Resource Allocation Optimisation for Vehicle-to-Everything Communications Underlaying Cellular Networks

Bintao Hu

First Supervisor: Prof. Xiaoli Chu

Second Supervisor: Prof. Jie Zhang

Department of Electronic and Electrical Engineering
University of Sheffield

This dissertation is submitted for the degree of
Doctor of Philosophy

Submission Date April 2022

I would like to dedicate this thesis and everything I do to my loving parents. I would not be who I am today without their love and support.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Bintao Hu
April 2022

Acknowledgements

I am thankful to my supervisors, Prof. Xiaoli Chu and Prof. Jie Zhang, who guided me tirelessly throughout my PhD. I want to especially thank my first supervisor, Prof. Xiaoli Chu, for her selfless dedication in helping me with my research works. Prof. Xiaoli Chu and Prof. Jie Zhang are both brilliant researchers who were always able to push ideas one step forward, provide valuable suggestions needed to accomplish a task, and share experiences about the research process. I want especially fortunate to work both of them and benefit from both of their perspectives.

The work in this thesis is result of collaboration with other people, Dr. Jianbo Du. Without her valuable suggestions and comments, their would not be this thesis.

Beyond direct collaborators on the works in this thesis, many other people contributed to my graduate work and made the four years PhD studies become an unforgettable experience. I want to thank the staff of Iquadrat team, who were provided helpful suggestions and comments on how to improve my system model for the social-aware vehicular networks. I would also thank Dr. Haonan Hu, who gave me many constructive suggestions on my research works when I visited Chongqing University of Posts Telecommunications in Chongqing, China, in 2020. Finally, I would thank the Wireless Communications Group of the University of Sheffield, which truly inspired me in my four years PhD studies and provided the opportunity for me to be a researcher in this area.

Last but not least, I would like to thank my lovely family: my mother Mrs Liping Jiao and my grandparents for unwavering support me throughout my life.

Abstract

With the ever-increasing demands for wireless communications from vehicles, vehicle-to-everything (V2X) communications have emerged to enable vehicle user equipment (V-UE) to communicate with other vehicles, pedestrians, and communication infrastructures. In addition, V-UEs are able to offload their computation-intensive applications to a cloud centre or a fog node for processing via a vehicle-to-infrastructure (V2I) link or vehicle-to-vehicle (V2V) link. In this case, resource management and offloading optimisation of V2X communications in conjunction with other advanced technologies, such as social networking and cloud/fog computing technologies, are the main paramount challenges for V2X communications underlaying cellular networks. In this thesis, we focus on two different vehicular system models related to the resource allocation and/or the offloading decisions optimisation.

The thesis presents three main research contributions. In the first contribution, we investigate how the social attributes of V-UEs in a vehicular network may affect the radio resource allocation mechanism, and the performance of V2X communications. We propose a social-aware clustering-based resource allocation mechanism for V2X communications, which maximises the sum V2I links capacity while guaranteeing the reliability of all V2V links.

In the second contribution, we study a vehicular network supported by a mixed cloud and fog computing system, where the queues at the fog node and the cloud centre are modelled following the M/M/1 and M/M/C queueing models, respectively. To minimise the maximum service delay (which includes the transmission delay, the queueing delay and the processing delay) among the V-UEs, we propose to jointly optimise the offloading decisions of all V-UEs and the computation resource allocation at the fog node while considering the V-UEs' mobility and queueing delays at the fog node and the cloud centre. This is achieved by devising a fireworks algorithm-based offloading decision optimisation algorithm in conjunction with a bisection method-based fog node computation resource allocation scheme.

In the third contribution, vehicle-carried fog nodes (V-FNs) are deployed in the fog layer and uplink communication resource blocks (RBs) are allocated among all cloud/fog

processing V-UEs. We jointly optimise the offloading decisions of all V-UEs, the computation resource allocation at all V-FNs, and the allocation of RB and transmission power for all V-UEs while considering the mobility of V-UEs and V-FNs. The joint optimisation is then solved by devising a fireworks algorithm-based offloading decisions optimisation scheme, in conjunction with a bisection method-based V-FNs computation resource allocation scheme and a clustering-based communication resource allocation scheme.

Table of contents

List of figures **xv**

List of tables **xvii**

List of abbreviations **xix**

1 Introduction **1**

1.1 Background and Motivation 1

1.1.1 Related Works 1

1.1.2 Motivation 4

1.2 Contributions 6

1.2.1 List of Publications 7

1.3 Thesis Organisation 8

2 Literature Review and Methodology **9**

2.1 LTE-V2X Communications Technologies 9

2.1.1 Types of LTE-V2X Application 10

2.1.2 Strengths of LTE-V2X 12

2.2 Vehicular Social Networks Technologies 13

2.2.1 Social Characteristics 14

2.3 Fog Computing Technologies 15

2.3.1 The Concept of Fog Computing 15

2.3.2 Characteristics of Fog Computing 15

2.3.3 Fog Computing in IoV Systems 17

2.4 Methodology 18

2.4.1 Basics of Social Theory 18

2.4.2 Queueing Theory 19

2.4.3 Matching Theory 22

2.4.4 Fireworks Algorithm 23

3 Social-Aware Resource Allocation Algorithm for V2X Communications 29
3.1 Introduction 29
3.2 System Model 30
3.2.1 Physical Domain 31
3.2.2 Social Domain 32
3.3 Problem Formulation 35
3.4 Algorithms of Matching under Preferences 36
3.4.1 The Proposed SACRA Algorithm 36
3.4.2 Social-Aware V2V Clustering 36
3.4.3 SACRA Algorithm 37
3.4.4 3-Dimensional Matching Problem 38
3.4.5 Complexity Analysis 41
3.5 Numerical Results and Analysis 41
3.6 Conclusions 45

4 Collaborative Computation Offloading and Computation Resource Allocation in Vehicular Networks Based on Mixed Cloud/Fog Computing Systems 47
4.1 Introduction 47
4.2 System Model 48
4.2.1 Vehicle-fog-cloud Architecture 48
4.2.2 Transmission Delay 49
4.2.3 Queueing Delay 50
4.2.4 Processing Delay 56
4.2.5 Service Delay 56
4.3 Problem Formulation and Proposed Algorithm 58
4.3.1 Problem Formulation 58
4.3.2 Mobility and Queueing Based Offloading Decision Optimisation Algorithm 59
4.3.3 Fog Computation Resource Allocation 62
4.4 Complexity Analysis 63
4.5 Simulation Results 64
4.6 Conclusion 68

5 Computation Offloading and Resource Allocation in Mixed Cloud/Vehicular-Fog Computing Systems 69
5.1 Introduction 69
5.2 System Model 70

5.2.1 Vehicle-fog-cloud Architecture 70
5.2.2 Local Processing 71
5.2.3 Fog Processing 72
5.2.4 Cloud Processing 75
5.3 Problem Formulation 76
5.4 Proposed Algorithm 78
5.4.1 Fireworks Algorithm based Offloading Optimisation and Resource Allocation Algorithm 78
5.4.2 Computation Resource Allocation 80
5.4.3 Communication Resource Allocation 83
5.5 Complexity Analysis 89
5.6 Simulation Results 89
5.7 Conclusion 102

6 Conclusions and Future Works 103
6.1 Main Findings of the Thesis 104
6.2 Future Works 105

References 109

List of figures

2.1 Types of LTE-V2X application 11
2.2 Sidelink communication scenarios 11
2.3 Cellular communication scenarios 11
2.4 Frame structure for 15kHz sub-carrier spacing 13
2.5 Illustration of social-aware V2X communication underlay cellular system . 15
2.6 Illustration of fog computing structure 16
2.7 A birth-and-death M/M/1 example 19

3.1 Illustration of V2X communications 30
3.2 Spectrum sharing between V2I link and V2V cluster of SACRA algorithm. 36
3.3 The sum V2I links capacity for all the V2I links versus vehicle velocity for different values of the strength of the social relationship δ. 43
3.4 The sum V2I links capacity for all the V2I links versus V2V SINR threshold γ_0. 44

4.1 System architecture of a three-layer computing system 49
4.2 Convergence of Algorithm 4.1, where $V = 5$. 65
4.3 Maximum service delay versus the number of V-UEs. 66
4.4 Maximum service delay versus the data size of the application. 67
4.5 Maximum service delay versus the processing density. 67

5.1 System architecture of a three-layer computing system. 71
5.2 Convergence of Algorithm 5.1, where $V = 5$. 91
5.3 Maximum service delay versus the number of V-UEs. 92
5.4 Maximum service delay versus the data size of the application. 93
5.5 Maximum service delay versus the processing density. 94
5.6 Maximum service delay versus the wired-link rate between BS and cloud server (Mbit/s). 95
5.7 Maximum total service delay versus the computation capability f_v^{cloud}. . . . 96

5.8 Maximum total service delay versus the computation capability F^{fog}. . . . 97
5.9 Maximum service delay versus the computation capability f_v^{local}. 98
5.10 Application processing success rate versus the velocity of each V-UE. . . . 99
5.11 Task processing success rate versus the velocity of each V-FN. 100
5.12 Comparison under different number of fireworks I. 101

List of tables

2.1 An overview of comparison between IEEE 802.11p and LTE-V2X 10

3.1 General Notation 34
3.2 Simulation Parameters 42

4.1 General Notation 57
4.2 Simulation Parameters 64

5.1 Channel Models for V2I/V2V link 72
5.2 Notation Definitions 77
5.3 Simulation Parameters 90

List of abbreviations

Acronyms / Abbreviations

3GPP	The 3rd Generation Partnership Project
5G	Fifth-generation
AWGN	Additive white Gaussian noise
BS	Base Station
C-V2X	Cellular Vehicle-to-everything
D2D	Device-to-device
DL	Downlink
FCFS	First-come first-served
FDM	Frequency Division Multiplexing
IoT	Internet-of-Thing
IoV	Internet-of-Vehicle
ITS	Intelligent Transportation System
LTE	Long Term Evolution
MIMO	Multiple-input and multiple-output
MINLP	Mixed integer nonlinear programming problem
mmWave	Millimeter Wave
OBU	On-Board Unit

QoS	Quality of Service
RB	Resource Block
RE	Resource Element
RP	Resource Pool
RSU	Roadside Unit
RTT	Road-trip time
Rx	Receiver
SDN	Software-Defined Networking
SINR	Signal-to-interference-plus-noise ratio
SL	Sidelink
TDM	Time Division Multiplexing
Tx	Transmitter
UL	Uplink
V-FN	Vehicle-carried fog node
V-UE	Vehicle user equipment
V2I	Vehicle-to-Infrastructure
V2N	Vehicle-to-Networks
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VFC	Vehicular fog computing
VSN	Vehicular social network

Chapter 1

Introduction

The utilisation of V2X communications technologies for ITS is regarded as one of the most promising solutions to meet the ever-increasing cellular traffic and future demands. Accordingly, V2X communications, which are expected to enable reliable and low-latency communication services for vehicles, are attracting a great amount of interest from both industry and academia. In this thesis, there will be a focus on optimal radio resource allocation, computation resource allocation and application offloading decisions in different system models based on V2X communication technologies. In this chapter, the related work will be introduced followed by a conclusion of the background and the motivation, then a summary of personal achievements. Finally, the thesis structure is outlined.

1.1 Background and Motivation

1.1.1 Related Works

In this thesis, there is a focus on the development of new resource allocation optimisation algorithms for V2X communications, along with optimisation algorithms that include application offloading decisions making. In this section, a summary of the latest research work in these areas is provided.

As presented in previous works [1–4], the problem of resource allocation and power control for V2X communications as well as including offloading decisions optimisation can be modelled as a MINLP problem, which is an NP-hard problem. Therefore, the analysis and performance enhancement of resource allocation optimisation algorithms, as well as including application offloading optimisation algorithms, has attracted attention and research in both academia and industry, which consists of two aspects (i.e., social-aware resource allocation; the joint optimisation of resource allocation and offloading decision making).

Social-aware resource allocation:

Early works include [5–7], in which the authors consider simplified radio resource allocation schemes for V2X communications under a vehicular network system. Generally, social-aware resource allocation schemes were only considered for D2D communications in [8–12]. In [5], many-to-one matching game based proximity-aware resource allocation scheme was proposed for V2V communications, where all V-UEs were first grouped into different zones according to their traffic patterns and proximity information. Afterwards, the resources were allocated to each V2V link within each zone. In [7], the authors proposed hypergraph-based resource allocation to enhance the throughput of V-UE, while in [6] the authors proposed a resource allocation for V2V communications based on the latency and reliability requirements of V2V links. The authors in [8] proposed a distributed resource allocation scheme for D2D communications which improves the system rate by considering interference graph in the physical domain and social domain. In [9], the authors proposed a social-aware D2D resource allocation scheme based on the bipartite graph matching theory to increase the D2D transmission rate. Where D2D links are formed according to the proximity between users in the physical domain, they correspond to user social relationships in the social domain. The authors in [10] proposed social-aware spectrum sharing [11] and caching helper selection scheme by adopting a genetic algorithm (GA) to maximise the capacity of D2D based multicast. The authors in [12] maximised the social-aware rate for D2D communications by exploiting the physical-social centrality in cluster formation and cluster head selection. However, the resource allocation schemes for V2X communications have ignored the influence of the social domain and the proposed social-aware resource allocation algorithms for D2D communications cannot be rapidly applied to high mobility vehicle users in V2X communications.

Joint optimisation of resource allocation and offloading decisions:

Resource allocation (including the allocation of transmit power, RB and computation resource) optimisation has been considered to reduce the latency and save energy for V2X communications systems [13, 14]. Earlier works, such as: [15–19], presented different cloud/fog/MEC based simplified offloading systems for V2X communications. In [15], the authors introduced a VFC system to reduce the average offloading latency of V-UEs while satisfying the application-specific requirements. In [16], the authors proposed a computation offloading scheme to minimise the utility of the MEC servers in a vehicular network while guaranteeing the quality of experience for the vehicular terminals, where the utility is a designed function of the latency and computation resources required for processing the tasks.

In [17], the authors proposed a distributed iterative scheme to minimise task completion delay and energy consumption at each UE side, while satisfying the constraints of task offloading deadline in a MEC system. The authors in [18] proposed a branch-and-bound algorithm to minimise the average response time for V-UEs in a fog-enabled real-time traffic management system. In [19], the authors introduced a mixed fog/cloud system for mobile applications offloading and proposed to minimise the maximum delay among all mobile devices by jointly optimising the offloading decisions, resource allocation and energy consumption. Despite their thoroughness, the above existing works have not sufficiently considered the mobility of vehicles that may increase the delay and energy consumption of computation offloading.

Considering the high mobility of vehicles, the authors in [20] proposed a one-to-many matching-based task offloading algorithm to minimise the total offloading delay of all the V-UEs in a vehicular edge computing system. In [21], the authors suggested a computation offloading algorithm based on ellipsoid method to jointly minimise the sum computation delay and the energy consumption of V-UEs in an MEC enabled vehicular network. Furthermore, the authors considered a non-dominated genetic scheme to minimise the total offloading delay of all V-UEs by jointly optimising the computation resource utilisation of multiple edge nodes in [22]. However, the queueing delay in their offloading models have been ignored because tasks may need to wait in queues at the computing servers before they can be processed, which is a limitation of the above existing works. Advanced offloading system models for V2X communications were considered in [23–26]. In [23], the authors suggested a greedy heuristic based computation offloading scheme to minimise the total latency of all tasks in a VFC system. In [24], the authors proposed a cloud-MEC based collaborative computation offloading scheme to maximise the system utility by jointly optimising the offloading decisions and computation resource allocation. In [25], a contract-matching-based task assignment algorithm was presented to minimise the total delay of all V-UEs by optimising the computation resource utilisation. The authors in [26] proposed a priority-aware task offloading strategy to minimise the total offloading time and waiting time of all tasks, where the computation offloading decisions were made according to the priority of each task. Since all offloaded tasks have to be transmitted over a wired or wireless channel to the cloud or fog servers in order to be processed, where the communication resource allocation should be considered to reduce the transmission delay for all tasks. However, in [23–26], communication resource allocation has been neglected in their computation offloading models, which may affect the offloading decisions for all V-UEs in the proposed system models. More complex scenarios were considered in [27–29] to jointly optimising the offloading decisions of all V-UEs and resource allocation (i.e., communication resource and computation resource). In [27], the authors designed a mixed cloud/fog computing system for mobile applications offloading and proposed to minimise the maximum cost of delay and energy consumption among all UEs by jointly optimising the offloading decisions and resource allocation. The authors in [29] advocated a scheme to maximise the utility of an edge computing system by jointly optimising the offloading decisions and computation resources of all the UEs, where the utility is designed as a sub-modular set function of the offloading decisions and resources required for processing the tasks. However, the interference management in multiple V-UE scenarios were not considered, which are limitation of these existing works.

1.1.2 Motivation

Based on the above research we identify the following V2X communications research challenges in respect to VSN and VFC systems, respectively.

Research Challenges for VSNs

Based on the previous research works, challenges in VSN can be broadly classified into three categories:

- **Network Management:** VSNs are dynamic in nature, not only are they all mobile users in the process, but V-UEs may actively subscribe to different online social networking services [30]. VSNs need to generate, collect, and share different contents rapidly. Taking account into share and forward similar contents according to the corresponding V-UEs over the imitated network bandwidth is still open issue, which may affected by the requirements, diversity, and selfishness variability of users, etc [31]. Furthermore, based on different subscribed services of users, VSNs may allow different third parties to utilise users' information, such as location, diversity, and mobility, which may lead to some potential security issues, like malicious attacks [32].
- **Mobility Management:** Mobility modelling is one of the important challenges in VSNs, which impacts connectivity links among nodes to share contents. The authors in [33, 34] investigated the network performance in VSNs, where the mobility of vehicles is restricted around specific social sports. In addition, as mentioned in [35], different mobility models in vehicular networks were proposed, i.e., car-following models and queue models, which have neglected to consider the social characteristics of users. It can be noted that, new mobility models should be designed for V2X communications in order to consider social attributes of users and make it compatible for future vehicular networks [36, 37].

- **Application and Resource Management:** Previous studies on social networks [38] identified that the connections among users are created based on their social attributes, such as friends, family members and so on. The VSN modelling and the corresponding performance can be evaluated according to the relevant social connections. Once connections among users have been established, interference management and resource allocation become the most significant problem for V2X communications in VSNs. Therefore, proper resource management for V2X communications in VSNs is the most effective solution to this problem [7, 39].

Research Challenges for VFC Systems

Using the above research work as a foundation, challenges in VFC systems can be broadly classified into two categories:

- **Performance Evaluation:** Many studies have evaluated application processing latency in a cloud/edge mixed computing system model, however, in most of the above works, the performance analysis is based on simulation results which neglect the queuing latency of the system model. The performance evaluation should consider the queuing delay of the FNs. In addition, a mobility model for V-UE should be applied to the VFC system models, which would enhance the effectiveness of the offloading scheme. The mobility model was also considered to analyse the performance in [40], however, the model is too simplified to track and update all information such as location and requirements for all V-UEs.

- **Performance Optimisation:** Previous research on offloading schemes have shown that it is not always optimal. Various improvement of offloading schemes under a VFC system have been proposed through the modified matching algorithm or genetic algorithm [21, 28]. However, all of the above works have concentrated on considering different features on VFC systems, such as service prices, while neglecting communication and computation resource allocation. Furthermore, in [24] performance evaluation and offloading optimisation scheme was based only on computation resource allocation. In [13, 21, 27], optimisation problems, which were formulated as several MINLP problems with different objectives (e.g., minimal latency, minimal energy consumption, etc), are NP-hard.

1.2 Contributions

In Chapter three, the thesis studies communication resource allocation in a social-aware cellular vehicular network scenario, which is in conjunction with V2X communications technologies. This research extended the radio resource allocation schemes for D2D communications to evaluate the impact of social attributes and high mobility of V-UEs in terms of resource allocation. Most existing works are aimed at improving the performance of D2D communications, and proposed radio resource allocation optimisation algorithms, while the high mobility of V-UEs has not been studied [8–12]. Then, we derive explicit expressions for the relationship between the physical layer and the social layer. Based on these expressions, we developed a two-step optimisation algorithm to maximise the sum V2I links capacity while guaranteeing that all V2V links are above a threshold. The first step is a clustering scheme whereby all V2V links are grouped into different clusters, and the second step is a resource allocation scheme based on a many-to-one matching algorithm to maximise the sum capacity of V2I links. Simulation results reveal that a stronger social relationship between vehicles in the same social community would result in a higher total V2I links capacity of the community.

In Chapter four, there is a focus on computation resource allocation and application offloading problems in a cloud/fog computing mixed vehicular networks system, where each V-UE may process its application by itself or offload its application to the fog node or a cloud centre. This is the first instance of the formulation of a multi-objective optimisation (i.e., the computation resource allocation and the offloading decisions) in conjunction with considering the high mobility of V-UEs and the queueing delays at the fog node and the cloud centre. Some of the previous works [15–19, 29] on cloud/fog based offloading of delay-sensitive applications neglected the mobility of vehicles, which may lead to increased latency and energy consumption. Other existing works [20–22, 25] have not adequately considered queuing latency and an application may need to wait at a fog node or a cloud server until all previous applications in the same queue have been processed by the corresponding server. Whereas in this chapter, the mobility of all V-UEs, as well as the queuing delays at the fog node and the cloud centre, are taken into account when investigating the offloading optimisation problem. The aim in this chapter is to minimise the maximum service delay (including the transmission delay, the queueing delay and the processing delay) among all V-UEs. To solve this problem, an optimisation algorithm is developed to find the optimal offloading decisions for all V-UEs, combined with a bisection method-based algorithm to optimise the computation resource allocation of fog node for all fog processing V-UEs. The performance of the proposed algorithms is evaluated through simulations in comparison with benchmarks, including pure local processing, fog processing, cloud processing and random processing. The simulation results have demonstrated that the maximum service delay is further reduced according to the proposed algorithms, which enables its potential implementation in real vehicular communications systems.

In Chapter five, there is a prioritisation of resource allocation and intensive-application offloading problems in a mixed cloud/VFC three-layer scenario, where each application could be either processed locally or offloaded to a V-FN or a cloud server for processing. Different from the most existing works have considered static fog nodes only, e.g., those embedded in RSUs, but fog nodes can be carried by moving vehicles in IoV systems, which can relieve pressure on stationary fog nodes at the roadside, provide more diverse services and extra computation resources for smart vehicles. Thus, in this chapter, all V-FNs are deployed in the fog layer instead of static fog nodes, while the offloading problem is studied with respect to each application of V-UE. It is intent of this part to minimise the maximum service delay (which includes the transmission delay, the queueing delay and the processing delay) of all V-UEs, thus guaranteeing the fairness among V-UEs. To solve this optimisation problem, we propose a novel iterative algorithm which decouples the joint optimisation problem into three sub-problems that optimise the offloading decisions for all V-UEs, the uplink communication resource allocation for all cloud/fog-computing V-UEs, and the computation resource allocation at each V-FNs, separately. This chapter demonstrates the optimality and convergence of the proposed fireworks-based algorithm and evaluate its performance through simulations. Simulation results show that our proposed schemes achieve a much lower maximum service delay than the benchmarks.

1.2.1 List of Publications

Publications

[1] B. Hu, and X. Chu, “Social-aware resource allocation for vehicle-to-everything communications underlaying cellular networks,” in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*. Apr. 2021, pp. 1–6.

Accepted

[2] B. Hu, J. Du, and X. Chu, "Enabling Low-latency Applications in Vehicular Networks Based on Mixed Fog/Cloud Computing Systems," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. Apr. 2022.

Ready to Submit

[3] Hu, Bintao, Du, Jianbo, and Chu, Xiaoli, "Computation Offloading and Resource Allocation in Mixed Cloud/Vehicular-Fog Computing Systems," in *2022 IEEE*

1.3 Thesis Organisation

The thesis is organised as follows: Chapter two reviews the existing works on LTE-V2X communications technologies, vehicular social networks technologies, fog computing technologies and the methodologies used in this thesis. Chapter three studies the radio resource management problems in a vehicular social network scenario. Social-aware clustering-based algorithm and matching-based algorithm are developed to optimise the RB utilisation and energy consumption for each V2V link. Chapter four focuses on the offloading decisions optimisation for all V-UEs, which is combined with the computation resource allocation optimisation for a fog node. Various queuing models are considered in a VFC system model in order to establish an effective low-latency offloading optimisation scheme. Chapter five extends the work in Chapter four, which involves the joint optimisation of offloading decisions, communication, and computation resource allocation for all V-UEs in a VFC system. A summary of the thesis, and discussion of future directions, is provided in Chapter six.

Chapter 2

Literature Review and Methodology

2.1 LTE-V2X Communications Technologies

In recent years, the emergent issues of road safety, urban congestion, traffic efficiency and environmental protection, due to the dramatic increase in the number of vehicles have become increasingly prominent [41]. In this context, V2X technology, which enables the connection of humans, vehicles, infrastructures and cloud/networks, has been proposed. Due to the lack of high rate and low latency internet connectivity on roadways (e.g., real-time road planning services), the use of existing low-cost communication links (i.e., V2V and V2I communications) to create, share, transmit information and manage resources, especially on motorways and in rural areas, remains one of the critical issues for 5G based vehicular cellular networks. The idea of LTE-V2X is first proposed by Datang Telecom Technology [42]. In 2005, Datang, Huawei and LG were driving the development of the LTE-V2X international standard under the auspices of 3GPP, which was adopted in the standardisation of 3GPP Rel. 14 and Rel. 15 [43]. Moreover, with the development of the 5G, the new radio V2X (NR-V2X) technology has been proposed, which is adopted in the standardisation of 3GPP Rel. 16, Rel. 17 and its evolution [44].

Compared with IEEE 802.11p technology, C-V2X as the core technology of LTE-V2X has attracted worldwide attention to meet the low latency, efficient resource utilisation, transmission rate and so on, which can be explained as follows [45–49]:

- **Low latency:** At low densities, IEEE 802.11p could satisfy transmissions with delay requirement of less than 100ms. However, at high densities, IEEE 802.11p cannot guarantee a low latency transmission service, while LTE-V2X Rel. 14 supports 20ms and LTE-V2X Rel. 15 can meet the service requirements with a latency of 10ms.

In addition to this, NR-V2X can support service latency requirements of 3ms in the future.

- **Efficient resource utilisation:** IEEE 802.11p only supports the TDM method in the time domain, while LTE-V2X may offer the TDM method and FDM method. Furthermore, compared with IEEE 802.11p, LTE-V2X supports multiple antenna technology, which further improves resource utilisation.
- **Transmission range:** IEEE 802.11p can support transmission services within a range of 100 meters, while LTE-V2X Rel.14 supports transmission within a range of 320 meters and LTE-V2X Rel.15 supports services within a range of 500 meters.
- **Transmission rates:** Between IEEE 802.11p and LTE-V2X, IEEE 802.11p can reach 6Mbit/s, LTE-V2X Rel.14 offers 31Mbit/s on a single carrier, while LTE-V2X Rel.15 can reach 300Mbit/s on multiple carriers.

The details of the comparison between IEEE 802.11p and LTE-V2X can be found in Table 2.1 [45–49].

Table 2.1 An overview of comparison between IEEE 802.11p and LTE-V2X

	IEEE 802.11p	**LTE-V2X**
Low latency	100ms (low density)	Rel. 14: 20 ms; Rel. 15: 10 ms
Reliability	Cannot guarantee	Rel. 14: >90%; Rel. 15: >95%
Coding	Convolution code	Turbo
Resource utilisation	TDM	TDM and FDM
Multiple antennas	Based on UE	Rel. 14: Not supported; Rel. 15: 2Tx/2Rx
Transmission range	100 m	Rel. 14: 320 m; Rel. 15: 500 m
Transmission rate	6 Mbit/s	Rel. 14: 30 Mbit/s; Rel. 15: 300 Mbit/s

Through the above analysis, it can be seen that LTE-V2X is better than IEEE802.11p in terms of system performance, larger communication range and less system interference for V-UEs moving at high speeds while ensuring low latency, high reliability and high transmission rate of large data volumes.

2.1.1 Types of LTE-V2X Application

3GPP TS 22.185 proposed that LTE-V2X contains four different types (i.e., V2P, V2I, V2V and V2N), which is shown in Fig. 2.1 [50]. V2V is the communication link between two different V-UEs, V2I is the communication link between a V-UE and a RSU or other infrastructures, V2P is the communication link between a V-UE and a pedestrian, and V2N is the communication link between a V-UE and an application server.

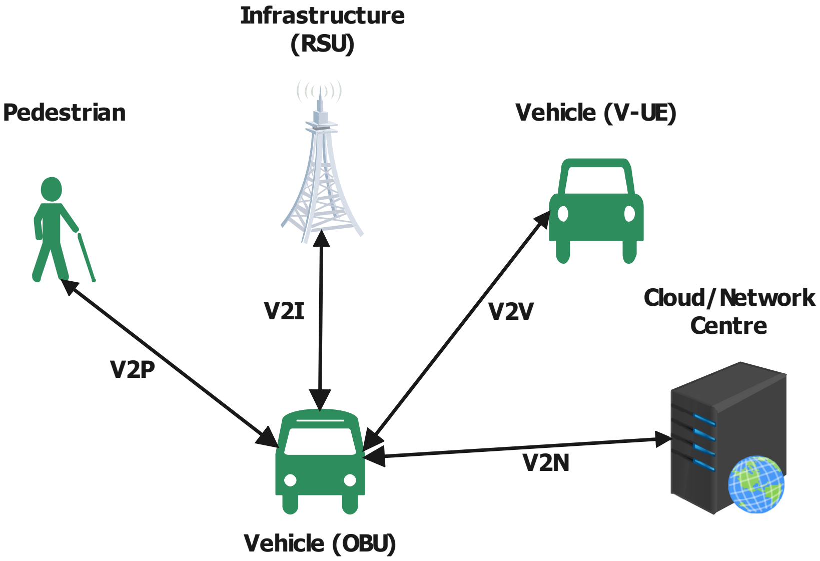

Fig. 2.1 Types of LTE-V2X application

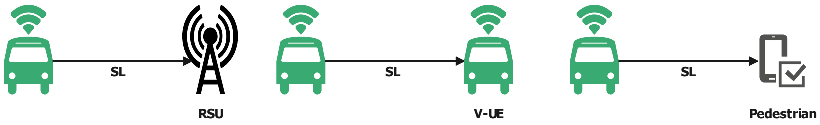

Fig. 2.2 Sidelink communication scenarios

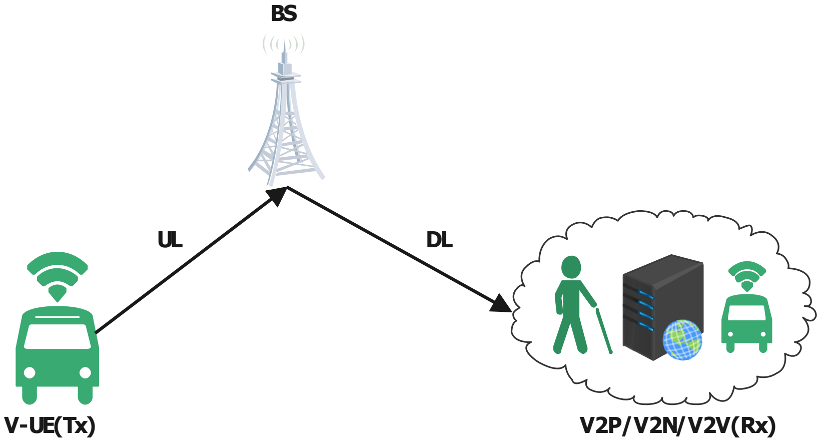

Fig. 2.3 Cellular communication scenarios

In addition, as shown in Fig. 2.2 and Fig. 2.3, there are two different communication mechanisms based on different V2X links or services, i.e., SL communication and cellular communication [51, 52]. SL communication is a direct communication between two terminals, without a BS involved in the transmission and reception of data traffic. Therefore, it can support the demand for low latency, high reliability V2X communication over short distances. In the case of the cellular communication mechanism, a V-UE transmits data to a BS via UL with the BS and forwards this data via DL between the BS and the receiver, thus allowing long-distance, large data size task and latency insensitive V2X communications.

2.1.2 Strengths of LTE-V2X

- **Frame Structure:** LTE-V2X frame structure is illustrated in Fig. 2.4, the hyperframes period is 10240ms and consists of 1024 frames of 10ms length. Each frame is made up of 10 sub-frames, each lasting 1 ms, and each sub-frame is made up of two 0.5 ms slots. The spectrum resources can be further divided into RBs, each of which has a bandwidth of 180 KHz and consists of 12 sub-carriers (i.e. a sub-carrier = 15 KHz) and 7 symbols. Continually LTE-V2X transmissions are scheduled over RBs among multiple V-UEs [53, 54].

- **Resource Allocation Scenario:** LTE-V2X has two resource allocation schedules, a decentralised resource allocation schedule and a centralised resource allocation schedule. For a decentralised resource allocation schedule, a V-UE selects resources from the RP while not allocating resources via the BS. Therefore, it supports V2X services with low latency and high-reliability requirements. For a centralised resource allocation schedule, a V-UE sends a request to the BS, which only allocates resources to the corresponding V-UE without sending information about the RP. It therefore increases the reliability of data transmission and the efficiency of resource allocation [49, 55, 56].

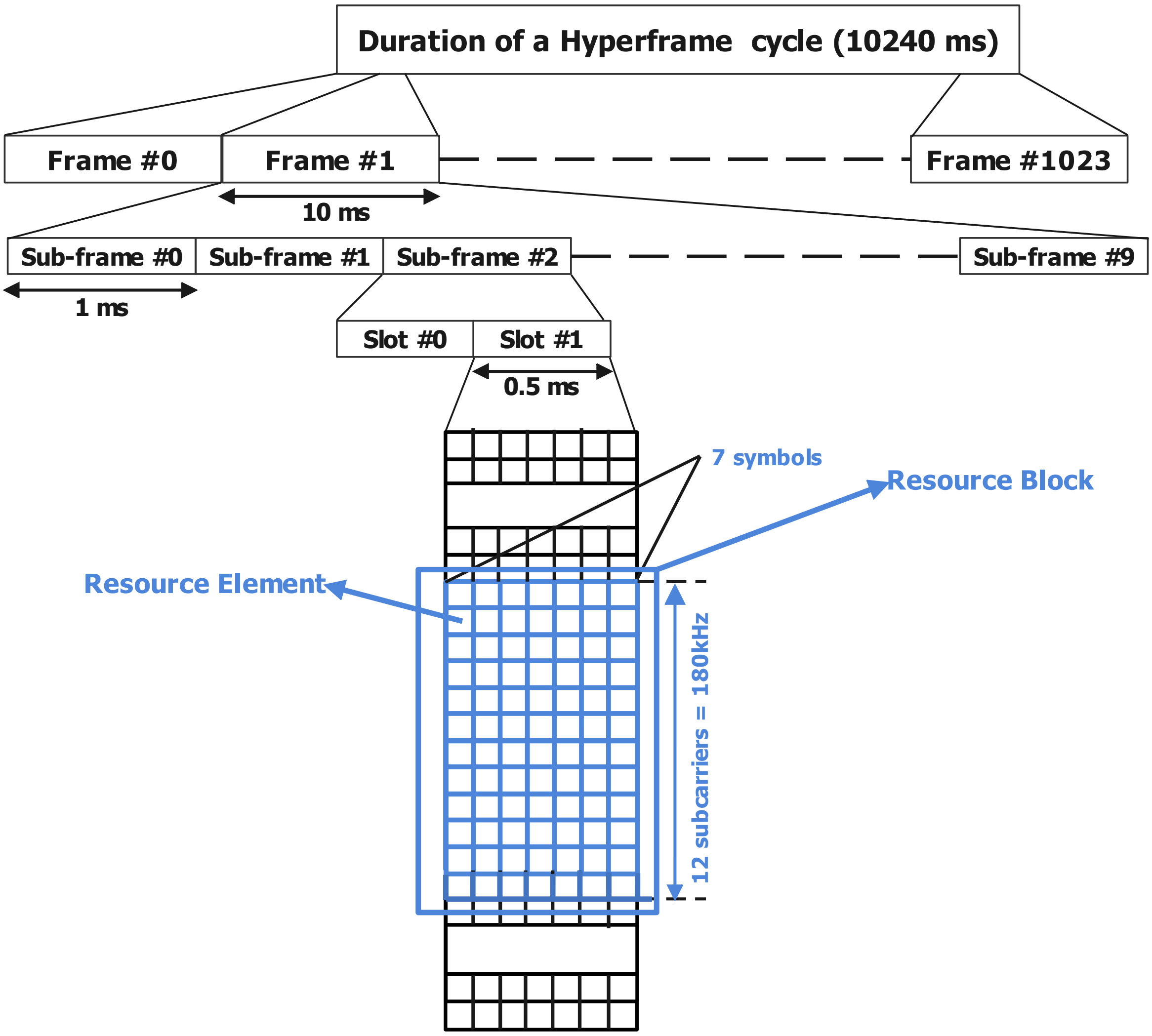

Fig. 2.4 Frame structure for 15kHz sub-carrier spacing

2.2 Vehicular Social Networks Technologies

By 2040, as vehicles are self-driving, it is expected that vehicle operators will not need a driving licence and that they will no longer be involved in controlling the speed, position, or direction of the vehicle [57]. In future, the efficient collection and allocation of network resources (i.e., communication and computation resources) from neighbouring V-UEs, based on a group of federated infrastructure providers cooperating to provide VSN services, will remain one of the key issues for VSNs.

2.2.1 Social Characteristics

VSNs are one of the main applications of mobile social networks (MSNs), which can be considered as a system for providing communication services involving social contact between V-UEs. And it is characterised as online social networks formed on Internet platforms such as Facebook and Twitter. As shown in Fig. 2.5, the social behaviours and structures of V2X-based VSNs include characteristics of social ties, social communities, degree centrality and bridges.

- **Social Ties:** In VSNs, social ties represent weak or strong relationships between different individuals (i.e., V-UEs, UEs, etc.) that are related to each other. Each social tie can be established between individuals through kinship, friendship, collegial cooperation and common behaviours or interests initiated by human users [58, 59].

- **Social Communities:** Social community is formed based on social relations among human users, and it defines clusters of V-UEs sharing the common social attributes (i.e., interests, behaviours and so on.). In VSNs, social communities may indicate different clusters by contents of locations, weather information, interests, or backgrounds . Thus, social community clustering for V-UEs may improve data transmission efficiency. Social communities are formed based on social relationships among human users, which define clusters of V-UEs with common social attributes (i.e. interests, behaviours, etc.) [60]. In VSNs, social communities represent different clusters through locations, weather information, interests, backgrounds, or other different contents, which in turn improves the efficiency of data transmission among all connected V-UEs [59].

- **Degree Centrality:** Degree centrality is a quantification of the relative structural importance of a central node in a social network [61]. A central node is a node that has a high capability to connect other nodes in the same social network. According to VSNs, selecting a node with a high degree of centrality can improve the chances of transmitting relative data to its corresponding destination. The most widely used measures are closeness centrality, betweenness centrality and bridging centrality [62, 63].

- **Bridges:** A bridge structure is a connection between two neighbouring social communities. In VSNs, each social community may have a large group of V-UEs, and a bridge is the edge of interaction between two social communities, which provides a potential way to connect these two social communities for information exchange.

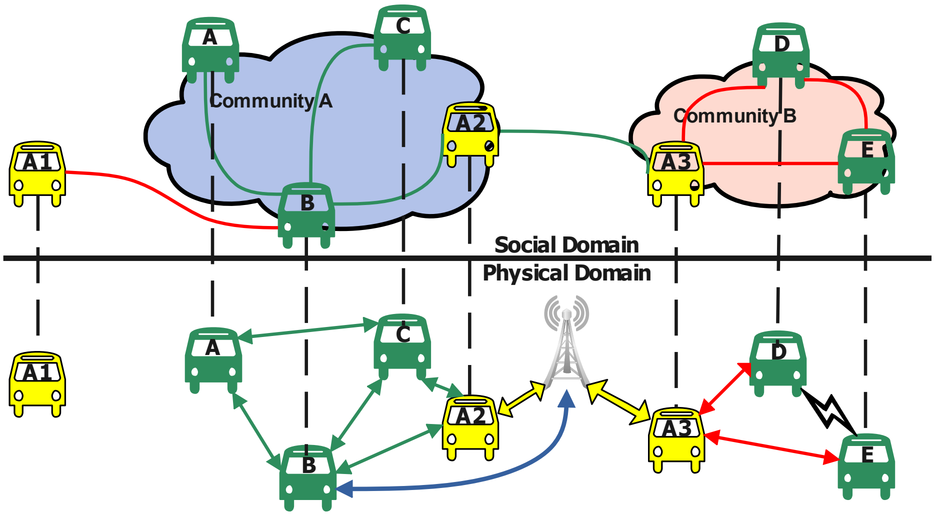

Fig. 2.5 Illustration of social-aware V2X communication underlay cellular system

2.3 Fog Computing Technologies

Internet-connected devices will triple in the next decade, from 11 billion in 2019 to 30 billion by 2030. This growth will also mean significant bandwidth consumption and higher latency for data processing [64]. Fog computing is a new technology released by Cisco in 2012 that can effectively overcome problems such as low latency and high bandwidth requirements [65].

2.3.1 The Concept of Fog Computing

Fog computing is defined as an extension of cloud computing and provides computation, data storage and application services closer to the end-user [66]. As shown in Fig. 2.6 fog computing acts as an arbitrator between the end-users and could centre which provides extra computing services, storage services and other networking services. In the fog layer, it includes various fog nodes which have less processing facilities and temporary storage. Based on using these fog nodes, end-users might obtain a real-time response for latency-sensitive applications. In addition, fog computing, like cloud computing, offers two computing mechanisms, namely infrastructure or platform as a service (I/PaaS) and software as a service (SaaS).

2.3.2 Characteristics of Fog Computing

- **Security:** Due to the current complex and dynamic wireless communication environment, i.e., bandwidth limitations and interference strength, distant backend servers

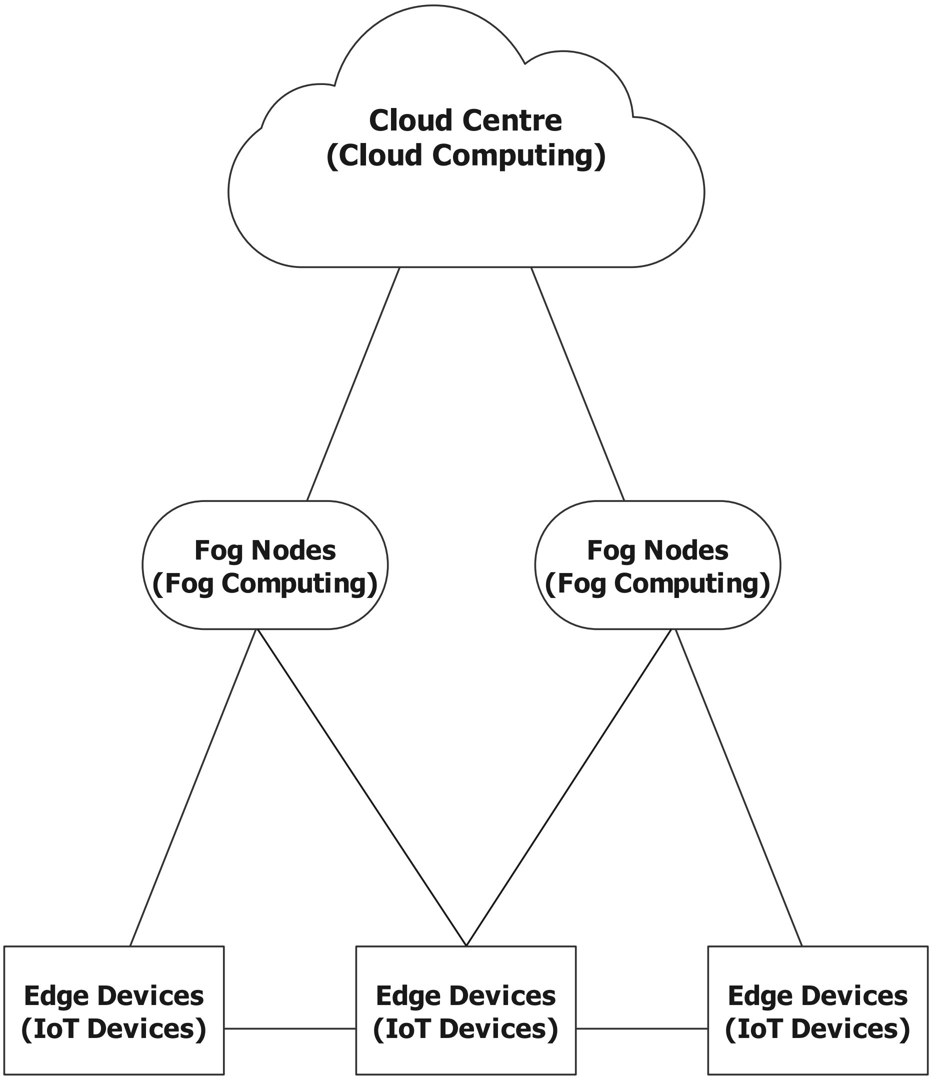

Fig. 2.6 Illustration of fog computing structure

are extremely susceptible to network security attacks. Fog computing can therefore configure optimal routing paths across the entire network based on different fog nodes to rapidly update the software security information and send it to the wireless sensors.

- **Cognition:** Fog computing enables clients to awareness of their customers' objectives to support autonomous decision-making in the deployment of compute and storage functions and network control functions. This advantage of fog computing changes the role of IoT devices from passive devices to active smart devices that can meet the requirements of clients without relying on distant cloud centres.

- **Efficiency:** Fog computing enables gateway devices to collect and redistribute computing and storage resources while processing most tasks, which reduces unnecessary communication bandwidth costs, energy consumption, and so on.
- **Agility:** Fog computing enhances the agility of IoT system deployment. Compared to traditional cloud computing, fog computing opens up more opportunities for individuals and small business parties to innovate, develop, deploy, and provide new services using overwhelmingly open software platforms.
- **Latency:** The proximity of fog nodes to edge devices reduces waiting latency when processing applications for edge devices. Moreover, fog computing delivers a highly flexible platform for the rapid reconfiguration of IoT devices as far as its softwarisation is concerned. As a result, fog computing provides better synchronous communication and real-time services than cloud computing.

2.3.3 Fog Computing in IoV Systems

Based on the advantages of fog computing, IoV-based ITS can be classified into three application scenarios as follows [67–69]:

- **Autonomous Driving:** By applying fog computing to autonomous driving systems, latency can be reduced via collecting information and processing tasks at a lower level in the fog layer. Conserve limited and expensive resources (e.g., bandwidth and storage). Furthermore, data applications, extraction and analysis are set up and performed closer to the V-UE to minimise the amount of raw data transferred to the cloud centre.
- **Cooperative Driving:** Cooperative driving based on fog computing enables a group of vehicles to update useful information or alerts, such as road conditions, congestion, traffic updates and weather information, from the leading vehicle via V2X connections. All this information can be collected and shared from multiple vehicles, which saves energy consumption and expensive cellular network connections.
- **Share Vehicles:** Shared vehicle services with fog computing allow data to be shared among different commercial companies that provide real-time services to respond to the V-UE's requests and allocate a potential parking space to the V-UE. In addition, the shared vehicle service provides real-time emergency traffic information, enabling incident reports to be sent immediately and securely to the required authorities.

2.4 Methodology

In this section, social networks theory, queueing theory, matching algorithm theory and fireworks algorithm theory are briefly introduced. In Chapter three, social attributes or interests similarity and social link duration are quite powerful to generate the performance of V2V links in the social domain, including the social community assignment and so forth. In addition, matching-based frameworks are developed to solve the resource allocation problems between all clusters and V2I links. In Chapters four and five, queuing theory affects the performance of offloading decisions for all V-UEs. For example, applications may need to wait in queues at the computing servers before they can be processed, which increases the offloading latency of V-UEs. Furthermore, fireworks algorithm-based frameworks are developed to solve resource allocation for all cloud/fog-computing V-UEs and optimise the offloading decisions for all V-UEs, which are NP-hard to solve. The resource allocation and offloading decisions obtained by using the modified fireworks algorithms are proved to be stable and optimal.

2.4.1 Basics of Social Theory

Social Attribute Similarity

In VSNs, common social attribute similarity quantifies the relative importance of a V-UE to other V-UEs. The higher the social attribute similarity, the more likely two V-UEs will be socially connected [32, 70].

Let H_i denotes the social attributes of a V-UE v_i, where each item in H_i is a binary variable representing whether v_i is interested in the corresponding item or not. For example, assuming the following social attributes {News, Music} and $H_i = \{1,0\}$. It indicates that v_i likes reading news, rather than listening to music. Based on the vector space model (VSM) [71], let $H_i = \{S_{i,x}\}$, where $x \in 1,...,V$ and $S_{\{i,x\}} \in \{0,1\}$ are the social attributes profile of v_i. Therefore, the social attributes similarity between two V-UEs v_i and v_j can be given as follows [59, 72]:

$$SAH_{i,j} = \frac{\sum\limits_{c=1}^{V} S_{i,c}S_{j,c}}{\sqrt{\sum\limits_{a=1}^{V} S_{i,a}^2}\sqrt{\sum\limits_{b=1}^{V} S_{j,b}^2}} \tag{2.1}$$

where $SAH_{i,j} = 1$ indicates that the V-UEs v_i and v_j have exactly the same social attributes and they are more likely to form a social connection in the VSNs. Otherwise, $SAH_{i,j} = 0$

indicates that both V-UEs do not share any common social attributes and they are less likely to form a social connection in VSNs.

Social Link Duration

Once the communication link between two V-UEs is connected, the social link duration between the two V-UEs depends on the communication link duration. Therefore, the social link duration $T_{SL}(t)$ at time t can be calculated as follows:

$$T_{SL}(t) = \frac{R - \theta_1 \sqrt{(x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2}}{|(s_i - \theta_2 s_j(t)|} \tag{2.2}$$

where R is the wireless transmission range; $x_i(t), x_j(t), y_i(t), y_j(t)$ are the locations of two V-UEs; $s_i(t), s_j(t)$ denote the velocities of two V-UEs; $\theta_1 = 1, \theta_2 = 1$ denote v_i moves forward in front of v_j; $\theta_1 = -1, \theta_2 = 1$ denote v_j overtakes v_i; $\theta_1 = 1, \theta_2 = -1$ denote v_i and v_j are moving away from each other; $\theta_1 = -1, \theta_2 = -1$ denote v_i and v_j are moving toward each other.

2.4.2 Queueing Theory

Queueing theory is the mathematical study of queueing systems which is composed of customers who arrive at a service facility for service or waiting for service [73]. A birth-and-death M/M/1 example is shown in Fig. 2.7, which is a simple Markovian birth-and-death queueing model and consisted of Poisson-input, exponential-service, single-server.

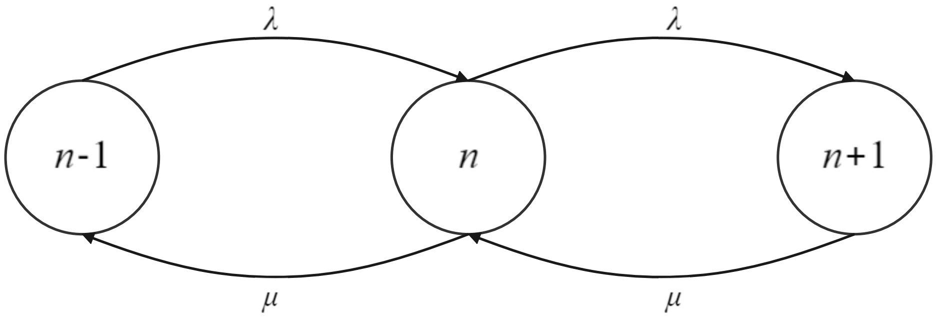

Fig. 2.7 A birth-and-death M/M/1 example

For M/M/1 queueing systems, the arrival rate is λ which follows a Poisson distribution, and the service rate is μ which follows an exponential distribution. Once an arrival occurs, the state of the system is changed from n to $n+1$, similarly, if a departure occurs, the system is in state n changes the system down one to $n-1$, where for all $n, \lambda_n = \lambda$ and $\mu_n = \mu$. Therefore, a steady-state solution exists and can be determined as

$$\begin{aligned} 0 &= -(\lambda+\mu)p_n + \mu p_{n+1} + \lambda p_{n-1} \; (n \geq 1), \\ 0 &= -\lambda p_0 + \mu p_1, \end{aligned} \tag{2.3}$$

where $p_n, \{n = 0, 1, 2, ...\}$ is the steady-state probability.

To solve the steady-state difference equations for p_n according to (2.3), an iterative method is considered to obtain a sequence of state probabilities, $p_1, p_2, p_3, ...$, each in terms of p_0. It follows that

$$\begin{aligned} p_1 &= \frac{\lambda}{\mu} p_0, \\ p_2 &= \frac{\lambda+\mu}{\mu} p_1 - \frac{\lambda}{\mu} p_0 \\ &= \frac{\lambda+\mu}{\mu}\frac{\lambda}{\mu} p_0 - \frac{\lambda}{\mu} p_0 \\ &= \frac{\lambda^2}{\mu^2} p_0, \\ p_3 &= \frac{\lambda+\mu}{\mu} p_2 - \frac{\lambda}{\mu} p_1 \\ &= \frac{\lambda+\mu}{\mu}\frac{\lambda^2}{\mu^2} p_0 - \frac{\lambda}{\mu}\frac{\lambda}{\mu} p_0 \\ &= \frac{\lambda^3}{\mu^3} p_0. \end{aligned} \tag{2.4}$$

The pattern which can be emerging is given by

$$\begin{aligned} p_n &= p_0 \prod_{i=1}^{n} \left(\frac{\lambda}{\mu}\right) \quad (n \geq 1) \\ &= p_0 \left(\frac{\lambda}{\mu}\right)^n . \end{aligned} \tag{2.5}$$

As the sum of the probabilities of every state being steady is 1, we have

$$\sum_{n=0}^{\infty} (\rho)^n p_0 = 1, \tag{2.6}$$

where $\rho = \dfrac{\lambda}{\mu}$ indicates the utilisation for single-server queues, and $\sum\limits_{n=0}^{\infty} (\rho)^n$ converges if and only if $\rho < 1$. Based on the sum of the terms of a geometric progression, we have

$$\sum_{n=0}^{\infty} \rho^n = \frac{1}{1-\rho},\ \rho < 1. \tag{2.7}$$

Thus, the solution of the steady-state for the M/M/1 queueing model can be expressed as a geometric probability function,

$$p_n = (1-\rho)\rho^n, \rho = \frac{\lambda}{\mu} < 1. \tag{2.8}$$

Based on the steady-state probability distribution, two measures of effectiveness, the mean queue length L_q and the expected waiting time W_q, can be calculated. Let N_q is the random variable number in steady-state, then we have

$$\begin{aligned} L_q = E[N_q] &= \sum_{n=1}^{\infty} (n-1)\, p_n \\ &= \sum_{n=1}^{\infty} n p_n - \sum_{n=1}^{\infty} p_n \\ &= (1-\rho) \sum_{n=1}^{\infty} n\rho^n - (1-p_0)\,. \end{aligned} \tag{2.9}$$

According to (2.5) and the summation of $\sum\limits_{n=1}^{\infty} n\rho^n$, we have

$$\begin{aligned} \sum_{n=1}^{\infty} n\rho^{n-1} &= \frac{d[1/(1-\rho)]}{d\rho} \\ &= \frac{1}{(1-\rho)^2}. \end{aligned} \tag{2.10}$$

Substituting (2.10) into (2.9), the expect queue length is L_q given by

$$L_q = \frac{\rho}{1-\rho} - \rho = \frac{\rho^2}{1-\rho}. \tag{2.11}$$

Based on Little's formulas and (2.11), the expected waiting time W_q can be given by

$$W_q = \frac{L_q}{\lambda} = \frac{\rho}{\mu-\lambda}. \tag{2.12}$$

Similarly, the expect waiting time in M/M/S queueing system can be given by

$$W_{qs} = \frac{\alpha^s p_0}{s!s\mu(1-\rho)^2}. \tag{2.13}$$

where $\alpha = \frac{\lambda}{\mu}$ and λ denotes the arrival rate; μ denotes the service rate; s denotes the number of servers and $\rho = \frac{\lambda}{s\mu} < 1$ defined as the traffic intensity for the system [74].

2.4.3 Matching Theory

The matching theory was published in 1962 and was first applied to economics research works [75]. Based on past research work, matching theory can be fairly classified into the following three types: bipartite matching problems with two-sided preferences, bipartite matching problems with one-sided preferences, and non-bipartite matching problems with preferences [76]. The classical stable marriage (SM) problem and the capacitated house allocation (CHA) problem are the central matching problems in bipartite matching problems with two-sided preferences and bipartite matching problems with one-sided preferences, respectively [77].

- **SM problem:** This is a typical one-to-one matching problem consisting of two sets of the same size (i.e., male and female) in which each member has its own preference list which includes all members of the opposite gender in order of preference. A pair is considered stable if no two members of the opposite gender would have both each other rather than their current partner [78]. Based on the algorithm was given by Gale and Shapley, the description of the procedure and general structure of the algorithm to solve the SM problem is detailed in Algorithm 2.1.

Algorithm 2.1 SM matching algorithm [75, 78].

```
To initialise sets of the same size for men and women;
To collect the information of the preference list of all individuals;
while All men and women are paired do
    All free men propose to their favourite women according to their preference lists, and remove the rest women from the list;
    A woman accepts the most preferred man according to their preference list and rejects the other men on the list;
end while
Output: Matching $\mu_{SM}$;
```

- **CHA problem:** CHA problem is the variant of SM problem and extends as a many-to-one problem in which the houses do not have their preference lists over the applicants

[79, 80]. Each applicant has preference list of the houses that can choose from, while the houses have a preference list for applicant-house pairs. The maximum number of applicants can be allocated to each house is limited and is denoted as the quota [81]. The Gale and Shapley based algorithm for CHA problem is detailed as follows:

Algorithm 2.2 CHA matching algorithm [76, 81].

```
To initialise sets for houses and applicants;
To collect the information of the preference list of all individuals;
while Each applicant is allocated to a house do
    All free applicants propose their favourite house according to their preference lists, and remove the rest houses from the list;
    Each house accepts the limited numbers of the most preferred applicants according to its preference list and rejects the other applicants on the list;
end while
Output: Matching $\mu_{CHA}$;
```

The basic graph method is the foundational method for proposing different matching algorithms, whether it is used to solve SM or CHA problems. Let $G = (V,E)$ is a given undirected hypergraph which includes a set V of vertices and a set E of edges. A matching in G is a subset $M_0 \subseteq E$ of edges such that for any distinct edges $e_1, e_2 \in M_0, e_1 \cap e_2 = \emptyset$. Inspired by the CHA problem, social-aware radio resource allocation problem in vehicular cellular networks can be transformed into a 3-dimensional matching which is to find a matching in a 3-partite hypergraph with the maximum number of edges.

The CHA algorithm is modified according to the above similarities in Chapter three.

2.4.4 Fireworks Algorithm

The fireworks algorithm theory was first published of International Conference on Sustainable Infrastructure conference in 2010 [82]. The simple fireworks algorithm (FWA) for optimisation is stated as follows: randomly generate fireworks in the feasible space, where each firework represents a solution for the optimise function. Calculate fitness value of each firework according to the fitness function. Based on the optimisation problem and to keep the diversity of the population, sparks and mutations fireworks are generated, where each spark and mutation firework is a solution in the feasible space. Finally, if the best fitness has been found, stop the algorithm. Otherwise, keep doing the iteration process until the best value is met.

Explosion Strength

For each firework, the number of sparks can be calculated by

$$S_i = m \frac{F_{max} - f(x_i) + \varepsilon_1}{\sum\limits_{i=1}^{N} (F_{max} - f(x_i)) + \varepsilon_1}. \tag{2.14}$$

where m is given constraints for the total number of sparks; F_{max} is the worst individual fitness value among N fireworks; function $f(x_i)$ denotes the fitness value of the firework x_i, while ε_1 is an extremely small number to avoid zero division errors.

In order to limit the number of sparks for each firework, which can be given as

$$\hat{s}_i = \begin{cases} round(am), & if\ S_i < am \\ round(bm), & if\ S_i > bm, a < b < 1 \\ round(S_i), & otherwise \end{cases} \tag{2.15}$$

where $\hat{s}_i$ is the number of explosion sparks that generated by the ith firework; $round(\cdot)$ is the rounding off function; a and b are given constraints.

Explosion Amplitude

The explosion amplitude of each spark is defined by

$$A_i = \hat{A} \frac{f(x_i) - F_{min} + \varepsilon_2}{\sum\limits_{i=1}^{N} (f(x_i) - F_{min}) + \varepsilon_2}. \tag{2.16}$$

where $\hat{A}$ is the given maximum amplitude constraints; F_{min} denotes the best individual fitness value among N fireworks; ε_2 is an extremely small number to avoid zero division errors.

According to the results of the explosion amplitude of each spark, to make displacement on each dimension of a firework and it can be defined by

$$\Delta x_i^k = x_i^k + rand(0, A_i). \tag{2.17}$$

where $rand(0, A_i)$ denotes the uniform random number within the amplitude A_i.

Based on (2.14) - (2.17), the description of the procedure and general structure of the sparks generation algorithm is detailed in Algorithm 2.3.

Algorithm 2.3 Sparks generation algorithm [83].

```
Randomly generate $N$ fireworks in the feasible space, and calculated their fitness values $f(x_i)$;
Calculate the number of explosion sparks for each firework by (2.14)-(2.15);
Calculate the explosion amplitude of each spark by (2.16);
Randomly choose $z = rand(0, dimension)$ dimensions;
for $k = 1 : dimensions$ do
    if $k \in z$ then
        Displacement operation by (2.17);
    end if
end for
```

Mutation Sparks

To guarantee the diversity of a population, Gaussian mutation is adopted into fireworks algorithm. Thus, the mutation sparks of Gaussian explosion are given by

$$x_i^k = x_i^k g. \tag{2.18}$$

where k indicates the current dimension; g is a random number follows Gaussian distribution, $g = \mathcal{N}(1,1)$.

The description of the procedure and general structure of the mutation sparks generation algorithm is detailed in Algorithm 2.4.

Algorithm 2.4 Mutation Sparks generation algorithm [83].

```
Randomly generate $N$ fireworks in the feasible space, and calculated their fitness values $f(x_i)$;
Calculate Gaussian coefficient by $g = \mathcal{N}(1,1)$;
Randomly choose $z = rand(0, dimension)$ dimensions;
for $k = 1 : dimensions$ do
    if $k \in z$ then
        Gaussian mutation sparks generation by (2.18);
    end if
end for
```

Fireworks Algorithm (FWA)

Based on the results of fireworks, explosion sparks and sudden change sparks, selection strategy is determined based on Euclidean distance measurements [84] and the roulette method [82].

- **Euclidean distance measurements :** In FWA, the Euclidean distance can be calculated by

$$R(x_i) = \sum_{j=1}^{K} d(x_i, xj) = \sum_{j=1}^{K} \parallel x_i - xj \parallel . \tag{2.19}$$

  where $d(x_i, xj)$ denotes the Euclidean distance between any two individuals x_i and x_j; K denotes a combined set of explosion and mutation.

- **Roulette wheel method :** In FWA, under each iteration, the probability for choosing individual can be calculated by

$$p(x_i) = \frac{R(x_i)}{\sum\limits_{j \in K} R(x_j)}. \tag{2.20}$$

Based on (2.19) - (2.20) and Algorithms 2.3 and 2.4, the description of the procedure and general structure of the FWA is detailed in Algorithm 2.5.

Algorithm 2.5 Fireworks Algorithm (FWA) [83].

```
Randomly generate $N$ fireworks in the feasible space;
while current iteration < maximum allowed iteration do
    Set off $N$ fireworks;
    for all fireworks $x_i$ do
        Calculate the number of sparks of each fireworks;
        Calculate the amplitude of sparks;
    end for
    Calculate the number of mutation sparks, $\hat{m}$;
    for $k = 1 \rightarrow \hat{m}$ do
        Randomly select a firework $x_i$ and generate a mutation spark;
    end for
    Select the best firework and other sparks according to the selection strategy;
end while
```

Inspired by the FWA, the offloading decisions optimisation problem in vehicular networks can be transformed into a fireworks algorithm based optimisation problem. However, our offloading decisions optimisation differs from the basic fireworks algorithm in the following aspects:

- **Total service delay as the fitness function:** The offloading decisions for all V-UEs are first initialised as a swarm of fireworks. After the procedure of explosion, mutation and selection is repeated until the algorithm reaches convergence, or the maximum iteration index is reached. The best fitness value is selected as the solution to the constructed problem.

- **Resource management:** In each iteration, some explosive sparks are generated for each firework, followed by some variant sparks due to maintaining the diversity of the population. Meanwhile, communication and computation resource are allocated under each firework, explosion spark and mutation spark, which are potential offloading decisions for all V-UEs. Afterwards, the fireworks, explosion sparks and mutation sparks with the best fitness values are selected for the next iteration.

The basic fireworks algorithm is modified according to the above differences and similarities in Chapters four and five.

Chapter 3

Social-Aware Resource Allocation Algorithm for V2X Communications

3.1 Introduction

In this chapter, we present social-aware resource allocation schemes for V2X communications. We start with designing a V2X communications system that includes both the physical and social domains, and further consider the optimal resource allocation schemes for V2X communications underlaying cellular networks.

As we mentioned in Chapter 1, social-aware resource allocation schemes enable to enhance the performance of D2D communications while guaranteeing the QoS requirements of conventional cellular communications. In addition, profound studies have been developed in order to propose an optimal resource allocation scheme in the physical domain for V2X communications. However, we note that, before our work [59] has been published, only few works have considered resource allocation schemes for V2X communications, where the social domain is commonly considered. As the ever-increasing for wireless connections from vehicles, and most communication demands are initiated by human users, whose social activities would influence their communication demands and requirements, as discussed in [85]. In existing works for D2D communications [86–89], authors assumed all devices to be either stationary or with a pedestrian speed, which cannot be readily applied to high mobility vehicular users in V2X communications.

In our work, we formulate the resource allocation problem based on the social and the physical domains, and aim to maximise the total capacity of V2I links while guaranteeing the reliability (in terms of outage probability) of all V2V links by jointly optimising the resource allocation and transmission power for all the links. To solve this problem, we propose a two-step optimisation scheme, i,e., social-aware clustering resource allocation (SACRA) algorithm. The first step aims to group all V-UEs into different social communities according to their social attributes, and all V2V links in the same social community are divided into different clusters based on the minimised intra-cluster interference between different V2V links. The second step is designed to allocate different RBs to each cluster obtained in step one, aiming to maximise the sum V2I capacity based on a k-Dimensional many-to-one matching [90]. The stability and convergence of each step are proved. The proposed SACRA algorithm is evaluated through simulations and outperforms nonsocial-aware resource allocation schemes with relatively smaller computational complexity.

The rest of the chapter is organised as follows. The system model is introduced in section 3.2. In section 3.3 we present the problem formulation. In section 3.4, a description of the procedure and general structure of the optimisation algorithms for resource allocation among all V2V and V2I links. In section 3.5, the performance in terms of the total capacity of V2I links is analysed based on the simulation results. Finally, section 3.6 concludes the chapter.

3.2 System Model

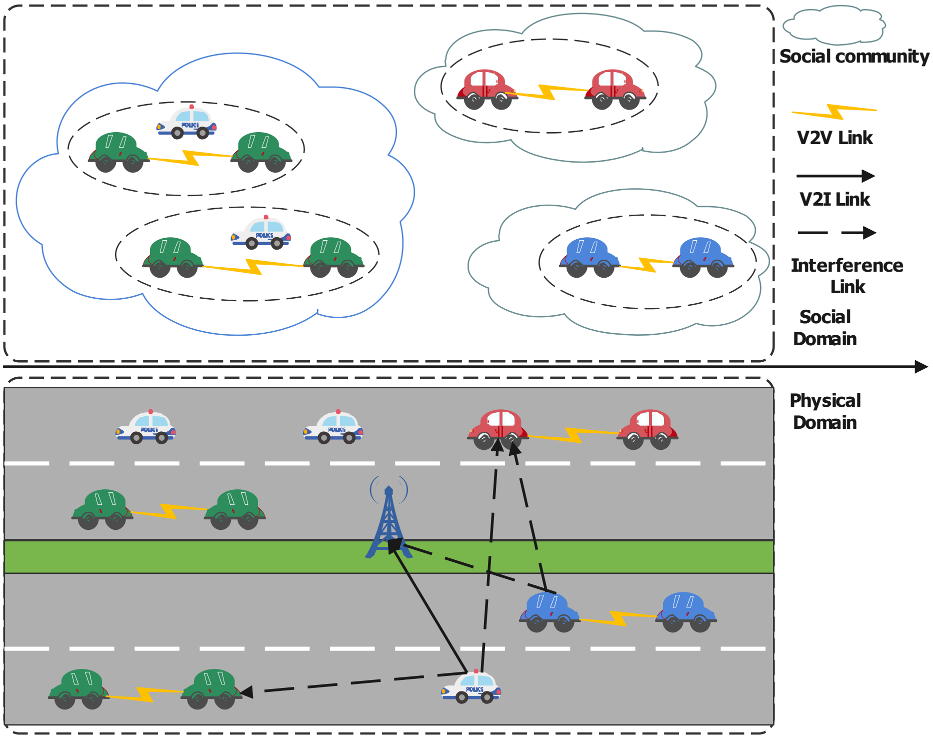

Fig. 3.1 Illustration of V2X communications

As shown in Fig. 3.1, we consider a single BS based cellular network for a V2X communication system, where the V2X system is divided into the physical domain and the social domain. In the physical domain, the total uplink bandwidth is divided into M orthogonal RBs which are assigned to F V2I links. The corresponding sets are denoted by $\mathscr{M} = \{1,...,M\}$ and $\mathscr{F} = \{1,..,F\}$, respectively. To avoid severe interference and reduce the high complexity of the system model, each V2I link uses a single RB and there is no interference between V2I links, i.e., $F = M$. In addition, V V2V links reuse the uplink resources of V2I links to improve spectrum utilisation, and the set is denoted by $\mathscr{V} = \{1,..,V\}$. It is noted that the number of V2V links is larger than the number of V2I links, i,e., $V \gg F$, and all V-UEs are deployed on the road randomly.

In the social domain, the social behaviors of V-UEs reflect their social connections, which can be obtained from different social medias (i.e., Facebook, Twitter and so on). Therefore, all V-UEs have to be grouped into Q social communities (i.e., the set is denoted by $\mathscr{Q} = \{1,...,Q\}$) based on their social attributes. Due to the V-UEs in the same social community are more likely to communicate with one another than those in different social communities, two V-UEs of each V2V link are belong to the same social community and there is no V2V link across two different social communities.

3.2.1 Physical Domain

In the physical domain, as Fig. 3.1 shows, the channel gain from the transmitter of the f-th V2I link to the BS over the m-th RB is given by

$$g_{f,B}^{m} = \alpha_{f,B}^{m} |h_{f,B}^{m}|^{2}, \tag{3.1}$$

where $\alpha_{f,B}^{m}$ is the large-scale fading of the channel from the transmitter of the f-th V2I link to the BS, which includes distance dependent path-loss and log-normal shadowing [91]; $h_{f,B}^{m}$ is the Raleigh fading channel coefficient that follows the complex Gaussian distribution $CN(0,1)$. Similarly, the channel gain from the transmitter of the f-th V2I link to the receiver of the v-th V2V link, and the channel gain of the v-th V2V link can be expressed as $g_{f,v}^{m}$ and g_{v}^{m}, respectively. Moreover, the interfering channel from the transmitter of the v-th V2V link to the BS over the m-th RB, and the interfering channel from the transmitter of the v'-th link to the receiver of the v-th V2V link over the m-th RB can be expressed as $g_{v,B}^{m}$ and $g_{v',v}^{m}$, respectively.

The received SINR of the f-th V2I link at the BS over the m-th RB is given by

$$\gamma_{f,B}^{m} = \frac{P_f g_{f,B}^{m}}{\sum\limits_{v \in V} a_{v,m} P_v g_{v,B}^{m} + \sigma^2}, \tag{3.2}$$

where P_f and P_v are the transmit powers of the f-th V2I transmitter and the v-th V2V link transmitter, respectively; σ^2 is AWGN power; and $a_{v,m}$ is the spectrum allocation indicator function, defined as

$$a_{v,m} = \begin{cases} 1 & \textit{if the v-th V2V link is transmitting over the m-th RB}, \\ 0 & \textit{otherwise}. \end{cases} \tag{3.3}$$

Similarly, the received SINR of the v-th V2V link at the receiver over the m-th RB is given by

$$\gamma_{v}^{m} = \frac{P_v g_{v}^{m}}{\sum\limits_{f \in F} a_{f,m} P_f g_{f,v}^{m} + \sum\limits_{v' \neq v, v' \in V} a_{v',m} P_{v'} g_{v',v}^{m} + \sigma^2}, \tag{3.4}$$

where $P_{v'}$ is the transmit power of the v'-th V2V transmitter; $a_{f,m}$ and $a_{v',m}$ are the spectrum allocation indicator functions, and they are similarly defined as $a_{v,m}$.

In order to maximise the sum capacity of V2I links, according to Shannon theorem [92], and guarantee the minimum reliability of all V2V links which according to the outage probabilities, the sum capacity of all V2I links is given by

$$r = \sum_{f \in F} \sum_{m \in M} log_2 \left(1 + \gamma_{f,B}^{m}\right). \tag{3.5}$$

3.2.2 Social Domain

In the social domain, according to the vehicular social networks theory [93], social community is defined as a set of V-UEs who have the same or similar social attributes or are interested in the same/similar information or contents (i.e., road traffic, weather information, video games, etc.). Based on the common interests or the shared social attributes between V-UEs, the V-UEs in the same social community more likely to communicate with one another or share contents via V2I link and V2V links than V-UEs in the completely different social communities.

According to the cosine similarity function [93], the similarity of between the v-th V-UE and the q-th social community is given

$$SH_{q,v} = \frac{\sum\limits_{k1=1}^{Q} Z_{q,k1} S_{v,k1}}{\sqrt{\sum\limits_{k2=1}^{Q} Z_{q,k2}^2} \sqrt{\sum\limits_{k3=1}^{Q} S_{v,k3}^2}}, \tag{3.6}$$

where $Z_{q,k1}, S_{v,k1}, Z_{q,k2}, S_{v,k3}$ are indicator functions, defined as

$$Z_{q,k1} = \begin{cases} 1 & \textit{if content } k1 \textit{ is involved in the } q\textit{-th community}, \\ 0 & \textit{otherwise}, \end{cases} \tag{3.7}$$

$$S_{v,k1} = \begin{cases} 1 & \textit{if the } v\textit{-th V-UE is interested in content } k1, \\ 0 & \textit{otherwise}. \end{cases} \tag{3.8}$$

And $Z_{q,k2}, S_{v,k3}$ are similarly defined as $Z_{q,k1}, S_{v,k1}$. Based on the result of the social similarity, each V-UE is allocated into the corresponding social community that has the highest value with it. For example, V-UE v is allocated into the q-th social community if $SH_{q,v} > SH_{q',v} \forall q, q' \in Q$ and $q \neq q'$. Moreover, the strength of the social relationship between the v-th V2V link and the f-th V2I link is defined by δ_f and the value is chosen in the range $[0,1]$ [93], where $\delta_f = 1$ indicates that the strongest social connection between V2I link f and V2V link v, and $\delta_f = 0$ indicates that the weakest social connection.

The notations used in this chapter can be found in Table 3.1.

Table 3.1 General Notation

Notation	Definition
$\mathscr{V}/V$	The set/number of V2V communication links
$\mathscr{F}/F$	The set/number of V2I communication links
$\mathscr{M}/M$	The set/number of uplink RBs
$\mathscr{Q}/Q$	The set/number of social communities
N	The total number of clusters
$g_{f,B}^{m}$	The channel from the f-th V2I Tx to the BS over RB m
g_{v}^{m}	The channel of the v-th V2V link over RB m
$g_{v,B}^{m}$	The interfering channel from the v-th V2V Tx to the BS over RB m
$g_{f,v}^{m}$	The channel from the f-th V2I Tx to the v-th V2V Rx over RB m
$g_{v',v}^{m}$	The interfering channel from the v'-th V2V Tx to the v-th V2V Rx over RB m
$\alpha_{f,B}^{m}$	The large-scale fading of V2I link f
$h_{f,B}^{m}$	The Raleigh fading channel coefficient
$\gamma_{f,B}^{m}$	SINR of the f-th V2I link at the BS over the m-th RB
γ_{v}^{m}	SINR of the v-th V2V link at the receiver over the m-th RB
P_f, P_v	The transmit power of the f-th V2I/v-th V2V transmitter
$a_{v,m}$	Equals 1 if v-th V2V link reuses the same spectrum with the f-th V2I link
σ^2	The additive white Gaussian noise power
r	The sum capacity of all V2I links
$SH_{q,v}$	The social similarity between the v-th V-UE and the q-th social community
$Z_{q,k1}$	Equals 1 if some V-UEs in the q-th community has cached content $k1$
$S_{v,k1}$	Equals 1 if the v-th V-UE is interested in content $k1$
ψ_n	The total intra-cluster interference
P_f^{max}, P_v^{max}	The maximum transmission power of V2I/V2V transmitter
δ_f	The social connection between V2I link f and V2V link v

3.3 Problem Formulation

The purpose of the work aims to maximise the sum V2I capacity to support contents sharing over the vehicular social networks. Therefore, the optimisation problem includes the physical domain and the social domain, and we formulated the communication resource allocation optimisation problem as

$$\max_{\{P_f,P_v\}} \sum_{m=1}^{M}\sum_{f=1}^{F} a_{f,m}\delta_f log_2(1+\gamma_{f,B}^{m}) \tag{3.9}$$

$$s.t.\ a_{v,m}Pr\{\gamma_v^m \leq \gamma_0\} \leq p_0,\ \forall v \in \mathscr{V}, \forall m \in \mathscr{M} \tag{3.9a}$$

$$0 < P_f \leq P_f^{max},\ \forall f \in \mathscr{F} \tag{3.9b}$$

$$0 < P_v \leq P_v^{max},\ \forall v \in \mathscr{V} \tag{3.9c}$$

$$a_{f,m}, a_{v,m} \in \{0,1\},\ \forall f \in \mathscr{F}, \forall v \in \mathscr{V}, \forall m \in \mathscr{M} \tag{3.9d}$$

$$\sum_{m=1}^{M} a_{f,m} = 1,\ \forall f \in \mathscr{F} \tag{3.9e}$$

$$\sum_{m=1}^{M} a_{v,m} = 1,\ \forall v \in \mathscr{V} \tag{3.9f}$$

$$\sum_{f=1}^{F} a_{f,m} = 1,\ \forall m \in \mathscr{M} \tag{3.9g}$$

$$\delta_f \geq \delta_0,\ \forall f \in \mathscr{F} \tag{3.9h}$$

where constraints (3.9a) is the minimum reliability requirements for all V V2V links, γ_0 is the minimum SINR to guarantee the reliability of an arbitrarily V2V link, and p_0 is the maximum tolerable outage probability, P_f^{max} and P_v^{max} in (3.9b) and (3.9c) are the maximum transmit powers of the V2I and V2V transmitters, constraint (3.9d) is the spectrum indicator with $a_{f,m} = 1$ sharing the V2I link is transmitting over the m-th RB and $a_{f,m} = 0$ otherwise. And the spectrum indicator for v-th V2V link is $a_{v,m}$, which is similarity defined; constraints (3.9e) and (3.9f) ensure an arbitrarily V2I link and V2V links can access single RB, constraint (3.9g) restricts orthogonal spectrum to be allocated among F V2I links. Constraint (3.9h) ensures that the social relationship between V2V link and V2I link f is no less than a threshold δ_0.

3.4 Algorithms of Matching under Preferences

3.4.1 The Proposed SACRA Algorithm

In this section, we propose the two-step SACRA algorithm to solve the problem (3.9). In the first step, all vehicular users are grouped into different social communities according to their social attributes. We assume that each V2V link is formed of two vehicles belonging to the same social community and each vehicle can be part of at most one V2V link. The different V2V links in the same social community are divided into different clusters in a way that the resulting intra-cluster interference between V2V links is minimised (which we will present in section 3.4.2). In the second step, based on the clustering results, we will adopt the matching theory [94] to solve the resource allocation problem in (3.9) under the constraints of maximum allowed transmission powers (which we will present in section 3.4.3).

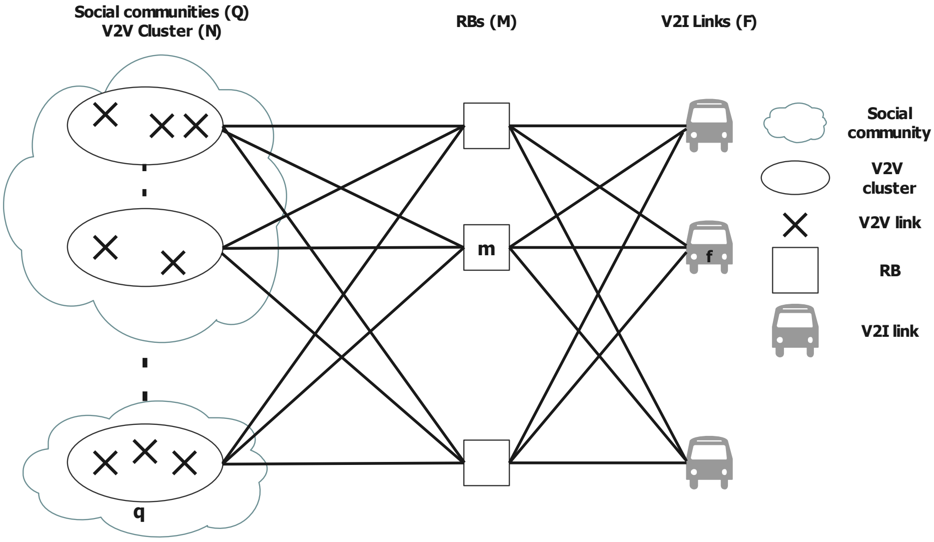

Fig. 3.2 Spectrum sharing between V2I link and V2V cluster of SACRA algorithm.

3.4.2 Social-Aware V2V Clustering

We first group all vehicular users into different social communities based on their social attributes, which can be calculated by using (3.6). Vehicular user v will be grouped into the q-th community that gives the largest value of similarity $SH_{q,v}(q \in Q)$ among all the communities. We assume that the two vehicles of each V2V link belong to a same community, and there is no V2V link across two different social communities.

In order to guarantee the minimum interference among different V2V links in the same social community, we propose a social-aware V2V clustering algorithm to divide the V2V links of each community into different clusters based on their mutual interference. This algorithm is given in Algorithm 3.1, which extends the simple heuristic algorithm [95], and is explained in the following. As shown in Fig.3.2, we assume that in the q-th (where $q \in \mathscr{Q}$) social community, there are V_q V2V links, which are divided into N_q clusters, where $V_q \gg N_q$. For Q social communities, we have N clusters and V V2V links in total, where $\sum_{q=1}^{Q} N_q = N$, $\sum_{q=1}^{Q} V_q = V$, and $V \gg N$. For simplicity, we assume that the total number of V2V clusters is equal to the total number of V2I links in the system, i.e., $N = F$. Thus, in the q-th social community, we divide V_q V2V links into N_q clusters which are denoted as $SC_{q,1},...,SC_{q,N_q}$. In order to avoid severe co-channel interference between different V2V links within the same cluster, for the v_q-th V2V link ($v_q \in \{1,...,V_Q\}, q \in Q$), we approximate the total intra-cluster interference that it may experience in the n-th cluster ($n \in \{1,...,N_q\}, q \in Q$) by $\psi_{v_q,n} = \sum_{v_q^{'} \in SC_{q,n}, v_q \neq v_q^{'}} (\alpha_{v_q,v_q^{'}} + \alpha_{v_q^{'},v_q})$, where $\alpha_{v_q,v_q^{'}}$ is the large-scale fading of the interference channel from the $v_q^{'}$-th V2V transmitter to the v_q-th V2V receiver in the q-th social community. Then, the v_q-th V2V link is grouped into the cluster that gives the smallest value of $\psi_{v_q,n} (n \in \{1,...,N_q\}, q \in Q)$ among all clusters in the q-th community.

According to the above, the social-aware V2V clustering scheme can be considered as a three partite graph, which is a NP-hard problem [96]. Algorithm 3.1 (lines 8-17) solves this NP-hard graph partitioning problem.

3.4.3 SACRA Algorithm

Based on the results of Algorithm 3.1, we consider to each V2I link share the same spectrum with all V2V links in the same cluster of each social community, while V2V links from different clusters cannot share spectrum resource. Thus, the RB sharing for the V2V links in the same cluster is also an NP-hard problem which has been proven in [96]. For simplicity, we assume M = F so that there is no spectrum resource sharing among V2I links, and to reduce the complexity of the algorithm

Accordingly, the resource allocation problem in (3.9) is rewritten as follows,

$$\max_{\{P_m,P_v\}} \sum_{q=1}^{Q} \sum_{m=1}^{M} \sum_{f=1}^{F} \sum_{n_q=1}^{N_q} a_{f,m} c_{n_q} \delta_f log_2(1+\gamma_{f,B}^{m}) \tag{3.10}$$

$$s.t.\ (3.9a) \sim (3.9h)$$

$$\sum_{q=1}^{Q} c_{n_q} = 1,\ \forall n_q \in \{1,...,N_q\} \tag{3.10i}$$

Algorithm 3.1 Social-Aware V2V Clustering Scheme

```
for $q = 1 : Q$ do
    for $v = 1 : V$ do
        Use (3.6) to calculate the social similarity among vehicles and social communities;
    end for
    To assign the $v$th V2V link to the $q$-th social community with $arg\ max$ (3.6);
end for
Return the social community result;
for $q = 1 : Q$ do
    Randomly assign one V2V link to each of the $N_q$ clusters in the $q$-th social community;
    for $v_q \in V_q$ do
        for $n = 1 : N_q$ do
            To calculate the intra-cluster interference by using $\sum_{v_q' \in SC_{q,n}, v_q \neq v_q'} (\alpha_{v_q,v_q'} + \alpha_{v_q',v_q})$
        end for
        To assign the $v$-th V2V link into $n_*^q$-th cluster in the $q$-th social community with $n_*^q = arg\ min \sum_{v_q' \in SC_{q,n}, v_q \neq v_q'} (\alpha_{v_q,v_q'} + \alpha_{v_q',v_q})$.
    end for
end for
Return the social-aware V2V clustering result.
```

where constraint (3.10i) ensures that each cluster belongs to one social community.

The problem in (3.10) is solved by Algorithm 3.2.

3.4.4 3-Dimensional Matching Problem

As we mentioned in Section 3.4.3, inspired by the CHA problem and the k-dimensional matching game in [90], the k-dimensional matching problem is to find a matching in a k-partite hypergraph with the maximum number of edges. Therefore, we model the radio resource allocation problem in (3.9) as a weighted 3-dimensional matching problem, where each possible resource allocation sharing pattern is considered as V2I-RB-V2V. The 3-dimensional resource allocation matching problem in our work can be defined as follows:

Definition 3.1. Let $H = (A,E)$ denotes a 3-partite hypergraph.

- A is a set of vertices, where $A = \{[f,0,0], 1 \leq f \leq F\} \cup \{[0,m,0], 1 \leq m \leq M\} \cup \{[0,0,n], 1 \leq n \leq N\}$;

Algorithm 3.2 SACRA for V2X Communications

```
According to the cluster results of Algorithm 1.
for $f = 1 : F$ do
    for $q = 1 : Q$ do
        for $n_q = 1 : N_q$ do
            for $m = 1 : M$ do
                To calculate the capacity of V2I link by using formula (3.9).
            end for
        end for
    end for
end for
To construct a three-partite graph, where the vertices are formed in term of V2I links,
    RBs and $N_q$ clusters in $q$-th social community.
To modify the k-Dimensional matching algorithm [90] and adopt Hungarian algorithm
    [97] to find matching solutions.
Return the corresponding resource allocation results.
```

- E is a set of edges and each edge is a non-empty subset of A, where $E = \{(f,m,n), 1 \leq f \leq F, 1 \leq m \leq M, 1 \leq n \leq N\}$;
- $\omega(f,m,n)$ is defined as the weight function, where $\omega(f,m,n) = log_2(1+\gamma_{f,B}^{m})$, $\forall f,m,n, 1 \leq f \leq F, 1 \leq m \leq M, 1 \leq n \leq N$.

Based on the definition 3.1, the weighted 3-dimensional matching problem can be converted to an integer program as follow:

$$\max \sum_{e \in E} \omega_e x_e \tag{3.11}$$

$$s.t. \sum_{e \in \Delta(a)} x_e \leq 1, \; \forall a \in A \tag{3.11a}$$

$$x_e \in \{0,1\}, \; \forall e \in E, \tag{3.11b}$$

where $\Delta(a)$ is the set of edges containing a, and the integer problem can be solved by Algorithms 3.3 and 3.4, which are obtained from the weighted 3-dimensional matching algorithm and the local ratio algorithm, respectively [90]. Let $M[e]$ be the set of edges H having non-empty intersection with e, $e \in M[e]$. Based on Algorithms 3.3 and 3.4, the solution x of the linear program (3.11) can be obtained, and produce a maximal matching in H.

Algorithm 3.3 Iterative 3-Dimensional Matching Algorithm [90]

```
Input $H = (A, E)$.
Let $B \subseteq E$ with initialisation $B = \emptyset$.
**repeat**
Find a hyperedge $e \in E - B$ with $x(M[e]) \leq 2$.
Let $B = B \cup \{e_i\}$.
Let $i = |B| + 1$, and $e_i = e$.
**until** $E - B = \emptyset$.
To find a matching $M_0$ by using Local-Ratio algorithm in Algorithm 3.4 with input $B$ and the weight $\omega$.
**Return** $M_0$ is the maximal matching.
```

Algorithm 3.4 Local Ratio Algorithm [90]

```
Input hypergraph $H = (A, E), B \subseteq E$, and an ordering of the edges in $E$.
Let $B' = \{e \in B : \omega_e > 0\}$.
**if** $B' = \emptyset$ **then**
    **Return** $\emptyset$
**end if**
Choose hyperedge $e'$ from $B'$ with the smallest index.
Decompose the weight function $\omega = \omega_1 + \omega_2$, where
$\omega_1(e) = \begin{cases} \omega(e'), if\ e \in M[e'], \\ 0, otherwise. \end{cases}$
$M' \leftarrow$ Local-Ratio $(B', \omega_2)$. (Note: this is a recursion.)
**if** $M' \cup \{e'\}$ is a matching in $H$ **then**
    Return $M' \cup \{e'\}$.
**else**
    Return $M'$.
**end if**
```

3.4.5 Complexity Analysis

In Algorithm 3.1, the social community assignment has a complexity of $\mathscr{O}(QV)$, and the V2V clustering has a complexity of $\mathscr{O}(QVF)$. Therefore, the complexity of Algorithm 3.1 is $\mathscr{O}(QV(1+F))$.

In Algorithm 3.2, the computational complexity mainly comes from the weighted 3-partite graph construction, where the complexity is $\mathscr{O}(QF^3)$. In Step 12, the complexity can be obtained from Algorithms 3.3 and 3.4. Therefore, the complexity of Algorithms 3.3 and 3.4 is detailed as follows: In Sections 3.4.4, we have a 3-partite hypergraph with $|A| = n$ and $|E| = f$. In Algorithm 3.3, the complexity mainly comes from Step 3 to Step 7, where the total number of iterations of the loop is f. In each iteration, an edge e can be searched with $x(M[e]) \leq 2$, where the complexity in each iteration is $\mathscr{O}(n^2 log_2 n)$. Thus, the complexity from Steps 3 to 7 is $\mathscr{O}(fn^2 log_2 n)$. In Step 8, the complexity can be obtained from Algorithm 3.4, which can be implemented $\mathscr{O}(fn^2)$.

Based on the above analysis, the total complexity of Algorithm 3.2 is $\mathscr{O}(QV(1+F) + QF^3 + F^5 logF)$.

3.5 Numerical Results and Analysis

In this section, we present the simulation results. We consider a single cell scenario covering a 500m × 500m highway area, with one BS located in the centre of the area. We assume that there are four lanes and each lane has a width of 6 meters. Without loss of generality, we suppose that there are three social communities in our scenario, i,e., $Q = 3$. We assume that all the vehicles are deployed on the road randomly and the total number of V2V clusters is equal to the number of V2I links. All the parameters used in the simulation are shown in Table 3.2.

Table 3.2 Simulation Parameters

Parameters	**Value**
Carrier frequency	2 GHz
Bandwidth	20 MHz
BS antenna height	15 m
Vehicle antenna height	1.5 m
BS Rx noise figure	5 dB
BS antenna gain	8 dBi
Vehicle antenna gain	3 dBi
Vehicle Rx noise figure	9 dB
Absolute vehicle speed v	60 km/h
Average inter-vehicle distance	$2.5 * v$
SINR threshold for V2V γ_0	5 dB
Maximum transmit power of V2I Tx	23 dBm
Maximum transmit power of V2V Tx	20 dBm
Noise power N_0	-174 dBm/Hz
Path-loss model (V2V link)	LOS in WINNER+B1
Path-loss model (V2I link)	$128.1 + 37.6 log_{10} d, d$ in km
Shadowing distribution (V2I/V2V link)	Log-normal
Shadowing standard deviation (V2I link)	8 dB
Shadowing standard deviation (V2V link)	3 dB

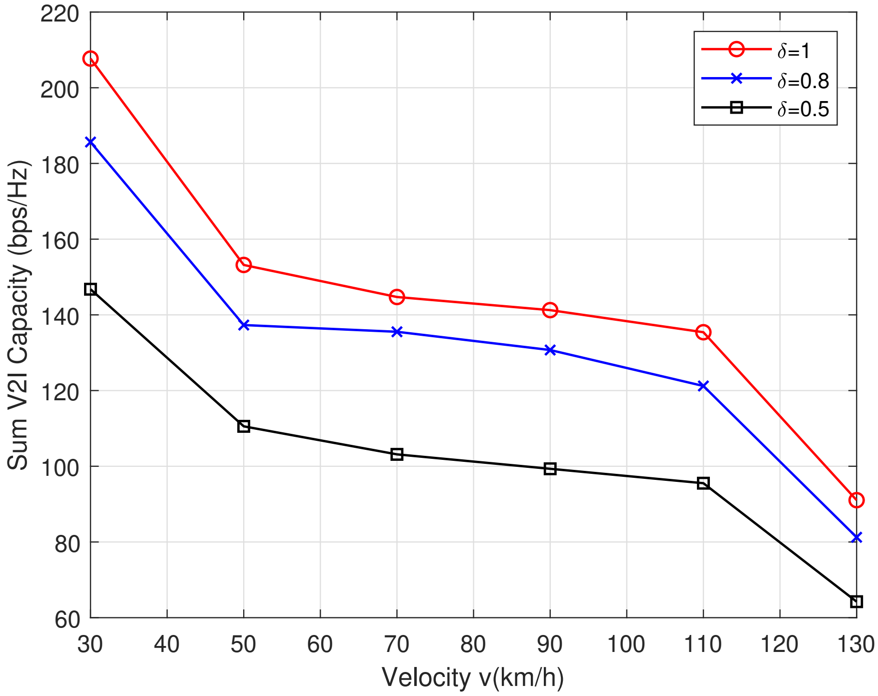

Fig. 3.3 The sum V2I links capacity for all the V2I links versus vehicle velocity for different values of the strength of the social relationship δ.

Fig. 3.3 shows the V2I links sum rate v.s. speed with different social relationship thresholds. To compare the impacts of the social relationships, we set the threshold of the strength of the social relationship values as $\delta_f = 0.5, \delta_f = 0.8$, and $\delta_f = 1$, when the threshold δ_f increases, the performance is getting better. The reason is when the threshold is lower, the more links with the weak intensity of social relationships can be satisfied, which lead to more interferences to influence on the sum rate. In Fig. 3.3, it is also illustrates that with the speeds increase, the performance also becomes worse, especially when the speed starts to exceed to 110 km/h. This is because the network topology becomes more dynamic, which leads to less reliable V2V links. Then the links are more vulnerable to be connected.

Fig. 3.4 illustrates that the sum capacity of V2I links decreases when the SINR thresholds for V2V link grows large, where $\delta_f = 1$ for the social-aware scenarios and non-social-aware scenarios neglect the social similarities among all V-UEs. This is because with the increasing of the SINR threshold, the interference tolerability of V2V link is reduced. It is also observed that the performance of the social-aware systems are better than the non-social-aware systems. With increasing of SINR threshold, the impact is getting smaller.

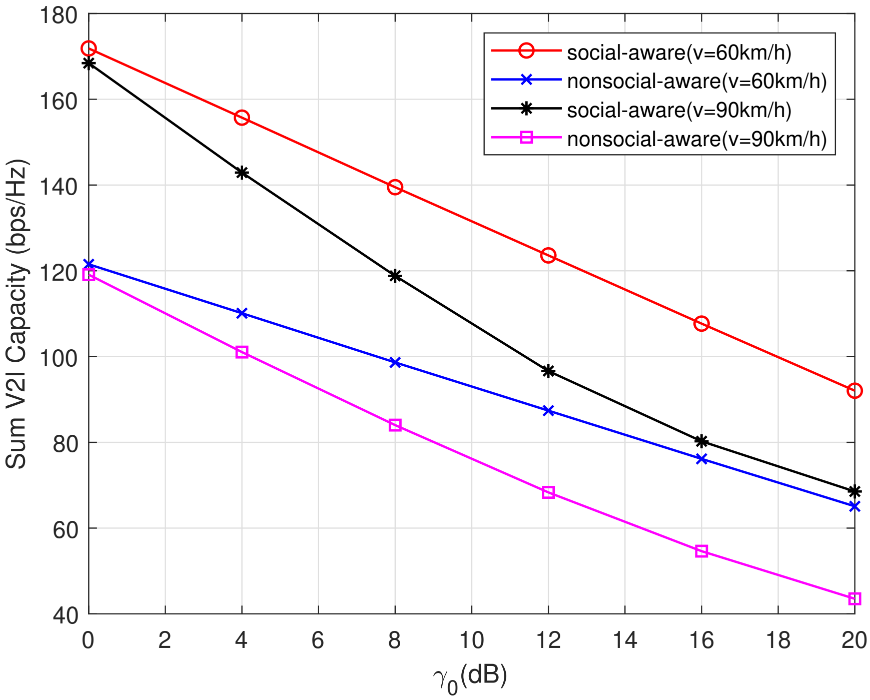

Fig. 3.4 The sum V2I links capacity for all the V2I links versus V2V SINR threshold γ_0.

3.6 Conclusions

In this chapter, we have studied the resource allocation problem of a social-aware V2X communication system, where each V2I link shares its RB with multiple V2V links. A V-UE is grouped into the social community that it has the strongest social interest similarity with. Then we have proposed a social-aware V2V clustering algorithm to divide all V2V links within each social community into different clusters according to the minimal intra-cluster interference among different V2V links. We also proposed a social-aware clustering resource allocation (SACRA) algorithm based on our social-aware V2V clustering algorithm to maximise the sum capacity of V2I links while guaranteeing the reliability for all V2V links. The simulation results demonstrate that a stronger social relationship between vehicles in a same social community would lead to a higher sum V2I capacity for the community. A larger SINR threshold for V2V link would lead to a lower sum V2I capacity for both social-aware and nonsocial-aware scenarios. However, the sum V2I capacity of the social-aware scenario is still higher than the one of nonsocial-aware scenarios.

Chapter 4

Collaborative Computation Offloading and Computation Resource Allocation in Vehicular Networks Based on Mixed Cloud/Fog Computing Systems

4.1 Introduction

In sections 1.1.1 and 1.1.2 we have introduced existing works on different offloading schemes for V2X communications while guaranteeing minimum offloading delays among all V-UEs. The proposed algorithms require accurate channel state information resource configuration for multiple channels and solve the MINLP problem, which is NP-hard. However, the impact of the mobility of V-UEs and/or the queuing delays of cloud centres or fog servers has not been fully studied.

In this chapter, we propose a three-layer (vehicle-fog-cloud) computation offloading framework for V2X communications to model the service delays of V-UEs locally processing or offloading their computationally intensive applications (e.g. online video games) to the cloud or fog servers, where the queues at the fog node and the cloud centre are modelled following the M/M/1 and M/M/C queueing models, respectively. We first identify the key constraints (i.e., the mobility of V-UEs, the maximum number of servers, the transmission distance and so on.) so that the QoS of V-UEs can be guaranteed. Since we aim to minimise the maximum service delay (which includes the transmission delay, the queueing delay and the processing delay) among all V-UEs in the proposed system. Then we propose a mobility and queueing-based offloading decisions optimisation algorithm, which jointly optimises the computation offloading decisions of all the V-UEs while considering their mobility and potential queueing delays at the fog node and the cloud centre in conjunction with a bisection method-based algorithm that optimises the fog node computation resource allocation for the fog-processing V-UEs. Finally, we discuss the performance of the proposed algorithms in comparison with the benchmarks, including pure local processing, fog processing, cloud processing, and random processing.

The rest of the chapter is organised as follows. The system model is introduced in section 4.2. In section 4.3 We first present the problem formulation, followed by a description of the procedure and general structure of the optimisation algorithms for offloading decisions and fog computation resource allocation. In section 4.4, complexity of the proposed schemes is analysed. In section 4.5, the performance in terms of total service latency is analysed based on the simulation results. Finally, section 4.6 concludes the chapter.

4.2 System Model

In this section, we first introduce a vehicle-fog-cloud three-layer networks, then derive the total service delay of the local, fog and cloud processing modes, respectively.

4.2.1 Vehicle-fog-cloud Architecture

As shown in Fig. 4.1, we consider a three-layer network model. The vehicle layer is composed of V V-UEs, the fog layer is composed of a fog node collected with a RSU, and the cloud layer mainly is distant cloud centre with C cloud servers. Denote the set of V-UEs and cloud servers as $\mathscr{V} = \{1,2,...,V\}$ and $\mathscr{C} = \{1,2,...,C\}$, respectively. All V-UEs in the coverage are connected to the fog node by V2I wireless links, while the fog node is connected to the cloud centre by a wired link.

For simplicity, we assume that each V-UE has only one application to process at the same time, and the RTT can be ignored. Within the communication coverage of the fog node, each V-UE may process its application by itself (i.e., local processing) or offload the application to the fog node or a cloud server for remote processing. Firstly, each V-UE sends an offloading request (including the information about its application size, location, velocity, heading direction and channel conditions) to the manager in the fog node [98]. After receiving the request from the V-UE, the manager optimises the offloading decisions of all V-UEs considering the offloading requests of other V-UEs and the available computational resources in the fog node and in the cloud centre.

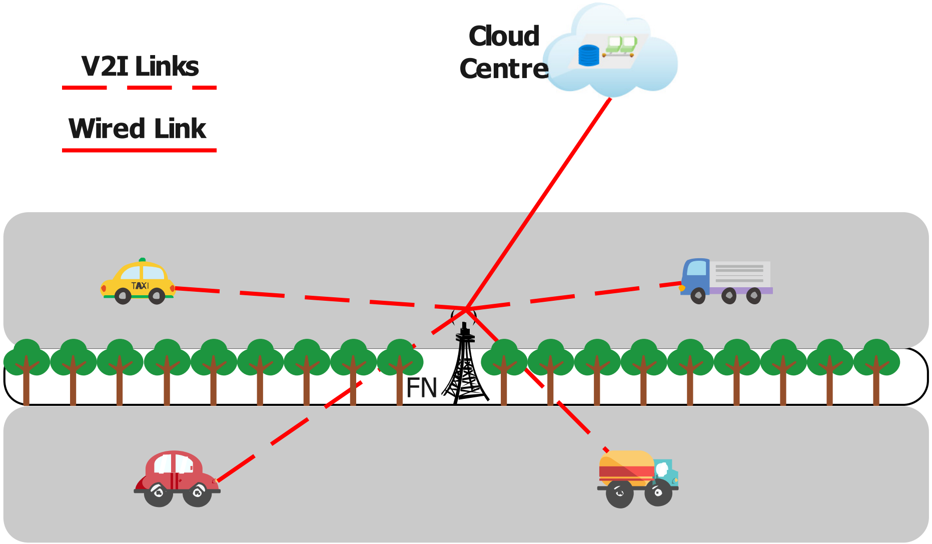

Fig. 4.1 System architecture of a three-layer computing system

The offloading decision for V-UE v is denoted by the offloading decision indicators, $x_v, y_v, z_v \in \{0,1\}$, we have

$$x_v + y_v + z_v = 1, \ \forall v \in \mathscr{V}. \tag{4.1}$$

where $x_v = 1, y_v = 1, z_v = 1$ indicate that the application is processed by V-UE v itself, by the fog node, or by a cloud server, respectively; otherwise, $x_v = 0, y_v = 0, z_v = 0$.

For all V-UEs, the offloading decisions matrix, i.e., $\mathbb{O}$, can be given by

$$\mathbb{O} = \begin{bmatrix} x_1, & y_1, & z_1 \\ \vdots & \vdots & \vdots \\ x_V, & y_V, & z_V \end{bmatrix}_{V \times 3}. \tag{4.2}$$

4.2.2 Transmission Delay

If the application of a V-UE is processed locally, there is no transmission delay of the application; If the application of a V-UE is for cloud/fog processing, then the transmission delay for uploading the input data of the application to the fog node or a cloud computing server can be calculated according to the offloading decision.

Fog Processing

If an application is to be processed by the fog node, the maximum achievable transmission rate (in bit/s) from V-UE v to the fog node under a selected RB can be given by

$$r_{v,f} = W_f log_2 \left(1 + \frac{P_v g_{v,f}}{\sigma^2}\right), \tag{4.3}$$

where W_f is the bandwidth of a selected RB (in Hz) between V-UE v and the fog node, in order to avoid severe interference, we assume that each V2I link is allocated an orthogonal RB, i.e., there is no interference between V2I links; P_v is the transmission power of a V-UE that is assumed to be the same for all V-UEs [26]; $g_{v,f}$ is the channel power gain from V-UE v to the fog node [99]; and σ_2 is the AWGN power at the fog node.

The transmission delay from V-UE v to the fog node is given by

$$T^{trans}_{v,f} = \frac{D_v}{r_{v,f}}, \tag{4.4}$$

where D_v is the data size (in bits) of V-UE v's application.

Cloud Processing

If the application is to be processed by a cloud server, then the application is forwarded by the fog node to the cloud centre by a high-speed wired link, where the transmission rate from the fog node to the cloud centre is denoted by $r_{f,c}$ (in bit/s). The transmission delay from the fog node to the cloud centre is given by

$$T^{trans}_{v,f,c} = \frac{D_v}{r_{f,c}}. \tag{4.5}$$

The total transmission delay from V-UE v to a cloud server can be calculated by

$$T^{trans}_{v,c} = T^{trans}_{v,f} + T^{trans}_{v,f,c}. \tag{4.6}$$

4.2.3 Queueing Delay

We consider a time-slotted model, where $\mathbf{t} = \{0, 1, ..., t\}$ denotes the set of time slots and the length of each time-slot is η. When $t < 0$, both the queues at the fog node and the cloud centre are empty.

Fog Processing

We assume the queue at the fog node follows the M/M/1 queueing model. Applications arrive at the queue of the fog node are assumed to follow a Poisson process with rate λ_f. It means that the number of fog processing applications, V_1, arriving at the fog node during a time slot has a Poisson distribution which is given by

$$P\left(V_1|t,\lambda_f\right)=\frac{\left(\lambda_f t\right)^{V_1}}{V_1!}exp(-\lambda_f t) \tag{4.7}$$

Moreover, the inter-arrival delays and the service delays have an exponential distribution with probability density as follow

$$a(t)=\lambda_f exp(-\lambda_f t),\ t>0. \tag{4.8}$$

$$b(t)=\mu_f exp(-\mu_f t),\ t>0, \tag{4.9}$$

where μ_f is the service rate of the fog node.

According to (4.7) - (4.9), the inter-arrival delays and the service delays can be calculated by

$$E\left[inter-arrival\ delay\right]=\frac{1}{\lambda_f}. \tag{4.10}$$

$$E\left[service\ delay\right]=\frac{1}{\mu_f}. \tag{4.11}$$

In addition, the utilisation rate of the fog node is given by

$$\rho_f=\frac{\lambda_f}{\mu_f}. \tag{4.12}$$

M/M/1 is one of simple general birth and death model, the generator matrix is given by

$$\mathbf{A}=\begin{bmatrix}-\lambda_f & \lambda_f & & \\ \mu_f & -(\lambda_f-\mu_f) & \lambda_f & \\ & \mu_f & -(\lambda_f-\mu_f) & \lambda_f \\ & & & \ddots\end{bmatrix}. \tag{4.13}$$

Based on Kolmogorov equations [100] for $P_{V_1}(t)$ $(V_1=0,1,2,\cdots)$, we have

$$P_0^{'}(t)=-\lambda_f P_0(t)+\mu_f P_1(t) \tag{4.14}$$

$$P_{V_1}^{'}(t)=-(\lambda_f+\mu_f)P_{V_1}(t)+\lambda_f P_{V_1-1}(t)+\mu_f P_{V_1+1}(t),\ V_1=1,2,3,... \tag{4.15}$$

Theorem 4.2.1 If the Markov process is irreducible (all states communicate), then the limiting distribution $\lim_{t\to\infty} P_{V_1}(t) = p_{V_1}$ exists and is independent of the initial conditions of the process. The limits $\{p_{V_1}, V_1 \in \mathbf{S}\}$ are such that they either vanish identically ($i.e.$, $p_{V_1} = 0$, $\forall V_1 \in \mathbf{S}$) or are all positive and form a probability distribution ($i.e.$, $p_{V_1} > 0, \forall V_1 \in \mathbf{S}$, $\sum\limits_{V_1 \in \mathbf{S}} p_{V_1} = 1$).

Theorem 4.2.2 The limiting distribution $\{p_{V_1}, V_1 \in \mathbf{S}\}$ of an irreducible recurrent Markov process is given by the unique solution of the equation $p\mathbf{A} = 0$ and $\sum\limits_{V_1' \in \mathbf{S}} p_{V_1'} = 1$, where $p = (p_0, p_1, p_2, ...)$.

For the limiting probabilities $\lim_{t\to\infty} P_{V_1}(t) = p_{V_1}$, the state balance equations can be expressed by

$$\lambda_f P_0 = \mu_f P_1. \tag{4.16}$$

$$(\lambda_f + \mu_f) P_{V_1} = \lambda_f P_{V_1 - 1} + \mu_f P_{V_1 + 1},\ V_1 = 1,2,3,... \tag{4.17}$$

As the sum of the probabilities of every state being steady is 1 (i.e., $\sum\limits_{0}^{\infty} P_{V_1} = 1$), we have

$$P_{V_1} = (1 - \rho_f)(\rho_f)^{V_1},\ V_1 = 1,2,3,... \tag{4.18}$$

where $\rho_f = \lambda_f / \mu_f < 1$.

When there are V_1 applications in the queue system at the fog node, the total queueing delay time of V_1 applications is Erlang with probability density is given by[73]

$$f_{V_1}(x) = exp(-\mu_f x) \frac{(\mu_f)^{V_1} (x)^{V_1 - 1}}{(V_1 - 1)!}. \tag{4.19}$$

Let $F_{fog}(t) = P(T_{fog} \leq t)$, we have

$$F_{fog}(0) = P(T_{fog} = 0) = 1 - \rho_f. \tag{4.20}$$

Thus, for $t > 0$, we have

$$\begin{aligned} dF_{fog}(t) &= P(t < T_{fog} < t+dt) \\ &= \sum_{V_1=1}^{\infty} P_{V_1} exp(-\mu_f t)\frac{(\mu_f)^{V_1}(t)^{V_1-1}}{(V_1-1)!}dt \\ &= (1-\rho_f)\sum_{V_1=1}^{\infty}(\rho_f)^{V_1} exp(-\mu_f t)\frac{(\mu_f)^{V_1}(t)^{V_1-1}}{(V_1-1)!}dt \\ &= \lambda_f(1-\rho_f)exp[-\mu_f(1-\rho_f)t]dt. \end{aligned} \tag{4.21}$$

The distribution of T_{fog} with the discontinuity at 0 can be expressed by

$$\begin{aligned} F_{fog}(t) &= P(t=0) + \int_0^t dF_{fog}(t) \\ &= 1-\rho_f exp[-\mu_f(1-\rho_f)t]. \end{aligned} \tag{4.22}$$

Let $E(T_{fog}) = T_f^{wait}$ and based on (4.21) and (4.22), the average queueing delay of an application in the fog node queue is given by

$$T_f^{wait} = E(T_{fog}) = \frac{\rho_f}{\mu_f(1-\rho_f)} = \frac{\lambda_f}{\mu_f(\mu_f-\lambda_f)} \tag{4.23}$$

Cloud Processing

The queue at the cloud centre follows the M/M/C model [73]. We assume the same service rate for each cloud server, i.e., μ_c. The applications arrive at the cloud centre also following a Poisson process with the rate of λ_c. Then, the utilisation rate of the cloud centre is given by $\rho_c = \lambda_c/(C\mu_c) < 1$, where it is assumed that each of the C cloud servers has enough capacity to process the received applications.

Similarly as M/M/1 model, according to the transform methods [101] and the solutions of Kolmogorov equations [100], we have

$$\lambda_c P_0 = \mu_c P_1, \tag{4.24}$$

$$(\lambda_c + V_c\mu_c)P_{V_c} = \lambda_c P_{V_c-1} + (V_c+1)\mu_c P_{V_c+1},\ 0 < V_c < C \tag{4.25}$$

$$(\lambda_c + C\mu_c)P_{V_c} = \lambda_c P_{V_c-1} + C\mu_c P_{V_c+1},\ C \le V_c < \infty \tag{4.26}$$

where $V_c = V_1 - V_2$ is the number of cloud-computing V-UEs, and V_2 is the number of remote-processing V-UEs.

A recursive procedure that used in the model of M/M/1, we have,

$$V_c\mu_c P_{V_c} = \lambda_c P_{V_c-1},\ V_c = 1,2,3,...,C \tag{4.27}$$

$$C\mu_c P_{V_c} = \lambda_c P_{V_c-1},\ V_c = C+1, C+2, C+3, ... \tag{4.28}$$

(Also see (4.18)). Therefore,

$$\begin{aligned} P_{V_c} &= \frac{1}{V_c!}(C\rho_c)^{V_c}P_0,\ 0 \le V_c < C \\ &= \frac{1}{C!}(C\rho_c)^C(\rho_c)^{V_c-C}P_0,\ C \le V_c < \infty, \end{aligned} \tag{4.29}$$

where $\rho_c = \dfrac{\lambda_c}{C\mu_c} < 1$ and under the condition $\sum\limits_{0}^{\infty} P_{V_c} = 1$, we have

$$P_0 = \left[\sum_{V_c=0}^{C-1}\frac{(C\rho_c)^{V_c}}{V_c!} + \frac{(C\rho_c)^C}{C!(1-\rho_c)}\right]^{-1} \tag{4.30}$$

$$\begin{aligned} P_{V_c} &= \frac{(C\rho_c)^{V_c}}{V_c!}P_0,\ 0 \le V_c < C \\ &= \frac{(C)^C(\rho_c)^{V_c}}{C!}P_0,\ C \le V_c < \infty, \end{aligned} \tag{4.31}$$

When the number of applications in the system is $V_c \ge C$, let $F_{cloud}(t) = P[T_{cloud} \le t]$,we have,

$$\begin{aligned} F_{cloud}(0) &= P[T_{cloud} = 0] \\ &= \sum_{V_c=0}^{C-1} P_{V_c} \\ &= P_0 \sum_{V_c=0}^{C-1} \frac{(\alpha_c)^{V_c}}{V_c!}, \end{aligned} \tag{4.32}$$

where $\Omega_c = C\rho_c$, according to (4.30) we have,

$$\sum_{V_c=0}^{C-1} \frac{(\Omega_c)^{V_c}}{V_c!} = \frac{1}{P_0} - \frac{(\Omega_c)^C}{C!}(1-\rho_c)^{-1} \tag{4.33}$$

Following the arguments for the M/M/1 queue model, in the C servers model we have,

$$\begin{aligned} dF_{cloud}(t) &= \sum_{V_c=C}^{\infty} P_{V_c} exp(-C\mu_c t) \frac{(C\mu_c t)^{(V_c-C)!}}{(V_c-C)!} C\mu_c dt \\ &= P_C exp(-C\mu_c t) \sum_{V_c=C}^{\infty} \rho_c^{(V_c-C)} \frac{(C\mu_c t)^{(V_c-C)}}{(V_c-C)!} C\mu_c dt \\ &= C\mu_c P_C exp(-C\mu_c(1-\rho_c)t)dt \end{aligned} \tag{4.34}$$

The distribution of T_{cloud} can be solved according to (4.32) and (4.34), we have,

$$\begin{aligned} F_{cloud}(t) &= F_{cloud}(0) + \int_0^t \frac{C\mu_c(\Omega_c)^C}{C!} P_0 exp(-C\mu_c(1-\rho_c)t)dt \\ &= 1 - \frac{(\Omega)^C P_0}{C!(1-\rho_c)} exp(-C\mu_c(1-\rho_c)t). \end{aligned} \tag{4.35}$$

Let $E(T_{cloud}) = T_c^{wait}$ and based on (4.34) and (4.35), the average queueing delay of an application in the cloud queue is given by

$$T_c^{wait} = E(T_{cloud}) = \frac{(\Omega c)^C P_0}{C!C\mu_c(1-\rho_c)^2}. \tag{4.36}$$

4.2.4 Processing Delay

The application being processed by the V-UE locally, the fog node, or a cloud server will lead to a different processing delay.

Local Processing

Denote the local processing capability of V-UE v by f_v^{local} (in CPU cycle/s) , then the processing delay of local processing can be expressed as

$$T_v^{proc} = \frac{A_v}{f_v^{local}}, \tag{4.37}$$

where A_v represents the total number of CPU cycles are required to process the application of V-UE v and it is given by $A_v = D_v \Lambda_v$, where Λ_v is the processing density (in CPU cycles/bit) of the application.

Fog Processing

If the application of V-UE v is processed by the fog node, given the fog processing capability of V-UE v as f_v^{fog} (in CPU cycles/s), the fog-processing delay is given by

$$T_{v,f}^{proc} = \frac{A_v}{f_v^{fog}}. \tag{4.38}$$

Cloud Processing

If the application is processed by a cloud server, denoting the cloud processing capability for V-UE v by f_v^{cloud} (in CPU cycles/s), then the cloud-processing delay can be expressed as

$$T_{v,c}^{proc} = \frac{A_v}{f_v^{cloud}}. \tag{4.39}$$

4.2.5 Service Delay

The service delay of an application may include the transmission delay, the queueing delay and the processing delay according to the offloading decision. Thus, the service delay of an application of V-UE v is given by

$$T_v = x_v T_v^{local} + y_v T_v^{fog} + z_v T_v^{cloud}, \tag{4.40}$$

where T_v^{local} is given in (4.37) as the application does not need to be transmitted to any remote server, and

$$T_v^{fog} = T_{v,f}^{trans} + T_{v,f}^{proc} + T_f^{wait}, \tag{4.41a}$$

$$T_v^{cloud} = T_{v,c}^{trans} + T_{v,c}^{proc} + T_c^{wait}. \tag{4.41b}$$

The notations used in this chapter can be found in Table 4.1.

Table 4.1 General Notation

Notation	**Definition**
$V/\mathscr{V}$	The number/set of V-UEs
$V_1/\mathscr{V}_1$	The number/set of fog-processing V-UEs
$V_2/\mathscr{V}_2$	The number/set of remote-processing V-UEs
$C/\mathscr{C}$	The number/set of cloud servers
$\mathbf{t}$	The set of time slots
$\mathbb{O}$	The offloading decisions matrix
x_v, y_v, z_v	The offloading decision of V-UE v
$r_{v,f}$	The transmission rate between V-UE v and the fog node
$r_{f,c}$	The transmission rate of the wired link
W_f	The bandwidth of a selected RB
P_v	The transmission power of each V-UE
$g_{v,f}$	The channel power gain between V-UE v and the fog node
σ^2	The additive white Gaussian noise power
D_v	The data size of application v
$T_{v,f,c}^{trans}$	The transmission delay from the fog node to the cloud centre
$T_{v,f}^{trans}/T_{v,c}^{trans}$	The transmission delay from V-UE v to the fog node/cloud centre
T_f^{wait}/T_c^{wait}	The average queueing delay at the fog node/cloud centre queue
λ_f/λ_c	The application arrival rate at the queue of the fog/cloud server
μ_f/μ_c	The service rate at the fog node/cloud server
A_v	Total required number of CPU cycles of V-UE v's application
Λ_v	Processing density of V-UE v's application
τ_v	The service delay threshold of V-UE v for remote-processing
s_v	The velocity of V-UE v
d_v	The distance between V-UE v and the fog node' s coverage edge
$\hat{s}$	The number of explosion sparks
$\hat{m}$	The number of mutation sparks
f^{fog}	Total computation capability of the fog node
I	The number of fireworks

$T_v^{proc}/T_{v,f}^{proc}/T_{v,c}^{proc}/$	The processing delay of local/fog/cloud processing
$T_v^{local}/T_{v,f}^{fog}/T_{v,c}^{cloud}/$	The service delay of local/fog/cloud processing
$f_v^{local}/f_v^{fog}/f_v^{cloud}/$	Processing ability of V-UE v in local/fog/cloud processing
η	The length of each time slot
L	Total number of iterations of MQA

4.3 Problem Formulation and Proposed Algorithm

In this section, we first formulate the min-max service delay problem, then propose a mobility and queueing based offloading decision optimisation algorithm, in conjunction with a bisection method based fog node computation resource allocation algorithm to solve the min-max problem.

4.3.1 Problem Formulation

We propose to minimise the maximum service delay among all V-UEs by jointly optimising the offloading decisions $\mathbb{O}$ and the fog node computation resource allocation $\mathrm{f}^{fog} = \{f_1^{fog}, ..., f_V^{fog}\}$. To reduce the computation complexity, we offload applications via selected channels to neglect the communication resource allocation issue [26] and assume that the cloud computation resources for each V-UE are constant at the cloud centre [98]. Accordingly, we formulate the optimisation problem as follows,

$$\mathscr{P}1: \min_{\mathbb{O}, \mathbf{f}^{fog}} \max_{v \in \mathscr{V}} T_v \tag{4.42}$$

$$s.t. \ \ x_v, y_v, z_v \in \{0, 1\}, \ \forall v \in \mathscr{V}, \tag{4.42a}$$

$$x_v + y_v + z_v = 1, \ \forall v \in \mathscr{V}, \tag{4.42b}$$

$$\sum_{v \in \mathscr{V}} y_v f_v^{fog} \le F^{fog}, \tag{4.42c}$$

$$0 \le f_v^{local} \le f_v^{fog} \ll f_v^{cloud}, \ \forall v \in \mathscr{V}, \tag{4.42d}$$

$$y_v T_v^{fog} + z_v T_v^{cloud} \le \tau_v, \ \forall v \in \mathscr{V}, \tag{4.42e}$$

where F^{fog} is the total computation capability in the fog node and τ_v is the estimated service delay threshold for V-UE v; (4.42a) and (4.42b) are the constraints on the binary offloading decision indicators for each V-UE; (4.42c) requires that the total allocated computation resources at the fog node cannot exceed its computation capability; (4.42d) indicates that for each V-UE, the amount of computation resource available at the cloud centre is the largest, followed by that at the fog node, while that available for local processing is the smallest but should be non-negative; and (4.42e) indicates that the service delay should be kept below the estimated threshold for each V-UE.

4.3.2 Mobility and Queueing Based Offloading Decision Optimisation Algorithm

For any given allocation of computation resources at the fog node, we propose a mobility and queueing based offloading decision optimisation algorithm (MQA), based on the traditional fireworks algorithm [83], to solve the formulated problem (4.42). The MQA is summarised in Algorithm 4.1. Firstly, we initialise I random offloading decisions for all V-UEs ($\mathbb{O}_1^{(0)},...,\mathbb{O}_I^{(0)}$) in the solution space. In the meantime, we calculate the service delay threshold of V-UE v for remote processing based on its information (i.e., its position with respect to the fog node, its moving direction and velocity) as follows,

$$\tau_v = \frac{d_v}{s_v}, \; \forall v \in \mathscr{V}, \tag{4.43}$$

where d_v is the distance between V-UE v and the coverage edge of the fog node in its direction of moving, and s_v is the velocity of V-UE v.

For all $v \in \mathscr{V}$ and $i = 1,...,I$, the estimated service delay of V-UE v based on the corresponding initial offloading decision in $\mathbb{O}_i^{(0)}$ for remote-processing, i.e., T_v^{fog} in (4.41a) and T_v^{cloud} in (4.41b), will be compared with its service delay threshold τ_v. If $max\{T_v^{fog}, T_v^{cloud}\} \leq \tau_v$, then the smallest estimated service delay between T_v^{fog} and T_v^{cloud} will be selected and the offloading decision for V-UE v in $\mathbb{O}_i^{(0)}$ will be updated accordingly. If $T_v^{fog} \leq \tau_v$ and $T_v^{cloud} > \tau_v$ ($T_v^{cloud} \leq \tau_v$ and $T_v^{fog} > \tau_v$), then T_v^{fog} (T_v^{cloud}) will be selected and the offloading decision for V-UE v in $\mathbb{O}_i^{(0)}$ will be updated to be fog processing (cloud processing). Otherwise, V-UE v can only process its application locally and the offloading decision for V-UE v in $\mathbb{O}_i^{(0)}$ is updated accordingly. After the all the initial offloading decision matrices ($\mathbb{O}_1^{(0)},...,\mathbb{O}_I^{(0)}$) have been updated, they become the initial fireworks, $\mathbb{O}_i^{(1)}, i = 1,...,I$.

In the lth iteration ($l = 1,...,L$, where L is the maximum allowed iteration), each firework $\mathbb{O}_i^{(l)}$ generates $\hat{s}_i^{(l)}$ new offloading decision matrices, which are called explosion sparks [82]. Each explosion spark of firework $\mathbb{O}_i^{(l)}$ is generated by randomly choosing j rows (where $j \in [1,V)$) of $\mathbb{O}_i^{(l)}$ and performing left circular shift by 1 position on each chosen row, while the other $(V-j)$ rows of the explosion spark are the same as the corresponding ones of firework $\mathbb{O}_i^{(l)}$. We take the objective function of (4.14) as the fitness function, i.e., $F(\mathbb{O}_i^{(l)}) = \max_{v \in \mathscr{V}} T_v(\mathbb{O}_i^{(l)})$, and the number of explosion sparks generated by firework $\mathbb{O}_i^{(l)}$ is given by

$$\hat{s}_i^{(l)} = ceil\left(S \frac{F_{max} - F(\mathbb{O}_i^{(l)}) + \varepsilon_1}{\sum\limits_{i=1}^{I} (F_{max} - F(\mathbb{O}_i^{(l)})) + \varepsilon_1} \right), \tag{4.44}$$

where $ceil(\cdot)$ denotes the ceiling function, S is a constant parameter for constraining the number of explosion sparks, $F_{max} = \max\limits_i(F(\mathbb{O}_i^{(l)}))$, and ε_1 is an extremely small number to avoid zero division errors.

In addition to the explosion sparks, $\hat{m}$ $(1 \leq \hat{m} \leq I)$ mutation sparks are generated by randomly selecting $\hat{m}$ fireworks from the I fireworks $(\mathbb{O}_1^{(l)}, ..., \mathbb{O}_I^{(l)})$ and randomly resetting some offloading decisions therein.

For each firework, explosion spark and mutation spark, the fog node computation resource allocation is obtained by using the bisection method [102] (which will be presented in Section III-C), and accordingly the fitness function value is calculated. Among all the fireworks, explosion sparks and mutation sparks, the one with the smallest fitness value is selected as firework $\mathbb{O}_1^{(l+1)}$ for the next iteration. Denoting the set of all fireworks, explosion sparks and mutation sparks excluding $\mathbb{O}_1^{(l+1)}$ in the lth iteration by $R_{est}^{(l)}$ (i.e., $\mathbb{O}_1^{(l+1)} \notin R_{est}^{(l)}$), the other ($I$ - 1) fireworks $(\mathbb{O}_2^{(l+1)}, ..., \mathbb{O}_I^{(l+1)})$ are selected from $R_{est}^{(l)}$ according to the roulette wheel selection method [83], where the probability of $\mathbb{A}_n^{(l)} (\mathbb{A}_n^{(l)} \in R_{est}^{(l)}, n = 1, ..., |R_{est}^{(l)}|)$ being selected is determined based on the Manhattan distance [84] as follows,

$$p(\mathbb{A}_n^{(l)}) = \frac{R(\mathbb{A}_n^{(l)})}{\sum\limits_{m=1}^{|R_{est}^{(l)}|} R(\mathbb{A}_m^{(l)})}, \tag{4.45}$$

where $R(\mathbb{A}_n^{(l)})$ is the sum of Manhattan distances between matrix $\mathbb{A}_n^{(l)}$ and all the other matrices in set $R_{est}^{(l)}$, which is given by

$$R(\mathbb{A}_n^{(l)}) = \sum_{m=1, m \neq n}^{|R_{est}^{(l)}|} \parallel \mathbb{A}_n^{(l)} - \mathbb{A}_m^{(l)} \parallel . \tag{4.46}$$

When the iteration converges or reaches the maximum allowed iteration, among all the fireworks, explosion sparks and mutation sparks, the one with the smallest fitness value is chosen as the optimal offloading decision $\mathbb{O}^*$ and the corresponding computation resource allocation at the fog node returns the optimal fog computation resource allocation $\mathbf{f^{fog}}*$.

Algorithm 4.1 Mobility and Queueing Based Offloading Decision Optimisation Algorithm (MQA)

```
Generate $I$ random fireworks $\{\mathbb{O}_1^{(0)}, ..., \mathbb{O}_I^{(0)}\}$.
For each firework, randomly allocate fog computation resources.
for $v = 1 : V$ do
    Calculate the service delay threshold of V-UE $v$ using (4.43).
    for $i = 1 : I$ do
        Calculate the estimated service delay of V-UE $v$ using (4.40).
        if $min\{T_v^{fog}, T_v^{cloud}\} > \tau_v$ then
            The $(v,1)$th element of $\mathbb{O}_i^{(0)}$ is substituted by $x_v = 1$.
        else if $T_v^{fog} < T_v^{cloud}$ then
            The $(v,2)$th element of $\mathbb{O}_i^{(0)}$ is substituted by $y_v = 1$.
        else
            The $(v,3)$th element of $\mathbb{O}_i^{(0)}$ is substituted by $z_v = 1$.
        end if
    end for
end for
Return: The updated fireworks as $\{\mathbb{O}_1^{(1)}, ..., \mathbb{O}_I^{(1)}\}$.
Input: $l = 1, F^{(0)} = 0, \varepsilon = 10^{-6}$.
while $l \leq L$ do
    for $i = 1 : I$ do
        For firework $\mathbb{O}_i^{(l)}$, run Algorithm 4.2 and calculate its fitness value.
        Calculate $\hat{s}_i^{(l)}$ according to (4.44).
        Generate $\hat{s}_i^{(l)}$ explosion sparks from firework $\mathbb{O}_i^{(l)}$.
        For each explosion spark, run Algorithm 4.2.
        Calculate the fitness value of each explosion spark.
    end for
    Generate $\hat{m}$ mutation sparks.
    For each mutation spark, run Algorithm 4.2.
    Calculate the fitness value of each mutation spark.
    The firework, explosion spark or mutation spark with the smallest fitness value is chosen as $\mathbb{O}_1^{(l+1)}$, and the smallest fitness value is denoted by $F^{(l)}$.
    if $|F^{(l)} - F^{(l-1)}| < \varepsilon$ then
        break;
    else
        Fireworks $(\mathbb{O}_2^{(l+1)}, ..., \mathbb{O}_I^{(l+1)})$ are selected according to (4.45), (4.46).
        $l = l + 1$;
    end if
end while
Return: The optimal offloading decision $\mathbb{O}^* = \mathbb{O}_1^{(l+1)}$ if $l < L$, otherwise $\mathbb{O}^* = \mathbb{O}_1^{(L+1)}$, and the corresponding fog computation resource allocation $\mathbf{f^{fog}}^*$.
```

4.3.3 Fog Computation Resource Allocation

In the lth iteration, for each offloading decision matrix $(\mathbb{O}_1^{(l)},...,\mathbb{O}_I^{(l)})$, the problem in (4.42) reduces to the optimisation of computation resource allocation at the fog node, i.e.,

$$\mathscr{P}2: \min_{\mathbf{f^{fog}}} \max_{v\in\mathscr{V}_1} \frac{A_v}{f_v^{fog}} + \Delta_v \tag{4.47}$$

$$s.t.\ (4.42c),(4.42d),$$

where $\mathscr{V}_1$ denotes the set of fog-processing V-UEs, for $\forall v \in \mathscr{V}_1, x_v = z_v = 0, y_v = 1$, and $\Delta_v = D_v/r_{v,f} + T_f^{wait}$ is a constant. Letting $\Theta = \max\limits_{v\in\mathscr{V}_1}\{A_v/f_v^{fog} + \Delta_v\}$, the problem $\mathscr{P}2$ is converted to

$$\mathscr{P}3: \min_{\mathbf{f^{fog}},\Theta} \Theta \tag{4.48}$$

$$\begin{aligned} s.t.\ & (4.42c),(4.42d), \\ & \frac{A_v}{f_v^{fog}} + \Delta_v \leq \Theta,\ \forall v \in \mathscr{V}_1. \end{aligned} \tag{4.48a}$$

Since $A_v/f_v^{fog} \geq 0$ and based on (4.42c) and (4.48a), we have $\sum\limits_{v\in\mathscr{V}_1} A_v/(\Theta - \Delta_v) \leq \sum\limits_{v\in\mathscr{V}_1} f_v^{fog} \leq F^{fog}$. As $A_v/(\Theta - \Delta_v)$ is a monotonically decreasing function of Θ, the maximum service delay among all the fog-processing V-UEs is minimised when $\sum\limits_{v\in\mathscr{V}_1} A_v/(\Theta - \Delta_v) = \sum\limits_{v\in\mathscr{V}_1} f_v^{fog} = F^{fog}$, and $\mathscr{P}3$ can be converted to

$$\mathscr{P}4: \min_{\Theta} \Theta \tag{4.49}$$

$$s.t.\ \sum_{v\in\mathscr{V}_1} \frac{A_v}{\Theta - \Delta_v} = F^{fog}. \tag{4.49a}$$

We adopt the bisection method to solve problem $\mathscr{P}4$ as summarised in Algorithm 4.2, where V_1 is the number of fog-processing V-UEs; Θ^* denotes the minimum value of Θ, and the optimal fog computation resource allocation for V-UE v is given by $f_v^{fog*} = A_v/(\Theta^* - \Delta_v)$.

Algorithm 4.2 Fog Computation Resource Allocation

```
Initialise the precision $\varepsilon_2 > 0$, $\Theta_{down} = \max_{v \in \mathscr{V}_1} \Delta_v$
and $\Theta_{up} = \sum_{v \in \mathscr{V}_1} (A_v V_1 / F^{fog} + \Delta_v)$
**Repeat**
$\Theta = (\Theta_{up} + \Theta_{down})/2.$
**if** $\sum_{v \in \mathscr{V}_1} (A_v/\Theta - \Delta_v) > F^{fog}$ **then**
    $\Theta_{down} = \Theta.$
**else**
    $\Theta_{up} = \Theta.$
**end if**
**until** $|\Theta_{up} - \Theta_{down}| \leq \varepsilon_2.$
$\Theta^* = |\Theta_{up} - \Theta_{down}|/2.$
**Output**: $\mathbf{f}^{\mathbf{fog}^*}$
```

4.4 Complexity Analysis

In Algorithm 4.1, the computational complexity mainly comes from the resource (which including the computation resource) allocation procedures in Steps 20, 23 and 27 and the number of sparks (which including the explosion sparks according to each fireworks and the mutation sparks) in Steps 21 and 26. We first describe the complexity from resource allocation procedures then discuss the complexity from sparks, respectively.

In each computation resource allocation procedure under Algorithm 4.2, the time complexity can be given by $\mathscr{O}(log_2(\Theta_{up} + \Theta_{down}/2))$, which based on the iterations for the bisection method to converge [102].

According to Steps 9, the number of explosion sparks under each iteration has a complexity of $\mathscr{O}(\sum_{o=1}^{O} o\hat{s}_o)$. Similarly, the number of mutation sparks under each iteration has a complexity of $\mathscr{O}(\hat{m})$. Therefore, the total complexity of the total number of explosion and mutation sparks is $\mathscr{O}(\sum_{o=1}^{O} o\hat{s}_o + \hat{m})$.

Based on the previous discussion, the complexity of MQA under L times iteration is $\mathscr{O}(LVI(\sum_{o=1}^{O} o\hat{s}_o + \hat{m})(log_2(\Theta_{up} + \Theta_{down}/2))) = \mathscr{O}(LVI(\sum_{o=1}^{O} o\hat{s}_o + \hat{m}))$.

4.5 Simulation Results

In this section, we present the simulation results. We assume a single cell scenario with one fog node located in the centre of a 500 m × 500 m urban area as illustrated in Fig.4.1. There is a straight two-lane road (with one lane in each direction) passing through the middle of the considered square area, dividing the area into two equal rectangles. The width of each lane is 6 metres. Moreover, we assume that all the V-UEs are uniformly distributed in the rectangular area of 12m x 500m spanned by the two-lane straight road, where the movement direction of each V-UE is determined by the direction of the line that it locates in, and the local processing capability f_v^{local} is uniformly distributed in [50,400] M cycles/s. All parameter values used in the simulation are given in TABLE 4.2 [27, 98, 99], unless otherwise specified.

Table 4.2 Simulation Parameters

Parameters	**Value**
Transmit bandwidth, W_f	180 kHz
Transmit power of V-UE v, P_v	200 mW
The noise power density at the fog node, N_0	-174 dBm/Hz
Data size of an application of V-UE v, D_v	0.42 MB
Processing density of the application of V-UE v, A_v	297.62 cycles/bit
Total computation capability of the fog node, F^{fog}	2 G cycles/s
Cloud processing capability for V-UE v, f_v^{cloud}	5 G cycles/s
The service rate of the fog node/cloud server, μ_f/μ_c	6
Wired link rate between the fog node and the cloud, $r_{f,c}$	1 Mb/s
The number of cloud servers, C	2
The average vehicular velocity	70 km/h
The number of fireworks, I	6
The number of total explosion sparks, M	4
The number of mutation sparks, $\hat{m}$	1
The maximum number of iterations of FA, L	100

Fig.4.2 plots the maximum service delay $T_{service}$ versus the iterations of the outer loop in Algorithm 4.1. We can see that Algorithm 4.1 converges after the third iteration.

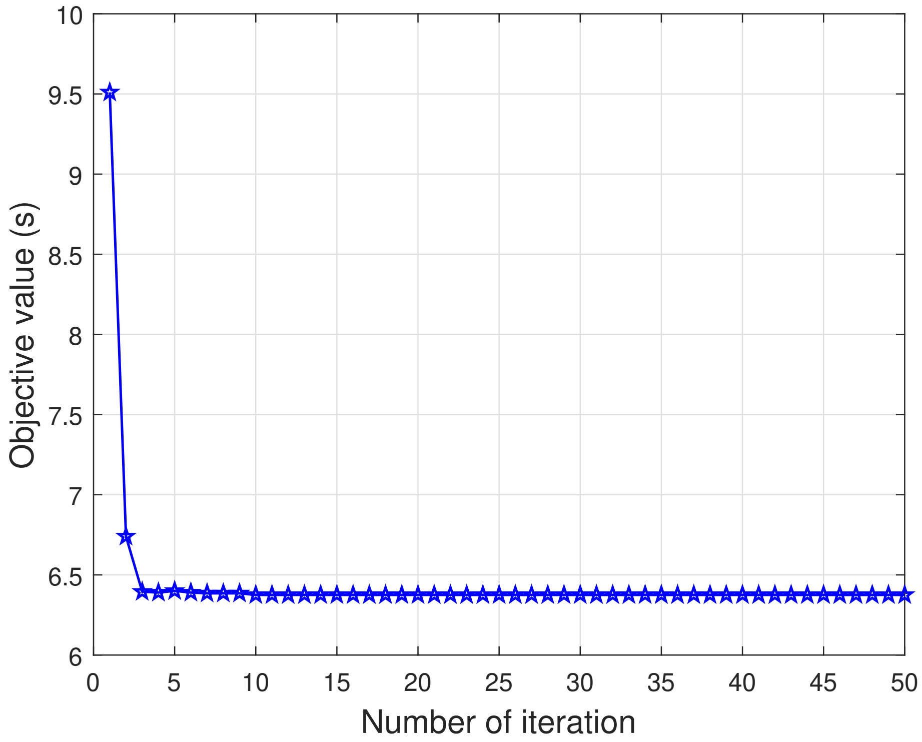

Fig. 4.2 Convergence of Algorithm 4.1, where $V = 5$.

Fig. 4.3 shows the maximum service delay versus the number of V-UEs, where 'MQA' denotes our proposed Algorithm 4.1, 'Local-Processing', 'Fog-Processing' and 'Cloud-Processing' denote the cases where all applications of the V-UEs are processed locally, by the fog node, or by the cloud servers, respectively, and 'Random-Processing' denotes the case where each V-UE's application has the equal probability of being processed by itself locally, the fog node, or a cloud server. We can see that the maximum service delay increases with the number of V-UEs in all the considered cases, among which MQA performs the best for any given number of V-UEs due to the joint optimization of offloading decisions for all the V-UEs while considering their mobility and queueing delays at the fog node and cloud centre. When the number of V-UEs is lager than 7, fog-processing leads to the highest maximum service delay. This is due to the long queueing and processing delays caused by many applications sharing the limited computation capacity of the fog node.

Fig. 4.4 and Fig. 4.5 show how the individual application's data size D_v and processing density A_v affect the maximum service delay, respectively, where $V = 5$. We can see that the larger the data size or the higher processing density of each application, the higher the

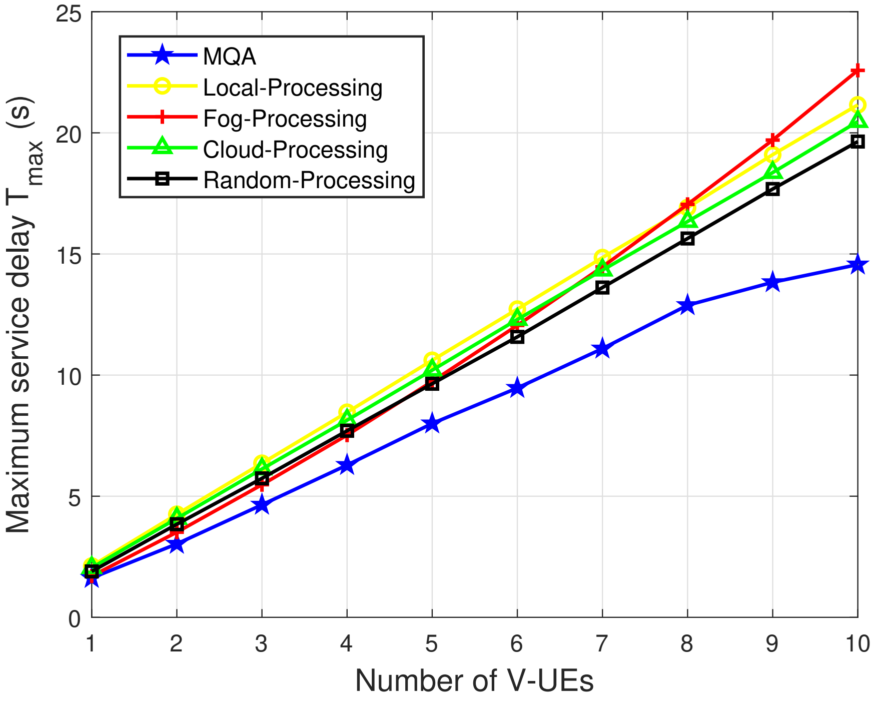

Fig. 4.3 Maximum service delay versus the number of V-UEs.

maximum service delay in each considered case. MQA always performs the best among all the considered cases. Moreover, local-processing is most significantly affected by a large data size or a high processing density of an application, due to the limited computation capability at each V-UE.

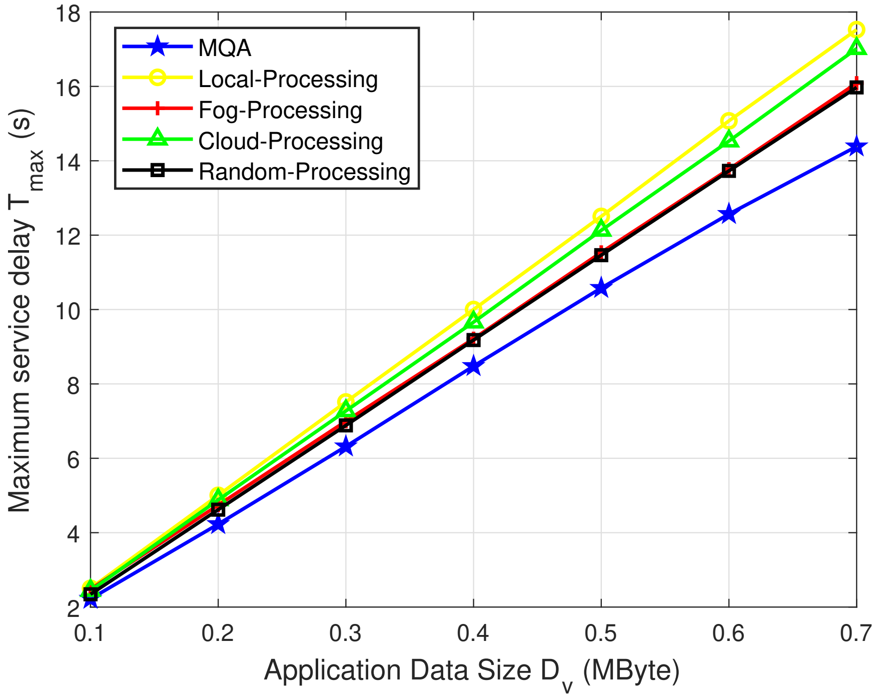

Fig. 4.4 Maximum service delay versus the data size of the application.

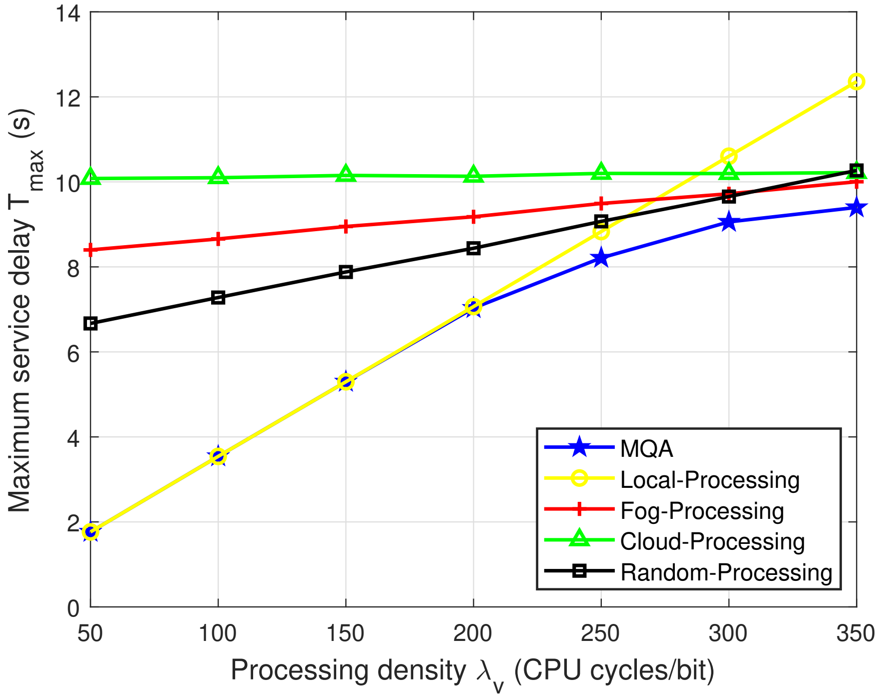

Fig. 4.5 Maximum service delay versus the processing density.

4.6 Conclusion

In this chapter, we have proposed a mobility and queueing-based offloading decision optimisation algorithm, in conjunction with a bisection method-based fog node computation resource allocation algorithm to minimise the maximum service delay of all V-UEs in an IoV system, where each V-UE may offload its task to a fog or cloud computing server or process it locally. The simulation results demonstrate that the proposed algorithms achieve a much lower maximum service delay than local-processing, fog-processing, cloud-processing, and random-processing.

Chapter 5

Computation Offloading and Resource Allocation in Mixed Cloud/Vehicular-Fog Computing Systems

5.1 Introduction

In this Chapter, we extend the offloading decisions optimisation problem in Chapter 4 from the computation resource allocation based offloading decisions optimisation problem to communication and computation resource allocation based offloading decisions optimisation problem, where V-UE applications can be offloaded to a movable V-FN for processing, instead of a static fog node. A primary objective of developing V-FNs in a mixed cloud/VFC system is to relieve the pressure on roadside fixed fog nodes, make the computing system more flexible, reduce the response latency of vehicle applications and provide more diverse services for smart vehicles.

We aim to minimise the maximum service delay (which includes the transmission delay, the queueing delay and the processing delay) of all V-UEs, thus guaranteeing the fairness among all V-UEs. Accordingly, we formulate a multi-objective optimisation problem, which is further decoupled into three sub-problems that optimise offloading decisions for all V-UEs, the communication resource allocation for all cloud/fog-computing V-UEs, and the computation resource allocation for all fog-computing V-UEs at V-FNs. In contrast to Chapter 4, we jointly consider the communication resource allocation for all cloud/fog computing V-UEs through a cluster-based algorithm. The efficiency and convergence of the proposed algorithms are proved. The effectiveness of the proposed fireworks algorithm-based algorithm is validated by comparing with the pure local-processing, fog-processing, cloud-processing, and random-processing schemes, and is further used to evaluate the impact of different parameters such as data size and V-UE mobility on system performance.

The remainder of this Chapter is organised as follows. The system model is introduced in section 5.2. The problem formulation is introduced in section 5.3. In section 5.4, we first describe the procedure and the general structure of the offloading decision optimisation algorithm, i.e., the fireworks algorithm based offloading optimisation and resource allocation algorithm (FORA). Then, followed by the details of the computation resource allocation algorithm and the communication resource allocation. In section 5.5, complexity of the proposed schemes is analysed. Simulation results are presented in section 5.6. Finally, the paper is concluded in section 5.7.

5.2 System Model

In this section, we first introduce a VFC architecture, then derive the service delay for local, fog and cloud processing, respectively.

5.2.1 Vehicle-fog-cloud Architecture

As shown in Fig. 5.1, we consider a VFC system with one RSU located in the centre of a square urban area, where a straight two-lane road (with one lane in each direction) passing through the middle of the considered square area, and the width of each lane is same. The VFC system consists of three layers, i.e., the V-UE layer, fog layer, and cloud layer. The V-UE layer is composed of V V-UEs, the fog layer includes K V-FNs, and the cloud layer is mainly a distant cloud centre. Denote the sets of V-UEs and V-FNs by $\mathscr{V} = \{1,2,...,V\}$ and $\mathscr{K} = \{1,2,...,K\}$, respectively. All V-UEs and V-FNs are connected to a RSU via V2I wireless links, while the RSU is connected to the cloud centre by a fiber link.

Without loss of generality, we assume that the time is divided into equal-length time slots, each of a duration η, and each V-UE has one application to be either processed locally or offloaded to a V-FN or a cloud server for processing through the following procedure. At the beginning of each time slot, each V-UE sends an offloading request (containing information about its location, velocity and local processing capacity, etc. [29]) to the RSU [103]. After the offloading requests of all V-UEs and the instantaneous V2I wireless channel gains have been collected by the RSU [27], the RSU decides where the application of each V-UE should be processed, i.e., at the V-UE locally, at a V-FN, or at the cloud centre. Then, the offloading decisions will be sent to the corresponding V-UEs. For simplicity, we assume that the delay consumption of computation outcome transmission from a V-FN or a cloud server to a V-UE can be neglected [104].

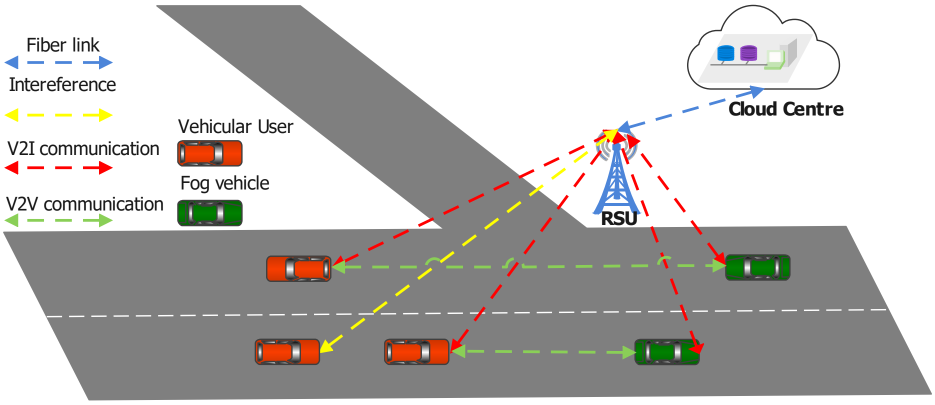

Fig. 5.1 System architecture of a three-layer computing system.

The offloading decision for V-UE v at the t-th time slot is denoted by $x_v(t), y_v(t), z_v(t) \in \{0,1\}$, where $t = 1,2,...; x_v(t) = 1, y_v(t) = 1, z_v(t) = 1$ indicate that the application is processed by V-UE v itself, by a V-FN, or by the cloud, respectively; otherwise, $x_v(t) = 0, y_v(t) = 0, z_v(t) = 0$; and we have

$$x_v(t) + y_v(t) + z_v(t) = 1, \forall v \in \mathscr{V}. \tag{5.1}$$

The offloading decisions of all V-UEs at time slot t are denoted by a $V \times 3$ matrix $\mathbb{O}(t)$, where the v-th row contains the offloading decision of V-UE v, i.e.,

$$\mathbb{O}(t) = \begin{bmatrix} x_1(t), & y_1(t), & z_1(t) \\ \vdots & \vdots & \vdots \\ x_V(t), & y_V(t), & z_V(t) \end{bmatrix}_{V\times 3}. \tag{5.2}$$

The application of V-UE v is described by $App_v = \{D_v, \Lambda_v, s_v, d_{vR}\}, v \in \mathscr{V}$, where D_v denotes the input data size (in bits) of the application, Λ_v is the required processing density (in CPU cycle/bit), s_v is the velocity (in km/h) of V-UE v, d_{vR} is the distance (in km) between V-UE v and the coverage edge of the RSU in its direction of movement [105]. The total required number of CPU cycles to process the application is given by $A_v = D_v\Lambda_v$ [4].

5.2.2 Local Processing

Let f_v^{local} denote the local computation capability (in CPU cycles/s) of V-UE v. Since the application does not need to be sent to any server, the service delay at the t-th time slot only consists of the processing delay, and we have

$$T_v^{local}(t) = \frac{A_v}{f_v^{local}}. \tag{5.3}$$

5.2.3 Fog Processing

In fog processing mode, V-UE v will offload it application to a nearby V-FN k, and the input data of App_v should be transmitted from V-UE v to V-FN k. Thus, the transmission delay and the energy consumption incurred during transmission have to be considered.

Divide the total available uplink bandwidth into M RBs, which are denoted by $\mathscr{M} = \{1,...,M\}$, where the bandwidth (in Hz) of the m-th RB is denoted by W_m. The channel power gain from the v-th V-UE to the k-th V-FN at the t-th time slot over the m-th RB is given by

$$g_{v,k}^m(t) = \alpha_{v,k}^m(t)|h_{v,k}^m(t)|^2 \tag{5.4}$$

where $\alpha_{v,k}^m(t)$ is the large-scale fading of the channel from the v-th V-UE to the k-th V-FN, which consists of the distance dependent path-loss and log-normal shadowing [106]; $h_{v,k}^m(t)$ denotes the Rayleigh fading, which follows the complex Gaussian distribution $CN(0,1)$ [99]. The channel power gain from the v-th V-UE to the RSU at the t-th time slot over the m-th RB is denoted by $g_{v,R}^m(t) = \alpha_{v,R}^m(t)|h_{v,R}^m(t)|^2$. A summary of the channel model for V2I/V2V link is given in TABLE 5.1 [99], [106]

Table 5.1 Channel Models for V2I/V2V link

Parameter	**The type of a link**
Pathloss model - V2I	$128.1 + 37.6 log_{10} d_{vR}$ (d_{vR} in km)
Pathloss model - V2V	LOS in WINNER + B1
Shadowing distribution	Log-normal
Shadowing standard deviation - V2I	8 dB
Shadowing standard deviation - V2V	3 dB
Fast fading	Rayleigh fading

Then, the received SINR from the v-th V-UE to the k-th V-FN over the m-th RB at the t-th time slot can be expressed as

$$\gamma_{v,k}^m(t) = \frac{P_v^m(t)g_{v,k}^m(t)}{\sigma^2 + \beta_{v,k}^m(t)}, \tag{5.5}$$

where $P_v^m(t)$ denotes the transmission power of V-UE v at the t-th time slot; σ^2 is the AWGN power; $\beta_{v,k}^m(t)$ is the sum interference caused by other V2V and/or V2I links which share the same RB with V-UE v, and it can be formulated as

$$\beta_{v,k}^m(t) = \sum_{v' \in \mathscr{V}_2} [y_{v'} a_{v',m} P_{v'}^m(t) g_{v',k}^m(t) + z_{v'} a_{v',m} P_{v'}^m(t) g_{v',R}^m(t)], \tag{5.6}$$

where $\mathscr{V}_2$ is the set of remote-processing V-UEs; $a_{v',m}$ is the binary spectrum allocation indicator, where $a_{v',m} = 1$ if the v'-th V2V/V2I link shares the same RB with the v-th V2V link, otherwise $a_{v',m} = 0$; $g_{v',k}^m(t)$ is the channel power gain from the v'-th V-UE to the k-th V-FN at the t-th time slot over the m-th RB; $g_{v,R}^m(t)$ is the channel power gain from the v-th V-UE to the RSU at the t-th time slot over the m-th RB.

Then, the corresponding transmission rate from V-UE v to the k-th V-FN at the t-th time slot over the m-th RB can be expressed as

$$r_{v,k}^m(t) = W_m log_2(1 + \gamma_{v,k}^m(t)), \tag{5.7}$$

and the transmission delay of application App_v at the t-th time slot can be given by

$$T_{v,k}^{trans}(t) = \frac{D_v}{r_{v,k}^m(t)}. \tag{5.8}$$

Due to the limited computation capacity at each V-FN, the queueing delay at the V-FN cannot be ignored. At the t-th time slot, assume the application of V-UE v is assigned to be processed at the k-th V-FN, the application arrival rate at the transmission queue in the transmission channel of V-FN k is assumed to follow a Poisson process with rate λ_k, and a Poisson distribution during a time slot can be given by

$$P(V_{1,k}|t, \lambda_k) = \frac{(\lambda_k t)^{V_{1,k}}}{V_{1,k}!} exp(-\lambda_k t), \tag{5.9}$$

where $V_{1,k}$ is the number of fog processing applications at the k-th V-FN, and the total number of fog processing applications is $V_1 = \sum_{k=1}^{K} V_{1,k}$.

In addition, the corresponding service rate is $\mu_k = \eta r_{v,k}^m(t)/D_v$, which is assumed to follow a exponential process, and an exponential distribution during a time slot can be given by

$$P(r) = \mu_k exp(-\mu_k r), \tag{5.10}$$

where r is the time between service completions. To ensure the steady state of the system, the utilisation rate is required to be lower than 1, i.e.,

$$\begin{aligned}\rho_k &= \frac{\lambda_k}{\mu_k} \\ &= \frac{\lambda_k D_v}{\eta r_{v,k}^m(t)} < 1.\end{aligned} \tag{5.11}$$

According to the traditional FCFS M/M/1 queueing model, we have

$$(\lambda_k + \mu_k)P_{V_{1,k}} - \lambda_k P_{V_{1,k}-1} - \mu_k P_{V_{1,k}+1} = 0,\ V_{1,k} \geq 1 \tag{5.12}$$

$$\lambda_k P_0 - \mu_k P_1 = 0. \tag{5.13}$$

To consider the steady state flow rates at the boundary between two states, we have

$$\lambda_k P_{V_{1,k}-1} = \mu_k P_{V_{1,k}}. \tag{5.14}$$

Thus, according to (5.11), (5.12), (5.13) and (5.14), we have

$$P_0 = 1 - \rho_k. \tag{5.15}$$

$$P_{V_{1,k}} = (\rho_k)^{V_{1,k}}(1 - \rho_k). \tag{5.16}$$

Based on Little's theorem [100], the average queueing delay of application App_v in the transmission channel is given by [107]

$$T_{v,k}^{wait}(t) = \frac{\rho_k}{\mu_k - \lambda_k}. \tag{5.17}$$

After all the input data of application App_v has been received by the k-th V-FN, it starts to process the application, and the corresponding processing delay can be expressed as

$$T_{v,k}^{proc}(t) = \frac{A_v}{f_{v,k}^{fog}}. \tag{5.18}$$

Therefore, the total service delay of processing application App_v at the k-th V-FN can be given by

$$T_v^{fog}(t) = T_{v,k}^{trans}(t) + T_{v,k}^{wait}(t) + T_{v,k}^{proc}(t). \tag{5.19}$$

5.2.4 Cloud Processing

When there is no available V-FNs for task offloading, applications will be offloaded to the cloud centre for processing, where application App_v is first transmitted from V-UE v to the RSU via a V2I link, similarly defined as (5.4), and the corresponding channel power gain from the v-th V-UE to the RSU is given by

$$g^m_{v,R}(t) = \alpha^m_{v,R}(t)|h^m_{v,R}(t)|^2, \tag{5.20}$$

where $\alpha^m_{v,R}(t)$ and $h^m_{v,R}(t)$ are similarly defined as $\alpha^m_{v,k}(t)$ and $h^m_{v,k}(t)$, respectively.

The SINR at the RSU and the transmission rate from V-UE v to the RSU at the t-th time slot over the m-th RB are given by

$$\gamma^m_{v,R}(t) = \frac{P^m_v(t)}{\sigma^2 + \beta^m_{v,R}(t)}, \tag{5.21}$$

$$r^m_{v,R}(t) = W_m log_2(1 + \gamma^m_{v,R}(t)), \tag{5.22}$$

where $\beta^m_{v,R}$ is the total interference caused by other V2V and V2I links that share the same RB, and it can be similarly defined as (5.6).

After the whole application is cached at the RSU, and it then is transmitted from the RSU to the cloud centre via a high-speed fibre wired link. Therefore, the total transmission delay of application App_v at the t-th time slot for processing at the cloud centre includes two parts, i.e., $T^{trans}_{v,R}(t)$ and $T^{trans}_{R,C}(t)$, and it can be obtained by

$$T^{trans}_{v,c}(t) = T^{trans}_{v,R}(t) + T^{trans}_{R,C}(t), \tag{5.23}$$

$$T^{trans}_{v,R}(t) = \frac{D_v}{r^m_{v,R}(t)}, \tag{5.23a}$$

$$T^{trans}_{R,C}(t) = \frac{D_v}{r_{R,C}(t)}. \tag{5.23b}$$

where $r_{R,C}$ (in bits/s) denotes the transmission rate of the high-speed wired fibre link.

Cloud servers at the cloud centre usually have sufficient computation resources, energy and processing ability, we ignore the queueing delay and the energy consumption at the cloud centre [108], [109]. We assume the propagation delay, i.e,. T_{pd}, in the backbone network cannot be neglected, which is incurred for a longer response delay of cloud processing [40]. Similar to the local-processing and fog-processing, the processing delay of application App_v at the cloud centre at the t-th time slot is given by

$$T_{v,C}^{proc}(t)=\frac{A_v}{f_v^{cloud}}, \tag{5.24}$$

where f_v^{cloud} (in CPU cycles/s) denotes the cloud processing capability for V-UE v. Then, the total service delay of processing application App_v at the cloud centre can be expressed as

$$T_v^{cloud}(t)=T_{v,C}^{trans}(t)+T_{pd}(t)+T_{v,C}^{proc}(t). \tag{5.25}$$

5.3 Problem Formulation

For each time slot $t\in\{0,1,...,T\}$, we formulate the problem of jointly optimising computation offloading decision and resource allocation for a three-layer VFC system and show that it is NP-hard. Based on the delays in local, fog, and cloud processing, the service delay of V-UE v at the t-th time slot is given by

$$T_v(t)=x_v(t)T_v^{local}(t)+y_v(t)T_v^{fog}(t)+z_v(t)T_v^{cloud}(t). \tag{5.26}$$

where $T_v^{local}(t), T_v^{fog}(t)$, and $T_v^{cloud}(t)$ are given in (5.3), (5.19) and (5.25), respectively.

We aim to minimise the maximum service delay consumption among all V-UEs by jointly optimising the computation offloading decision $\mathbb{O}(t)$, the transmission power $\mathbf{P}=\left[\mathbf{P}_1^m,\cdots,\mathbf{P}_v^m\right]$, the RB assignment $\mathbf{a}=[\mathbf{a}_{v,1},\cdots,\mathbf{a}_{v,m}]$, the V-FN assignment $\mathbf{b}=\left[\mathbf{b}_{v,1},\cdots,\mathbf{b}_{v,k}\right]$, and the computation resources allocation $\mathbf{f}^{fog}=\left[\mathbf{f}_1^{fog},\cdots,\mathbf{f}_k^{fog}\right]$. For local processing V-UE v, we let $P_v=0$ and $a_{v,m}=0$. Note sets $\mathscr{V}_1$ and $\mathscr{V}_2$ denote applications are processed among V-FNs or among remote servers including V-FNs and the cloud centre, respectively. Then, the joint optimisation problem can be formulated as follows

$$\mathscr{P}1: \min_{\mathbb{O},\mathbf{a},\mathbf{P},\mathbf{b},\mathbf{f}^{fog}} \max_{v\in\mathscr{V}} \sum_k T_v(t), \tag{5.27}$$

$$\begin{aligned}
s.t.\ & x_v(t),y_v(t),z_v(t)\in\{0,1\},\ \forall v\in\mathscr{V}, && (5.27a)\\
& x_v(t)+y_v(t)+z_v(t)=1,\ \forall v\in\mathscr{V}, && (5.27b)\\
& 0\le b_v^k(t)f_{v,k}^{fog}(t),\ \forall v\in\mathscr{V}_1, && (5.27c)\\
& \sum_{v\in\mathscr{V}_1} b_v^k f_{v,k}^{fog}(t)\le F_k^{fog}(t),\ \forall k\in\mathscr{K}, && (5.27d)\\
& b_v^k(t)\in\{0,1\},\ \forall v\in\mathscr{V}_1,\ \forall k\in\mathscr{K}, && (5.27e)
\end{aligned}$$

$$\sum_{k=1}^{K} b_v^k(t) = 1, \ \forall v \in \mathscr{V}_1, \tag{5.27f}$$

$$0 \leq P_v^m(t) \leq P_v^{max}, \ \forall v \in \mathscr{V}_2, \tag{5.27g}$$

$$\sum_m a_{v,m} = 1, \ \forall v \in \mathscr{V}_2, \tag{5.27h}$$

$$\sum_m a_{k,m} = 1, \ \forall k \in \mathscr{K}, \tag{5.27i}$$

$$T_v(t) \leq \tau_v(t), \ \forall v \in \mathscr{V}, \tag{5.27j}$$

$$y_v T_v^{fog}(t) < b_v^k(t)\tau_k(t), \ \forall v \in \mathscr{V}_1, \tag{5.27k}$$

where F_k^{fog} is the available computation capacity of the k-th V-FN, and τ_v, τ_k are the maximum tolerable latency threshold for V-UE v and V-FN k, respectively; (5.27a) and (5.27b) are the constraints on the binary offloading decision indicators for each V-UE; (5.27c) indicates that the computation resource allocated to each fog-processing V-UE v should be non-negative; (5.27d) shows that the allocated computation resources cannot exceed the available computation capability of each V-FN; (5.27e) is the constraints on the binary computation allocation indicator for each V-UE; (5.27f) guarantees each application can be allocated to only one V-FN for processing at the t-th time slot; (5.27g) is the constraints on transmit power of each V-UE, respectively; (5.27h) and (5.27i) indicate each of the V2I link accesses a single RB; (5.27j) indicates that the service delay should be kept below the maximum tolerable latency threshold for each V-UE; and (5.27k) indicates that the service delay at a V-FN should be kept below its maximum tolerable latency threshold.

A summary of the used notations is presented in TABLE 5.2

Table 5.2 Notation Definitions

Notation	**Definition**
$\mathscr{V}, V$	The set/number of all V-UEs
$\mathscr{K}, K$	The set/number of all V-FNs
$V_1/\mathscr{V}_1$	The number/set of fog-processing V-UEs
$V_2/\mathscr{V}_2$	The number/set of remote-processing V-UEs
x_v, y_v, z_v	Offloading decisions of V-UE v
$\mathbb{O}$	Matrix of offloading decisions for all V-UEs
D_v	Input data size of App_v
Λ_v	Processing density App_v
A_v	Total required number of CPU cycles of App_v

s_v	The velocity of V-UE v
$\mathscr{M}, M$	The set/number of RBs
W_m	The bandwidth of each RB
$g_{v,k}^m(t)$	The channel power gain from V-UE v to V-FN k
$\alpha_{v,f}^m(t)$	The large-scale fading of the channel from V-UE v to V-FN k
$h_{v,f}^m(t)$	The Rayleigh fading of the channel from V-UE v to V-FN k
λ_k	Application arrival rate at the k-th transmission channel
μ_k	Application service rate at the k-th transmission channel
ρ_k	The utilisation rate
N	The toal number of clusters
η	The length of each time slot
s_k	The velocity of V-FN k
d_{vR}	The distance between V-UE v and the RSU
σ^2	The additive white Gaussian noise power
P_v^m	Transmit power of V-UE v
T_{pd}	Propagation delay in cloud processing
$r_{R,C}$	Transmission rate between the RSU and the cloud centre
I	The number of fireworks
$\hat{s}$	The number of explosion sparks
$\hat{m}$	The number of mutation sparks
$\gamma_{v,k}^m, r_{v,k}^m, \beta_{v,k}^m$	SINR/Transmission rate/Interference between V-UE v and V-FN k
$\gamma_{v,R}^m, r_{v,R}^m, \beta_{v,k}^m$	SINR/Transmission rate/Interference between V-UE v and the RSU
$f_v^{local}, f_{v,k}^{fog}, f_v^{cloud}$	Processing ability of V-UE v in local/fog/cloud processing
$T_v^{local}, T_v^{fog}, T_v^{cloud}$	Service delay of V-UE v in local/fog/cloud processing

5.4 Proposed Algorithm

In this section, we propose a fireworks algorithm based offloading optimisation and resource allocation algorithm (FORA), in conjunction with a bisection method based V-FN computation resource allocation algorithm and a clustering method based communication resource allocation algorithm to solve the min-max problem.

5.4.1 Fireworks Algorithm based Offloading Optimisation and Resource Allocation Algorithm

Since problem $\mathscr{P}1$ is difficult to solve, we propose a fireworks algorithm based offloading optimisation and resource allocation algorithm (FORA), based on the traditional fireworks algorithm [82], to solve the formulated problem (5.14). The FORA is summarised in Algorithm 5.1. Firstly, we initialise I random offloading decisions for all V-UEs $(\mathbb{O}_1^{(0)}, ..., \mathbb{O}_I^{(0)})$, and they become the swarm of initial fireworks in the solution space. Then, in the l-th iteration ($l = 1,...,L$, where L is the maximum allowed iteration), we take the objective function of (5.20) as the fitness function, and the fitness value can be calculated by $f(\mathbb{O}_i^{(l)}) = \max\limits_{v\in\mathscr{V}} \sum\limits_{k} T_v(\mathbb{O}_i^{(l)}),\ i = 1,...,I$. In the meantime, each firework $\mathbb{O}_i^{(l)}$ generates $\hat{s}_i^{(l)}$ new offloading decisions matrices which are called explosion sparks, and the number of explosion sparks of each firework is given by [82]

$$\hat{s}_i^{(l)} = ceil\left(S \frac{f_{max} - f(\mathbb{O}_i^{(l)}) + \varepsilon_1}{\sum\limits_{i=1}^{I} (f_{max} - f(\mathbb{O}_i^{(l)})) + \varepsilon_1} \right), \tag{5.28}$$

where $ceil(\cdot)$ denotes the ceiling function, S is a constant parameter used for constraining the total number of explosion sparks, $f_{max} = \max\limits_{i}(f(\mathbb{O}_i^{(l)}))$, and ε_1 is an extremely small number to avoid zero division errors. In order to avoid each firework generates too many or too less explosion sparks, the number of explosion sparks of the i-th fireworks can be constrained by

$$\widetilde{s}_i^{(l)} = \begin{cases} round(\theta_1 S), & if\ \hat{s}_i^{(l)} < \theta_1 S \\ round(\theta_2 S), & if\ \hat{s}_i^{(l)} > \theta_2 S, \theta_1 < \theta_2 < 1 \\ round(\hat{s}_i^{(l)}), & otherwise \end{cases} \tag{5.29}$$

where $\widetilde{s}_i^{(l)}$ is the number of explosion sparks that generated by the i-th firework, $round(\cdot)$ is the rounding off function, ψ_1 and ψ_2 are given constraints. After $\hat{s}_i^{(l)}$ has been confirmed, each explosion spark of firework $\mathbb{O}_i^{(l)}$ can be generated by first randomly choosing z rows (where $z \in [1,V)$) from firework $\mathbb{O}_i^{(l)}$, and then perform left circular shift by one position on each chosen row, while the rest of rows $(V - z)$ of the explosion spark are kept the same as the corresponding ones of firework $\mathbb{O}_i^{(l)}$.

To guarantee the spark diversity and the searching capability, in the l-th iteration, $\hat{m}^{(l)}(0 \leq \hat{m}^{(l)} \leq I)$ mutation sparks are generated by randomly selecting $\hat{m}^{(l)}$ fireworks from the I fireworks $(\mathbb{O}_1^{(l)},...,\mathbb{O}_I^{(l)})$ and randomly resetting some offloading decisions therein.

For each firework, explosion spark and mutation spark, the V-FN assignment and the corresponding computation resource allocation are obtained by using the bisection method [102], which will be presented in section 5.4.2. The communication resource allocation is obtained by the clustering method [106], which will be introduced in section 5.4.3, and accordingly the fitness function value is calculated. Among all the fireworks, explosion sparks and mutation sparks, the one with the smallest fitness value is selected as a new firework $\mathbb{O}_1^{(l+1)}$ for the upcoming iteration. Denoting the set of all fireworks, explosion sparks and mutation sparks excluding $\mathbb{O}_1^{(l+1)}$ in the l-th iteration by $R_{est}^{(l)}$ (i.e., $\mathbb{O}_1^{(l+1)} \notin R_{est}^{(l)}$), the other ($I$ - 1) fireworks ($\mathbb{O}_2^{(l+1)},...,\mathbb{O}_I^{(l+1)}$) are selected from $R_{est}^{(l)}$ according to the roulette wheel selection method [82], where the probability of $\mathbb{A}_y^{(l)}$ ($\mathbb{A}_y^{(l)} \in R_{est}^{(l)}, y = 1,...,|R_{est}^{(l)}|$) being selected is determined based on the Manhattan distance [84] as follows,

$$p(\mathbb{A}_y^{(l)}) = \frac{R(\mathbb{A}_y^{(l)})}{\sum\limits_{m=1}^{|R_{est}^{(l)}|} R(\mathbb{A}_{y'}^{(l)})}, \tag{5.30}$$

where $R(\mathbb{A}_y^{(l)})$ is the sum of Manhattan distances between matrix $\mathbb{A}_y^{(l)}$ and all the other matrices in set $R_{est}^{(l)}$, which is given by

$$R(\mathbb{A}_y^{(l)}) = \sum_{y'=1, y' \neq y}^{|R_{est}^{(l)}|} \parallel \mathbb{A}_y^{(l)} - \mathbb{A}_{y'}^{(l)} \parallel . \tag{5.31}$$

When the iteration converges or reaches the maximum allowed iteration, among all the fireworks, explosion sparks and mutation sparks, the one with the smallest fitness value is chosen as the optimal offloading decision $\mathbb{O}^*$ and the corresponding RB assignment, V-FN assignment , transmit power of each V2I link transmitter and the computation resource allocation at each V-FN returns the optimal results $\mathbf{a}^*, \mathbf{b}^*, \mathbf{P}^*$ and $\mathbf{f_k^{fog}}^*$, respectively.

5.4.2 Computation Resource Allocation

When computation offloading decision I has been obtained, in problem $\mathscr{P}1$, the communication resource allocation which includes the uplink transmission power $\mathbf{P}$ and the RB assignment $\mathbf{a}$, and the computation resource allocation (including the V-FN assignment $\mathbf{b}$ and the computation resources $\mathbf{f^{fog}}$) are decoupled from each other in both the objective and the constraints. Thus, problem $\mathscr{P}1$ can be solved by optimising communication and computation resources independently. Moreover, for analytical tractability, we assume each application of a V-UE cannot be partitioned into sub-applications, and at each time slot, RSU can receive global information of all applications, V-UEs as well as V-FNs. In this section, we discuss the computation resource allocation, while the communication resource assignment will be presented in the next section.

For computation resource allocation among all V-UEs in $\mathscr{V}_1$, assuming that the RB assignment $\mathbf{a}$ and the transmission power control $\mathbf{P}$ are given, we have

$$\mathscr{P}2: \min_{\mathbf{b}, \mathbf{f}^{fog}} \max_{v \in \mathscr{V}_1} \sum_k T_v(t), \tag{5.32}$$

Algorithm 5.1 Fireworks Algorithm based Offloading Optimisation and Resource Allocation Algorithm (FORA)

```
$N, K, F^{fog}, B, OE, OM, \theta_r, \theta_1, \theta_2$.
Initialise $l = 1, F^{(0)} = 0, \varepsilon_1 = 10^{-6}$.
**for** $t = 1 : T$ **do**
    Generate $I$ random fireworks at each time slot $\{\mathbb{O}_1^{(0)}, ..., \mathbb{O}_I^{(0)}\}$.
    **while** $l \leq L$ **do**
        **for** $i = 1 : I$ **do**
            For firework $\mathbb{O}_i^{(l)}$, perform resource allocation by Algorithm 5.2 and Algorithm 5.3.
            Calculate the fitness value of firework $\mathbb{O}_i^{(l)}$ by using (5.27).
            Calculate the number of explosion sparks $\hat{s}_i^{(l)}$ according to (5.28) and (5.29).
            Generate $\hat{s}_i^{(l)}$ explosion sparks from firework $\mathbb{O}_i^{(l)}$.
            For each explosion spark, run Algorithm 5.2 and Algorithm 5.3.
            Calculate the fitness value of each explosion spark.
        **end for**
        Generate $\hat{m}^{(l)}$ mutation sparks.
        For each mutation spark, run Algorithm 5.2 and Algorithm 5.3.
        Calculate the fitness value of each mutation spark.
        The firework, explosion spark or mutation spark with the smallest fitness value is chosen as $I_1^{(l+1)}$, and the smallest fitness value is denoted by $f^{(l)}$.
        **if** $|f^{(l)} - f^{(l-1)}| < \varepsilon$ **then**
            break;
        **else**
            Fireworks $(\mathbb{O}_2^{(l+1)}, ..., \mathbb{O}_I^{(l+1)})$ are selected according to (5.30), (5.31).
            $l = l + 1$;
        **end if**
    **end while**
**end for**
**Return:** The optimal offloading decision $\mathbb{O}^* = \mathbb{O}_1^{(l+1)}$ if $l < L$, otherwise $\mathbb{O}^* = \mathbb{O}_1^{(L+1)}$, and the corresponding RB assignment $\mathbf{a}^*$, V-FN assignment $\mathbf{b}^*$, transmit power allocation $\mathbf{P}^*$ and the fog computation resource allocation $\mathbf{f}^{\mathbf{fog}^*} = \left[\mathbf{f}_\mathbf{1}^{\mathbf{fog}^*}, ..., \mathbf{f}_\mathbf{k}^{\mathbf{fog}^*}\right]$.
```

$$s.t.\ \ (5.27c)-(5.27f),(5.27j)-(5.27k)$$

To obtain the sub-optimal solution to problem $\mathscr{P}2$, we propose a bisection method based application and computation resource allocation algorithm (BACA). We first construct a V-by-K matrix $\mathbf{H}$ to assign a V-FN for each V-UE to process its application. Each element in $\mathbf{H}$ records the service delay thresholds difference at the t-th time slot between V-UE v and V-FN k (i.e., $\tau_v(t)-\tau_k(t)$), and they can be calculated as follows,

$$\tau_v(t)=\frac{d_{vR}(t)}{s_v(t)},\ \forall v\in\mathscr{V}_1, \tag{5.33}$$

$$\tau_k(t)=\frac{d_{kR}(t)}{s_k(t)},\ \forall k\in\mathscr{K}, \tag{5.34}$$

where $\tau_v(t)$ indicates the service delay threshold of V-UE v for remote processing, $\tau_k(t)$ indicates the maximum tolerable latency threshold of V-FN k; and $d_{kR}(t)$ is the distance between V-FN k and the coverage edge of the RSU in its direction of motion, and $s_k(t)$ is the velocity of V-FN k.

In the l-th iteration, comparing all the results of V-UE v in matrix $\mathbf{H}_l$, the k-th V-FN is chosen for processing the application of V-UE v according to the smallest non-zero value among V-UE v and all V-FNs. Based on assignment $\mathbf{b}$ for all V-UEs, for the k-th V-FN, problem $\mathscr{P}2$ is converted to

$$\mathscr{P}3:\ \min_{\mathbf{f}_k^{fog}}\ \max_{v\in\mathscr{V}_{1,k}}\frac{A_v}{f_{v,k}^{fog}(t)}+B_v(t), \tag{5.35}$$

$$s.t.\ \ (5.27c)-(5.27f),$$

where $\mathscr{V}_{1,k}$ denotes the set of fog-processing V-UEs shares the computation resources of V-FN k, and $\mathscr{V}_{1,1}\bigcap\mathscr{V}_{1,2},\ldots,\bigcap\mathscr{V}_{1,k}=\varnothing$; $B_v=\dfrac{D_v}{r_{v,R}(t)}+\dfrac{D_v}{r_{R,k}(t)}+\dfrac{\lambda_k D_v^2}{\delta r_{R,k}(t)(\delta r_{R,k}(t)-\lambda_k D_v)}$ is a constant; Letting $\Delta_B(t)=\max\limits_{v\in\mathscr{V}_{1,k}}\{\dfrac{C_v}{f_{v,k}^{fog}(t)}+B_v(t)\}$, and problem $\mathscr{P}3$ is converted to

$$\mathscr{P}4:\ \min_{\mathbf{f}_k^{fog},\Delta_B(t)}\ \Delta_B(t), \tag{5.36}$$

$$s.t.\ \ (5.27c)-(5.27d),$$
$$\frac{A_v}{f^{fog}_{v,k}(t)}+B_v(t)\leq\Delta_B(t),\ \forall v\in\mathscr{V}_{1,k}. \tag{5.36a}$$

Since $\frac{C_v}{f^{fog}_{v,k}(t)}\geq 0$ and based on (5.27d) and (5.36a), we have

$$\sum_{v\in\mathscr{V}_{1,k}}\frac{A_v}{(\Delta_B(t)-B_v(t))}\leq\sum_{v\in\mathscr{V}_{1,k}}f^{fog}_{v,k}(t)\leq F^{fog}_k(t). \tag{5.37}$$

After few iterations until all the computation resources of the k-th V-FN will be distributed among all $v\in\mathscr{V}_{1,k}$. Then we have $\sum\limits_{v\in\mathscr{V}_{1,k}}\frac{A_v}{(\Delta_B(t)-B_v(t))}=\sum\limits_{v\in\mathscr{V}_{1,k}}f^{fog}_{v,k}(t)=F^{fog}_k(t)$ and problem $\mathscr{P}4$ can be converted to

$$\mathscr{P}5:\ \min_{\Delta_B(t)}\Delta_B(t), \tag{5.38}$$

$$s.t.\ \sum_{v\in\mathscr{V}_{1,k}}\frac{A_v}{(\Delta_B(t)-B_v(t))}=F^{fog}_k(t). \tag{5.38a}$$

We utilise the bisection method to deal with problem $\mathscr{P}5$ as summarised in Algorithm 5.2, where $V_{1,k}$ is the number of fog-processing V-UEs at the k-th V-FN. And Δ_B^* indicates the minimum value of Δ_B and the optimal computation resource allocation for V-UE v at the k-th V-FN can be obtained by $f^{fog*}_{v,k}(t)=\frac{A_v}{(\Delta^*_B(t)-B_v(t))}$ and the V-FN assignment $\mathbf{b}^*$.

5.4.3 Communication Resource Allocation

After computation resource allocation $\mathbf{f}^{fog}$ and V-FN assignment $\mathbf{b}$ are obtained, problem of $\mathscr{P}5$ degrades to the joint optimisation of RB assignment $\mathbf{a}^m$ and transmit power allocation among all remote-processing V-UEs $\mathbf{P}^m$, then the communication resource allocation subproblem can be converted to

$$\mathscr{P}6:\ \min_{\mathbf{a},\mathbf{P}}\ \max_{v\in\mathscr{V}_2}T_v(t), \tag{5.39}$$

$$s.t.\ \ (5.27g)-(5.27j).$$

In remote-processing mode, each application will be sent to the RSU first, after the assignment of V-FN for all $v\in\mathscr{V}_1$, the application will be sent to the corresponding V-

Algorithm 5.2 Bisection Method based Application and Computation Resource Allocation Algorithm (BACA)

```
for $k = 1 : K$ do
    Initialise the precision $\varepsilon_2 > 0$, $\Delta_{B_{down}}(t) = \max_{v \in \mathcal{V}_{1,k}} B_v(t)$

    and $\Delta_{B_{up}}(t) = \sum_{v \in \mathcal{V}_{1,k}} (A_v V_{1,k}/F_k^{fog} + B_v(t))$
    Repeat
    $\Theta = (\Theta_{up} + \Theta_{down})/2$.
    if $\sum_{v \in \mathcal{V}^{fog}} (B_v/\Theta - \Delta_v) > F^{fog}$ then
        $\Theta_{down} = \Theta$.
    else
        $\Theta_{up} = \Theta$.
    end if
    until $|\Theta_{up} - \Theta_{down}| \leq \varepsilon_2$.
    $\Theta^* = |\Theta_{up} - \Theta_{down}|/2$.
    Return: $\mathbf{f_k^{fog}}^*$.
end for
Return: $\mathbf{f^{fog}}^* = [f_1^{fog^*}, ..., f_K^{fog^*}]$, $\mathbf{b}^*$.
```

FN for processing. Therefore, a fog-processing application may use two different RBs, and all remote-processing applications may need V_2 RBs, where V_2 is the total number of remote-processing V-UEs. To this end, we propose a cluster-based communication resource allocation algorithm (CCAA) to solve problem $\mathscr{P}6$ as summarised in Algorithm 5.3. We first calculate the required total number of RBs according to the total number of remote-processing V-UEs, V_2. Then we degrade the communication resource allocation problem into two cases (which detailed in section 5.4.3), basing on the total number of RBs M and the total required number of RBs V_2.

The Case of $V_2 \leq M$

If the total required number of RBs V_2 is less than the total number of RBs M, each V-UE may be allocated with at least one for data transmission, and there is no interference among all remote-processing applications during the transmission period. Then, problem $\mathscr{P}6$ degrades to a power allocation problem for each V2I/V2V link. To solve this problem, each V2I link has to guarantee the minimum SINR threshold γ_0, such as $\gamma_{v,k}^{m}(t) \geq \gamma_0$ and $\gamma_{v,R}^{m}(t) \geq \gamma_0$, and the transmission power for the v-th V-UE can be formulated as,

Algorithm 5.3 Cluster-Based Communication Resource Allocation Algorithm (CCAA)

```
Initialise $V_2$. Set $l = 0$.
if $V_2 \leq M$ then
    Arbitrarily assign one V2I/V2V link to one RB.
    Allocate transmit power $P_j^m$ according to (5.33) and (5.34).
else
    Arbitrarily assign one V2I/V2V link to each of the $N$ clusters.
    for $j \in \mathcal{V}_2$ do
        for $n = 1 : N$ do
            To calculate the intra-cluster interference according to $\psi_{j,n} = \sum_{j' \in C_n, j' \neq j} (\alpha_{j',j} + \alpha_{j,j'})$.
        end for
        To assign the $j$-th V2I link into $n^*$-th cluster with $\psi_{j,n^*} = arg\ min \sum_{j' \in C_n, j' \neq j} (\alpha_{j',j} + \alpha_{j,j'})$.
    end for
    Return: V2I/V2V clustering result.
    for $n = 1 : N$ do
        Randomly select a target V2I/V2V link and marked it as V2I link $i$ in the $n$-th cluster.
        for $j = 1 : C_n$ do
            Allocate transmit power $P_{j,i}^m$ according to (5.41).
            Update Lagrange multipliers $\omega_j, \varphi_j$ according to (5.42) and (5.43), respectively.
        end for
    end for
    Return: $P_{j,i}^m, \forall j \in C_n, n \in N$.
end if
Return: $\mathbf{a}^*$, $\mathbf{P}^*$.
```

$$P^m_{v,k}(t) \leq \min\{P^{max}_{v,k}, \frac{\gamma_0 \sigma^2}{g^m_{v,k}(t)}\} \tag{5.40}$$

$$P^m_{v,R}(t) \leq \min\{P^{max}_{v,R}, \frac{\gamma_0 \sigma^2}{g^m_{v,R}(t)}\} \tag{5.41}$$

The Case of $V_2 > M$

If the total required number of RBs V_2 is lager than the total number of RBs M, we first need to group the V_2 V2I links into N clusters, which are denoted as $C_1, ..., C_N$, meanwhile, we need to minimise the intra-interference across all clusters (i.e., $\sum\limits_n (\sum\limits_{v',v \in C_n} \alpha_{v',v})$, where $\alpha_{v',v}$ is the large-scale fading from the v'-th V2I transmitter to the v-th V2I receiver) [106]. For simplicity, we approximate the total intra-cluster interference that it may experience in the n-th cluster by $\psi_{v,n} = \sum\limits_{v' \in C_n} (\alpha_{v',v} + \alpha_{v,v'})$. The v-th V2I link can be grouped into the cluster that gives the smallest value of $\psi_{v,n}$ among all clusters in the system. In addition, we assume that the total number of clusters is equal to the total number of RBs in the system, i.e., $N = M$. If there is only one V2I link in a cluster, the power resource allocation follows rules as shown in section the case of $V_2 \leq M$. Based on the result of the clustering, in each cluster, we randomly select a V2I/V2V link (i.e., $v' \in C_n, n = 1, ..., N$) and re-marked it as V2I/V2V link i, its corresponding SINR and the transmit power can be re-marke as $\gamma^m_i(t)$ and $P^m_i(t)$, respectively. For simplify, the rest of V2I/V2V links in the same cluster which share the same RB with the target user can be re-marked as $j(i \neq j, j \in C_n)$. Thus, problem $\mathscr{P}6$ can be degraded as a power allocation problem in each cluster as follows

$$\mathscr{P}7: \max_{\mathbf{P}} \{W_m log_2(1 + \gamma^m_{i,j}(t)) + W_m log_2(1 + \gamma^m_{j,i}(t))\} \tag{5.42}$$

$$\begin{aligned} s.t.\ & (5.27g) - (5.27j), \\ & \gamma^m_{i,j}(t) \geq \gamma_0,\ \forall j \in \mathscr{C}_n && (5.42a) \\ & \gamma^m_{j,i}(t) \geq \gamma_0.\ \forall i \in \mathscr{C}_n && (5.42b) \end{aligned}$$

Substituting (5.5) and (5.21) into (5.42a) and (5.42b), we have

$$P^m_{i,j}(t) \leq \{P^{max}_v, \frac{\gamma_0(\sigma^2 + \sum\limits_{j \in C_n} p^m_{j,i}(t) g^m_{j,i}(t))}{g^m_{i,j}(t)}\}, \tag{5.43}$$

$$P_{j,i}^{m}(t) = \min\{P_{v}^{max}, \frac{P_{i,j}^{m}(t)g_{i,j}^{m}(t)}{\gamma_0 g_{j,i}^{m}(t)} - \frac{\sigma^2}{g_{j,i}^{m}(t)} - \sum_{j' \in C_n, j' \neq j} \frac{P_{j',i}^{m}(t)g_{j',i}^{m}(t)}{g_{j,i}^{m}(t)}\}. \tag{5.44}$$

Then, in each cluster, problem $\mathscr{P}7$ can be converted as the transmit power control for all $j \in C_n$, and we have

$$\mathscr{P}8: \max_{P_{j,i}^{m}} \sum_{j \in C_n} W_m log_2(1 + \gamma_{j,i}^{m}(t)) \tag{5.45}$$

$$s.t. \ (5.42a), (5.42b), (5.43), (5.44).$$

For the j-th V2I/V2V link, we assume $P_{j,i}^{m} = exp(\xi_{j,i}^{m})$ and substituting it into (5.45) we have

$$\sum_{j \in C_n} W_m log_2 \left(\frac{exp(\xi_{j,i}^{m})g_{j,i}^{m}}{\sigma^2 + P_{i,j}^{m} g_{i,j}^{m} + \sum_{j' \in C_n, j' \neq j} exp(\xi_{j',i}^{m}) g_{j',i}^{m}} \right) = \frac{W_m}{ln(2)} \sum_{j \in C_n} \left[\frac{\xi_{j',i}^{m}}{ln(10)} + lg(g_{j,i}^{m}) - lg\left(\sigma^2 + p_{i,j}^{m} g_{i,j}^{m} + \sum_{j' \in C_n, j' \neq j} exp(\xi_{j',i}^{m} + ln(g_{j',i}^{m}))\right)\right] \tag{5.46}$$

Since $\mathscr{P}8$ is convex, the problem could be solved by using typical constrained optimisation method, such as Karush-Kuhn-Tucker (KKT) conditions [110]. Then, the Lagrange dual function of problem $\mathscr{P}8$ is given by (5.47), where $\omega_j \geq 0, \varphi_j \geq 0$ are Lagrange dual variables corresponding to (5.44) and (5.42b), respectively, and $T_{j,0}^{m} = W_m log_2 (1 + \gamma_0), \forall j \in C_n$.

$$\begin{aligned}\mathscr{L}(\xi_{j,i}^{m},\omega_j,\varphi_j) = {} & \frac{W_m}{ln(2)}\sum_{j\in C_n} ln\left(\frac{exp(\xi_{j',i}^{m})g_{j,i}^{m}}{\sigma^2+P_{i,j}^{m}g_{i,j}^{m}+\sum\limits_{j'\in C_n,j'\neq j} exp(\xi_{j',i}^{m})g_{j',i}^{m}}\right)\\ & -\sum_{j\in C_n}\omega_j\left(exp(\xi_{j',i}^{m})-P_v^{max}\right)\\ & +\frac{W_m}{ln(2)}\sum_{j\in C_n}\varphi_j\left[ln\left(1+\frac{exp(\xi_{j',i}^{m})g_{j,i}^{m}}{\sigma^2+p_{i,j}^{m}g_{i,j}^{m}+\sum\limits_{j'\in C_n,j'\neq j} exp(\xi_{j',i}^{m})g_{j',i}^{m}}\right)-T_{j,0}^{m}\right],\end{aligned} \tag{5.47}$$

$$P_{j,i}^{m}=\left[\frac{\varphi_j}{W_m}ln(2)+\frac{1+\omega_j}{\sum\limits_{j'\in C_n}\left[g_{j',i}^{m}(1+\varphi_{j'})/(\sigma^2+P_{i,j}^{m}g_{i,j}^{m}+\sum\limits_{j''\in C_n,j''\neq j'}P_{j'',i}^{m}g_{j'',i}^{m})\right]}\right]^{+}, \tag{5.48}$$

$$\omega_j^{l+1}=\left[\omega_j^{l}+\phi_1\left(T_{j,0}^{m}-W_m log_2\left(\frac{P_{j,i}^{m}g_{j,i}^{m}}{\sigma^2+P_{i,j}^{m}g_{i,j}^{m}+\sum\limits_{j'\in C_n,j'\neq j}P_{j',i}^{m}g_{j',i}^{m}}\right)\right)\right]^{+}, \forall j\in C_n, \tag{5.49}$$

By differentiating $\mathscr{L}(\xi_{j,i}^{f},\omega_j,\varphi_j)$ with respect to $\xi_{j,i}^{m}, j\in C_n$, and letting it equal to 0, we have the optimal transmit power allocation for the transmitter of the j-th V2I/V2V link, which is given by (5.44), where $[x]^{+}\triangleq \max(0,x)$. Then, we use the gradient method to update the values of the Lagrange multipliers as follows,

$$\varphi_j^{l+1}=\left[\varphi_j^{l}-\phi_2\left(p^{max}-p_{j,i}^{f}\right)\right]^{+}, \forall j\in C_n, \tag{5.50}$$

where l is iertation index; ϕ_1 and ϕ_2 are the positive step sizes of the associated constraints and they need to be properly initialised for guaranteeing the convergence and optimality. In this paper, we set the step size as $\phi_1=\phi_2=10^{-5}$ for the Lagrange multipliers update.

5.5 Complexity Analysis

In Algorithm 5.1, the computational complexity mainly comes from the resource (which including the computation resource and the communication resource) allocation procedures in Steps 7,11 and 15 and the number of sparks (which including the explosion sparks according to each fireworks and the mutation sparks) in Steps 9 and 14. We first describe the complexity from resource allocation procedures then discuss the complexity from sparks, respectively.

In each computation resource allocation procedure under Algorithm 5.2, based on the iterations for the bisection method to converge [102], the time complexity can be given by $\mathcal{O}(log_2(\Theta_{up}+\Theta_{down}/2))$. Similarly, in each communication resource allocation procedure under Algorithm 5.3 in Steps 7 - 13, the V2I clustering has a complexity of $\mathcal{O}(VN)$, where $V_2 = V$ because there are at most remote-processing V-UEs. In Steps 14 - 21 of Algorithm 5.3, it has a complexity of $\mathcal{O}(LNV)$, where L is the total number of iterations. Therefore, the total complexity of each communication resource allocation procedure under Algorithm 5.3 is $\mathcal{O}(VN(L+1))$.

According to Steps 9, the number of explosion sparks under each iteration has a complexity of $\mathcal{O}(\sum_{o=1}^{O} o\hat{s}_o)$. Similarly, the number of mutation sparks under each iteration has a complexity of $\mathcal{O}(\hat{m})$. Therefore, the total complexity of the total number of explosion and mutation sparks is $\mathcal{O}(\sum_{o=1}^{O} o\hat{s}_o+\hat{m})$.

Based on the previous discussion, the complexity of FROA under L times iteration is $\mathcal{O}(L(\sum_{o=1}^{O} o\hat{s}_o+\hat{m})(VN(L+1)+log_2(\Theta_{up}+\Theta_{down}/2))) = \mathcal{O}(L(\sum_{o=1}^{O} o\hat{s}_o+\hat{m})(VN(L+1)))$.

5.6 Simulation Results

In this section, we present the simulation results. We assume a single cell scenario with one BS located in the centre of a 500 m × 500 m urban area as illustrated in Fig. 5.1. There is a straight four-lane road (with one lane in each direction) passing through the middle of the considered square area, dividing the area into two equal rectangles. The width of each lane is 6 metres. Moreover, we assume that all the V-UEs and the V-FEs are uniformly distributed in the rectangular area of 24 m × 500 m spanned by the four-lane straight road, where the movement direction of each V-UE or V-FN is determined by the direction of the line that it locates in, and the local processing capability f_v^{local} is uniformly distributed in [50,400] M cycles/s. All parameter values used in the simulation are given in TABLE 5.3 [99, 27, 19], unless otherwise specified.

Table 5.3 Simulation Parameters

Parameters	**Value**
Transmit bandwidth, W_m	180 kHz
Transmit power of V-UE v, $p_{v,R}$	23 dBm
The noise power density at the BS, σ^2	-114 dBm
Cell radius,	500 m
BS antenna height,	25 m
BS antenna gain,	8 dBi
BS receiver noise figure,	5 dB
Vehicle antenna height,	1.5 m
Vehicle antenna gain,	3 dBi
Vehicle receiver noise figure,	9 dB
Reliability for V2I p_0,	0.01
Data size of an application of V-UE v, D_v	0.42 MB
Processing density of the application of V-UE v, Λ_v	297.62 cycles/bit
Total computation capability of each V-FN, F^{fog}	2 G cycles/s
Cloud processing capability for V-UE v, f_v^{cloud}	5 G cycles/s
Wired link rate between the BS and the cloud, $r_{R,C}$	15 Mb/s
The average vehicular velocity of each V-UE s_v	60 km/h
The average vehicular velocity of each V-FN s_k	60 km/h
The number of V-FN	3
The number of fireworks, I	2
The number of total explosion sparks, M	4
The number of mutation sparks, $\hat{m}$	1
The maximum number of iterations of FA, L	100

Fig. 5.2 plots the maximum service delay $T_{service}$ versus the iterations of the outer loop in Algorithm 5.1. We can see that Algorithm 5.1 converges after the third iteration.

Fig. 5.3 shows the maximum service delay versus the number of V-UEs, where ‘FORA’ denotes our proposed Algorithm 5.1, ‘Local-Processing’, ‘Fog-Processing’, and ‘Cloud-Processing’ denote the cases where all applications of the V-UEs are processed locally, by a V-FN, by the cloud servers, respectively, and ‘Random offloading decisions’ denotes the case where each V-UE’s application has the equal probability of being processed by itself locally, a V-FN, or a cloud server. We can see that the maximum service delay increases with the number of V-UEs in all the considered cases, among which FORA performs the best for any given number of V-UEs due to the joint optimisation of offloading decisions for all the V-UEs while considering their mobility, queueing delays, resources allocation at a V-FN and

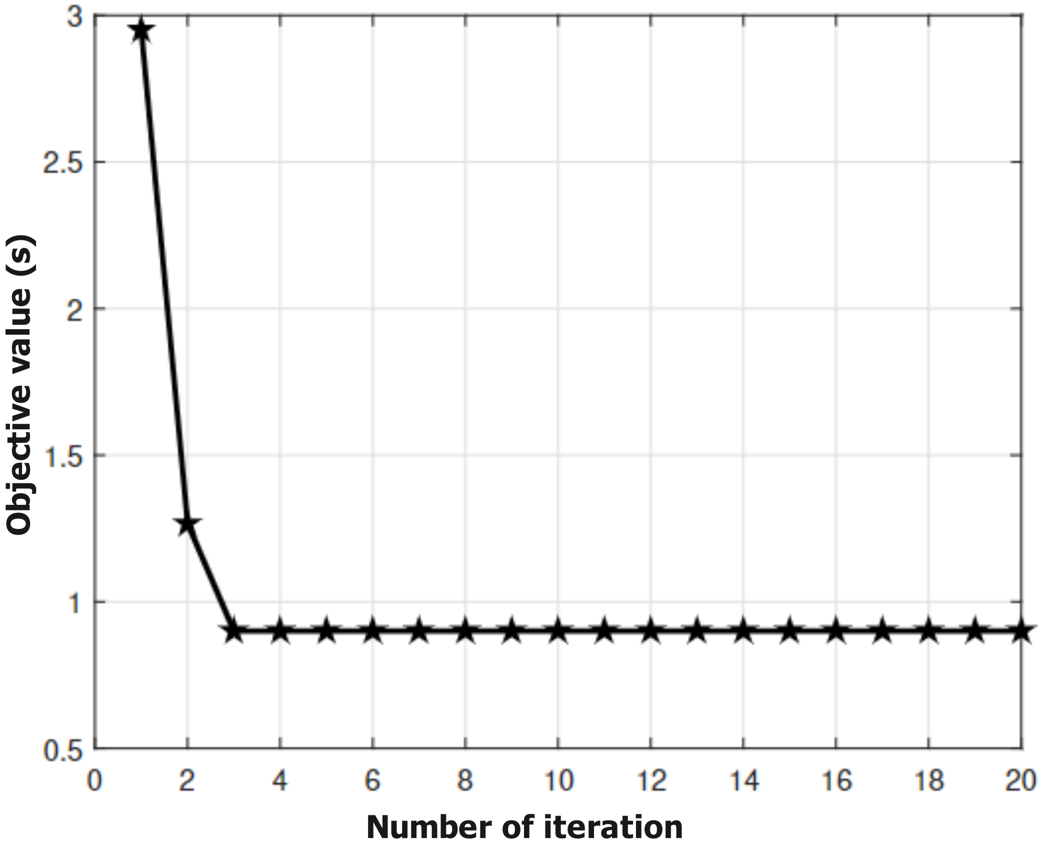

Fig. 5.2 Convergence of Algorithm 5.1, where $V = 5$.

cloud centre. When the number of V-UEs is lager than 3, fog-processing leads to a higher maximum service delay than FORA case, cloud-processing and random offloading decisions cases. This is due to the long queueing and processing delays caused by many applications sharing the limited computation capacity of V-FNs.

Fig. 5.4 and Fig. 5.5 show how the individual application's data size D_v and processing density Λ_v affect the maximum service delay, respectively, where $V = 6$. We can see that the larger the data size or the higher processing density of each application, the higher the maximum service delay in each considered case. FORA always performs the best among all the considered cases. Moreover, local-processing is most significantly affected by a large data size or a high processing density of an application, due to the limited computation capability at each V-UE.

Fig. 5.6 shows the channel state affects the total maximum service delay, especially the wired transmission rate $r_{R,C}$ between the BS and the cloud centre. As the wired channel state has no effect on local and fog processing, the total maximum service delay remains unchanged. However, as the channel state becomes better, and data transmission becomes

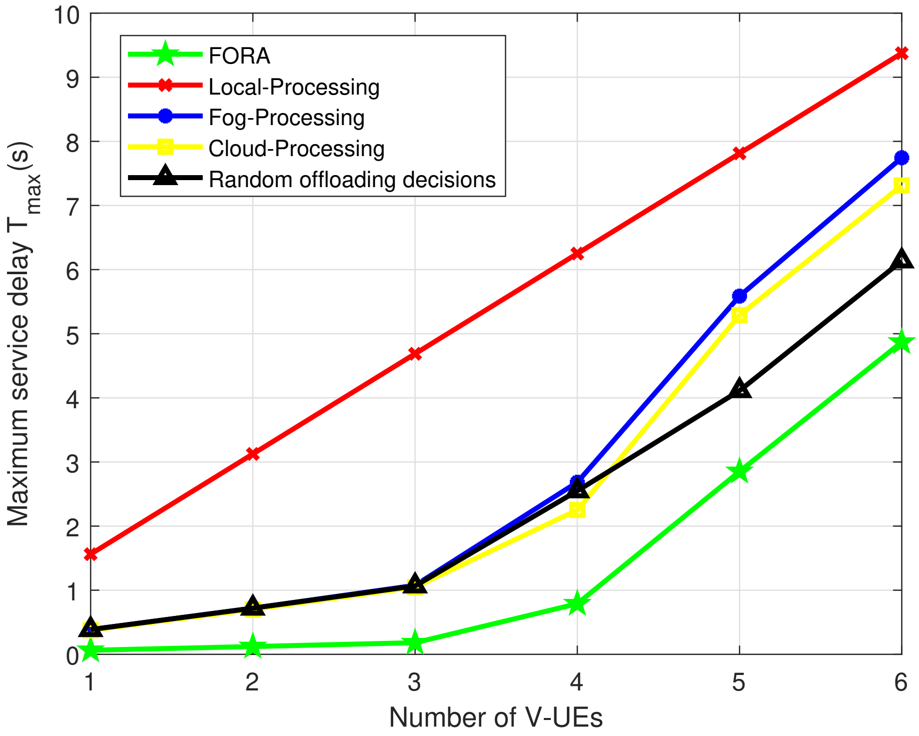

Fig. 5.3 Maximum service delay versus the number of V-UEs.

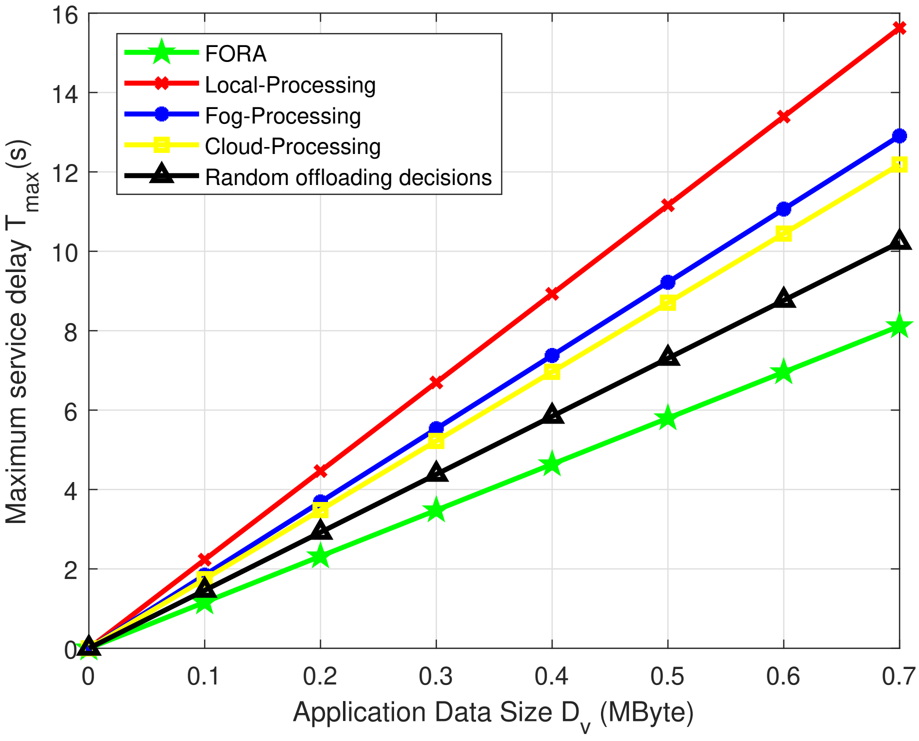

Fig. 5.4 Maximum service delay versus the data size of the application.

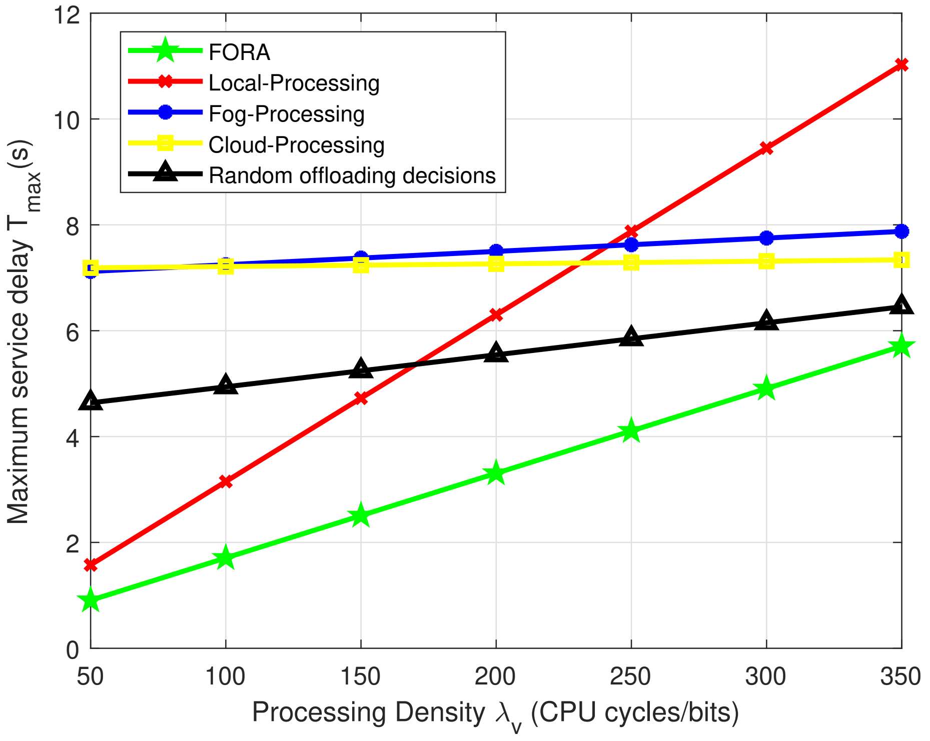

Fig. 5.5 Maximum service delay versus the processing density.

more time efficient, the total maximum service delay becomes smaller and smaller for cloud-processing, random-processing and optimisation case. In addition to this, we can also find that FORA always performs the best among all cases.

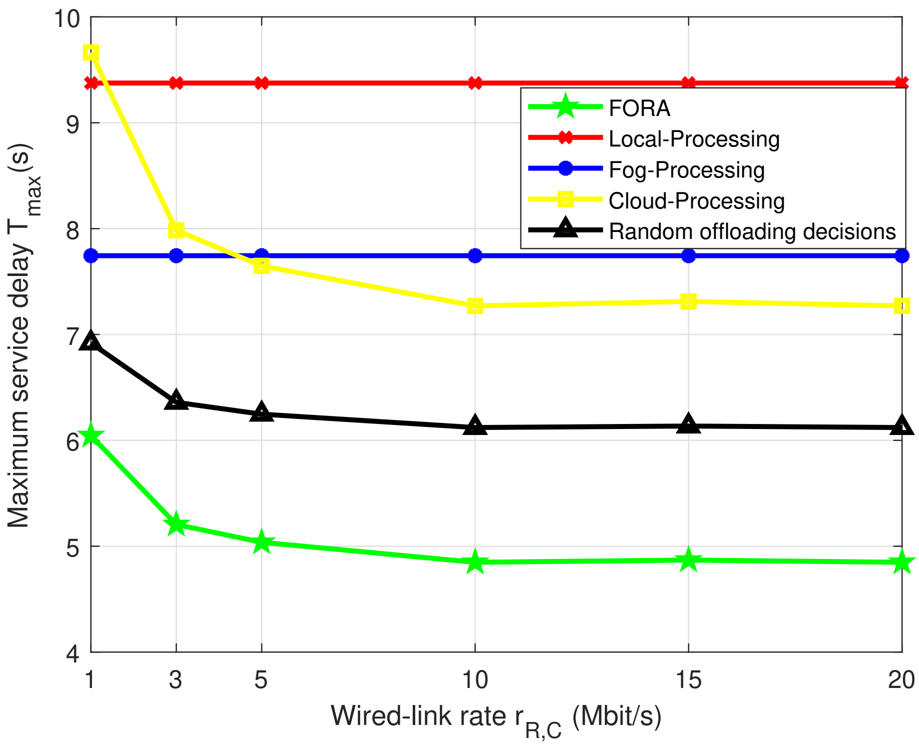

Fig. 5.6 Maximum service delay versus the wired-link rate between BS and cloud server (Mbit/s).

Fig. 5.7 and Fig. 5.8 plot the impact of the fog processing capability F^{fog} and cloud processing capability f_v^{cloud} on the total maximum service delay. It is easy to find that, these two parameters have no effect on local-processing, and the total service delay for local-processing always keeps still. However, when either of the two parameters increases, the total maximum service delays of the rest algorithms decrease. Moreover, FORA still performs the best in delay reduction and far outdistances other algorithms.

Fig. 5.9 shows the how the local processing capability f_v^{local} affect the maximum total service delay, where $V = 6$. When the value of f_v^{local} increases, fog-processing and cloud-processing still keep the same values. It means that the local processing capability has no effect on fog or cloud processing. However, the maximum total service delay decrease quickly with the increase of f_v^{local} for local-processing, random-processing and FORA algorithms. The local-processing algorithm decays more quickly than other algorithms (including random-processing and FORA algorithms). It is because when the local processing capability of

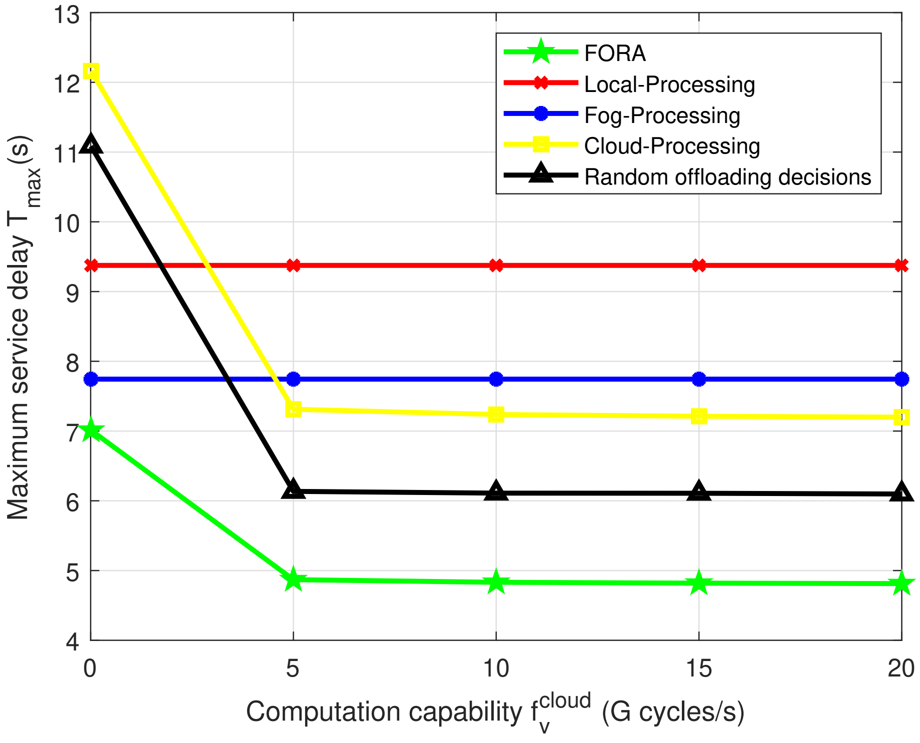

Fig. 5.7 Maximum total service delay versus the computation capability f_v^{cloud}.

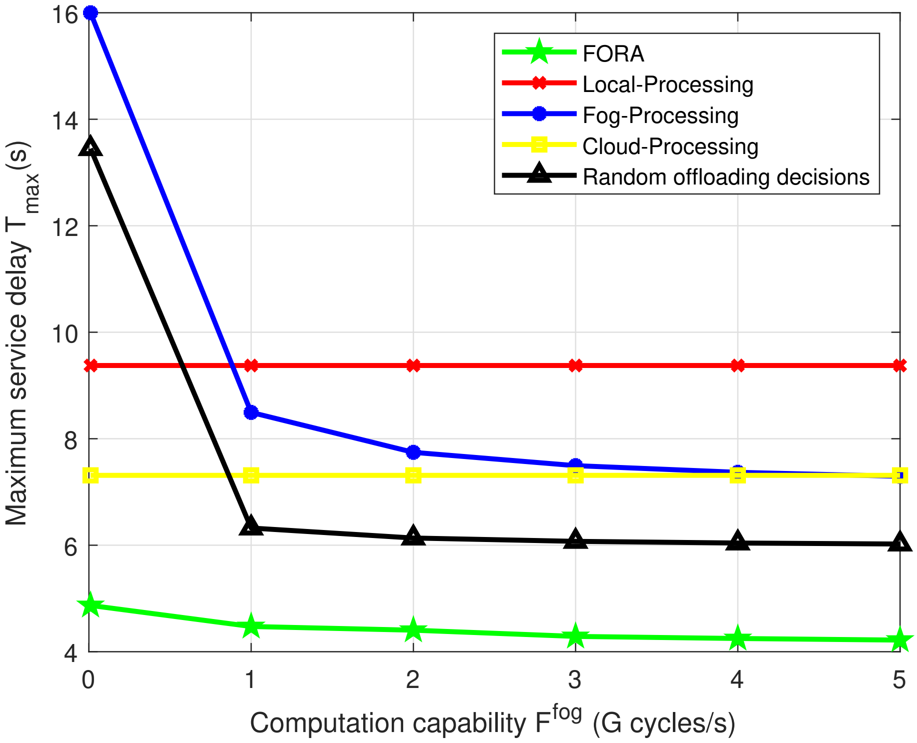

Fig. 5.8 Maximum total service delay versus the computation capability F^{fog}.

each V-UE is large enough, the application of each V-UE can be processed locally with good performance, and the application does not need to be offloaded to a remote server. Compared between FROA and local-processing algorithms, when f_v^{local} is very small (i.e., $f_v^{local} = 50 * 10^6$ cycles/s), FORA performs very well. However, FORA is only a little better than local-processing algorithm when $f_v^{local} = 400 * 10^6$ (cycles/s). It indicates that FORA still performs well in this case.

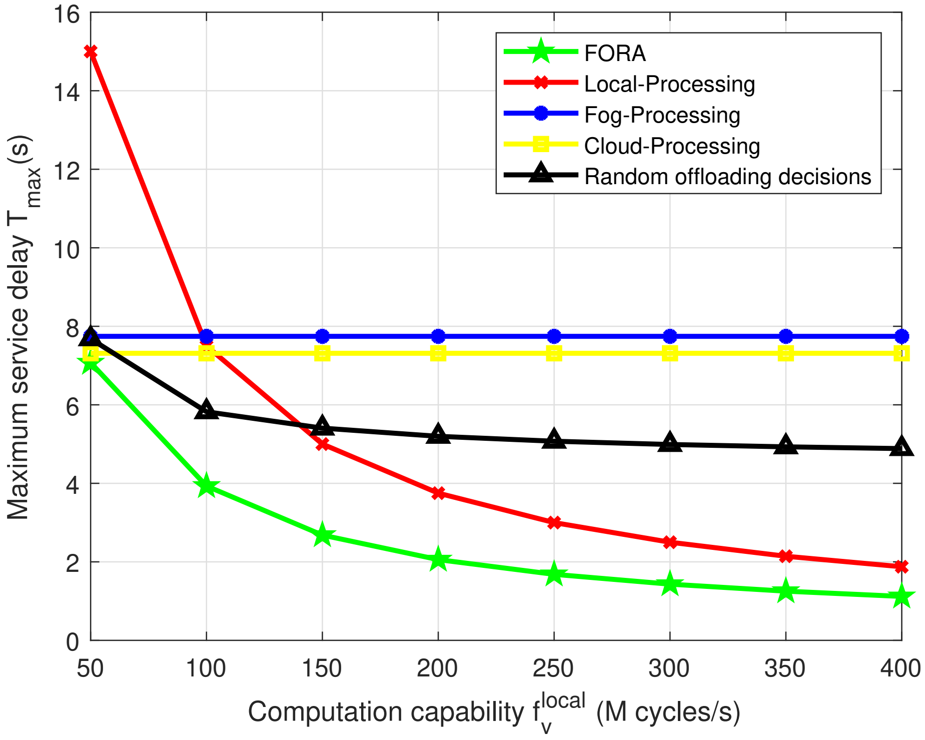

Fig. 5.9 Maximum service delay versus the computation capability f_v^{local}.

In Fig. 5.10 and Fig. 5.11, we evaluate the impact of the velocity of V-UE and V-FN on application processing success rate, including FORA, fog-processing and random-processing algorithms. As velocities of V-UEs and V-FNs barely have an effect on local-processing and cloud-processing algorithms, the application processing success rates of these two cases have been ignored with the comparison. We can find that when the velocity of V-UE increases (where the velocity of V-FN is $s_k = 60km/h$) or when the velocity of V-FN decreases (where the velocity of V-UE is $s_v = 60km/h$), the application processing success rate increases for fog-processing algorithm. This is because when the V-FN maintains a relatively high vehicle speed, it stays within the communication range of the BS for a relatively short period of time. As a result, if the V-UE vehicle speed is under a very small value (i.e., $s_v = 20km/h$), the application offloaded by the V-UE to the corresponding V-FN cannot be completely processed, and the corresponding V-FN has already driven out of communication range. On the contrary, when the V-FN maintains a relatively low value of its velocity (i.e. $s_k = 20km/h$), it will stay in the communication range of the BS for a relatively enough time, which the V-FN will have sufficient time to process the corresponding offloaded application by a V-UE, and the application processing success rate is increased. Moreover, comparing with other algorithms, FORA still performs the best in application processing success rate.

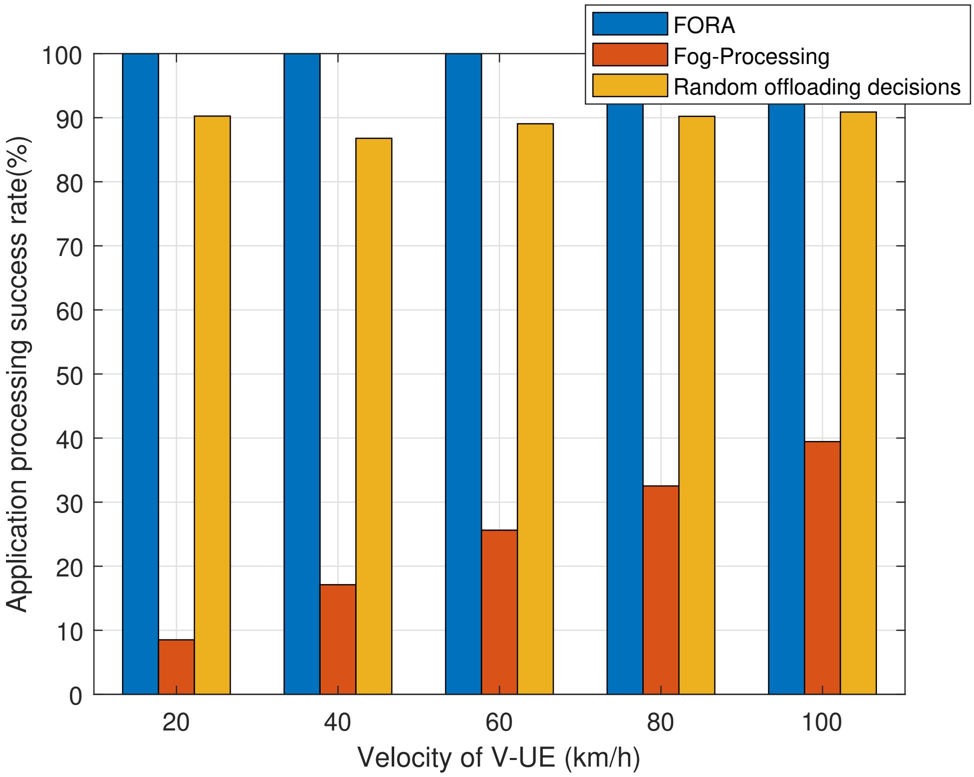

Fig. 5.10 Application processing success rate versus the velocity of each V-UE.

Fig. 5.12 indicates the three different delay metrics T_{max}, T_{min} and $T_{average}$ as influenced by the parameter (i.e., the number of fireworks I) of the fireworks algorithm. From Fig 5.12, we can see that as the number of fireworks increases, the three delays decrease accordingly. This is because each firework or spark is an offloading decision. Therefore, the increase in the number of fireworks or sparks results in more different resource allocation schemes accordingly, and it also increases the searching capabilities and opportunities to find the optimal solution. However, the corresponding computation complexity will also be increased. The Fig 5.12 shows that when the number of fireworks is set as $I = 2$, the complexity of the computation and the searching capability to find the optimal solution can be relatively balanced.

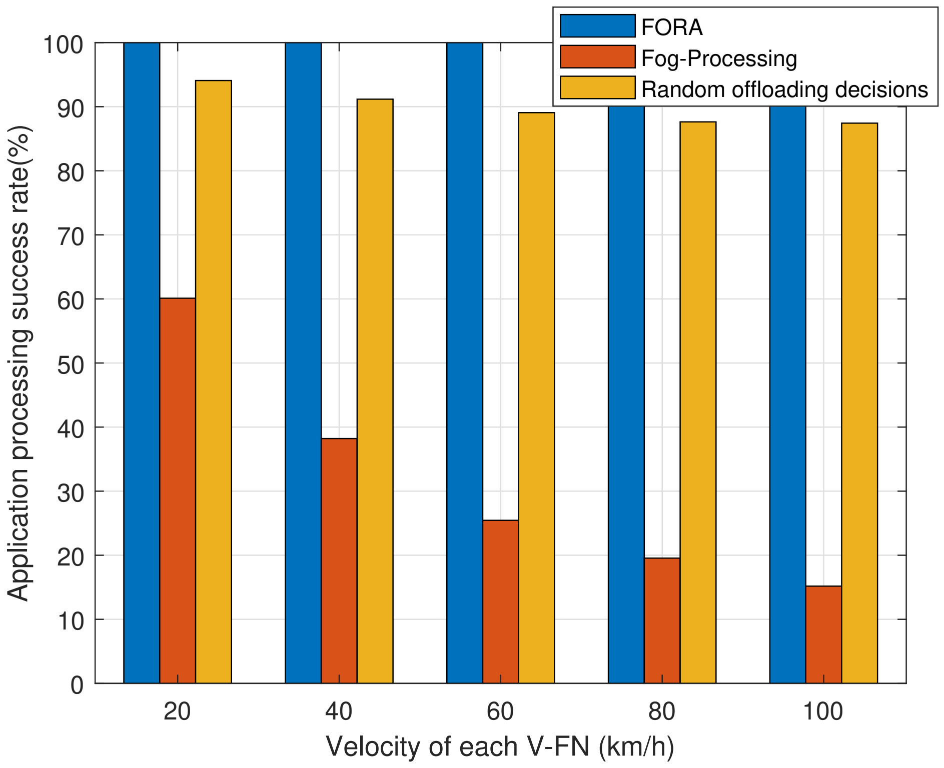

Fig. 5.11 Task processing success rate versus the velocity of each V-FN.

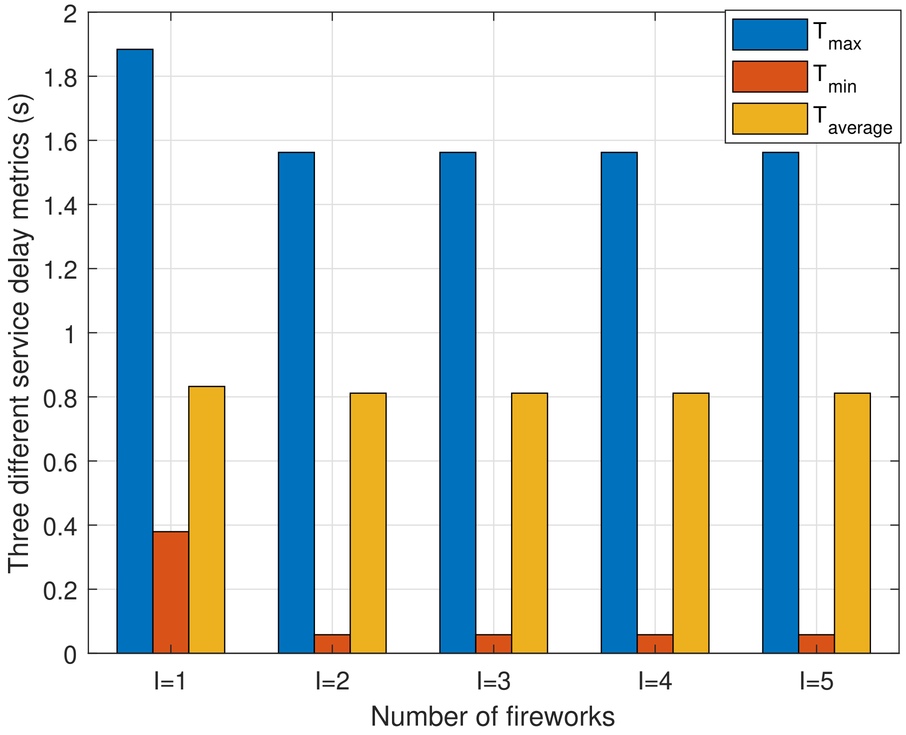

Fig. 5.12 Comparison under different number of fireworks *I*.

5.7 Conclusion

In this Chapter, we have proposed a mobility and queueing-based algorithm to optimise computation offloading decision, in conjunction with a bisection method-based V-FN computation resource allocation algorithm, and a cluster-based communication resource allocation algorithm to minimise the maximum service delay of all V-UEs in an IoV system, where each V-UE may offload its application to a V-FN or cloud computing server or process it locally. The simulation results demonstrate that the proposed algorithms achieve a much lower maximum service delay than local-processing, fog-processing, cloud-processing, and random-processing.

Chapter 6

Conclusions and Future Works

How we can provide the optimal resource allocation and offloading decisions for V2X communications in different scenarios remains as an unanswered question. This thesis studied two promising schemes in different scenarios: 1) an optimised resource allocation for V2X communications in a VSN; 2) an optimised offloading decisions in conjunction with resource allocation optimisation for V2X communications in a mixed cloud/VFC system.

This thesis has shown that optimised resource allocation for V2X communications can significantly improve the performance of V2X communications. However, such optimisation schemes are difficult to be achieved in many cases. For example, the algorithm for the joint power control, RB allocation and/or computation resource allocation optimisation problem is a MINLP problem that lacks a mathematical solution with low computational complexity and requires careful design.

In Chapter three, it is presented that the social attributes/interests of V-UEs can affect the performance of a conventional V2X communications system. To overcome this trade-off, we can define a V2X communications system that includes both the physical and social domains. We develop a two-step optimisation scheme to maximise the sum capacity of V2I links and guarantee the reliability of all V2V links. The results demonstrate that optimal resource allocation can be obtained by using a matching-based algorithm.

In addition, it is possible to successfully combine V2X communications with other technologies to enable low-latency applications to be offloaded to the cloud/fog servers in vehicle networks and their corresponding resource allocation. The mixed cloud/fog computing system proposed in Chapter four enables V-UEs to offload their intensive applications to a fog/cloud server for processing. The results show that the mixed cloud/fog computing system can significantly reduce the total service latency and optimise offloading decisions for all V-UEs. In Chapter five, we consider the joint allocation of communication and computation resources, along with the optimisation of offloading decisions for all V-UEs. We show that offloading strategies can effectively improve the efficiency of resource allocation and increase the throughput of V2X communications.

In this chapter, we first summarise the main findings of this thesis. Then, we list future research directions for V2X communications that are relevant to the scope of this thesis.

6.1 Main Findings of the Thesis

- **Optimisation of resource allocation involving social attributes is effective in improving throughput performance:** As described in Section 2.2, most communication demands are initiated by human users, whose social activities and attributes influence their communication demands and requirements. Based on this definition, in Section 3.2 we designed a vehicular social network system including both the social and physical domains, where each V2V link is not only grouped into different social communities according to their social attributes, but also with minimal internal interference guaranteed among all clusters. In addition, we develop a matching-based social-aware radio resource allocation optimisation scheme in Section 3.5 which shows better performance than the non-social-aware scheme.

- **Mixed cloud/fog computing systems are effective in reducing service latency for V2X communications:** Cloud computing and fog computing technologies have many advantages in terms of improving network performance, such as providing additional computing capabilities and sufficient transmission power. However, there is doubt whether it is practical and efficient to mix cloud and fog computing in V2X communications. In this thesis, we design a mixed cloud/VFC offloading system in Sections 4.2 and 5.2. In Sections 4.3 and 5.4, the mixed cloud/fog computing system enables significantly reduce the maximum service latency for processing V-UEs applications by applying different joint optimisation schemes for V2X communication, including resource allocation optimisation and offloading decisions optimisation.

- **Fair and efficient radio resource allocation can be achieved by Matching-based frameworks:** The theory of matching algorithms has been developed for many years since it was proposed for solving economic problems. Based on the similarities between the resource allocation problems in wireless communications and the investment problems in economics, we develop a matching-based scheme to achieve efficient society-aware radio resource allocation for V2X communications underlaying cellular networks in Chapter three. We note that based on the consideration of auxiliary information (e.g., social attributes and mobility of V-UEs), we modify the classical matching algorithm with a clustering-based approach in Section 3.4, which may help us to transform the NP-hard problem into a solution with low computational complexity.

- **Fireworks algorithm is efficient to solve MINLP problem in V2X communications:** In both Chapter four and Chapter five, it was possible to utilise improved fireworks algorithms to solve MINLP problems that are joint optimisation problems, including offloading decisions optimisation, radio resource allocation and/or computation resource allocation for V2X communications. In contrast to traditional approaches to solving NP-hard MILNP problems, as mentioned in Section 2.4, fireworks algorithm frameworks solve NP-hard MINLP problems with much lower computational complexity. Note the modified conventional fireworks algorithm in Sections 4.3 and 5.4 to improve its search efficiency, and our proposed algorithms converge less than five iterations.

6.2 Future Works

In this thesis, one focus was the optimised offloading schemes and resource allocation for V2X communications in different scenarios and applying LTE for V2X communications. Part of the future research directions related to the topics of the thesis were able to be summarised.

As evaluated in Chapter three, there remain many limitations in our research on social-aware resource allocation for V2X communications.

- **Radio resource allocation in 5G VSNs:** As mentioned in Chapters one and two, several advanced technologies have been proposed for next generation cellular communications [111]. For example, to study how Massive MIMO, mmWave and other new technologies affect V2X communications in VSNs and thus propose relevant optimised radio resource allocation schemes.

- **Real-time messages dissemination in VSNs:** In this piece only data transmission under traditional network standards were considered such as Dedicated Short Range Communication (DSRC) and 3G/4G cellular networks. This leads to more serious problems such as network congestion due to broadcast storms and so on [112]. To cope with these problems, an advanced real-time dissemination algorithm based on 5G/6G VSNs should be investigated. For example, a recursive evolutionary algorithm [113] is proposed to evaluate the performance of network parameters in terms of delivery rate and message overhead for different vehicle densities and speed values based on realistic vehicle movement trajectories. However, this scheme is complex and lack complexity analysis while requiring further research to develop different road network typologies.

- **Trustworthy transmission:** In our current work, we assume that all V-UEs and V2X links are reliable, which is a common assumption in existing works. As mentioned in Chapter one, all services may be provided by different third parties, and therefore trust, security, and privacy of communications between V2X V-UEs should be considered in the future.

Chapters four and five focused on the offloading schemes in conjunction with resource allocation for V2X communications in a cloud/fog mixed VFC system. We list the limitations as follows.

- **Machine learning for V2X resource allocation:** One of the possible future studies is to reduce the complexity of the proposed algorithm. It has been observed that the computational complexity of the proposed algorithm is approaching a theoretical upper limit due to the increasing number of input parameters and/or redundant iterations. Machine learning frameworks, such as reinforcement learning and Q-learning are promising solutions to improve convergence speed and learning efficiency by training the system with raw inputs (i.e., application size, etc.) and outputs (offloading decisions and resource allocation).

- **Leverage communication-computation resources:** Another improvement is developing a comprehensive offloading scheme which can be satisfied with different requirements. For example, an offloading scheme can be divided into single-application offloading and multi-application offloading based on the total number of offloading applications. Similarly, it can also be based on static-dynamic, priority and other aspects.

- **V2X communications channel modelling:** In our current works, it was considered an ideal transmission channel with strong assumptions (e.g., the channel fading is assumed as Rayleigh fading). However, in V2X communications, most transmitters and receivers have high mobility characteristics, different from traditional mobile cellular channels, which may lead to dealing with more complex Doppler shifts and severe multipath fading. To remove the assumptions in our works, we should consider more practical channel models in the future. Some related works and potential research directions are listed in [114].

Concludingly, the combination of V2X communications with other state-of-the-art technologies is promising and this has great potential in several emerging research directions, such as ITS, IoV, etc.

- **Big data in V2X:** Big Data is a hot topic for cost reduction and decision making [115, 116], and we can combine this with V2X communications to enable analysing information and making decisions immediately, while reducing costs and conducting business in more efficient ways.
- **V2X communications with SDN:** Combining V2X communications with SDN will provide V2X communications with the capability to support different business models and enable efficient network configuration [117], which could improve system performance.
- **Business model in V2X communications:** Business model, such as pricing strategies, is a core point for a technology to achieve commercial success. Therefore, more complex incentive algorithms involving requirements such as V2X user charging are required in real-life communication systems.

There is great hope that continued research in V2X communications will improve its usability and performance. advancement in V2X communications in terms of access design, resource allocation schemes, offloading schemes, etc. can inspire researchers in wireless communications and lead to new products and technologies that will benefit the entire human society.

References

[1] Y. Li, M. Sheng, X. Wang, Y. Zhang, and J. Wen, “Max–min energy-efficient power allocation in interference-limited wireless networks,” *IEEE Transactions on Vehicular Technology*, vol. 64, no. 9, pp. 4321–4326, 2014.

[2] P. Phunchongharn, E. Hossain, and D. I. Kim, “Resource allocation for device-to-device communications underlaying lte-advanced networks,” *IEEE wireless communications*, vol. 20, no. 4, pp. 91–100, 2013.

[3] L. Zhao, J. Wang, J. Liu, and N. Kato, “Optimal edge resource allocation in iot-based smart cities,” *IEEE Network*, vol. 33, no. 2, pp. 30–35, 2019.

[4] Y. Yang, X. Luo, X. Chu, and M. Zhou, “Fog computing architecture and technologies,” in *Fog-Enabled Intelligent IoT Systems*. Springer, 2020, pp. 39–60.

[5] M. I. Ashraf, M. Bennis, C. Perfecto, and W. Saad, “Dynamic proximity-aware resource allocation in vehicle-to-vehicle (v2v) communications,” in *2016 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2016, pp. 1–6.

[6] W. Sun, E. G. Ström, F. Brännström, K. C. Sou, and Y. Sui, “Radio resource management for d2d-based v2v communication,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 6636–6650, 2015.

[7] Q. Wei, W. Sun, B. Bai, L. Wang, E. G. Ström, and M. Song, “Resource allocation for v2x communications: A local search based 3d matching approach,” in *2017 IEEE International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–6.

[8] Y. Zhao, Y. Li, Y. Cao, T. Jiang, and N. Ge, “Social-aware resource allocation for device-to-device communications underlaying cellular networks,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 12, pp. 6621–6634, 2015.

[9] Z. Feng, Z. Feng, and T. A. Gulliver, “Effective small social community aware d2d resource allocation underlaying cellular networks,” *IEEE Wireless Communications Letters*, vol. 6, no. 6, pp. 822–825, 2017.

[10] N.-S. Vo, T.-M. Phan, M.-P. Bui, X.-K. Dang, N. T. Viet, and C. Yin, “Social-aware spectrum sharing and caching helper selection strategy optimized multicast video streaming in dense d2d 5g networks,” *IEEE Systems Journal*, 2020.

[11] X. Zhang, Y. Li, Y. Zhang, J. Zhang, H. Li, S. Wang, and D. Wang, “Information caching strategy for cyber social computing based wireless networks,” *IEEE Transactions on Emerging Topics in Computing*, vol. 5, no. 3, pp. 391–402, 2017.

[12] Y. Wu, D. Wu, L. Yang, X. Shi, L. Ao, and Q. Fu, “Matching-coalition based cluster formation for d2d multicast content sharing,” *IEEE Access*, vol. 7, pp. 73 913–73 928, 2019.

[13] U. Saleem, Y. Liu, S. Jangsher, X. Tao, and Y. Li, “Latency minimization for d2d-enabled partial computation offloading in mobile edge computing,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4472–4486, 2020.

[14] Z. Ning, X. Wang, J. J. Rodrigues, and F. Xia, “Joint computation offloading, power allocation, and channel assignment for 5g-enabled traffic management systems,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 3058–3067, 2019.

[15] C. Zhu, J. Tao, G. Pastor, Y. Xiao, Y. Ji, Q. Zhou, Y. Li, and A. Ylä-Jääski, “Folo: Latency and quality optimized task allocation in vehicular fog computing,” *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4150–4161, 2018.

[16] Q. Liu, Z. Su, and Y. Hui, “Computation offloading scheme to improve qoe in vehicular networks with mobile edge computing,” in *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*. IEEE, 2018, pp. 1–5.

[17] H. Feng, S. Guo, L. Yang, and Y. Yang, “Collaborative data caching and computation offloading for multi-service mobile edge computing,” *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, pp. 9408–9422, 2021.

[18] X. Wang, Z. Ning, and L. Wang, “Offloading in internet of vehicles: A fog-enabled real-time traffic management system,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4568–4578, 2018.

[19] J. Du, L. Zhao, X. Chu, F. R. Yu, J. Feng, and I. Chih-Lin, “Enabling low-latency applications in lte-a based mixed fog/cloud computing systems,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1757–1771, 2018.

[20] P. Liu, J. Li, and Z. Sun, “Matching-based task offloading for vehicular edge computing,” *IEEE Access*, vol. 7, pp. 27 628–27 640, 2019.

[21] J. Wang, D. Feng, S. Zhang, J. Tang, and T. Q. Quek, “Computation offloading for mobile edge computing enabled vehicular networks,” *IEEE Access*, vol. 7, pp. 62 624–62 632, 2019.

[22] X. Xu, Y. Xue, X. Li, L. Qi, and S. Wan, “A computation offloading method for edge computing with vehicle-to-everything,” *IEEE Access*, vol. 7, pp. 131 068–131 077, 2019.

[23] C. Tang, X. Wei, C. Zhu, Y. Wang, and W. Jia, “Mobile vehicles as fog nodes for latency optimization in smart cities,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 9, pp. 9364–9375, 2020.

[24] J. Zhao, Q. Li, Y. Gong, and K. Zhang, “Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7944–7956, 2019.

[25] Z. Zhou, P. Liu, J. Feng, Y. Zhang, S. Mumtaz, and J. Rodriguez, “Computation resource allocation and task assignment optimization in vehicular fog computing: A contract-matching approach,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3113–3125, 2019.

[26] M. Adhikari, M. Mukherjee, and S. N. Srirama, “Dpto: A deadline and priority-aware task offloading in fog computing framework leveraging multilevel feedback queueing,” *IEEE IoTJ*, vol. 7, pp. 5773–5782, 2019.

[27] J. Du, L. Zhao, J. Feng, and X. Chu, “Computation offloading and resource allocation in mixed fog/cloud computing systems with min-max fairness guarantee,” *IEEE TCOM*, vol. 66, no. 4, pp. 1594–1608, 2017.

[28] X. Wei, C. Tang, J. Fan, and S. Subramaniam, “Joint optimization of energy consumption and delay in cloud-to-thing continuum,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2325–2337, 2019.

[29] X. Lyu, H. Tian, C. Sengul, and P. Zhang, “Multiuser joint task offloading and resource optimization in proximate clouds,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3435–3447, April. 2017.

[30] F. Esposito, “Architectures for social vehicular network programmability,” in *Vehicular Social Networks*. CRC Press, 2017, pp. 79–99.

[31] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, “Vehicular social networks: Enabling smart mobility,” *IEEE Communications Magazine*, vol. 55, no. 5, pp. 16–55, 2017.

[32] A. Rahim, X. Kong, F. Xia, Z. Ning, N. Ullah, J. Wang, and S. K. Das, “Vehicular social networks: A survey,” *Pervasive and Mobile Computing*, vol. 43, pp. 96–113, 2018.

[33] N. Lu, T. H. Luan, M. Wang, X. Shen, and F. Bai, “Bounds of asymptotic performance limits of social-proximity vehicular networks,” *IEEE/ACM transactions on networking*, vol. 22, no. 3, pp. 812–825, 2013.

[34] ——, “Capacity and delay analysis for social-proximity urban vehicular networks,” *2012 Proceedings IEEE INFOCOM*, pp. 1476–1484, 2012.

[35] J. Harri, F. Filali, and C. Bonnet, “Mobility models for vehicular ad hoc networks: a survey and taxonomy,” *IEEE Communications Surveys & Tutorials*, vol. 11, no. 4, pp. 19–41, 2009.

[36] Y. Zhong, K. Hua, P. Li, D. Deng, X. Liu, and Y. Chen, “Dynamic periodic location encounter network analysis for vehicular social networks,” *IEEE Transactions on Vehicular Technology*, 2021.

[37] A. T. Akabane, R. Immich, R. W. Pazzi, E. R. M. Madeira, and L. A. Villas, “Exploiting vehicular social networks and dynamic clustering to enhance urban mobility management,” *Sensors*, vol. 19, no. 16, p. 3558, 2019.

[38] G. Aloi, M. Di Felice, V. Loscrì, P. Pace, and G. Ruggeri, "Spontaneous smartphone networks as a user-centric solution for the future internet," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 26–33, 2014.

[39] L. Feng, P. Zhao, F. Zhou, M. Yin, P. Yu, W. Li, and X. Qiu, "Resource allocation for 5g d2d multicast content sharing in social-aware cellular networks," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 112–118, 2018.

[40] S. S. Shah, M. Ali, A. W. Malik, M. A. Khan, and S. D. Ravana, "vfog: A vehicle-assisted computing framework for delay-sensitive applications in smart cities," *IEEE Access*, vol. 7, pp. 34 900–34 909, 2019.

[41] "Inrix 2020 global traffic scorecard," March 2020, [online] https://www.boston.com/wp-content/uploads/2021/03/2020_INRIX_Scorecard_-6046893ec91f4.pdf.

[42] "Frontier report on cellular vehicle networking (c-v2x) technology and industry development trends," November 2020, [online] https://www.china-cic.cn/upload/202012/09/de38bbae1ea14b70b512606ecd59671c.pdf.

[43] "3gpp rp-170798. new wid on 3gpp v2x phase 2," March 2017, [online] https://portal.3gpp.org/ngppapp/TdocList.aspx?meetingId=31939.

[44] "3gpp rp-181429. new sid: Study on nr v2x," June 2018, [online] https://portal.3gpp.org/ngppapp/CreateTDoc.aspx?mode=view&contributionUid=RP-181429.

[45] "The case for cellular v2x for safety and cooperative driving," November 2016, [online] https://5gaa.org/wp-content/uploads/2017/10/5GAA-whitepaper-23-Nov-2016.pdf.

[46] "5g americas: V2x cellular solutions," October 2016, [online] https://www.5gamericas.org/v2x-cellular-solutions/.

[47] "White paper: cellular v2x communications towards 5g," March 2018, [online] https://www.5gamericas.org/wp-content/uploads/2019/07/2018_5G_Americas_White_Paper_Cellular_V2X_Communications_Towards_5G__Final_for_Distribution.pdf.

[48] "Liaison statement on technology evaluation of lte-v2x and dsrc," October 2017, [online] https://www.ngmn.org/wp-content/uploads/Publications/2017/171011_NGMN_LS_on_Technology_Evaluation_of_LTE-V2X_and_DSRC_to_5GAA.pdf.

[49] S. Chen, J. Hu, Y. Shi, L. Zhao, and W. Li, "A vision of c-v2x: technologies, field testing, and challenges with chinese development," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 3872–3881, 2020.

[50] "Service requirements for v2x services," July 2018, [online] https://www.etsi.org/deliver/etsi_ts/122100_122199/122185/14.04.00_60/ts_122185v140400p.pdf.

[51] M. Harounabadi, D. M. Soleymani, S. Bhadauria, M. Leyh, and E. Roth-Mandutz, "V2x in 3gpp standardization: Nr sidelink in release-16 and beyond," *IEEE Communications Standards Magazine*, vol. 5, no. 1, pp. 12–21, 2021.

[52] "Study on lte-based v2x services," June 2016, [online] https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2934.

[53] M. Wang, J. Zhang, B. Ren, W. Yang, J. Zou, M. Hua, and X. You, “The evolution of lte physical layer control channels,” *IEEE Communications Surveys Tutorials*, vol. 18, no. 2, pp. 1336–1354, 2016.

[54] J. Zhang, L.-L. Yang, L. Hanzo, and H. Gharavi, “Advances in cooperative single-carrier fdma communications: Beyond lte-advanced,” *IEEE Communications Surveys Tutorials*, vol. 17, no. 2, pp. 730–756, 2015.

[55] S. Chen, J. Hu, Y. Shi, and L. Zhao, “Lte-v: A td-lte-based v2x solution for future vehicular network,” *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 997–1005, 2016.

[56] S. Chen, J. Hu, Y. Shi, Y. Peng, J. Fang, R. Zhao, and L. Zhao, “Vehicle-to-everything (v2x) services supported by lte-based systems and 5g,” *IEEE Communications Standards Magazine*, vol. 1, no. 2, pp. 70–76, 2017.

[57] A. R. Swanson, “Somebody grab the wheel: State autonomous vehicle legislation and the road to a national regime,” *Marq. L. Rev.*, vol. 97, p. 1085, 2014.

[58] S. Aral and D. Walker, “Identifying influential and susceptible members of social networks,” *Science*, vol. 337, no. 6092, pp. 337–341, 2012.

[59] B. Hu and X. Chu, “Social-aware resource allocation for vehicle-to-everything communications underlaying cellular networks,” in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, April 2021, pp. 1–6.

[60] L. A. Maglaras and D. Katsaros, “Social clustering of vehicles based on semi-markov processes,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 318–332, 2015.

[61] Y. Li, T. Wu, P. Hui, D. Jin, and S. Chen, “Social-aware d2d communications: Qualitative insights and quantitative analysis,” *IEEE Communications Magazine*, vol. 52, no. 6, pp. 150–158, 2014.

[62] M. H. Eiza and Q. Shi, “Social evolving graph-based connectivity model for vehicular social networks,” in *Vehicular Social Networks*. CRC Press, 2017, pp. 41–57.

[63] N. Kayastha, D. Niyato, P. Wang, and E. Hossain, “Applications, architectures, and protocol design issues for mobile social networks: A survey,” *Proceedings of the IEEE*, vol. 99, no. 12, pp. 2130–2158, 2011.

[64] H. R. Abdulqadir, S. R. Zeebaree, H. M. Shukur, M. M. Sadeeq, B. W. Salim, A. A. Salih, and S. F. Kak, “A study of moving from cloud computing to fog computing,” *Qubahan Academic Journal*, vol. 1, no. 2, pp. 60–70, 2021.

[65] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the internet of things,” in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, 2012, pp. 13–16.

[66] H. Sabireen and V. Neelanarayanan, “A review on fog computing: Architecture, fog with iot, algorithms and research challenges,” *ICT Express*, vol. 7, no. 2, pp. 162–176, 2021.

[67] C. Yu, B. Lin, P. Guo, W. Zhang, S. Li, and R. He, "Deployment and dimensioning of fog computing-based internet of vehicle infrastructure for autonomous driving," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 149–160, 2018.

[68] K. Kai, W. Cong, and L. Tao, "Fog computing for vehicular ad-hoc networks: paradigms, scenarios, and issues," *the journal of China Universities of Posts and Telecommunications*, vol. 23, no. 2, pp. 56–96, 2016.

[69] M. Li, L. Zhu, and X. Lin, "Efficient and privacy-preserving carpooling using blockchain-assisted vehicular fog computing," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4573–4584, 2018.

[70] A. M. Vegni and V. Loscrí, "A survey on vehicular social networks," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2397–2419, 2015.

[71] X. Li, L. Guo, and Y. E. Zhao, "Tag-based social interest discovery," in *Proceedings of the 17th international conference on World Wide Web*, 2008, pp. 675–684.

[72] T. H. Luan, R. Lu, X. Shen, and F. Bai, "Social on the road: Enabling secure and efficient social networking on highways," *IEEE Wireless Communications*, vol. 22, no. 1, pp. 44–51, 2015.

[73] J. F. Shortle, J. M. Thompson, D. Gross, and C. M. Harris, *Fundamentals of queueing theory*. John Wiley & Sons, 2018, vol. 399.

[74] U. N. Bhat, *An introduction to queueing theory: modeling and analysis in applications*. Springer, 2008, vol. 36.

[75] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *The American Mathematical Monthly*, vol. 69, no. 1, pp. 9–15, 1962.

[76] D. Manlove, *Algorithmics of matching under preferences*. World Scientific, 2013, vol. 2.

[77] D. Gusfield and R. W. Irving, *The stable marriage problem: structure and algorithms*. MIT press, 1989.

[78] D. E. Knuth, "Marriages stables," *Technical report*, 1976.

[79] A. Abdulkadiroğlu and T. Sönmez, "Random serial dictatorship and the core from random endowments in house allocation problems," *Econometrica*, vol. 66, no. 3, pp. 689–701, 1998.

[80] L. Zhou, "On a conjecture by gale about one-sided matching problems," *Journal of Economic Theory*, vol. 52, no. 1, pp. 123–135, 1990.

[81] A. Hylland and R. Zeckhauser, "The efficient allocation of individuals to positions," *Journal of Political economy*, vol. 87, no. 2, pp. 293–314, 1979.

[82] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," in *International conference in swarm intelligence*. Springer, 2010, pp. 355–364.

[83] Y. Tan, *Fireworks algorithm.* Springer, 2015, vol. 10.

[84] D.-J. Chang, A. H. Desoky, M. Ouyang, and E. C. Rouchka, “Compute pairwise manhattan distance and pearson correlation coefficient of data points with gpu,” in *ACIS SNPD*, 2009, pp. 501–506.

[85] W. Qi, Q. Song, X. Wang, L. Guo, and Z. Ning, “Sdn-enabled social-aware clustering in 5g-vanet systems,” *IEEE Access*, vol. 6, pp. 28 213–28 224, 2018.

[86] P. Zhao, L. Feng, P. Yu, W. Li, and X. Qiu, “A social-aware resource allocation for 5g device-to-device multicast communication,” *IEEE Access*, vol. 5, pp. 15 717–15 730, 2017.

[87] Z. Huang, H. Tian, S. Fan, Z. Xing, and X. Zhang, “Social-aware resource allocation for content dissemination networks: An evolutionary game approach,” *IEEE Access*, vol. 5, pp. 9568–9579, 2016.

[88] L. Wang, C. Li, Y. Zhang, and G. Gui, “Game-theoretic social-aware resource allocation for device-to-device communications underlaying cellular network,” *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[89] E. Ahmed, I. Yaqoob, A. Gani, M. Imran, and M. Guizani, “Social-aware resource allocation and optimization for d2d communication,” *IEEE wireless communications*, vol. 24, no. 3, pp. 122–129, 2017.

[90] Y. H. Chan and L. C. Lau, “On linear and semidefinite programming relaxations for hypergraph matching,” *Mathematical programming*, vol. 135, no. 1, pp. 123–148, 2012.

[91] X. Zhang, Y. Li, Y. Zhang, J. Zhang, H. Li, S. Wang, and D. Wang, “Information caching strategy for cyber social computing based wireless networks,” *IEEE Transactions on Emerging Topics in Computing*, vol. 5, no. 3, pp. 391–402, 2017.

[92] C. E. Shannon and W. Weaver, “The mathematical theory of communication [1949]. urbana, il,” 1959.

[93] A. M. Vegni, V. Loscrì, and A. V. Vasilakos, *Vehicular social networks.* CRC Press, 2017.

[94] A. Al-Dulaimi, X. Wang, and I. Chih-Lin, *5G networks: fundamental requirements, enabling technologies, and operations management.* John Wiley & Sons, 2018.

[95] S. Sahni and T. Gonzalez, “P-complete approximation problems,” *Journal of the ACM (JACM)*, vol. 23, no. 3, pp. 555–565, 1976.

[96] B. Aydin, M. Henning, S. Ilya, S. Peter, and S. Christian. Recent advances in graph partitioning. (2013). [Online]. Available: http://arxiv.org/abs/1311.3144

[97] H. W. Kuhn, “The hungarian method for the assignment problem,” *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[98] J. Du, L. Zhao, X. Chu, R. Yu, J. Feng, and C.-L. I, "Enabling low-latency applications in lte-a based mixed fog/cloud computing systems," *IEEE Transactions on Vehicular Technology*, pp. 1757–1771, 2018.

[99] *Technical Specification Group Radio Access Network: Study on LTEBased V2X Services (Rel. 14) 3GPP*, Std. TR 36.885 V2.0.0, Jul. 2016.

[100] G. Da Prato, *Kolmogorov equations for stochastic PDEs*. Birkhäuser, 2012.

[101] T. L. Saaty, "A scaling method for priorities in hierarchical structures," *Journal of mathematical psychology*, vol. 15, no. 3, pp. 234–281, 1977.

[102] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[103] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular fog computing: A viewpoint of vehicles as the infrastructures," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3860–3873, 2016.

[104] C. Wang, C. Liang, F. R. Yu, Q. Chen, and L. Tang, "Computation offloading and resource allocation in wireless cellular networks with mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 16, no. 8, pp. 4924–4938, 2017.

[105] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: Partial computation offloading using dynamic voltage scaling," *IEEE Transactions on Communications*, vol. 64, no. 10, pp. 4268–4282, 2016.

[106] B. Hu and X. Chu, "Social-aware resource allocation for vehicle-to-everything communications underlaying cellular networks," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*. IEEE, April 2021, pp. 1–6.

[107] D. Gross and C. M. Harris, "Simple markovian birth-death queuein models," in *Fundamentals of queueing theory, 3rd ed*, 1998, p. 53–115.

[108] M. Whaiduzzaman, M. Sookhak, A. B. Gani, and R. Buyya, "A survey on vehicular cloud computing," *Journal of Network and Computer Applications*, 2014.

[109] C. Huang, R. Lu, and K. K. R. Choo, "Vehicular fog computing: Architecture, use case, and security and forensic challenges," *IEEE Communications Magazine*, vol. 55, no. 11, pp. 105–111, 2017.

[110] H.-C. Wu, "The karush–kuhn–tucker optimality conditions in an optimization problem with interval-valued objective function," *European Journal of Operational Research*, vol. 176, no. 1, pp. 46–59, 2007.

[111] S. Kim and R. Shrestha, "Internet of vehicles, vehicular social networks, and cybersecurity," in *Automotive Cyber Security*. Springer, 2020, pp. 149–181.

[112] A. Mchergui, T. Moulahi, B. Alaya, and S. Nasri, "A survey and comparative study of qos aware broadcasting techniques in vanet," *Telecommunication Systems*, vol. 66, no. 2, pp. 253–281, 2017.

[113] N. Ullah, X. Kong, L. Lin, M. Alrashoud, A. Tolba, and F. Xia, "Real-time dissemination of emergency warning messages in 5g enabled selfish vehicular social networks," *Computer Networks*, vol. 182, p. 107482, 2020.

[114] Z. Huang, X. Zhang, and X. Cheng, "Non-geometrical stochastic model for non-stationary wideband vehicular communication channels," *IET Communications*, vol. 14, no. 1, pp. 54–62, 2020.

[115] F. Mehdipour, B. Javadi, A. Mahanti, G. Ramirez-Prado, and E. Principles, "Fog computing realization for big data analytics," *Fog and edge computing: Principles and paradigms*, pp. 259–290, 2019.

[116] R. K. Naha, S. Garg, and A. Chan, "Fog computing architecture: Survey and challenges," *arXiv preprint arXiv:1811.09047*, 2018.

[117] K. Benzekki, A. El Fergougui, and A. Elbelrhiti Elalaoui, "Software-defined networking (sdn): a survey," *Security and communication networks*, vol. 9, no. 18, pp. 5803–5833, 2016.