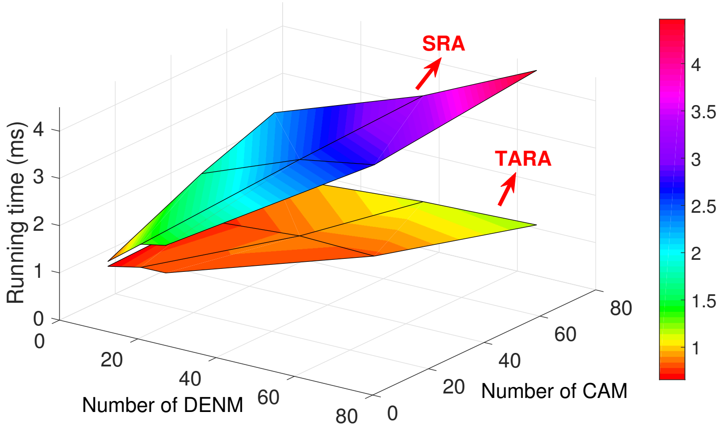

Figure 4.24: Running time of the resource allocation schemes.

a high traffic density and a low DENM request probability, the optimal CPM period is 1, since the fact that the sensing gain achieved by selecting more SCVs is lower than the penalty of a longer CPM update latency. In Fig. 4.23, the average sensing error for each block is evaluated. Considering the penalty error for the area out of sensing coverage, the CP error with different traffic densities has a similar trend as that of the coverage rate. Meanwhile, a longer CPM upload latency leads to a higher CP error, which can be observed through the similar increasing trends of the CPM period and the CP error.

4.6.3 Time Complexity of Resource Allocation Schemes

To compare the time complexity of the SRA and TARA, we evaluate the running time of their wireless resource allocation algorithms introduced in Subsection 4.4.1 and Algorithm 5, respectively. Both are run on an Intel i7-8750U CPU@2.2GHz. The running time is evaluated with varying traffic density and DENM request probabilities, i.e., the varying numbers of DENM and CAM transmissions handled by resource allocation algorithms. In Fig. 4.24, an almost linear increasing trend is observed for the average running time of the SRA algorithm, while the increasing trend of the TARA is negligible. Specifically, in the high traffic density scenario with the DENM probability of 0.9, the time consumed by the SRA is three times longer than that of the TARA. According to the evaluation results, the TARA can make resource allocation decisions more efficiently than the SRA, especially during the traffic peak hours with a large number of transmission requests. The simulation results indicate the low complexity of the proposed TARA framework in large-scale vehicular networks with satisfied delay and CP service requirements.

4.7 Summary

In this chapter, we have proposed TARA framework to support safety message transmissions in RSU-assisted urban vehicular network. Specifically, TARA framework consists of a group-level resource reservation module and a vehicle-level resource allocation module to enable adaptive multi-dimensional resource allocation. For vehicles with DENM, CAM, or CPM requests, the multi-dimensional resource allocation decisions, including the V2I/V2V mode selection, RB allocation, and the CP seed vehicle selection, can be made to optimize the distinct performance metrics for different message types. The trace-driven simulation results have been provided to demonstrate the low latency and high successful reception achieved for DENM and CAM transmissions, the high sensing quality achieved for CPM transmissions, and the robustness of the framework to adapt to dynamic traffic densities. Moreover, the proposed TARA framework can achieve real-time satisfying performance and be readily applied into large-scale vehicular networks. For the future work, considering the computation-intensive vehicular services, we will further investigate the joint allocation of computing, wireless, and sensing resources in vehicular networks for efficient service provisioning.

Chapter 5

Learning-Based Proactive Resource Sharing for Delay-Sensitive Packet Transmissions

In this chapter, we investigate forwarding resource sharing scheme to support interaction intensive applications in HVNets, especially for the delay-sensitive packet transmission between vehicles and management controllers. A learning-based proactive resource sharing scheme is proposed for core communication networks, where the available forwarding resources at a switch are proactively allocated to the traffic flows in order to maximize the efficiency of resource utilization with delay satisfaction. The resource sharing scheme consists of two joint modules, estimation of resource demands and allocation of available resources. Considering the distinct features of each traffic flow, a linear regression scheme is developed for resource demand estimation, utilizing the mapping relation between traffic flow status and required resources. To learn the implicit relation between the allocated resources and delay, a multi-armed bandit learning-based resource allocation scheme is proposed, which enables fast resource allocation adjustment to traffic arrival dynamics. Extensive simulation results are presented to demonstrate the effectiveness of the proposed resource sharing scheme in terms of delay satisfaction, traffic adaptiveness, and resource allocation gain.

5.1 Background and Motivations

The proliferation of new interaction intensive vehicular applications has placed significant pressure on service provisioning in future core communication networks beyond the 5G. The delay-sensitive packet transmission between vehicles and management controllers is required, which is challenging to satisfy. For instance, trajectory planning requires low latency and high reliability to ensure interactivity between vehicles and traffic management controller. To meet these requirements, the utilization of buffer spaces at each network switch needs to be properly controlled. Specifically, the dominant contributing factor for E2E packet transmission delay is the delay for packet queuing at network switches, and the packet loss is mainly caused by buffer overflow during network congestion [90].

Recently, some congestion control methods are proposed to meet the stringent QoS requirements, e.g., performance-oriented congestion control [91] and BBR [92], both of which are implemented at the packet source node, to adjust sending rate based on the observed E2E performance (e.g., achieved goodput, packet loss rate, and average latency). On the other hand, in-network congestion control schemes (e.g., active queue management (AQM)) adjust the queue lengths at switches by dropping or marking packets [93]. In [94], the interplay of end congestion control and in-network AQM is investigated to enable real-time Web browsing. A variation of BBR is proposed for a multi-path transmission scenario [95]. Congestion control schemes in general assume a fixed amount of forwarding resources for the traffic flow of each service (i.e., an aggregation of packets of the same service type being transmitted from a source node to a destination node). To guarantee QoS satisfaction for different applications, resource over-provisioning is usually the case to support the peak traffic volume, while resource multiplexing is exploited among traffic flows of different services to improve QoS with efficient resource utilization.

To support interaction intensive vehicular applications, packet transmission delay in core networks should be minimized, in order to meet the E2E delay requirement. The delivery delay of each service flow in core networks is determined by the routing path and the allocated forwarding resources at each switch on route. Software-defined networking (SDN) is an emerging technology to enhance the routing performance in the next-generation core networks, in which the routing path for packet transmission can be customized and optimized for different services [66, 96]. Specifically, the SDN controller calculates the routing path for each traffic flow and distributes the flow tables to all the related switches, thereby guiding the packet forwarding. In addition, the SDN controller can calculate the average delay of packets traversing each switch [97, 98], which can be utilized to configure the per-hop delay requirement. Given the routing path, the amount of allocated forwarding resources at each passing switch collectively determines the E2E delay of a traffic flow. The decomposition of E2E delay requirement enables the development of a per-hop resource allocation solution. Compared with an E2E solution, a per-hop resource allocation solution can be more flexible when dealing with varying network conditions.

To satisfy the decomposed per-hop delay requirement with efficient resource utilization, each switch makes decision on resource sharing among traffic flows. Delay requirements for per-hop packet transmission are taken into account by the WTP and EDD algorithms. In using the algorithms, the switch makes decision each time that a packet is forwarded. Considering the high forwarding rate of core network switches, flow-level resource sharing schemes with lower computational complexity are investigated for fairness and delay requirements, such as the WFQ and DRR algorithms. Different from the packet-level decision made based on traffic characteristics, estimating resource demand is required for flow-level resource allocation decisions. Thus, to accommodate the large data volume and traffic fluctuations in future core communication networks while achieving efficient resource utilization, we resort to generalized traffic prediction and resource demand estimation to develop a proactive resource sharing scheme.

To support delay-sensitive packet transmission in a dynamic networking environment, learning-based scheduling algorithms are investigated. Due to the distinct delay requirements of applications, different models (e.g., parameters of neural networks) are trained to capture the relations between resource allocation and QoS performance separately, which leads to increased computational cost. To incorporate a large number of flows in the core communication networks, a resource allocation algorithm implemented at in-network switches needs to be effective in achieving satisfactory QoS performance with low computational complexity.

In this chapter, we aim at developing a lightweight online resource demand estimation model for adaptive resource sharing. The resources are for packet forwarding at the egress port of a network switch. On the basis of resource demand estimation to accommodate differentiated delay requirements of different flows, a learning module is developed to learn the relation between allocated resources and achieved QoS for distinct applications. Based on the information of flows (including estimated resource demand, predicted traffic load and resource occupancy), a resource allocation module facilitates resource sharing among flows to achieve maximal overall QoS satisfaction, with the consideration of tradeoff between exploitation and exploration. The multi-armed bandit (MAB) framework has a potential to balance the exploration-exploitation tradeoff, in resource allocation [99] and data offloading decision [100] in wireless networks. In the MAB framework, the player makes sequential decisions, choosing one from the arm set each time to maximize the obtained reward. The player focuses on exploiting the most rewarding arms based on historical performance or pulling new arms for exploration. Upper confidence bound (UCB) [101] is used to balance the exploitation-exploration tradeoff for bandit problems, which provides theoretical confidence bound of regret. In the resource sharing problem, the amount of resources allocated to each flow can be formulated as an arm, while the obtained QoS satisfaction is the reward for pulling the arm. The resource demand can also be considered as the guide for resource sharing, which provides context information of each arm. Thus, the resource sharing is a feature-based exploration-exploitation problem, which can be formulated as a contextual bandit problem [102].

In this chapter, we aim at maximizing the efficiency of resource sharing at a switch (i.e., the ratio of delay satisfaction to the allocated resources), and propose a bandit learning-based proactive forwarding resource sharing scheme, which maps the delay requirement and packet arrivals of each traffic flow to forwarding resource demand. Then, the available resources are allocated to those traffic flows accordingly. The resource allocation is formulated as a bandit learning problem and a regret bounded allocation strategy is developed, which is shown to be efficient in resource utilization. We consider the resource sharing at a software-programmable switch, which is capable of supporting virtual network functions and packet processing functions, in addition to packet forwarding [103]. The main contributions of this chapter are summarized in the following:

1. *Resource sharing framework* – We develop a learning-based framework to make resource sharing decisions at each switch. In this framework, the resource demand is firstly extracted as a service feature, by considering both traffic arrivals and delay requirements. Then, the switch allocates an optimal amount of available resources to different traffic flows based on their features;

2. *Resource demand estimation* – We propose an online resource demand estimation module in the resource sharing framework, which combines a linear regression model with an online gradient descent method. Utilizing the linear mapping relation between traffic loads and the required forwarding resources, the resource demand for each flow at a switch is estimated based on traffic prediction and its per-hop delay requirement;

3. *Allocation of available resources* – We design a MAB-based allocation of available resources scheme in the resource sharing framework. Based on the estimated resource demand, the available resources are allocated to the flows accordingly. Using the measured delay as feedback, the parameters of the proposed learning module are updated.

The remainder of this chapter is organized as follows. The system model is described and the research problem is formulated in Section 5.2. In Section 5.3, the learning-based available resource sharing solution is presented. The performance evaluation and comparison are given in Section 5.4. Finally, conclusions are drawn in Section 5.5.

5.2 System Model and Problem Formulation

5.2.1 Network Model

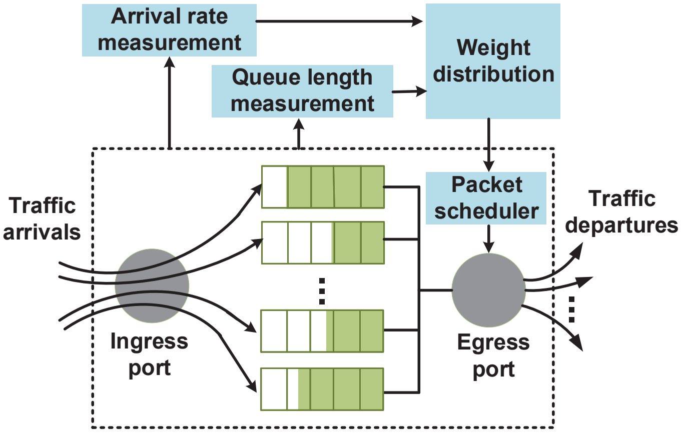

Figure 5.1: An illustration of packet transmission at a switch.

For the vehicles accessing to one edge node, their required data for each service formulate as one traffic flow, with specific E2E transmission delay requirement. Traffic flows of different services traverse a sequence of network switches in core networks before reaching their destinations for E2E service delivery. A network switch refers to a software switch, e.g., an openflow switch, centrally managed by an SDN control module in the core networks [104, 105]. An openflow switch is capable of both layer 2 and layer 3 network functionalities, depending on whether the device is located within a local area network or is interconnecting two public network segments. The E2E delay requirement for each traffic flow is considered, which is composed of the per-hop packet delays at all passing switches along the routing path from the source to the destination. This per-hop delay consists of packet queuing delay and transmission delay at a switch.

At each switch, packets from different flows enter corresponding transmission queues of the packet scheduler, as shown in Fig. 5.1. The packet scheduler makes packet forwarding decisions based on the resource allocation for traffic flows. If more forwarding resources are allocated to a flow, the forwarding rate becomes higher, leading to a reduced queuing delay and transmission delay for this flow. Hence, delay requirements of the traffic flows can be satisfied through adjusting the amount of allocated resources. Consider a set, $\mathcal{N}$, of in-network switches. The set of flows traversing an intermediate switch $n \in \mathcal{N}$ is denoted by $\mathcal{F}_n$, and the set of switches on the routing path of flow $f \in \mathcal{F}_n$ is denoted by $\mathcal{N}_f \subseteq \mathcal{N}$. We assume the flow set, $\mathcal{F}_n$, and the switch set, $\mathcal{N}_f$, remain stable in the course of scheduling. The amount of forwarding resources of switch n is denoted by $C^{(n)}$, and the amount of pre-allocated forwarding resources for flow f is denoted by $\bar{C}_f^{(n)}$, which is determined based on the long-term QoS satisfaction and is assumed to be always guaranteed. The amount of available resources of switch n is $C_I^{(n)} = C^{(n)} - \sum_{f \in \mathcal{F}_n} \bar{C}_f^{(n)}$.

Time is partitioned into resource sharing intervals of constant duration. At the beginning of the mth interval, switch n makes decision on the amount of available resources allocated to flow f, denoted by $\Delta C_{f,m}^{(n)} (\geq 0)$. Then, the allocated forwarding resource for flow f is $C_{f,m}^{(n)} = \bar{C}_f^{(n)} + \Delta C_{f,m}^{(n)}$, and $\left[C_{f,m}^{(n)}\right]_{f \in \mathcal{F}_n}$ are the allocated forwarding resources for all the flows at the mth interval. The weighted round-robin scheme is adopted for packet forwarding, where the weights of flow f, denoted by $v_{f,m}^{(n)}$, is proportional to its allocated forwarding resources. That is,

$$v_{f_1,m}^{(n)} : v_{f_2,m}^{(n)} = C_{f_1,m}^{(n)} : C_{f_2,m}^{(n)}, \forall f_1, f_2 \in \mathcal{F}_n \tag{5.1}$$

where $\sum_{f \in \mathcal{F}_n} v_{f,m}^{(n)} = 1$.

5.2.2 Per-Hop Delay Requirements

We consider traffic flows with differentiated delay requirements and packet arrival patterns. Suppose the E2E packet delay for flow f is decomposed to per-hop delay requirements based on the pre-allocated forwarding resources at each switch. To support the applications with strict delay requirements, we consider SDN enabled core networks, in which an SDN controller has global network information. Upon service requests, the SDN controller configures routing paths and distributes the flow tables to the passing switches for packet forwarding [106, 107]. In addition, the controller can calculate the average queuing delay and average transmission delay for packets traversing each switch [97, 98], the summation of which can be utilized to configure the per-hop delay requirement for the switch. Specifically, the delay requirement for flow $f \in \mathcal{F}_n$ at switch $n \in \mathcal{N}$ is denoted by $D_f^{(n)}$.

We denote the overall delay satisfaction ratio of switch n in interval m as $\rho_m^{(n)} = \sum_{f \in \mathcal{F}_n} \rho_{f,m}^{(n)}$, where $\rho_{f,m}^{(n)}$ is the delay satisfaction ratio for flow f, i.e., the ratio of number of packets with experienced delay smaller than $D_f^{(n)}$ over total number of transmitted packets belonging to flow f in the mth interval. Let ε be the tolerance of per-hop delay violation ratio, i.e., $\rho_{f,m}^{(n)} \geq 1 - \varepsilon$.

5.2.3 Problem Formulation

We consider the resource allocation ratio when making the resource sharing decision, which is denoted by $\eta_m^{(n)}$ for switch n in interval m, $\eta_m^{(n)} = \sum_{f \in \mathcal{F}_n} C_{f,m}^{(n)} / C^{(n)}$. As shown in (5.1), the weighted resource sharing is conducted proportionally to the assigned weight for each flow.

Considering resource allocation for delay satisfaction, we describe the resource sharing efficiency as the delay satisfaction ratio achieved by per unit of allocated resources, denoted by $\rho_m^{(n)} / \eta_m^{(n)}$. The resource sharing optimization problem is formulated as (**P1**), in which our objective is to maximize the resource sharing efficiency at switch n under the resource constraints to determine the optimal decision of allocated available resources ($\left\{\Delta C_{f,m}^{(n)}\right\}_{f \in \mathcal{F}_n}$).

$$(\mathbf{P1}): \max_{\left\{\Delta C_{f,m}^{(n)}\right\}_{f \in \mathcal{F}_n}} \quad \rho_m^{(n)} / \eta_m^{(n)}$$

$$\text{s.t.} \begin{cases} \Delta C_{f,m}^{(n)} \geq 0, f \in \mathcal{F}_n & (5.2\text{a}) \\ \displaystyle\sum_{f \in \mathcal{F}_n} \Delta C_{f,m}^{(n)} \leq C_I^{(n)}. & (5.2\text{b}) \end{cases}$$

Delay satisfaction ratio $\rho_m^{(n)}$ is determined based on the forwarding resources and the packet-level traffic information. Due to the uncertainty of traffic arrival patterns, it is difficult to establish an analytical model for the ratio. Thus, a model-free learning-based method is expected to solve the optimization problem, in which the delay satisfaction ratio can be measured as feedback to the scheme.

5.3 Learning-based Proactive Resource Sharing

In this section, we present a bandit learning-based proactive resource sharing framework to improve delay satisfaction of different services while achieving efficient utilization of network resources.

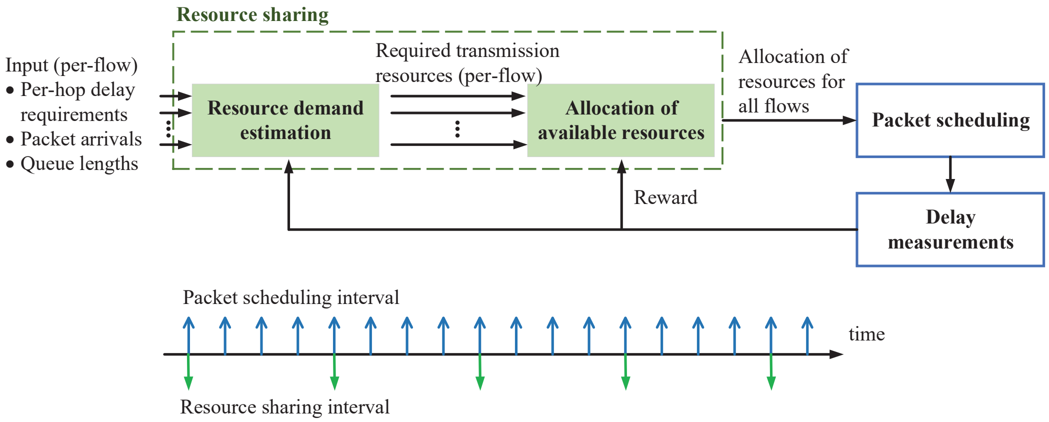
Resource sharing
Input (per-flow)
• Per-hop delay requirements
• Packet arrivals
• Queue lengths
Resource demand estimation
Required transmission resources (per-flow)
Allocation of available resources
Allocation of resources for all flows
Packet scheduling
Delay measurements
Reward
Packet scheduling interval
time
Resource sharing interval

Figure 5.2: An illustration of resource sharing for packet transmission at a switch.

5.3.1 Proactive Resource Sharing Framework

For proactive resource sharing among all flows at each switch, we propose two modules, the resource demand estimation and allocation of available resources, as shown in Fig. 5.2. Within each resource sharing interval, there are integer multiple packet scheduling intervals. The resource sharing framework for packet transmission at a switch consists of the following four functionalities:

1. *Resource demand estimation* – To make the resource allocation decisions, the resource demands from different flows are estimated, based on their delay requirements for per-hop packet transmission, the queue lengths for each flow, and the predicted numbers of arrived packets;

2. *Allocation of available resources* – At the beginning of each resource sharing interval, the allocation of resources for all flows passing through the switch is updated to maximize the efficiency of resource sharing at the switch, based on the estimated resource demands;

3. *Packet scheduling* – For packet transmission, the switch schedules packets from different flows at the egress port according to their allocated resource shares. The decision on packet scheduling is made at each scheduling interval, which is shorter than the resource sharing interval;

4. *Delay measurements* – When a packet is being transmitted at the egress port, its delay at the switch is measured by comparing the time difference between the packet transmission instant and the packet arrival instant [104]. Based on the delay requirements, the overall delay satisfaction ratio at the switch is calculated at the end of each resource sharing interval. With the resource allocation ratio, the resource sharing efficiency is evaluated, which is fed back to the resource demand estimation module and the resource allocation module for updating the resource sharing decisions in the following intervals.

For each flow, the more allocated forwarding resources, the higher delay satisfaction ratio. However, the improvement of delay satisfaction achieved by allocating the same amount of resources to different flows may be different. Considering a flow with sufficient allocated resources, which already has a high delay satisfaction ratio, its performance improvement upon additional allocated resources is limited. On the contrary, allocating resources to a flow with low delay satisfaction ratio will lead to more significant performance improvement. Thus, when allocating resources to different flows, their resource demands should be considered as a feature of the flows, such that a higher resource demand indicates a larger potential improvement of overall delay satisfaction ratio.

The resource demands of flows in the resource sharing problem is analogous to user preferences in an advertisement click problem, referred to as context information. This type of context-based decision making problems can be formulated as contextual bandit problem, and solved through UCB methods [108]. Hence, we propose a MAB formulation with context information to describe the resource sharing problem in (**P1**). In our MAB-based resource sharing problem, an arm represents a potential resource allocation decision for all flows in each resource sharing interval. To maximize the resource sharing efficiency, the reward of the MAB problem is represented by the delay satisfaction ratio, and the optimal arm represents the resource allocation decision which achieves the highest ratio of the reward over the allocated resources. Before arm selection, the rewards of pulling different arms are estimated, considering the context information (i.e., the estimated resource demands in our problem). The achieved reward depends on both arm selection and context information, and the resource sharing among flows is iteratively converged through the learning process with reward feedback.

At the beginning of each resource sharing interval, the learning module makes resource sharing decision. Decision variables $\Delta C_{f,m}^{(n)}$ in (**P1**) is continuous, leading to an infinite number of arms. Since the reward distribution knowledge is learned through pulling different arms, it is necessary to make the arm set finite, i.e., a finite number of arms in the set. Thus, we discretize the available resources at switch n into $I^{(n)} = \left\lfloor C_I^{(n)}/B \right\rfloor$ resource blocks (RBs), each with the same amount of forwarding resources of B (in unit of bits per second). The decision variables $\Delta C_{f,m}^{(n)}$ are also discretized correspondingly as $\Delta I_{f,m}^{(n)}$. Hence, (**P1**) is transformed to (**P2**).

$$(\mathbf{P2}): \max_{\left\{\Delta I_{f,m}^{(n)}\right\}_{f\in\mathcal{F}_n}} \quad \rho_m^{(n)}/\eta_m^{(n)}$$

$$\text{s.t.} \begin{cases} \Delta I_{f,m}^{(n)} \in \mathbb{N}, f \in \mathcal{F}_n & \text{(5.3a)} \\ \Delta C_{f,m}^{(n)} = \Delta I_{f,m}^{(n)} \times B & \text{(5.3b)} \\ \sum\limits_{f\in\mathcal{F}_n} \Delta C_{f,m}^{(n)} \leq C_I^{(n)}. & \text{(5.3c)} \end{cases}$$

A contextual-bandit scheme proceeds in the discrete resource sharing intervals. At the beginning of the mth interval, switch n estimates its current resource demand, $\left[\hat{C}_{f,m}^{(n)}\right]_{f\in\mathcal{F}_n}$, based on reward information from the previous $(m-1)$ intervals $\left[\rho_i^{(n)}\right]_{i=1,2,...,m-1}$. The arm is a vector of $|\mathcal{F}_n|$ integers, each element representing the number of allocated available resource blocks to a flow. Thus, each selected arm $\left[\Delta I_{f,m}^{(n)}\right]_{f\in\mathcal{F}_n}$ determines the allocation of available resources. To avoid redundant resource allocation for a flow, its resource demand is considered. Thus, the number of RBs that can be allocated to flow f at the mth interval should be less than $\left\lceil \frac{\hat{C}_{f,m}^{(n)} - \bar{C}_{f,m}^{(n)}}{B} \right\rceil$. As the total amount of available resources at switch n is $I^{(n)}$, for each flow, we obtain the upper bound of the number of allocated resource blocks $\Delta I_{f,m}^{(n)}$, as $U_{f,m}^{(n)} = \min\left[I^{(n)}, \max\left(\left\lceil \frac{\hat{C}_{f,m}^{(n)} - \bar{C}_{f,m}^{(n)}}{B} \right\rceil, 0\right)\right]$. Since the arm describes the allocation of available resources for all flows traversing the switch, the set of potential arms is denoted by $\mathcal{A}_{f,m}^{(n)}|_{f\in\mathcal{F}_n}$, where $\mathcal{A}_{f,m}^{(n)} = \left\{0, 1, 2, ..., U_{f,m}^{(n)}\right\}$, and the size of the arm set is $\prod_{f\in\mathcal{F}_n}\left(U_{f,m}^{(n)} + 1\right)$. In each interval, an arm is selected from $\mathcal{A}_{f,m}^{(n)}|_{f\in\mathcal{F}_n}$ to achieve the maximal potential reward $\hat{\rho}_m^{(n)}$, i.e., the estimated value of overall delay satisfaction ratio, $\rho_m^{(n)}$.

We propose the resource sharing framework to solve this contextual-bandit problem, with the resource demand estimation module for context information extraction and allocation module of available resource blocks for arm selection. Fig. 5.3 shows the operation procedure of allocating available resource blocks to different flows based on their estimated resource demands. The switch executes the following steps in the mth interval:

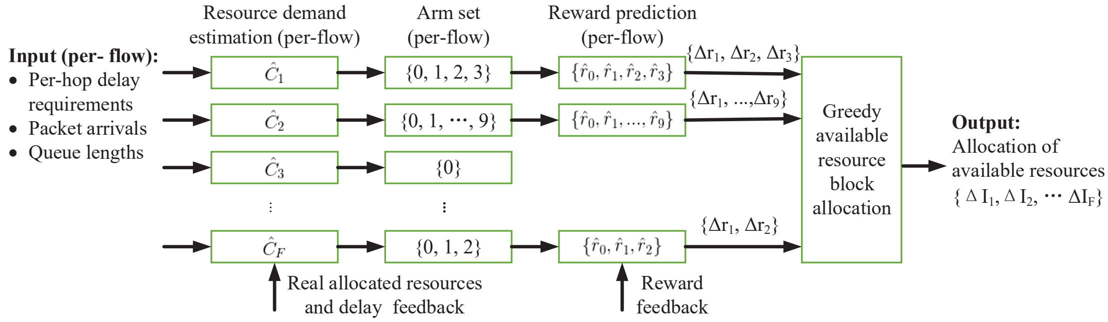
Resource demand estimation (per-flow)
Arm set (per-flow)
Reward prediction (per-flow)
Input (per- flow):
• Per-hop delay requirements
• Packet arrivals
• Queue lengths
$\hat{C}_1$
$\hat{C}_2$
$\hat{C}_3$
$\hat{C}_F$
{0, 1, 2, 3}
{0, 1, ⋯, 9}
{0}
{0, 1, 2}
$\{\hat{r}_0, \hat{r}_1, \hat{r}_2, \hat{r}_3\}$
$\{\hat{r}_0, \hat{r}_1, ..., \hat{r}_9\}$
$\{\hat{r}_0, \hat{r}_1, \hat{r}_2\}$
{Δr1, Δr2, Δr3}
{Δr1, ...,Δr9}
{Δr1, Δr2}
Greedy available resource block allocation
Output:
Allocation of available resources
{ Δ I1, Δ I2, ⋯ ΔIF}
Real allocated resources and delay feedback
Reward feedback

Figure 5.3: Details of resource sharing framework.

1. For flow $f \in \mathcal{F}_n$, switch n estimates its current resource demand $\hat{C}_{f,m}^{(n)}$, and determines its arm set $\mathcal{A}_{f,m}^{(n)}$ accordingly, as discussed in Subsection 5.3.2. Only the flows with $U_{f,m}^{(n)} > 0$ are processed in the following steps, to reduce computing complexity in a large-scale network scenario;

2. Based on rewards $\left[\rho_i^{(n)}\right]_{i=1,2,\ldots,m-1}$ in the previous $(m-1)$ intervals and $\hat{C}_{f,m}^{(n)}$, switch n estimates the potential value of reward, $\hat{\rho}_m^{(n)}$, for each arm from $\mathcal{A}_{f,m}^{(n)}|_{f\in\mathcal{F}_n}$, as discussed in Subsection 5.3.3. Then, the arm with the maximal potential reward is selected and the available resources are allocated;

3. At the end of the interval, $\rho_m^{(n)}$ is observed and used as the feedback reward. With this new observation, the tuple, $\left(\left[\hat{C}_{f,m}^{(n)}\right]_{f\in\mathcal{F}_n}, \left[\Delta I_{f,m}^{(n)}\right]_{f\in\mathcal{F}_n}, \rho_m^{(n)}\right)$, including the context information, the selected arm, and reward, can be used to improve the arm-selection strategy. Note that only the selected arm has the feedback of reward.

5.3.2 Resource Demand Estimation

At the end of the mth resource sharing interval, to obtain delay satisfaction ratio $\rho_{f,m}^{(n)}$ for flow f, switch n measures the delay of staying at the switch for all x_R packets from flow f arriving within interval m. Denote the delay of each packet staying at the switch as d_i,

$i = 1, 2, ..., x_R$. The delay satisfaction ratio $\rho^{(n)}_{f,m}$ is calculated as

$$\rho^{(n)}_{f,m} = \frac{1}{x_R} \sum_{i=1}^{x_R} \mathbf{1}\left(D^{(n)}_f - d_i\right) \tag{5.4}$$

where $\mathbf{1}(x) = 1$ if $x \geq 0$, otherwise $\mathbf{1}(x) = 0$. The delay measurement can be accomplished through pipelined packet processing, which includes the functionalities of reading and writing packet headers [104]. At the end of interval m, current queue length (i.e., initial queue length of interval $(m + 1)$) and the number of arrived packets during interval m, denoted by $b^{(n)}_{f,m+1}$ and $\lambda^{(n)}_{f,m}$ respectively, are measured by the switch.

Based on (5.1), through adjusting the weights (i.e., $v^{(n)}_{f,m}$ for flow f at interval m), the forwarding resources can be allocated for flows at switch n. In the WFQ model, the ratio of the weight for flow f_1 over that for flow f_2 is $v^{(n)}_{f_1,m} : v^{(n)}_{f_2,m} = \delta_{f_2} : \delta_{f_1}$, where δ_{f_1} is the per hop queuing delay bound of flow f_1. The weights designed in WFQ only depend on delay requirements, which are unchangeable with network. Hence, in dynamic WFQ [52], the traffic arrival rate ($\lambda^{(n)}_{f,m}$) and queue length ($b^{(n)}_{f,m}$) of flow f at interval m are included in the weight design, and the ratio of weights for flow f_1 and flow f_2 is

$$v^{(n)}_{f_1,m} : v^{(n)}_{f_2,m} = \frac{b^{(n)}_{f_1,m} + \frac{1}{2}\lambda^{(n)}_{f_1,m}U}{\delta_{f_1}} : \frac{b^{(n)}_{f_2,m} + \frac{1}{2}\lambda^{(n)}_{f_2,m}U}{\delta_{f_2}}, \tag{5.5}$$

where time is partitioned into intervals of constant duration of U sec. If traffic prediction is conducted, resources can be proactively allocated to the flows, according to dynamic WFQ. However, the required resource estimation in dynamic WFQ is accurate only if the packets have equal inter-arrival time. Due to the uncertainty of packet arrivals, we employ an online linear regression (LR) method to directly estimate the required resources, instead of using the determined mapping function. Then, the weights are determined based on the resource demand estimation, which enables proactively resource sharing to deal with the predicted traffic arrival.

At switch n, for packets from flow f arriving within interval m, the required forwarding resources for delay satisfaction is determined by a linear combination of $\lambda^{(n)}_{f,m}/\delta_f$ and $b^{(n)}_{f,m}/\delta_f$ according to (5.5). For supporting applications with stringent delay requirements, resource allocation decisions are made by considering the delay requirements, $D^{(n)}_f$, for individual packet transmission. At switch n, the resource demand for flow f in interval m, $\hat{C}^{(n)}_{f,m}$, is defined as the minimal amount of forwarding resources to satisfy $\rho^{(n)}_{f,m} \geq 1 - \varepsilon$. Since both new packet arrivals and the packets waiting in the queue are to be processed in the next resource sharing interval, the resource demand is dependent on the predicted number of arrived packets $\hat{\lambda}_{f,m}^{(n)}$, $b_{f,m}^{(n)}$, and $D_f^{(n)}$. To estimate $\hat{C}_{f,m}^{(n)}$, we use a LR model to approximate the mapping relation between $\left(D_f^{(n)}, \hat{\lambda}_{f,m}^{(n)}, b_{f,m}^{(n)}\right)$ and $\hat{C}_{f,m}^{(n)}$ [109]. The LR model is widely used in engineering areas, such as signal processing and financial engineering [110]. To approximate the mapping relation, the number of arrived packets, the observed queue length, and the measured delay bound to satisfy the per-hop delay violation ratio are collected at each resource sharing interval to train the model parameters. For better approximation accuracy, we combine an online weight update method (e.g., gradient decent algorithm in [111]) with linear regression. The detailed description of the linear regression based resource demand estimation is given in Algorithm 6, in which the initial model parameters are obtained in the training stage.

Algorithm 6 Online Linear Regression based Resource Demand Estimation Scheme.

```
Initialize $\mathbf{W}_1$
for $m = 1, 2, 3, ...$ do
    $\mathbf{I}_m \leftarrow \left[\hat{\lambda}_m, \frac{\hat{\lambda}_m}{D}, b_m, \frac{b_m}{D}, 1\right]$
    Estimate $\hat{C}_m = \mathbf{W}_m^T \mathbf{I}_m$
    Give $\hat{C}_m$ to allocation of available resources module
    At the end of interval, observe $\lambda_m$, $C_m$, and $D_m^R$
    $\mathbf{I}_m^R \leftarrow \left[\lambda_m, \frac{\lambda_m}{D_m^R}, b_m, \frac{b_m}{D_m^R}, 1\right]$
    Update $\mathbf{W}_{m+1} \leftarrow \mathbf{W}_m - \eta_{m+1} \left(\mathbf{W}_m^T \mathbf{I}_m^R - C_m\right) \mathbf{I}_m^R$
end for
```

The resource demand estimation module takes $\left(D_f^{(n)}, \hat{\lambda}_{f,m}^{(n)}, b_{f,m}^{(n)}\right)$ as input, and estimates corresponding $\hat{C}_{f,m}^{(n)}$ as output. We omit n and f from the symbols used in Algorithm 6 for clarity. Based on the LR model, $\hat{C}_m$ is estimated as

$$\hat{C}_m = w_m^1 \hat{\lambda}_m + w_m^2 \frac{\hat{\lambda}_m}{D} + w_m^3 b_m + w_m^4 \frac{b_m}{D} + w_m^5 \tag{5.6}$$

where the weight vector in the m th resource sharing interval is denoted as $\mathbf{W}_m = [w_m^1, ..., w_m^5]$, and the input vector is $\mathbf{I}_m = \left[\hat{\lambda}_m, \frac{\hat{\lambda}_m}{D}, b_m, \frac{b_m}{D}, 1\right]$.

As shown in Fig. 5.3, at the beginning of each resource sharing interval, say interval m, the amount of required resources is estimated and used as contextual information for the following allocation of available resources module. At the end of the interval, the switch can observe the actual number of arrived packets, λ_m, and the utilized forwarding resources, C_m. In addition, based on the measurement for the delay of each packet staying at the switch, d_i, we can determine D_m^R that satisfies $\frac{1}{x_R}\sum_{i=1}^{x_R}\mathbf{1}\left(D_m^R - d_i\right) = 1-\varepsilon$, with the allocated forwarding resources C_m in interval m. Hence, $\left(D_m^R, \lambda_m, b_m\right)$ and C_m are used to iteratively refine the weight parameters of the LR model in (5.6). According to [111], $\mathbf{W}_m$ is updated as

$$\mathbf{W}_{m+1} = \mathbf{W}_m - \eta_{m+1}\left(\mathbf{W}_m^T\mathbf{I}_m^R - C_m\right)\mathbf{I}_m^R \tag{5.7}$$

where $\mathbf{I}_m^R = \left[\lambda_m, \frac{\lambda_m}{D_m^R}, b_m, \frac{b_m}{D_m^R}, 1\right]$, $\eta_{m+1} = \frac{1}{m+1}$.

5.3.3 Allocation of Available Resource Blocks

To determine the amount of allocated available resources, switch n first estimates the potential reward ($\hat{\rho}_m^{(n)}$) by selecting each possible arm in $\mathcal{A}_{f,m}^{(n)}|_{f\in\mathcal{F}_n}$, which has $\prod_{f\in\mathcal{F}_n}\left(U_{f,m}^{(n)}+1\right)$ arms. However, the increasing number of flows leads to a growth of arm set with high computational complexity for reward estimation. To make the scheme scalable in a large scale network, we estimate the reward on a per-flow basis. As illustrated in Fig. 5.3, switch n estimates the potential per-flow reward ($\hat{\rho}_{f,m}^{(n)}$) with the corresponding number of allocated available resource blocks ($a \in \mathcal{A}_{f,m}^{(n)}$), i.e., $\hat{\rho}_{f,m}^{(n)}$ is estimated under the action of $\Delta I_{f,m}^{(n)} = a$. After that, a greedy method is used to select the arm that achieves the highest ratio of potential reward $\hat{\rho}_m^{(n)}$ over the allocated resources.

In the following, we explain in detail how the per-flow performance is estimated. This is similar to the reward estimation in a contextual bandit problem, which observes the user feature and potentially selected arm as prior knowledge. The problem is well solved by LinUCB, a variation of UCB method proposed as a generic contextual bandit algorithm in [108]. It is proved that a closed-form confidence interval (i.e., the deviation of reward estimation) can be computed efficiently when the reward model is linear. For each flow f, switch n runs one LinUCB to estimate per-flow reward. For clarity, we present the symbols used in the LinUCB based reward estimation scheme, where indexes n and f are omitted. In the mth interval, we have

1. Two-dimensional feature: $\mathbf{x}_m = \left[1, \hat{C}_{f,m}^{(n)}\right]$, $\mathbf{x}_m \in \mathbb{R}^2$;
2. Arm: $a = \Delta I_{f,m}^{(n)}, a \in \mathcal{A}_{f,m}^{(n)}$;
3. Reward: $r_{m,a} = \hat{\rho}_{f,m}^{(n)}$, with the action of choosing arm a.

Under the assumption that the potential reward of a given arm is linear versus its observed feature $\mathbf{x}_m$ with unknown parameter vector θ_a^*, we have

$$\mathbf{E}\left[r_{m,a}|\mathbf{x}_m\right] = \mathbf{x}_m^T\theta_a^*. \tag{5.8}$$

If arm a was selected m' ($m' < m$) times in the previous $(m-1)$ time intervals, the associated reward feedback can be used to improve the estimation of θ_a^*. Let $\mathbf{X}_{m,a}$ and $\mathbf{R}_{m,a}$ denote the previous m' feature vectors and observed rewards, where we have $\mathbf{X}_{m,a} \in \mathbb{R}^{m'\times 2}$ and $\mathbf{R}_{m,a} \in \mathbb{R}^{m'}$. With training data $\mathbf{X}_{m,a}$ and $\mathbf{R}_{m,a}$, parameter vector θ_a^* is estimated as

$$\hat{\theta}_{m,a} = \left(\mathbf{X}_{m,a}^T\mathbf{X}_{m,a} + \mathbf{I}_2\right)^{-1}\mathbf{X}_{m,a}^T\mathbf{R}_{m,a} \tag{5.9}$$

where $\mathbf{I}_2$ is the 2×2 identity matrix. For clarity, we denote $\mathbf{A}_{m,a} = \mathbf{X}_{m,a}^T\mathbf{X}_{m,a} + \mathbf{I}_2$, and $\mathbf{b}_{m,a} = \mathbf{X}_{m,a}^T\mathbf{R}_{m,a}$. Based on the confidence interval given in [108], with the probability of at least $(1-\delta)$, we have

$$\left|\mathbf{x}_m^T\hat{\theta}_{m,a} - \mathbf{E}\left[r_{m,a}|\mathbf{x}_m\right]\right| \leq \alpha\sqrt{\mathbf{x}_m^T\mathbf{A}_{m,a}^{-1}\mathbf{x}_m} \tag{5.10}$$

for $\forall\delta > 0$, where $\alpha = 1 + \sqrt{\ln\left(2/\delta\right)/2}$ is a constant. Based on the analysis in [108], we set $\alpha = 1.5$. The inequality in (5.10) gives an upper bound of the reward estimated by (5.8)-(5.9). In the mth interval, the potential reward for arm a is

$$\hat{r}_{m,a} = \mathbf{x}_m^T\hat{\theta}_{m,a} + \alpha\sqrt{\mathbf{x}_m^T\mathbf{A}_{m,a}^{-1}\mathbf{x}_m}. \tag{5.11}$$

We define the marginal performance gain of allocating one more available RB to flow f as $\Delta r_{m,a}^f = \hat{r}_{m,a} - \hat{r}_{m,a-1}, a = 1, 2, ..., U_{f,m}^{(n)}$, where $\hat{r}_{m,a}$ is obtained by the LinUCB for flow f. Since a better delay satisfaction ratio should be achieved with more allocated forwarding resources, we have $\Delta r_{m,a}^f > 0$. Due to the constant size of RB, a greater $\Delta r_{m,a}^f$ indicates a larger delay satisfaction improvement achieved by the allocated RB. Thus, based on $\Delta r_{m,a}^f$, switch n allocates available RBs greedily for all flows in $\mathcal{F}_n$. Algorithm 7 gives a detailed description of how to allocate available RBs at switch n, and the details of greedy RB allocation are given in Algorithm 8.

To demonstrate how to allocate available RBs, we use the resource sharing among 5 flows as an example shown in Fig. 5.4. The thick red line indicates the upper bound of the amount of RBs that can be allocated to each flow, and the green blocks indicate the amount of RBs already allocated to the flows. The blocks highlighted in orange indicate the potential RB allocation to the flows in each algorithm iteration, if the highest marginal gain (e.g., $\Delta r_{m,a}^f$ for flow f) is observed. In Algorithm 8, one RB is allocated in each iteration, until all the available RBs at switch n have been allocated or all the flows get sufficient RBs (i.e., their upper bounds are reached).

Algorithm 7 Proposed Allocation Scheme for Available Resource Blocks.

```
for $m = 1, 2, 3, ...$ do
    Estimate resource demand (by Algorithm 6) for all flows $f \in \mathcal{F}_n$, $\hat{C}_{f,m}^{(n)}$
    for $f = 1, 2, 3, ... |\mathcal{F}_n|$ do
        Determine $\mathbf{x}_m = \left[1, \hat{C}_{f,m}^{(n)}\right]$ and $U_{f,m}^{(n)}$
        for $a = 0, 1, 2, 3, ...U_{f,m}^{(n)}$ do
            if $a$ is new then
                $\mathbf{A}_{m,a} \leftarrow \mathbf{I}_2$
                $\mathbf{b}_{m,a} \leftarrow \mathbf{0}_{2\times 1}$ (zero vector)
            end if
            $\hat{\theta}_{m,a} \leftarrow \mathbf{A}_{m,a}^{-1}\mathbf{b}_{m,a}$
            $\hat{r}_{m,a} \leftarrow \mathbf{x}_m^T\hat{\theta}_{m,a} + \alpha\sqrt{\mathbf{x}_m^T\mathbf{A}_{m,a}^{-1}\mathbf{x}_m}$
            if $a = 0$ then
                $\Delta r_{m,a}^f = \hat{r}_{m,a}$
            else
                $\Delta r_{m,a}^f = \hat{r}_{m,a} - \hat{r}_{m,a-1}$
            end if
        end for
    end for
    Execute Algorithm 8
    Implement the resource sharing decision $\left[C_{f,m}^{(n)}\right]_{f\in\mathcal{F}_n}$ through packet scheduling
  module
    At the end of interval, $\rho_{f,m}^{(n)}$ is observed as the feedback to all flows $f \in \mathcal{F}_n$ as
  real-valued reward $r_{f,m}$
    for $f = 1, 2, 3, ... |\mathcal{F}_n|$ do
        $a = a_{f,m}$, $r = r_{f,m}$
        $\mathbf{A}_{m,a} \leftarrow \mathbf{A}_{m,a} + \mathbf{x}_m\mathbf{x}_m^T$
        $\mathbf{b}_{m,a} \leftarrow \mathbf{b}_{m,a} + r\mathbf{x}_m$
    end for
end for
```

Algorithm 8 Greedy Allocation of Available Resource Blocks.

```
**Input:** $\Delta r^{f}_{m,a}$
**Initialization:** $\mathcal{F}_t \leftarrow \{\}$
**for** $f = 1, 2, 3, ... |\mathcal{F}_n|$ **do**
    $a_{f,m} \leftarrow 0$
    **if** $U^{(n)}_{f,m} > 0$ **then**
        $\mathcal{F}_t \leftarrow \mathcal{F}_t + \{f\}$
    **end if**
**end for**
**for** $i = 1, 2, 3, ... I^{(n)}$ **do**
    **if** $\mathcal{F}_t$ is not empty **then**
        $f_i = \arg\max_{f \in \mathcal{F}_t} \Delta r^{f}_{m,a}$ where $a = a_{f,m} + 1$. Randomly select $f_i$ if multiple flows have the maximal $\Delta r_{m,a}$
        $a_{f_i,m} \leftarrow a_{f_i,m} + 1$
        **if** $a_{f_i,m} = A^{(n)}_{f_i,m}$ **then**
            $\mathcal{F}_t \leftarrow \mathcal{F}_t - \{f_i\}$
        **end if**
    **end if**
**end for**
**for** $f = 1, 2, 3, ... |\mathcal{F}_n|$ **do**
    $\Delta I^{(n)}_{f,m} \leftarrow a_{f,m}$
**end for**
Based on $\left[\Delta I^{(n)}_{f,m}\right]_{f \in \mathcal{F}_n}$, obtain the corresponding resource sharing decision $\left[C^{(n)}_{f,m}\right]_{f \in \mathcal{F}_n}$
**Output:** Resource sharing decision $\left[C^{(n)}_{f,m}\right]_{f \in \mathcal{F}_n}$ and the selected arm $[a_{f,m}]_{f \in \mathcal{F}_n}$
```

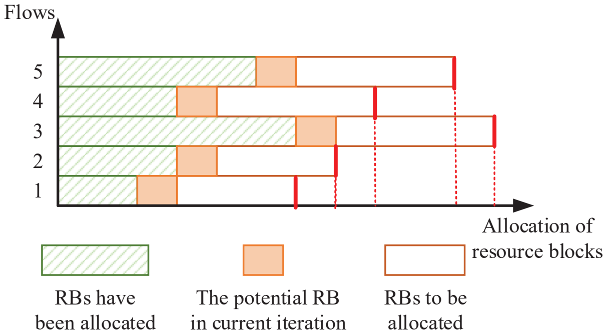
Flows
5
4
3
2
1
Allocation of resource blocks
RBs have been allocated
The potential RB in current iteration
RBs to be allocated

Figure 5.4: An illustration of the greedy available resource block allocation.

5.3.4 Discussion

Regret analysis

As shown in (5.8), the potential reward is linear with respect to the observed feature $\mathbf{x}_m$, and the true coefficient vector is θ_a^*. Without loss of generality, we assume $\|\mathbf{x}_m\| \le L$, where $\|\cdot\|$ represents the l_2-norm.

In the m th interval, denote the best arm $\left[a_{f,m}^*\right]_{f\in\mathcal{F}_n}$ that satisfies

$$\left[a_{f,m}^*\right]_{f\in\mathcal{F}_n} = \mathop{\arg\max}_{\left[a_f\right]_{f\in\mathcal{F}_n}} \sum_{f\in\mathcal{F}_n} \mathbf{x}_m^T \theta_{a_f}^* \tag{5.12}$$

where $a_f \in \mathcal{A}_{f,m}^{(n)}$. Then, comparing with the reward achieved by the selected arm $\left[a_{f,m}\right]_{f\in\mathcal{F}_n}$, the M-trail regret of this resource allocation scheme is calculated by

$$L_M = \sum_{m=1}^{M} \left(\sum_{f\in\mathcal{F}_n} \mathbf{x}_m^T \theta_{a_{f,m}^*}^* - \sum_{f\in\mathcal{F}_n} \mathbf{x}_m^T \theta_{a_{f,m}}^* \right). \tag{5.13}$$

Considering the confidence interval in (5.10), with probability at least $1-\delta$, we have

$$\sum_{f\in\mathcal{F}_n} \mathbf{x}_m^T \theta_{a_{f,m}}^* \ge \sum_{f\in\mathcal{F}_n} \left(\mathbf{x}_m^T \hat{\theta}_{m,a_{f,m}} - \alpha \sqrt{\mathbf{x}_m^T \mathbf{A}_{m,a_{f,m}}^{-1} \mathbf{x}_m} \right). \tag{5.14}$$

According to the proposed available resource block allocation scheme and (5.10), we obtain

$$\begin{aligned} \sum_{f\in\mathcal{F}_n} \mathbf{x}_m^T \theta_{a_{f,m}^*}^* &\le \sum_{f\in\mathcal{F}_n} \left(\mathbf{x}_m^T \hat{\theta}_{m,a_{f,m}^*} + \alpha \sqrt{\mathbf{x}_m^T \mathbf{A}_{m,a_{f,m}^*}^{-1} \mathbf{x}_m} \right) \\ &\le \sum_{f\in\mathcal{F}_n} \left(\mathbf{x}_m^T \hat{\theta}_{m,a_{f,m}} + \alpha \sqrt{\mathbf{x}_m^T \mathbf{A}_{m,a_{f,m}}^{-1} \mathbf{x}_m} \right). \end{aligned} \tag{5.15}$$

Thus,

$$L_M \leq \sum_{m=1}^{M} \left(\sum_{f \in \mathcal{F}_n} 2\alpha \sqrt{\mathbf{x}_m^T \mathbf{A}_{m,a_{f,m}}^{-1} \mathbf{x}_m} \right). \tag{5.16}$$

To simplify (5.16), we apply lemma 4.4 and lemma 4.5 in [112], as

$$\sum_{m=1}^{M} \mathbf{x}_m^T \mathbf{A}_{m,a_{f,m}}^{-1} \mathbf{x}_m \leq 2 \log \frac{\left|\mathbf{A}_{m,a_{f,m}}\right|}{\beta} \tag{5.17}$$

$$\left|\mathbf{A}_{m,a_{f,m}}\right| \leq \left(\beta + mL^2/d\right)^d \tag{5.18}$$

where $d \geq 1$ and $\beta \geq \max\left(1, L^2\right)$. Apply the lemmas in (5.16), we have

$$\begin{aligned} L_M &\leq 2\alpha \sum_{f \in \mathcal{F}_n} \left(\sum_{m=1}^{M} \sqrt{\mathbf{x}_m^T \mathbf{A}_{m,a_{f,m}}^{-1} \mathbf{x}_m} \right) \\ &\leq 2\alpha \sum_{f \in \mathcal{F}_n} \sqrt{M \left(\sum_{m=1}^{M} \mathbf{x}_m^T \mathbf{A}_{m,a_{f,m}}^{-1} \mathbf{x}_m \right)} \\ &\leq 2\alpha \sum_{f \in \mathcal{F}_n} \sqrt{2M \log \frac{\left|\mathbf{A}_{m,a_{f,m}}\right|}{\beta}} \\ &\leq 2\alpha \sum_{f \in \mathcal{F}_n} \sqrt{2Md \cdot \log \left(1 + \frac{ML^2}{\beta d}\right)} \\ &\leq 2\alpha F \sqrt{2Md} \cdot \sqrt{\log\left(1 + M/d\right)}. \end{aligned} \tag{5.19}$$

In (5.19), the M-trail regret bound, $O\left(\sqrt{Md \cdot \log\left(1 + M/d\right)}\right)$, indicates the zero-regret feature[1], i.e., $\lim_{M \to \infty} \frac{L_M}{M} = 0$. The zero-regret strategy is guaranteed to converge to an optimal strategy after enough rounds are played [113]. Thus, the proposed scheme is proved to be asymptotically approaching the optimal strategy [114].

Time complexity analysis

To analyze the scalability of the proposed resource sharing scheme in terms of the number of flows (F), we evaluate the time complexity for each stage in Fig. 5.3 in one interval.

[1] A strategy whose average regret per round tends to zero when the horizon tends to infinity is defined as zero-regret strategy [113].

First, the resource demand estimation module runs Algorithm 6 for each flow separately, which leads to $O\left(F\right)$ time complexity. Then, in Algorithm 7, the reward is calculated for all the arms, where flow f has $\left(U_{f,m}^{(n)}+1\right)$ arms, according to its resource demand. Thus, the complexity is proportional to the total number of arms, $\sum_{f\in\mathcal{F}_n}\left(U_{f,m}^{(n)}+1\right)$. The overall number of resource blocks required by flows should be comparable to the number of available resource blocks ($I^{(n)}$). Otherwise, the network would incur unstable queue accumulation. We obtain the complexity of reward calculation in Algorithm 7 as $O\left(I^{(n)}\right)$. After that, each resource block is allocated by comparing the marginal reward for candidate flows, as given in Algorithm 8. The maximal time complexity is $O\left(F\right)$, in which all the F flows belong to the candidate set. As the comparison is conducted at most for $I^{(n)}$ times, the time complexity of Algorithm 8 is $O\left(FI^{(n)}\right)$. Therefore, combining the complexity of three algorithms, the time complexity of the resource sharing scheme is $O\left(FI^{(n)}\right)$.

Since $I^{(n)}$ is determined by the amount of available resources and the resource block size, we can set the value of $I^{(n)}$ by adjusting the block size. If we set $I^{(n)}$ as a fixed value, the complexity is reduced to $O\left(F\right)$. However, when the number of flows increases, the resource sharing scheme with the fixed $I^{(n)}$ may have a degraded performance. In particular, if there are more flows than resource blocks, i.e., $F > I^{(n)}$, the resources allocated to different flows are distinct even if all the flows have same features. To avoid this inconsistency between the allocated resources and the observed features due to insufficient resource blocks, we set $I^{(n)} \propto F$, and the polynomial time complexity, $O\left(F^2\right)$, is achieved. Note that if the resource block size has to be set as a small value with the increase of F in order to satisfy $I^{(n)} \propto F$, the improvement on delay satisfaction by allocating one resource block could be insignificant, leading to a low marginal gain. Hence, a lower limit on the resource block size needs to be determined properly, considering the allocation efficiency, which remains an important and open issue for the proposed scheme.

5.4 Numerical Results

To demonstrate the effectiveness of the proposed resource sharing scheme, numerical results are presented.

5.4.1 Simulation Scenario

In this chapter, the resource allocation scheme is designed for each switch, aiming at satisfying the per-hop delay requirement. To evaluate the performance of the proposed

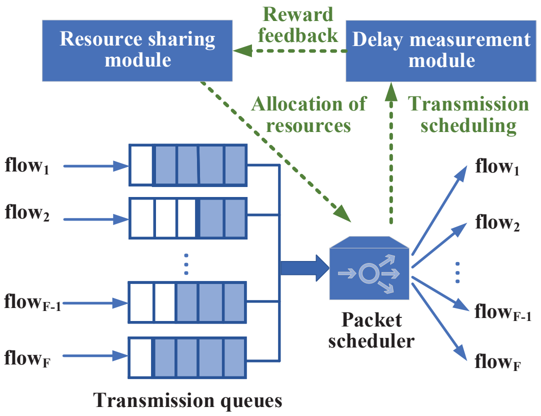

Figure 5.5: An illustration of the simulation scenario.

Table 5.1: Parameters of traffic flow

Number of flows (F)	50
Average traffic rate	15 Mbps
Delay violation tolerance (ε)	0.5 %
Per-hop delay requirement - number of flows	0.5 ms - 10
	0.7 ms - 15
	1.5 ms - 25

scheme, we consider the packet transmission at a network switch in the simulation. For the F traffic flows traversing the switch, the resource sharing among them is simulated and the delay of each packet passing the switch is measured, as shown in Fig. 5.5. The network switch, consisting of the transmission queues, packet scheduler, resource sharing module, and delay measurement module, is emulated on a high-performance computer server using a Python IDE called PyCharm [115]. For each traffic flow, its packets enter the corresponding transmission queue and wait for transmission scheduling. Given the amount of resources allocated to each traffic flow, the packet scheduler makes packet forwarding decisions, i.e., adjusting the forwarding rate for each flow. During the packet transmission, the delay measurement module records the packet reception time and departing time to measure the packet delay at this switch. At the end of each resource sharing interval, the delay measurement module calculates the delay satisfaction ratio for each flow, which is utilized by the resource sharing module to make future resource allocation decisions.

Table 5.2: Parameters of switch

Network traffic condition	Off-peak		Peak
Switch forwarding resources (Mbps)	1250		833
Available resource block size (Mbps)	3.5	5	0.83
Pre-allocated resources (Mbps / flow)	18	15	15
Minimal resource allocation ratio	0.72	0.6	0.9
Pre-allocated condition	Over-provisioning	Matching traffic	

The detailed parameters of the traffic flows are given in Table 5.1. To simulate flows of different services, we divided the 50 flows into 3 subsets with different per-hop delay requirements. The duration of each resource sharing interval is 5 ms. From Table 5.1, the average traffic rate at this switch is 750 Mbps, considering 50 flows traversing the switch with average traffic rate of 15 Mbps. In reality, the traffic volumes are different during peak and off-peak hours, which leads to varying average resource utilization of the switch. To simulate these variations, we set the switch forwarding resources for the off-peak hour case as 1250 Mbps (i.e., 60% resource utilization) and for the peak hour case as 833 Mbps (i.e., 90% resource utilization). In addition to traffic load variations, we consider different pre-allocated resource conditions that impact the demand of resource sharing, including over-provisioning and on-demand matching cases. The pre-allocated conditions are determined by both the pre-allocated forwarding resources and the average traffic rate of each flow (15 Mbps in our simulation). Parameters of these cases are given in Table 5.2, where the minimal resource allocation ratio is the ratio of overall pre-allocated resources over the switch forwarding resources.

5.4.2 Performance of Resource Demand Estimation

To evaluate the accuracy of resource demand estimation, we divide the traffic data trace into two sets, one is used as training data set (including $489,930$ observations) and the remaining is used for testing. Each observation represents the flow transmission status during one resource sharing interval, consisting of input vector (including the number of arrived packets, initial queue length, and the measured delay bound to satisfy the per-hop delay violation ratio), and the output (allocated forwarding resources). In the training stage, all the observations are fed into an LR module given by (5.6), and a weight vector $\mathbf{W} \in \mathbb{R}^5$ is obtained by the LR module. Then, we apply the online resource demand

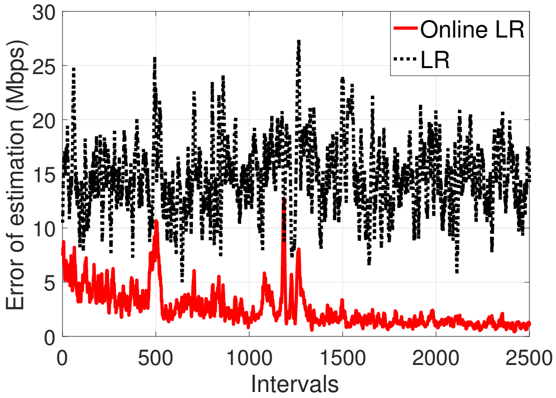

Figure 5.6: Error of resource demand estimation.

estimation scheme with the initial vector $\mathbf{W}$ to estimate the required resources. The estimation error in each interval represents the difference between an estimated amount and an actual amount of allocated resources. To show the performance of resource demand estimation, the moving average of errors during 10 consecutive intervals is shown in Fig. 5.6 where online LR refers to the proposed method and LR method utilizing $\mathbf{W}$ for estimation without weights update. Without the weights update, the error of LR method varies between 15% and 35% . However, due to the weights update in (5.7), the estimation error of online LR shows a decreasing trend in Fig. 5.6, and becomes stable around 1% after 1500 intervals. Therefore, the proposed resource estimation module is capable of providing an accurate resource demand estimation for the allocation of available resources module.

5.4.3 Delay Reduction via Resource Sharing

To evaluate the delay performance of the resource sharing scheme, we determine the proportion of flows that meet their delay requirements (i.e., satisfying $\rho_{f,m}^{(n)} \geq 1 - \varepsilon$), referred to as delay guaranteed proportion. In terms of cost, we measure the resource allocation fraction including both pre-allocated resources and allocated available resources. For comparison between different resource sharing schemes, the cumulative distribution functions (CDF) of packet delay, delay guaranteed proportion, and resource allocation gain which is the ratio of delay guaranteed proportion over resource allocation ratio are calculated for $4,000$ intervals.

The proposed MAB based resource sharing scheme is compared with an on-switch resource allocation approach, the dynamic WFQ (D-WFQ) method [52]. The D-WFQ is an enhanced version of WFQ under dynamic traffic conditions, considering the differentiated per-hop delay requirements of traffic flows. We also compare with the resource sharing scheme without resource demand estimation module (MAB-WO). In the MAB-WO method, the allocation of available resources module directly uses the per-hop delay requirements, the predicted traffic arrival, and the queue lengths as input, instead of using the estimated resource demands as shown in Fig. 5.3. To show the performance improvement achieved by utilizing available resources, we simulate the packet transmissions with only pre-allocated resources given in Table 5.2 (pre-allocated). If the forwarding resources are fully used, we make a fixed resource sharing decision to achieve the optimal accumulative delay guaranteed proportion during the simulation intervals (optimal).

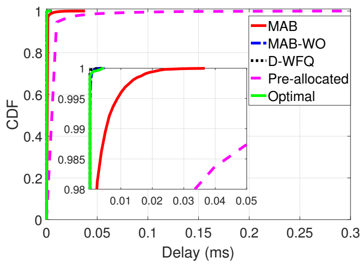

Figure 5.7: Packet delay in a off-peak hour condition with matching traffic resources.

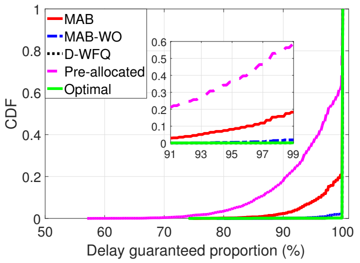

Figure 5.8: Delay guaranteed proportion in a off-peak hour condition with matching traffic resources.

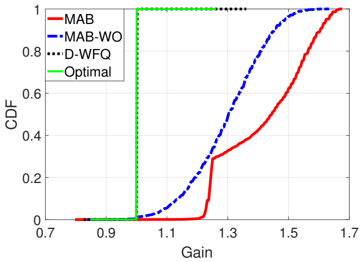

Figure 5.9: Resource allocation gain in a off-peak hour condition with matching traffic resources.

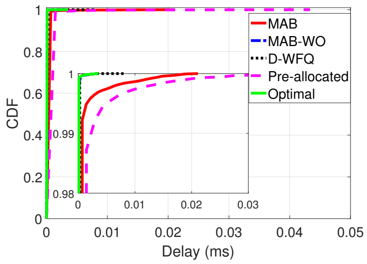

Figure 5.10: Packet delay in a off-peak hour condition with over-provisioning resources.

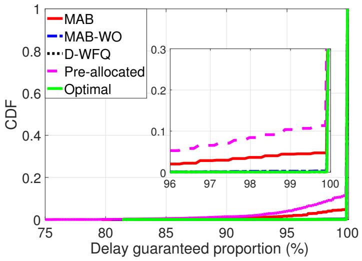

Figure 5.11: Delay guaranteed proportion in a off-peak hour condition with over-provisioning traffic resources.

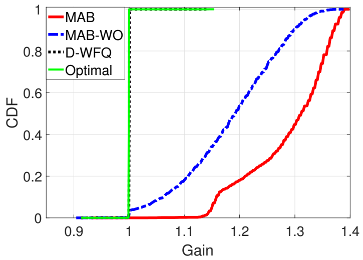

Figure 5.12: Resource allocation gain in a off-peak hour condition with over-provisioning traffic resources.

When the flows are pre-allocated with the resources matched their packet arrival rates, the packet delay experiences a long-tailed distribution as shown in Fig. 5.7. It leads to the delay guaranteed proportion less than 90% among 20% of intervals, as shown in Fig. 5.8. However, during the off-peak hours, all four resource sharing schemes achieve nearly 100% delay guaranteed proportion through allocating available resources. Since MAB and MAB-WO schemes require less forwarding resources than D-WFQ and optimal schemes, they have higher resource allocation gain, as shown in Fig. 5.9. This is because MAB schemes can learn to allocate the available resources to the flows for the highest marginal gain, through the greedy allocation scheme in Algorithm 8. Comparing MAB and MAB-WO schemes, the resource demand estimation module makes it possible to identify the flows with stringent delay requirements, and set higher upper bounds of allocated available resource blocks. As more available resources can be allocated to the flows with stringent requirements, MAB outperforms MAB-WO in most resource sharing intervals in terms of resource allocation gain. In over-provisioning situations, we simulate the case that each flow is pre-allocated with more resources than necessary on average, as shown in Fig. 5.10 to Fig. 5.12. Better delay guaranteed proportion is shown in Fig. 5.11, compared with the on-demand matching case. However, due to the high resource allocation ratio in the over-provisioning case, its resource allocation gain is lower than the on-demand matching case.

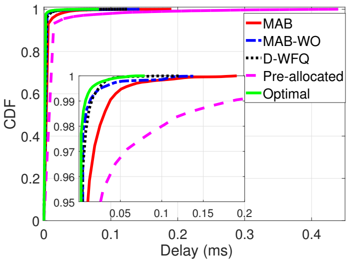

Figure 5.13: Packet delay in a peak hour condition with matching traffic resources.

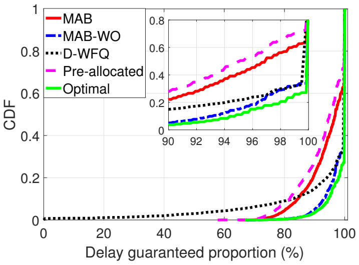

Figure 5.14: Delay guaranteed proportion in a peak hour condition with matching traffic resources.

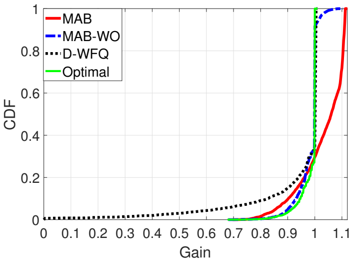

Figure 5.15: Resource allocation gain in a peak hour condition with matching traffic resources.

In the peak-hour scenario, all four resource sharing schemes tend to make full use of the resources to support the heavy traffic, and experience delay guaranteed proportion higher than 90% among 80% of the simulation intervals, as shown in Fig. 5.13 to Fig. 5.15. Comparing with transmission upon pre-allocated resources, the MAB method has a higher delay guaranteed proportion by avoiding the long-tailed distribution of packet delay. Due to the small amount of available resources during peak hours, it is unlikely to allocate the required amount of available resources to a flow, based on the estimated resource demand. Thus, comparing with the off-peak hour case, the difference between MAB and MAB-WO methods vanishes during peak hours.

The highest resource allocation gain is obtained when 100% delay guaranteed proportion is achieved with the pre-allocated resources. Thus, resource allocation gain is bounded by the reciprocal of minimal resource allocation ratio (i.e., the ratio of overall pre-allocated resources over the switch forwarding resources). Since the optimal and D-WFQ schemes make full use of available forwarding resources, the resource allocation gain is bounded by 1. However, MAB and MAB-WO schemes can achieve better delay guaranteed proportion with less resources, and their gains outperform the optimal and D-WFQ schemes. Due to the errors at the starting stage of resource demand estimation, the MAB scheme experiences over-allocation of available resources compared with the subsequent stable stages. Thus, there is a step change in the resource allocation gain, which becomes negligible with more fine-grained available resource block sizes. As shown in Fig. 5.9, Fig. 5.11, and Fig. 5.14, the step shrinks with the available resource block size decreases from 5 Mbps, to 3.5 Mbps, and to 0.83 Mbps, respectively.

5.4.4 Adaptive Resource Sharing

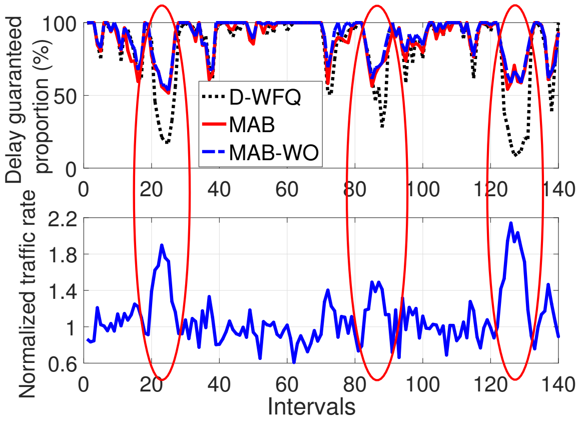

Figure 5.16: Resource sharing performance in a peak hour condition with matching traffic resources.

Due to insufficient forwarding resources, instantaneous delay degradation happens during traffic peaks, such as in intervals 25, 87, and 125, as shown in Fig. 5.16, where the real-time packet arrival rate is normalized to the average traffic rate. Although all three schemes experience degraded performance, both MAB and MAB-WO schemes outperform D-WFQ, due to the proactive resource allocation.

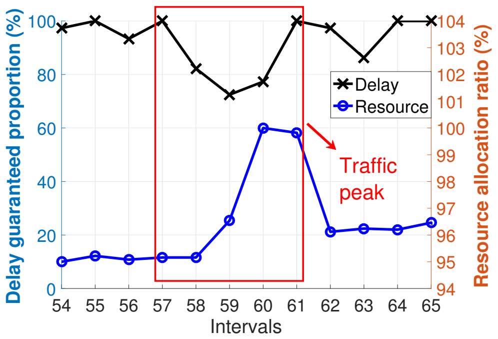

Figure 5.17: The response of the proposed resource sharing scheme to traffic peak 1.

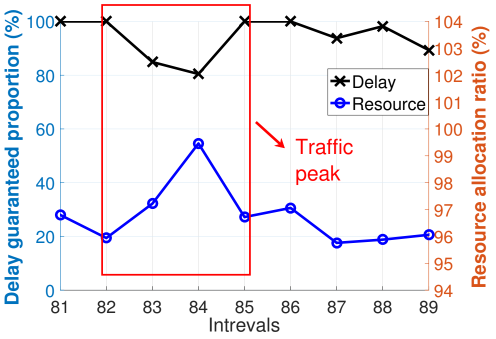

Figure 5.18: The response of the proposed resource sharing scheme to traffic peak 2.

Furthermore, to adapt to traffic dynamics, it requires a fast response of resource allocation when traffic peak comes. Fig. 5.17 and Fig. 5.18 show how the MAB scheme adaptively allocates resources. During traffic peak, packet delay increases, and more forwarding resources are needed. To obtain the response time to traffic peak, we focus on the time instants where resource allocation ratio starts to increase or falls back to a stable level. Comparing with the traffic peak duration, it is observed that the response time is at most one interval when traffic peak appears and disappears, which is 5 ms in our simulation. Therefore, the proposed resource sharing scheme demonstrates adaptation capability to traffic dynamics, by allocating suitable resources to the flows.

5.5 Summary

In this chapter, to support interaction intensive services in HVNets, we have investigated forwarding resource sharing scheme. We have proposed a novel learning-based proactive resource sharing scheme to maximize resource utilization efficiency with delay satisfaction. Two modules for estimating online resource demand and allocating available resources are developed jointly to achieve efficient resource sharing at each network switch. To learn the implicit relation between the allocated resources and differentiated delay requirements from traffic flows of different services, a multi-armed bandit learning-based resource allocation scheme is proposed, which enables fast and proactive resource adjustment upon traffic variations. During the data transmission, delay satisfaction ratios are measured as the reward feedback to refine the learning parameters for better convergence. The proposed scheme is proved to be asymptotically approaching the optimal strategy with the polynomial time complexity. Extensive simulation results are presented to demonstrate both the advantages of the proposed resource sharing scheme over conventional schemes and the robustness to traffic dynamics.

Chapter 6

Conclusions and Future Works

In this chapter, we summarize the main contributions of this thesis, and discuss future research directions.

6.1 Main Research Contributions

In this thesis, we have investigated the resource allocation scheme to guarantee three types of vehicular services: content downloading, safety message transmissions, and interaction-intensive services. In specific, to overcome the challenges faced by vehicular service provisioning, we have studied the cooperative caching placement for content downloading services, joint communication and sensing resource allocation for safety message transmissions, and forwarding resource sharing scheme in core networks for interaction-intensive services. The main research contributions of this thesis are summarized as follows:

1. We have designed a cooperative edge caching scheme to support various vehicular content downloading services. In particular, two types of vehicular content requests have been considered, i.e., location-based and popular contents, with different delay requirements. The proposed scheme allows vehicles to fetch one content from multiple caching servers cooperatively, which can be optimized by finding an optimal cooperative content placement that determines the placing locations and proportions for all contents. Based on the theoretical analysis of transmission delay and service cost, we have formulated an optimization problem of cooperative content placement to minimize the overall transmission delay and service cost. We have devised an ACO-based scheme to solve this multi-objective MMKP and achieve a near-optimal

solution. Simulation results have been provided to validate the performance of the proposed scheme, including its convergence and optimality of caching, while guaranteeing low transmission delay and service cost;

2. To support the vehicular safety message transmissions, we have proposed the TARA framework, including a group-level resource reservation module and a vehicle-level resource allocation module. Particularly, the resource reservation module is designed to allocate resources to support different types of message transmission for each vehicle group at the first level, where a supervised learning model is devised to learn the implicit relation between the resource demand and message transmission requests. To obtain the training data, we have proposed a SRA scheme, making the optimal allocation decisions on sensing resources and communication resources based on historical network information. Extensive simulation results have been provided to demonstrate the effectiveness of the proposed TARA framework in terms of the high reliability and low latency for message transmission and the high quality of collective perception service. In addition, the proposed TARA framework has been proved to be able to achieve a satisfying performance in a real-time manner and be readily applied into large-scale vehicular networks;

3. To support interaction intensive services in HVNets, we have designed a forwarding resource sharing scheme to guarantee delay-sensitive packet transmissions between vehicles and management controllers. A learning-based proactive resource sharing scheme has been proposed for the core communication networks, where the available forwarding resources at a switch are proactively allocated to the traffic flows in order to maximize the efficiency of resource utilization with delay satisfaction. The resource sharing scheme consists of two joint modules, estimation of resource demands and allocation of available resources. Considering the distinct features of each traffic flow, the resource demand estimation module has been developed based on linear regression scheme, mimicking the mapping relation between traffic flow status and required resources. Moreover, a multi-armed bandit learning-based resource allocation scheme has been proposed to enable fast resource allocation adjustment to traffic arrival dynamics. The proposed scheme has been proved to be asymptotically approaching the optimal strategy, with polynomial time complexity. Extensive simulation results have been presented to demonstrate the effectiveness of the proposed resource allocation scheme in terms of delay satisfaction, traffic adaptiveness, and resource allocation gain.

6.2 Future Works

For the future research, there are some interesting related topics as follows:

1. **Multi-dimensional resource allocation for computation-intensive applications:** We will investigate the multi-dimensional resource allocation scheme to support computation-intensive applications in HVNets, where computing and sensing capabilities are enabled at both vehicles and edge nodes. Considering the distinct sensing data source, data processing, and service latency requirements of multifarious applications, e.g., collective perception, cooperative positioning, and collision avoidance applications, a joint communication, computing, and sensing resource allocation is required for HVNets. Due to the coupling of different resource types in service provisioning, it is challenging to make an efficient resource allocation scheme, which is capable to keep pace with dynamic environments;

2. **Resource allocation for space-terrestrial networks:** Considering the urging growth of devices, terrestrial networks can hardly satisfy the stringent requirements, due to the limitations of geographic-constrained infrastructure deployment and scarce spectrum. Hence, the integration of space and terrestrial networks has attracted substantial attention from both academia and industry, which exploits complementary advantages of different network segments. In specific, the low latency requirement can be satisfied by terrestrial networks, while the globally seamless coverage can be achieved by space networks, i.e., satellite constellation. To support the stringent service requirements, the multi-dimensional resources, e.g., communication, caching, and computing resources, from different network segments should be efficiently managed, considering resource utilization with service requirement satisfaction. However, the high mobility of satellites leads to dynamic network topology, which makes it challenging to achieve an efficient resource allocation scheme.

References

[1] F. Tang, Y. Kawamoto, N. Kato, and J. Liu, "Future intelligent and secure vehicular network toward 6G: Machine-learning approaches," *Proc. IEEE*, vol. 108, no. 2, pp. 292–307, Feb. 2020.

[2] F. Lyu, H. Zhu, N. Cheng, H. Zhou, W. Xu, M. Li, and X. Shen, "Characterizing urban vehicle-to-vehicle communications for reliable safety applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2586–2602, Jun. 2020.

[3] J. E. Siegel, D. C. Erb, and S. E. Sarma, "A survey of the connected vehicle landscape—architectures, enabling technologies, applications, and development areas," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2391–2406, Aug. 2018.

[4] Y. Wang and J. Zheng, "Connectivity analysis of a highway with one entry/exit and multiple roadside units," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 11 705–11 718, Dec. 2018.

[5] J. Liu, H. Guo, J. Xiong, N. Kato, J. Zhang, and Y. Zhang, "Smart and resilient EV charging in SDN-Enhanced vehicular edge computing networks," *IEEE J. Select. Areas Commun.*, vol. 38, no. 1, pp. 217–228, Jan. 2020.

[6] C. M. Silva, B. M. Masini, G. Ferrari, and I. Thibault, "A survey on infrastructure-based vehicular networks," *Mobile Inform. Syst.*, vol. 2017, pp. 1–28, Aug. 2017.

[7] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang, and Y. Zhou, "Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2377–2396, Jun. 2015.

[8] N. Cheng, F. Lyu, J. Chen, W. Xu, H. Zhou, S. Zhang, and X. Shen, "Big data driven vehicular networks," *IEEE Netw.*, no. 99, pp. 1–8, Dec. 2018.

[9] G. Raja, A. Ganapathisubramaniyan, S. Anbalagan, S. B. M. Baskaran, K. Raja, and A. K. Bashir, “Intelligent reward-based data offloading in next-generation vehicular networks,” *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3747–3758, May 2020.

[10] N. Cheng, N. Lu, N. Zhang, X. Zhang, X. Shen, and J. W. Mark, “Opportunistic WiFi offloading in vehicular environment: A game-theory approach,” *IEEE Trans. Intell. Transp. Syst*, vol. 17, no. 7, pp. 1944–1955, Jul. 2016.

[11] R. F. Atallah, C. M. Assi, and M. J. Khabbaz, “Scheduling the operation of a connected vehicular network using deep reinforcement learning,” *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1669–1682, May 2019.

[12] J.-K. Bae, M.-C. Park, E.-J. Yang, and D.-W. Seo, “Implementation and performance evaluation for DSRC-based vehicular communication system,” *IEEE Access*, vol. 9, pp. 6878–6887, Dec. 2020.

[13] B. Abidi, F. M. Moreno, M. El Haziti, A. Hussein, A. Al Kaff, and D. M. Gomez, “Hybrid V2X communication approach using WiFi and 4G connections,” in *IEEE ICVES*, Sept. 2018, pp. 1–5.

[14] P. E. Ross, “Europe’s smart highway will shepherd cars from rotterdam to vienna,” https://spectrum.ieee.org/transportation/advanced-cars/europes-smart-highway-will-shepherd-cars-from-rotterdam-to-vienna/, accessed Dec. 30, 2014.

[15] H. Wu, M. Palekar, R. Fujimoto, R. Guensler, M. Hunter, J. Lee, and J. Ko, “An empirical study of short range communications for vehicles,” in *Proceedings of ACM VANET*, Sept. 2005, pp. 83–84.

[16] U. of Michigan, “connected ann arbor experiment website,” https://mcity.umich.edu/our-work/on-the-road/.

[17] “Towards 5G initiative welcomes qualcomm, shows fast results,” https://www.qualcomm.com/news/releases/2017/02/24/towards-5g-initiative-welcomes-qualcomm-shows-fast-results/, accessed Feb. 24, 2017.

[18] “UK CITE UK connected intelligent transport environment,” https://www.ukcite.co.uk/.

[19] 3GPP, “Study on enhancement of 3GPP support for 5G V2X services,” *TR 22.886*, 2018.

[20] L. Yao, A. Chen, J. Deng, J. Wang, and G. Wu, “A cooperative caching scheme based on mobility prediction in vehicular content centric networks,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5435–5444, Jun. 2018.

[21] Z. Zhao, L. Guardalben, M. Karimzadeh, J. Silva, T. Braun, and S. Sargento, “Mobility prediction-assisted over-the-top edge prefetching for hierarchical VANETs,” *IEEE J. Sel. Areas Commun.*, vol. 36, no. 8, pp. 1786–1801, Aug. 2018.

[22] L. Wang, H. Wu, Y. Ding, W. Chen, and H. V. Poor, “Hypergraph-based wireless distributed storage optimization for cellular D2D underlays,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 10, pp. 2650–2666, Oct. 2016.

[23] D. J. MacKay, “Fountain codes,” *IEE Proceedings-Communications*, vol. 152, no. 6, pp. 1062–1068, Dec. 2005.

[24] Y. Zhang, R. Wang, M. S. Hossain, M. F. Alhamid, and M. Guizani, “Heterogeneous information network-based content caching in the internet of vehicles,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 10 216–10 226, Oct. 2019.

[25] S. Zhang, P. He, K. Suto, P. Yang, L. Zhao, and X. Shen, “Cooperative edge caching in user-centric clustered mobile networks,” *IEEE Trans. Mobile Comput.*, vol. 17, no. 8, pp. 1791–1805, Aug. 2018.

[26] Z. Qu, B. Ye, B. Tang, S. Guo, S. Lu, and W. Zhuang, “Cooperative caching for multiple bitrate videos in small cell edges,” *IEEE Trans. Mobile Comput.*, vol. 19, no. 2, pp. 288–299, Feb. 2020.

[27] Q. Li, W. Shi, X. Ge, and Z. Niu, “Cooperative edge caching in software-defined hyper-cellular networks,” *IEEE J. Sel. Areas Commun.*, vol. 35, no. 11, pp. 2596–2605, Nov. 2017.

[28] W. Jiang, G. Feng, and S. Qin, “Optimal cooperative content caching and delivery policy for heterogeneous cellular networks,” *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1382–1393, May 2017.

[29] F. Lyu, H. Zhu, H. Zhou, L. Qian, W. Xu, M. Li, and X. Shen, “MoMAC: Mobility-aware and collision-avoidance MAC for safety applications in VANETs,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10 590–10 602, Nov. 2018.

[30] Y. Wang, J. Shi, L. Chen, B. Lu, and Q. Yang, “A novel capture-aware TDMA-based MAC protocol for safety messages broadcast in vehicular ad hoc networks,” *IEEE Access*, vol. 7, pp. 116 542–116 554, Aug. 2019.

[31] S. O. Gani, Y. P. Fallah, G. Bansal, and T. Shimizu, "A study of the effectiveness of message content, length, and rate control for improving map accuracy in automated driving systems," *IEEE Trans. Intell. Transport. Syst.*, vol. 20, no. 2, pp. 405–420, Feb. 2019.

[32] H. P. Luong, M. Panda, H. L. Vu, and B. Q. Vo, "Beacon rate optimization for vehicular safety applications in highway scenarios," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 524–536, Jan. 2018.

[33] M. Noor-A-Rahim, G. M. N. Ali, H. Nguyen, and Y. L. Guan, "Performance analysis of IEEE 802.11 p safety message broadcast with and without relaying at road intersection," *IEEE Access*, vol. 6, pp. 23 786–23 799, Apr. 2018.

[34] A. Ullah, S. Yaqoob, M. Imran, and H. Ning, "Emergency message dissemination schemes based on congestion avoidance in VANET and vehicular FoG computing," *IEEE Access*, vol. 7, pp. 1570–1585, Dec. 2018.

[35] H. D. R. Albonda and J. Pérez-Romero, "An efficient RAN slicing strategy for a heterogeneous network with eMBB and V2X services," *IEEE Access*, vol. 7, pp. 44 771–44 782, Mar. 2019.

[36] S. K. Tayyaba, H. A. Khattak, A. Almogren, M. A. Shah, I. U. Din, I. Alkhalifa, and M. Guizani, "5G vehicular network resource management for improving radio access through machine learning," *IEEE Access*, vol. 8, pp. 6792–6800, Jan. 2020.

[37] S. Zhang, W. Quan, J. Li, W. Shi, P. Yang, and X. Shen, "Air-ground integrated vehicular network slicing with content pushing and caching," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2114–2127, Sept. 2018.

[38] G. Sun, G. O. Boateng, D. Ayepah-Mensah, G. Liu, and J. Wei, "Autonomous resource slicing for virtualized vehicular networks with D2D communications based on deep reinforcement learning," *IEEE Syst. J.*, vol. 14, no. 4, pp. 4694–4705, Dec. 2020.

[39] Y. Chen, Y. Wang, M. Liu, J. Zhang, and L. Jiao, "Network slicing enabled resource management for service-oriented ultra-reliable and low-latency vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7847–7862, Jul. 2020.

[40] Y. Park, T. Kim, and D. Hong, "Resource size control for reliability improvement in cellular-based V2V communication," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 379–392, Jan. 2019.

[41] L. F. Abanto-Leon, A. Koppelaar, C. B. Math, and S. H. de Groot, "Impact of quantized side information on subchannel scheduling for cellular V2X," in *Proc. IEEE VTC Spring*, Jun. 2018, pp. 1–5.

[42] L. F. Abanto-Leon, A. Koppelaar, and S. H. de Groot, "Subchannel allocation for vehicle-to-vehicle broadcast communications in mode-3," in *Proc. IEEE WCNC*, Apr. 2018, pp. 1–6.

[43] S. Park, B. Kim, H. Yoon, and S. Choi, "RA-eV2V: Relaying systems for LTE-V2V communications," *J. Commun. Netw.*, vol. 20, no. 4, pp. 396–405, Aug. 2018.

[44] ETSI EN 302 637-3, "Intelligent transport systems (ITS); vehicular communications; basic set of applications; part 3: Specifications of decentralized environmental notification basic service," Aug. 2018.

[45] ETSI EN 302 637-2, "Intelligent transport systems (ITS); vehicular communications; basic set of applications; part 2: Specification of cooperative awareness basic service," Apr. 2019.

[46] ETSI TR 103 562 V2.1.1, "Intelligent transport systems (ITS); vehicular communications; basic set of applications; analysis of the collective perception service (CPS)," Dec. 2019.

[47] C. Dovrolis, D. Stiliadis, and P. Ramanathan, "Proportional differentiated services: Delay differentiation and packet scheduling," *IEEE/ACM Trans. Networking*, vol. 10, no. 1, pp. 12–26, Feb. 2002.

[48] E. Davies, M. A. Carlson, W. Weiss, D. Black, S. Blake, and Z. Wang, "An architecture for differentiated services," *IETF, RFC 2475*, Dec. 1998.

[49] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 19, no. 4, pp. 1–12, Sept. 1989.

[50] M. Shreedhar and G. Varghese, "Efficient fair queuing using deficit round-robin," *IEEE/ACM Trans. Networking*, vol. 4, no. 3, pp. 375–385, Jun. 1996.

[51] M. Menth, M. Mehl, and S. Veith, "Deficit round robin with limited deficit savings (DRR-LDS) for fairness among TCP users," in *Proc. MMB*, Jan. 2018, pp. 188–201.

[52] C.-C. Li, S.-L. Tsao, M. C. Chen, Y. Sun, and Y.-M. Huang, "Proportional delay differentiation service based on weighted fair queuing," in *Proc. IEEE ICCCN*, Oct. 2000, pp. 418–423.

[53] K. Bao, J. D. Matyjas, F. Hu, and S. Kumar, "Intelligent software-defined mesh networks with link-failure adaptive traffic balancing," *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 2, pp. 266–276, Jun. 2018.

[54] K. Chen and L. Huang, "Timely-throughput optimal scheduling with prediction," *IEEE/ACM Trans. Networking*, vol. 26, no. 6, pp. 2457–2470, Dec. 2018.

[55] M. D. F. De Grazia, D. Zucchetto, A. Testolin, A. Zanella, M. Zorzi, and M. Zorzi, "QoE multi-stage machine learning for dynamic video streaming," *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 1, pp. 146–161, Mar. 2018.

[56] J. Li, W. Shi, N. Zhang, and X. Shen, "Delay-aware VNF scheduling: A reinforcement learning approach with variable action set," *IEEE Trans. Cogn. Commun. Netw., DOI: 10.1109/TCCN.2020.2988908*, pp. 1–15, Apr. 2020.

[57] L. Wang, X. Wang, M. Tornatore, K. J. Kim, S. M. Kim, D.-U. Kim, K.-E. Han, and B. Mukherjee, "Scheduling with machine-learning-based flow detection for packet-switched optical data center networks," *IEEE J. Opt. Commun. Netw.*, vol. 10, no. 4, pp. 365–375, Apr. 2018.

[58] I.-S. Comşa, S. Zhang, M. E. Aydin, P. Kuonen, Y. Lu, R. Trestian, and G. Ghinea, "Towards 5G: A reinforcement learning-based scheduling solution for data traffic management," *IEEE Trans. Netw. Service Manag.*, vol. 15, no. 4, pp. 1661–1675, Dec. 2018.

[59] J. Luo, X. Su, and B. Liu, "A reinforcement learning approach for multipath TCP data scheduling," in *Proc. IEEE CCWC*, Jan. 2019, pp. 0276–0280.

[60] J. Zhu, Y. Song, D. Jiang, and H. Song, "A new deep-Q-learning-based transmission scheduling mechanism for the cognitive Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2375–2385, Aug. 2017.

[61] Y. Hui, Z. Su, T. H. Luan, and J. Cai, "A game theoretic scheme for optimal access control in heterogeneous vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 12, pp. 4590–4603, Jan. 2020.

[62] W. Zhuang, Q. Ye, F. Lyu, N. Cheng, and J. Ren, "SDN/NFV-empowered future IoV with enhanced communication, computing, and caching," *Proc. IEEE*, vol. 108, no. 2, pp. 274–291, Feb. 2020.

[63] Z. Su, Y. Hui, Q. Xu, T. Yang, J. Liu, and Y. Jia, "An edge caching scheme to distribute content in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5346–5356, Jun. 2018.

[64] L. T. Tan, R. Q. Hu, and L. Hanzo, "Twin-timescale artificial intelligence aided mobility-aware edge caching and computing in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3086–3099, Apr. 2019.

[65] Y. Zhang, C. Li, T. H. Luan, Y. Fu, W. Shi, and L. Zhu, "A mobility-aware vehicular caching scheme in content centric networks: Model and optimization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3100–3112, Apr. 2019.

[66] J. Chen, Q. Ye, W. Quan, S. Yan, P. T. Do, P. Yang, W. Zhuang, X. Shen, X. Li, and J. Rao, "SDATP: An SDN-based traffic-adaptive and service-oriented transmission protocol," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 2, pp. 756–770, Jun. 2020.

[67] T. H. Luan, X. Shen, and F. Bai, "Integrity-oriented content transmission in highway vehicular ad hoc networks," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2562–2570.

[68] H. Zhou, B. Liu, T. H. Luan, F. Hou, L. Gui, Y. Li, Q. Yu, and X. Shen, "Chaincluster: Engineering a cooperative content distribution framework for highway vehicular communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2644–2657, Dec. 2014.

[69] 3GPP TR 36.786 V14.0.0, "Vehicle-to-everything (V2X) services based on LTE; user equipment (UE) radio transmission and reception," Mar. 2017.

[70] W. Xu, W. Shi, F. Lyu, H. Zhou, N. Cheng, and X. Shen, "Throughput analysis of vehicular internet access via roadside WiFi hotspot," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3980–3991, Apr. 2019.

[71] IEEE standard 802.11, "Wireless lan medium access control (MAC) and physical layer (PHY) specifications," Jan. 1999.

[72] D. Ciullo, V. Martina, M. Garetto, and E. Leonardi, "How much can large-scale video-on-demand benefit from users' cooperation?" *IEEE/ACM Trans. Networking*, vol. 23, no. 6, pp. 1846–1861, Dec. 2014.

[73] M. Dorigo, G. D. Caro, and L. M. Gambardella, "Ant algorithms for discrete optimization," *Artificial life*, vol. 5, no. 2, pp. 137–172, 1999.

[74] J. Branke, J. Branke, K. Deb, K. Miettinen, and R. Słowiński, *Multiobjective optimization: Interactive and evolutionary approaches.* Springer Science & Business Media, 2008, vol. 5252.

[75] D. Alanis, P. Botsinis, Z. Babar, H. V. Nguyen, D. Chandra, S. X. Ng, and L. Hanzo, "A quantum-search-aided dynamic programming framework for pareto optimal routing in wireless multihop networks," *IEEE Trans. Commun.*, vol. 66, no. 8, pp. 3485–3500, Aug. 2018.

[76] M. Mavrovouniotis, F. M. Müller, and S. Yang, "Ant colony optimization with local search for dynamic traveling salesman problems," *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1743–1756, Jul. 2017.

[77] H. Pirkul, "A heuristic solution procedure for the multiconstraint zero-one knapsack problem," *NRL*, vol. 34, no. 2, pp. 161–172, Apr. 1987.

[78] Y. Ni, L. Cai, J. He, A. Vinel, Y. Li, H. Mosavat-Jahromi, and J. Pan, "Toward reliable and scalable internet-of-vehicles: Performance analysis and resource management," *Proc. of the IEEE*, vol. 108, no. 2, pp. 324–340, Feb. 2020.

[79] H. Wu, F. Lyu, C. Zhou, J. Chen, L. Wang, and X. Shen, "Optimal UAV caching and trajectory in aerial-assisted vehicular networks: A learning-based approach," *IEEE J. Select. Areas Commun.*, vol. 38, no. 12, pp. 2783–2797, Dec. 2020.

[80] J. Chen, H. Wu, P. Yang, F. Lyu, and X. Shen, "Cooperative edge caching with location-based and popular contents for vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 10 291–10 305, Sept. 2020.

[81] T. Mangel, O. Klemp, and H. Hartenstein, "5.9 GHz inter-vehicle communication at intersections: a validated non-line-of-sight path-loss and fading model," *EURASIP J. Wirel. Commun. Netw.*, vol. 2011, no. 182, pp. 1–11, Nov. 2011.

[82] T. Mangel, F. Schweizer, T. Kosch, and H. Hartenstein, "Vehicular safety communication at intersections: Buildings, non-line-of-sight and representative scenarios," in *Proc. IEEE WONS*, Jan. 2011, pp. 35–41.

[83] H. M. El-Sallabi, "Fast path loss prediction by using virtual source technique for urban microcells," in *Proc. IEEE VTC Spring*, May 2000, pp. 2183–2187.

[84] L. F. Abanto-Leon, A. Koppelaar, C. B. Math, and S. H. de Groot, "System level simulation of scheduling schemes for C-V2X mode-3," *arXiv:1807.04822*, 2018.

[85] "Didi chuxing GAIA initiative." https://gaia.didichuxing.com/.

[86] 3GPP TR 36.885 V14.0.0, "Study on LTE-based V2X services," *3GPP*, Jul. 2016.

[87] L. Xu, H. Wang, W. Lin, T. A. Gulliver, and K. N. Le, "GWO-BP neural network based OP performance prediction for mobile multiuser communication networks," *IEEE Access*, vol. 7, pp. 152 690–152 700, Oct. 2019.

[88] L. Zhang, F. Wang, T. Sun, and B. Xu, "A constrained optimization method based on BP neural network," *Neural. Comput. Appl.*, vol. 29, no. 2, pp. 413–421, Jan. 2018.

[89] Y. Wu, Y. Wang, W. Hu, X. Zhang, and G. Cao, "Resource-aware photo crowdsourcing through disruption tolerant networks," in *Proc. IEEE ICDCS*, Aug. 2016, pp. 374–383.

[90] S. Manjunath and G. Raina, "Stability and performance of compound TCP with a proportional integral queue policy," *IEEE Trans. Contr. Syst. Technol.*, vol. 27, no. 5, pp. 2139–2155, Sept. 2019.

[91] M. Dong, Q. Li, D. Zarchy, P. B. Godfrey, and M. Schapira, "PCC: Re-architecting congestion control for consistent high performance," in *Proc. USENIX*, May 2015, pp. 395–408.

[92] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "BBR: Congestion-based congestion control," *Commun. ACM*, vol. 60, no. 2, pp. 58–66, Feb. 2017.

[93] S. Floyd, K. Ramakrishnan, and D. L. Black, "The addition of explicit congestion notification (ECN) to IP," *IETF, RFC 3168*, Sept. 2001.

[94] G. Carlucci, L. De Cicco, and S. Mascolo, "Controlling queuing delays for real-time communication: the interplay of E2E and AQM algorithms," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 46, no. 3, pp. 1–7, Jul. 2018.

[95] J. Han, K. Xue, Y. Xing, P. Hong, and D. S. Wei, "Measurement and redesign of BBR-based MPTCP," in *Proc. ACM SIGCOMM*, Aug. 2019, pp. 75–77.

[96] C. Gao, V. Rajabian-Schwart, W. Zhang, G. Xue, and J. Tang, "How would you like your packets delivered? An SDN-enabled open platform for QoS routing," in *Proc. IEEE/ACM IWQoS*, Jun. 2018, pp. 1–10.

[97] J. W. Guck, A. Van Bemten, and W. Kellerer, "DetServ: network models for real-time QoS provisioning in SDN-based industrial environments," *IEEE Trans. Netw. Service Manag.*, vol. 14, no. 4, pp. 1003–1017, Sept. 2017.

[98] Q. Ye, W. Zhuang, X. Li, and J. Rao, "End-to-end delay modeling for embedded VNF chains in 5G core networks," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 692–704, Feb. 2019.

[99] O. Avner and S. Mannor, "Multi-user communication networks: A coordinated multi-armed bandit approach," *IEEE/ACM Trans. Networking*, vol. 27, no. 6, pp. 2192–2207, Dec. 2019.

[100] A. Mukherjee, S. Misra, V. S. P. Chandra, and M. S. Obaidat, "Resource-optimized multi-armed bandit based offload path selection in edge UAV swarms," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4889–4896, Jun. 2019.

[101] P. Auer, "Using confidence bounds for exploitation-exploration trade-offs," *J. Mach. Learn. Res*, vol. 3, pp. 397–422, Nov. 2002.

[102] W. Chu, L. Li, L. Reyzin, and R. Schapire, "Contextual bandits with linear payoff functions," in *Proc. AISTATS*, Apr. 2011, pp. 208–214.

[103] M. M. Tajiki, S. Salsano, L. Chiaraviglio, M. Shojafar, and B. Akbari, "Joint energy efficient and QoS-aware path allocation and VNF placement for service function chaining," *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 1, pp. 374–388, Mar. 2019.

[104] "OpenFlow switch specification," *Open Networking Foundation*, Mar. 2015.

[105] M. Shahbaz, S. Choi, B. Pfaff, C. Kim, N. Feamster, N. McKeown, and J. Rexford, "Pisces: A programmable, protocol-independent software switch," in *Proc. ACM SIGCOMM*, Aug. 2016, pp. 525–538.

[106] O. Alhussein, P. T. Do, Q. Ye, J. Li, W. Shi, W. Zhuang, X. Shen, X. Li, and J. Rao, "A virtual network customization framework for multicast services in NFV-enabled core networks," *IEEE J. Select. Areas Commun.*, vol. 38, no. 6, pp. 1025–1039, Jun. 2020.

[107] O. Alhussein and W. Zhuang, "Robust online composition, routing and NF placement for NFV-enabled services," *IEEE J. Select. Areas Commun.*, vol. 38, no. 6, pp. 1089–1101, Jun. 2020.

[108] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proc. ACM WWW*, Apr. 2010, pp. 661–670.

[109] X. Yan and X. Su, *Linear regression analysis: theory and computing.* World Scientific, 2009.

[110] Y. Feng and D. P. Palomar, "A signal processing perspective on financial engineering," *Found. Trends Signal Process.*, vol. 9, no. 1–2, pp. 1–231, 2016.

[111] E. Hazan, A. Rakhlin, and P. L. Bartlett, "Adaptive online gradient descent," in *Proc. NIPS*, Dec. 2007, pp. 65–72.

[112] L. Zhou, "A survey on contextual multi-armed bandits," *arXiv:1508.03326*, pp. 1–43, Aug. 2015.

[113] J. Vermorel and M. Mohri, "Multi-armed bandit algorithms and empirical evaluation," in *Proc. ECML*, Oct. 2005, pp. 437–448.

[114] P. Yang, N. Zhang, S. Zhang, L. Yu, J. Zhang, and X. Shen, "Content popularity prediction towards location-aware mobile edge caching," *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 915–929, Apr. 2019.

[115] "Pycharm." [Online]. Available: https://www.jetbrains.com/pycharm/

Appendix A

Delay Analysis of Content Delivery

A.1 Delay Analysis of LF Content Delivery

Since content of $\mathbb{LF}^w$ is requested right after the vehicle enters coverage of RSU W_w, the duration of vehicle staying in each RSU is determined by its average speed and the RSU coverage range, i.e., $T_{R_w} = D_R/E\left[v\right]$. According to Fig. 3.1, the time duration under the MBS can be divided into three segments, denoted by $T_{B_1} = D_{M_1}/E\left[v\right]$, $T_{B_2} = D_{M_2}/E\left[v\right]$ and $T_{B_3} = D_{M_3}/E\left[v\right]$.

For $f \in \mathbb{LF}^1$, its content placement scheme is $\left[s_{R_1}^f, s_{R_2}^f, s_B^f\right]$. Without loss of generality, we classify the caching states into four types, based on the caching conditions of two RSUs (i.e. $s_{R_w}^f > 0$ or $s_{R_w}^f = 0, w = 1, 2$). The effective transmission time segments for MBS and RSU can be defined correspondingly. For instance, if $s_{R_1}^f > 0$ and $s_{R_2}^f > 0$, we have $T_1^f = T_{R_1}$, $T_2^f = T_B^1 = T_{B_2}$, $T_3^f = T_{R_2}$, $T_4^f = T_B^2 = T_{B_3}$, where T_B^n means the duration of VU accessing to MBS at the n-th time slot. The volume of downloaded data from MBS during T_B^n is denoted by $s_{B,n}^f = \left\lfloor T_B^n/t_B^f \right\rfloor$. For $f \in \mathbb{LF}^2$, since file only needs to be cached at RSU W_2 and MBS, there are two types of caching states (i.e. $s_{R_2}^f > 0$ or $s_{R_2}^f = 0$).

In addition to the connection duration and transmission rate, the number of packets downloaded during the n-th time segment, $S_n^f, n = 1, 2, ..., N_t^f$, is bounded by the content placement scheme, $\left[s_{R_1}^f, s_{R_2}^f, s_B^f\right]$. Based on the time segments sequence and the content placement, the average transmission delay of file f, $\overline{D}^f$, can be calculated following the content downloading process shown in Fig. 3.4. Here, we calculate the $\overline{D}^f$ in the case of caching file f, $f \in \mathbb{LF}^1$, at both RSUs as an example, shown as (A.1).

$$\overline{D}^f = \begin{cases} S_f \cdot t^f_{R_1} & \text{if } s^f_{R_1} \geq S_f \\ T^f_1 + t^f_B \cdot \left(S_f - s^f_{R_1}\right) & \text{else if } s^f_{R_1} + \min\left(s^f_{B,1}, s^f_B\right) \geq S_f \\ T^f_1 + T^f_2 + t^f_{R_2} \cdot \left[S_f - s^f_{R_1} - \min\left(s^f_{B,1}, s^f_B\right)\right] & \text{else if } s^f_{R_1} + \min\left(s^f_{B,1}, s^f_B\right) + s^f_{R_2} \geq S_f \\ T^f_1 + T^f_2 + T^f_3 + & \text{else if } s^f_{R_1} + \min\left(s^f_{B,1} + s^f_{B,2}, s^f_B\right) \\ t^f_B \cdot \left(S_f - s^f_{R_1} - \min\left(s^f_{B,1}, s^f_B\right) - s^f_{R_2}\right) & + s^f_{R_2} \geq S_f \\ T^f_1 + T^f_2 + T^f_3 + t^f_B \cdot \max\left(s^f_B - s^f_{B,1}, 0\right) + t^f_{BL} \cdot & \text{else} \\ \left[S_f - s^f_{R_1} - s^f_{R_2} - \min\left(s^f_{B,1} + s^f_{B,2}, s^f_B\right)\right] & \end{cases} \tag{A.1}$$

In order to evaluate delay performance, we calculate the mean of total content download delay for LF files ($\overline{D}$)

$$\overline{D} = \sum_{w=1}^{W} \left[\sum^{f \in \mathbb{LF}^w} \left(N \cdot P_f \cdot P^f_{V2I} \cdot \overline{D}^f \right) \right], \tag{A.2}$$

where $N \cdot P_f \cdot P^f_{V2I}$ is the average number of VUs fetching file f through V2I connection. Through optimizing $S^f_{R_1}$, $S^f_{R_2}$, and S^f_B, we aim to minimize $\overline{D}$ for LF services.

A.2 Delay Analysis of PF Content Delivery

We assume that the number of vehicles driving into the coverage follows Poisson process and PF contents are requested according to content popularity. Thus, the locations where vehicles raise the request and start the transmission are uniformly distributed within MBS coverage. We consider the file download latency requirement for file f as D^f_R, and evaluate the average data volume that can be downloaded during D^f_R as S^f_D.

The content placement scheme for file f, $f \in \mathbb{PF}$, is $\left[s^f_{R_1}, s^f_{R_2}, s^f_B\right]$. Without loss of generality, we classify the caching states into four types, the possible handover locations for vehicles can be demonstrated by a duration sequence for handover segments in Table A.1, the duration of each segment is defined as H^f_n. Different from LF, since the PF transmission may start at any location, the time segment is not fixed for each case. Thus, we calculate the average T^f_n for each case, considering latency constraint, $\sum_{n=1}^{N^f_t} T^f_n = D^f_R$, where N^f_t is the number of segments that the vehicle can go through. To guarantee

Table A.1: Handover duration segments.

Case	Caching types	Handover segments
1	$s^f_{R_1} = s^f_{R_2} = 0$	$H^f_1 = (T_{B_1} + T_{R_1} + T_{B_2} + T_{R_2} + T_{B_3})$
2	$s^f_{R_1} = 0, s^f_{R_2} > 0$	$H^f_1 = (T_{B_1} + T_{R_1} + T_{B_2}), H^f_2 = T_{R_2}, H^f_3 = T_{B_3}$
3	$s^f_{R_1} > 0, s^f_{R_2} = 0$	$H^f_1 = T_{B_1}, H^f_2 = T_{R_1}, H^f_3 = (T_{B_2} + T_{R_2} + T_{B_3})$
4	$s^f_{R_1} > 0, s^f_{R_2} > 0$	$H^f_1 = T_{B_1}, H^f_2 = T_{R_1}, H^f_3 = T_{B_2}, H^f_4 = T_{R_2}, H^f_5 = T_{B_3}$

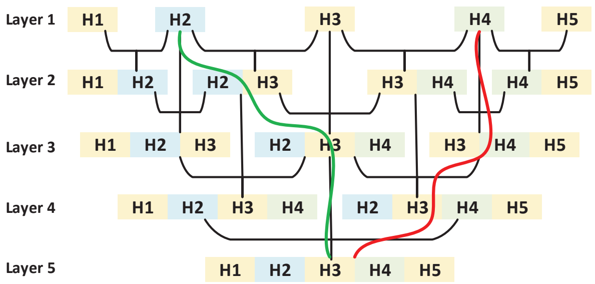

Figure A.1: An illustration of possible combinations of handover segments.

successful file transmission, we should make sure S^f_D for all possible combinations of time segments are larger than S_f, so we need to calculate the minimum value of S^f_D. Based on the handover duration sequences in Table A.1 and the content placement, the minimum downloaded data volume for file f within the delay constraint can be calculated for each case.

To evaluate the minimum downloaded data volume within D^f_R, we need to compare transmission rates between the MBS and RSU W_w $(w = 1, 2)$ to determine the worst case. To compare with RSU W_w, the average MBS transmission delay (t^f_{M,W_w}) of each packet can be calculated as the weighted average of t^f_B and t^f_{BL}

$$t^f_{M,W_w} = \frac{s^f_B \cdot t^f_B + \max\left(s^f_{R_w} - s^f_B, 0\right) \cdot t^f_{BL}}{s^f_B + \max\left(s^f_{R_w} - s^f_B, 0\right)}. \tag{A.3}$$

The minimum S^f_D corresponds to the download data volume of the worst case, which is defined as vehicle downloading from edge caching server with the lowest transmission rate, following by higher ones. We build a tree diagram to obtain the possible combinations of handover segments, with a given D^f_R. In Fig. A.1, we set a child node as the extension of its parent nodes. Then, each path from a layer 1 node to the last layer node is a possible segment combining process to achieve a given D_R^f. The tree diagram with 5 layers illustrates the handover for Case 4. Since there are 3 segments, (H_1^f,H_2^f, H_3^f), for Case 2 and Case 3, a 3-layer tree diagram can be built accordingly. For instance, the green line in Fig. A.1 represents a possible combination of handover segments for Case 4, when the ascending order of transmission time for one packet follows $W_1 > W_2 >$ MBS. The order indicates the worst case is downloading from RSU W_1, so the path starts from H_2^f. If $D_R^f > H_2^f$, the vehicle can drive out of the coverage of W_1 during the service, so it can extend to layer 2, i.e., $\left(H_2^f - H_3^f\right)$. Since the transmission from RSU W_2 is more time consuming than MBS, the extension directs to H_4^f, i.e., $\left(H_2^f - H_4^f\right)$ of layer 3. If $D_R^f > H_2^f + H_3^f + H_4^f$, we consider the combination of all handover segments, i.e., $\left(H_1^f - H_5^f\right)$ of layer 5. We compare D_R^f with the length of each node in one path from layer 1 to the last layer, the first node longer than D_R^f is one possible combination for transmission segments, which determines the specific transmission pattern. All possible combinations can be found by searching paths in the diagram. Next, we need to calculate the downloaded data volume of all possible transmission patterns and find the required $\left[s_{R_1}^f, s_{R_2}^f, s_B^f\right]$ for successful transmission.

A.2.1 Case 1

If $s_B^f > 0$, a vehicle firstly downloads content from caching server at MBS, then from the remote server if necessary. Otherwise, the vehicle directly downloads from the remote server. If $\left\lfloor D_R^f / t_B^f \right\rfloor < S_f$, file f cannot be successfully downloaded for this content placement case regardless of the value of s_B^f. Otherwise, file f may be successfully downloaded, which depends on s_B^f. Since $s_B^f \leq S_f$, we have $D_R^f \geq s_B^f \cdot t_B^f$ as the necessary condition for successful transmissions.

$$S_D^f = s_B^f + \left\lfloor \left(D_R^f - s_B^f \cdot t_B^f\right) / t_{BL}^f \right\rfloor . \tag{A.4}$$

In order to guarantee successful transmission, we let $S_D^f \geq S_f$, then we can obtain the required s_B^f.

$$s_B^f \geq \left(S_f + 1 - \frac{D_R^f}{t_{BL}^f}\right) \cdot \frac{t_{BL}^f}{t_{BL}^f - t_B^f} \tag{A.5}$$

where $D_R^f \geq S_f \cdot t_B^f$.

A.2.2 Case 2 and Case 3

For Case 2, vehicles can download content from caching server at the MBS or RSU W_2, then from the remote server if necessary. If $t_{R_2}^f \geq t_{M,W_2}^f$, the worst case is that vehicles download from W_2 as much time as possible, so the segments combination path is $\left(H_2^f \rightarrow \left(H_1^f + H_2^f + H_3^f\right)\right)$. Different ranges of D_R^f lead to different transmission patterns. For each pattern, the S_D^f can be estimated, then the required $s_{R_2}^f$ and s_B^f can be obtained.

1. *Pattern-1* – W_2: if $D_R^f \leq H_2^f$

$$\begin{cases} S_D^f = \left\lfloor\, D_R^f / t_{R_2}^f \right\rfloor \\ s_{R_2}^f \geq S_f \cdot \frac{H_2^f}{D_R^f}, s_B^f \geq 0 \end{cases} \tag{A.6}$$

2. *Pattern-2* – W_2 & MBS: if $H_2^f \leq D_R^f \leq s_B^f \cdot t_B^f + H_2^f$

$$\begin{cases} S_D^f = s_{R_2}^f + \left\lfloor \left(D_R^f - H_2^f\right) / t_B^f \right\rfloor \\ s_{R_2}^f \geq S_f + 1 - \frac{D_R^f - H_2^f}{t_B^f}, s_B^f \geq \frac{D_R^f - H_2^f}{t_B^f} \end{cases} \tag{A.7}$$

3. *Pattern-3* – W_2 & MBS & backhaul: if $D_R^f > s_B^f \cdot t_B^f + H_2^f$

$$\begin{cases} S_D^f = s_{R_2}^f + s_B^f + \left\lfloor \left(D_R^f - H_2^f - s_B^f \cdot t_B^f\right) / t_{BL}^f \right\rfloor \\ s_{R_2}^f + s_B^f \cdot \frac{t_{BL}^f - t_B^f}{t_{BL}^f} \geq S_f + 1 - \frac{D_R^f - H_2^f}{t_B^f} \end{cases} \tag{A.8}$$

If $t_{M,W_2}^f \geq t_{R_2}^f$, the worst case is that vehicles download from the MBS as much time as possible, so the segments combination path is $\max\left(H_1^f, H_3^f\right) \rightarrow \max\left(H_1^f, H_3^f\right) + H_2^f \rightarrow \left(H_1^f + H_2^f + H_3^f\right)$. The analysis of the required $s_{R_2}^f$ and s_B^f is similar to the case of $t_{R_2}^f \geq t_{M,W_2}^f$.

With a given D_R^f, we first find the possible transmission patterns for the worst case considering two possible relationships of t_{M,W_2}^f and $t_{R_2}^f$. Then, we get the required $s_{R_2}^f$ and s_B^f. For Case 3, the estimation of S_D^f and analysis of the required $s_{R_1}^f$ and s_B^f are similar to *Case 2*.

A.2.3 Case 4

Vehicles can download content from caching server at the MBS or RSU W_1, W_2, then from the remote server if necessary. By sorting $t^f_{R_1}$, $t^f_{R_2}$, and t^f_{M,W_w} in ascending order, we get the edge server order for the worst-case path. There are six possible paths for different orders. Two examples of segment combination paths are given as follows, in which *Path-1* (green line) and *Path-2* (red line) are shown in Fig. A.1. The corresponding transmission pattern path for each segment combination path can be obtained, we use W^W_w and W^P_w denote vehicle driving through whole or part of RSU W_w.

1. If $W_1 > W_2 >$ MBS:
   $H^f_2 \rightarrow \left(H^f_2 - H^f_3\right) \rightarrow \left(H^f_2 - H^f_4\right) \rightarrow \left(H^f_1 - H^f_5\right)$
   Pattern: $W^P_1 \rightarrow W^W_1 + MBS \rightarrow W^W_1 + MBS + W^P_2 \rightarrow W^W_1 + MBS + W^W_2$;

2. If $W_2 >$ MBS$> W_1$:
   $H^f_4 \rightarrow \left(H^f_3 - H^f_5\right) \rightarrow \left(H^f_2 - H^f_5\right) \rightarrow \left(H^f_1 - H^f_5\right)$
   Pattern: $W^P_2 \rightarrow W^W_2 + MBS \rightarrow W^W_2 + MBS + W^P_1 \rightarrow W^W_2 + MBS + W^W_1$.

Then, we can calculate the required content placement for guarantee of successful transmission.

Different from LF content caching, the objective of PF content placement is to satisfy latency requirements rather than to achieve the lowest download latency. Thus, when design the caching scheme for file $f \in \mathbb{PF}$ with specific latency requirement, the required content placement scheme for it is $\left[s^f_{R_1}, s^f_{R_2}, s^f_B\right]$, which can be obtained by the lower bound of content placement for each case.