Content Caching and Delivery in Heterogeneous Vehicular Networks

by

Huaqing Wu

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

Waterloo, Ontario, Canada, 2021

Examining Committee Membership

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

External Examiner:	Amiya Nayak Professor School of Electrical Engineering and Computer Science University of Ottawa
Supervisor(s):	Xuemin (Sherman) Shen University Professor Department of Electrical and Computer Engineering University of Waterloo
Internal Member:	Kshirasagar Naik Professor Department of Electrical and Computer Engineering University of Waterloo
Internal Member:	Liang-Liang Xie Professor Department of Electrical and Computer Engineering University of Waterloo
Internal-External Member:	Yaoliang Yu Assistant Professor School of Computer Science University of Waterloo

Author's Declaration

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

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

Abstract

Connected and automated vehicles (CAVs), which enable information exchange and content delivery in real time, are expected to revolutionize current transportation systems for better driving safety, traffic efficiency, and environmental sustainability. However, the emerging CAV applications such as content delivery pose stringent requirements on latency, throughput, reliability, and global connectivity. The current wireless networks face significant challenges to satisfy the requirements due to scarce radio spectrum resources, inflexibility to dynamic traffic demands, and geographic-constrained fixed infrastructure deployment. To empower multifarious CAV content delivery, heterogeneous vehicular networks (HetVNets), which integrate the terrestrial networks with aerial networks formed by unmanned aerial vehicles (UAVs) and space networks constituting of low Earth orbit (LEO) satellites, can guarantee reliable, flexible, cost-effective, and globally seamless service provisioning. In addition, edge caching is a promising solution to facilitate content delivery by caching popular files in the HetVNet access points (APs) to relieve the backhaul traffic with a lower delivery delay. The main technical issues are: 1) to fully reveal the potential of HetVNets for content delivery performance enhancement, content caching scheme design in HetVNets should jointly consider network characteristics, vehicle mobility patterns, content popularity, and APs' caching capacities; 2) to fully exploit the controllable mobility and agility of UAVs to support dynamic vehicular content demands, the caching scheme and trajectory design for UAVs should be jointly optimized, which has not been well addressed due to their intricate inter-coupling relationships; and 3) for caching-based content delivery in HetVNets, a cooperative content delivery scheme should be designed to enable the cooperation among different network segments with ingenious utilization of heterogeneous network resources.

In this thesis, we design the content caching and delivery schemes in the caching-enabled HetVNet to address the three technical issues. First, we study the content caching in HetVNets with fixed terrestrial APs including cellular base stations (CBSs), Wi-Fi roadside units (RSUs), and TV white space (TVWS) stations. To characterize the intermittent network connection caused by limited network coverage and high vehicle mobility, we establish an on-off model with service interruptions to describe the vehicular content delivery process. Content coding then is leveraged to resist the impact of unstable network connections and enhance caching efficiency. By jointly considering file characteristics and network conditions, the content placement is formulated as an integer linear programming (ILP) problem. Adopting the idea of the student admission model, the ILP problem is then transformed into a many-to-one matching problem between content files and HetVNet APs and solved by our proposed stable-matching-based caching scheme. Simulation results demonstrate that the proposed scheme can achieve near-optimal performances in

terms of delivery delay and offloading ratio with a low complexity. Second, UAV-aided caching is considered to assist vehicular content delivery in aerial-ground vehicular networks (AGVN) and a joint caching and trajectory optimization (JCTO) problem is investigated to jointly optimize content caching, content delivery, and UAV trajectory. To enable real-time decision-making in highly dynamic vehicular networks, we propose a deep supervised learning scheme to solve the JCTO problem. Specifically, we first devise a clustering-based two-layered (CBTL) algorithm to solve the JCTO problem offline. With a given content caching policy, we design a time-based graph decomposition method to jointly optimize content delivery and UAV trajectory, with which we then leverage the particle swarm optimization algorithm to optimize the content caching. We then design a deep supervised learning architecture of the convolutional neural network (CNN) to make online decisions. With the CNN-based model, a function mapping the input network information to output decisions can be intelligently learnt to make timely inferences. Extensive trace-driven experiments are conducted to demonstrate the efficiency of CBTL in solving the JCTO problem and the superior learning performance with the CNN-based model. Third, we investigate caching-assisted cooperative content delivery in space-air-ground integrated vehicular networks (SAGVNs), where vehicular content requests can be cooperatively served by multiple APs in space, aerial, and terrestrial networks. In specific, a joint optimization problem of vehicle-to-AP association, bandwidth allocation, and content delivery ratio, referred to as the *ABC* problem, is formulated to minimize the overall content delivery delay while satisfying vehicular quality-of-service (QoS) requirements. To address the tightly-coupled optimization variables, we propose a load- and mobility-aware *ABC (LMA-ABC)* scheme to solve the joint optimization problem as follows. We first decompose the *ABC* problem to optimize the content delivery ratio. Then the impact of bandwidth allocation on the achievable delay performance is analyzed, and an effect of diminishing delay performance gain is revealed. Based on the analysis results, the *LMA-ABC* scheme is designed with the consideration of user fairness, load balancing, and vehicle mobility. Simulation results demonstrate that the proposed *LMA-ABC* scheme can significantly reduce the cooperative content delivery delay compared to the benchmark schemes.

In summary, we have investigated the content caching in terrestrial networks with fixed APs, joint caching and trajectory optimization in the AGVN, and caching-assisted cooperative content delivery in the SAGVN. The proposed schemes and theoretical results should provide useful guidelines for future research in the caching scheme design and efficient utilization of network resources in caching-enabled heterogeneous wireless networks.

Acknowledgments

Pursuing the Ph.D. degree at the University of Waterloo is definitely one of the best decisions I have made. During my Ph.D. study, there are so many people who have greatly inspired and supported me, without whom this thesis would not be possible. I would like to take this opportunity to give my sincere thanks to them.

First and foremost, I would like to express my deepest and sincerest gratitude to my supervisor, Professor Xuemin Shen, for his invaluable supervision, enormous support, and great patience during my Ph.D. study at the University of Waterloo. His insightful advice, positive philosophy, and continuous encouragement have broadened my academic vision and improved my research abilities. From Prof. Shen, I have learned not only the rigorous research attitude and the dedicating spirit for work but also the enthusiasm for life and an open mind for different cultures and opinions. I have been extremely lucky to have Prof. Shen as my supervisor and he is and will always be my role model.

I would like to thank Professor Liang-Liang Xie, Professor Kshirasagar Naik, Professor Yaoliang Yu, and Professor Amiya Nayak for serving my thesis examination committee. Their insightful comments and valuable questions have significantly improved the quality of my thesis. I would also like to thank Professor Weihua Zhuang for helping me to build the knowledge base from her courses, which greatly benefits my research in this thesis.

In the past four years, the precious friendship, support, and help from BBCR members have made my life at the University of Waterloo unforgettable and enjoyable. I would like to thank Prof. Feng Lyu, Prof. Nan Cheng, Dr. Weisen Shi, Dr. Junling Li, Dr. Wenchao Xu, Dr. Wen Wu, Prof. Ning Zhang, Dr. Haibo Zhou, Dr. Peng Yang, Jiayin Chen, Dr. Haixia Peng, Conghao Zhou, Mushu Li, Mingcheng He, Mingyan Li, and Chenxi Li for their great help in both my research and life. I am also grateful for all the time spent with Dr. Nan Chen, Dr. Dongxiao Liu, Dr. Haohao Liao, Ziyu Zhao, Bo Yang, Dr. Dongxu Ma, Dr. Jianan Zhang, and all current and former BBCR members. I would like to specially thank every member in the BBCR-SAG subgroup, for the valuable meetings and discussions we had together, which are inspiring for my research works.

Finally, I would like to thank my parents for their love, understanding, and encouragement. A special thank to my fiance Yuhui Lin for the enduring love, unfailing support, and continuous encouragement throughout my undergraduate, Master, and Ph.D. studies. The eight years' accompaniment has supported me to overcome all the challenges and encouraged me to keep pursuing my dream.

Huaqing Wu
Mar. 15, 2021
Waterloo, Ontario, Canada

Dedication

This Ph.D dissertation is dedicated to my beloved parents,

my grandparents, my brother, and my fiance.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

CAV	Connected and Automated Vehicle
HetVNet	Heterogeneous Vehicular Network
RSU	Roadside Unit
BS	Base Station
UAV	Unmanned Aerial Vehicle
V2X	Vehicle-to-Everything
V2V	Vehicle-to-Vehicle
V2I	Vehicle-to-Infrastructure
VN	Vehicular Networks
SAG	Space-Air-Ground
V2R	Vehicle-to-roadside
DSRC	Dedicated Short-Range Communication
QoS	Quality of Services
SAGVN	Space-Air-Ground Integrated Vehicular Network
RAT	Radio Access Technology
TVWS	TV White Space
HAP	High Altitude Platform
LAP	Low Altitude Platform
LoS	Line-of-Sight
GEO	Geostationary-Earth-Orbit
MEO	Medium-Earth-Orbit
LEO	Low-Earth-Orbit
AP	Access Point
CBS	Cellular Base Station
ITU	International Telecommunication Union
ILP	Integer Linear Programming

AGVN	Aerial-Ground Vehicular Network
JCTO	Joint Caching and Trajectory Optimization
DSL	Deep Supervised Learning
CBTL	Clustering-Based Two-Layered
CNN	Convolutional Neural Network
ABC	User Association, Bandwidth allocation, and Content delivery ratio
LMA-ABC	Load- and Mobility-Aware ABC
DRL	Deep Reinforcement Learning
3GPP	3rd Generation Partnership Project
NR	New Radio
CR	Cognitive Radio
SDN	Software-Defined Networking
NFV	Network Function Virtualization
AI	Artificial Intelligence
CDN	Content Delivery Network
QoE	Quality of Experience
SE	Spectral Efficiency
ICN	Information-Centric Networking
PRAI	Partial Repeat-After-Interruption
SA	Student Admission
GS	Gale-Shapley
PSO	Particle Swarm Optimization
PSO	Particle Swarm Optimization
IoV	Internet of Vehicles
U2V	UAV-to-Vehicle
RCSP	Resource Constrained Shortest Path
SNR	Signal-to-Noise Ratio
S2V	Satellite-to-Vehicle
B2V	BS-to-Vehicle

Chapter 1

Introduction

The technique of connected and automated vehicles (CAVs) enables vehicles to interact with their internal and external environments to improve road safety, transportation efficiency, and the experience of both drivers and passengers [2]. To empower smart vehicular services especially in the future driverless era, high-bandwidth content delivery and reliable accessibility of multifarious applications are expected [3]. However, the limited radio spectrum resources, the inflexibility in accommodating dynamic traffic demands, and geographically-constrained fixed infrastructure deployment of current terrestrial networks pose great challenges in ensuring ubiquitous, flexible, and reliable network connectivity [4]. To address these challenges in a cost-effective way, heterogeneous vehicular networks (HetVNets) which integrate terrestrial networks with non-terrestrial networks can be leveraged to boost network capacity, enhance system robustness, and provide ubiquitous 3D wireless coverage. Furthermore, edge caching technologies can be utilized in HetVNets to further mitigate backhaul traffic burden and reduce vehicular content delivery delay. In this chapter, we first provide an overview of the vehicular content delivery networks, then elaborate the edge caching-assisted HetVNets with their specific communication characteristics. Finally, we present the three key research problems investigated in this thesis.

1.1 Overview of Vehicular Content Delivery Networks

With the tremendous technological development in advanced sensors, onboard processing, and wireless communications and networking, CAVs are expected to perform essential roles in diversified fields of human society. As predicted by International Data Corporation (IDC), the number of CAVs will continue to surge over the next several years, increas-

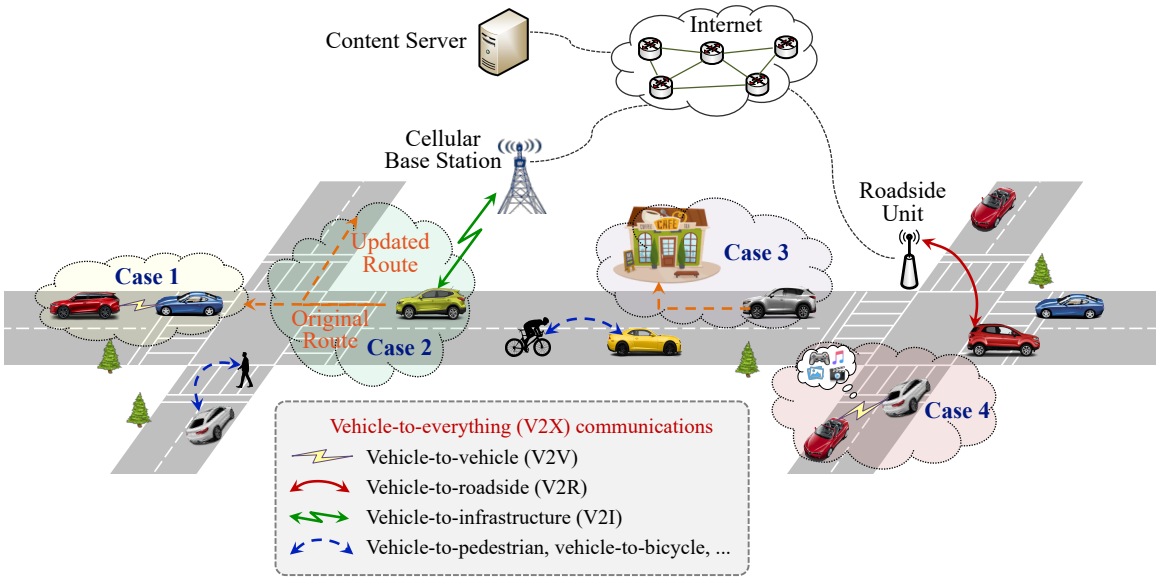


Figure 1.1: An overview of vehicular content delivery networks.

ing from 31.4 million units in 2019 to 54.2 million units (more than 50% of all vehicles produced) in 2024 [5]. CAVs can exchange information via both intra-vehicle communications and inter-vehicle communications [6]. Intra-vehicle communications happen within a vehicle, such as among different on-board sensors and systems. With inter-vehicle communications, a vehicle can communicate with other entities such as other vehicles, roadside units (RSUs), and base stations (BSs), collectively referred to as vehicle-to-everything (V2X) communications, as shown in Fig. 1.1. With V2X communications, ubiquitous information exchange and content delivery can be enabled to support multifarious CAV applications. In particular, content delivery in vehicular networks (VNs) has the following typical application scenarios [7]:

1. **Safety Information Delivery -** Road safety is of the utmost importance for connected vehicles. To reduce the frequency and severity of vehicle collisions, vehicles need to monitor the environmental data and collect vehicle state data to evaluate the vehicles' safety status. Vehicles' safety-related information includes available energy (e.g., fuel and electric), moving direction and speed, and distances among vehicles. By analyzing the collected safety-related information, the system can deliver alarm information when necessary and perform accident forewarning, shown as **Case 1** in Fig. 1.1;

2. **Traffic Efficiency Information Delivery -** To facilitate traffic management and transportation efficiency, useful information (including vehicles' locations, road conditions, and congestion) should be exchanged for system coordination and route planning. For example, an on-board map application can automatically reroute according to the received traffic condition information to avoid the congestion area and enhance traffic efficiency, shown as **Case 2** in Fig. 1.1;

3. **Infotainment Content Delivery -** To improve drivers' and passengers' traveling experiences, the infotainment content (such as videos, music, and maps) can be delivered and shared to provide informative or entertaining services. For example, service providers can collect vehicles' real-time locations and users' tastes and health conditions to recommend a restaurant and deliver the related information such as public evaluation and open hours, shown as **Case 3** in Fig. 1.1. Furthermore, vehicles in proximity may request similar location-based content such as traffic conditions and map data. In such cases, data transmission can be accomplished with direct delivery among adjacent vehicles, shown as **Case 4** in Fig. 1.1.

In this thesis, we mainly investigate the delivery of infotainment content. Notice that content delivery in VNs is different from that in conventional mobile networks since it is highly dependent on vehicles' mobility patterns, road conditions, and user behaviors [8]. Considering the massive amount of vehicular data, dynamic vehicular network conditions, and differentiated service requirements, there exist various technical challenges for content delivery in VNs:

1) Access network congestion - With an enormous increase in the number of vehicles on road and the proliferation of multifarious vehicular applications, the vehicular data requirement is soaring at a tremendous pace. According to the mobile data forecast from Cisco, the mobile data traffic will grow at a compound annual growth rate (CAGR) of 46 percent, increasing seven-fold from 2017 to 2022 [9]. Furthermore, connected cars will be the fastest-growing industry segment with a 28 percent CAGR. However, the cellular network capacity is not able to grow at a comparable pace to support the enormous data traffic due to the scarce spectrum resources and high cost of the infrastructure upgrade. The Federal Communications Commission (FCC) has allocated 75 MHz bandwidth at the 5.9 GHz spectrum band to dedicated short-range communications (DSRC) for vehicular communications. However, DSRC mainly focuses on enabling vehicular safety applications [10], i.e., supporting rapid short message exchange, and the limited spectrum resource is insufficient to satisfy the quality of services (QoS) of the bandwidth-intensive infotainment applications;

2) Backhaul network congestion - In addition to access network congestion, the transmission of massive vehicular data also increases the probability of blocking the backhaul networks, especially for wireless backhaul networks that have limited capacities and are easily congested. Although wired backhaul networks can provide a higher transmission data rate and have more spectrum resources, it is costly and sometimes difficult to deploy especially in some remote areas;

3) On-demand service provisioning - Network conditions (e.g., traffic density and request distribution) in VNs are highly dynamic in both temporal (e.g., peak hours or midnight) and spatial (e.g., urban or rural areas) domains. To guarantee uniform service coverage, the infrastructure needs to be densely deployed, which enhances the communication performance but also increases the deployment cost. Therefore, how to effectively utilize and deploy the infrastructure in VNs to provide on-demand content delivery services in different demanding areas is a challenging problem;

4) Globally connectivity and reliability - Global network connectivity, which is essential for ubiquitous vehicular service provisioning, can hardly be achieved by depending only on the current terrestrial networks due to the geographically-constrained infrastructure deployment. For example, it is cost-ineffective or even impossible to deploy infrastructure in sparsely populated or remote mountainous areas. Furthermore, how to guarantee reliable and uninterrupted network connections for vehicular content delivery regardless of the potential infrastructure outage (e.g., caused by natural disasters) is another critical yet challenging topic.

In summary, VNs can improve road safety and provide better travel experiences for drivers and passengers by enabling safety-related and infotainment content delivery. However, the massive traffic demands, the inherent characteristics of VNs, and the diversified vehicular service requirements make it challenging to design an efficient, flexible, and cost-effective content delivery system. To address the above-mentioned challenges, in this thesis, we investigate the edge caching-assisted content delivery in HetVNets to improve the service quality with enhanced resource utilization efficiency.

1.2 Edge Caching-Assisted Content Delivery in HetVNets

To support tremendous CAV content delivery, HetVNets are considered to expand the breadth and depth of communication coverage by utilizing multiple revolutionary networking techniques. Specifically, aerial networks based on unmanned aerial vehicles (UAVs) and

space networks consisting of satellites can be involved in HetVNets to assist vehicular content delivery, thus achieving a space-air-ground integrated vehicular network (SAGVN). With the integration of terrestrial, aerial, and space networks, the SAGVN can exploit the complementary advantages of different network segments to provide globally seamless, reliable, flexible, and cost-effective network access [11–13].

Despite the benefits and potentials brought by the SAGVN, the backhaul transmission of the vehicular traffic data still face some technical challenges. In specific, for terrestrial HetVNets, although the wireless cellular traffic burden can be relieved by utilizing other radio access technologies (RATs) like TV white space (TVWS) and Wi-Fi, the backhaul networks which support all vehicular traffic data still suffer a high congestion probability. For UAV-assisted network access, the wireless backhaul links are generally slow and unreliable. For satellite-based content delivery, an unacceptable delivery delay may occur if the satellite goes through backhaul links to retrieve the requested content due to the long propagation delay. To address these issues, edge caching technologies can be utilized to cache content files closer to the end users to alleviate backhaul congestion, reduce energy consumption, and decrease content retrieving delay.

1.2.1 Overview of the Heterogeneous SAGVN

As shown in Fig. 1.2, the SAGVN compromises three main network segments: ground networks, aerial networks, and space networks. The integration of these network segments has attracted increasing attention from both academia and industry. In this part, we first summarize recent advances in different network segments. Then, the advantages of the integrated SAGVN will be introduced.

A. Ground Networks

As the main solution to provide wireless network coverage in most scenarios, ground networks consist of heterogeneous terrestrial communication systems such as cellular networks, Wi-Fi, TVWS, and worldwide interoperability for microwave access (WiMAX). Ground networks can be characterized by the following features:

1) Ultra-dense small cells - As the key component in ground networks, cellular networks have evolved rapidly from the first generation (1G) to the fifth generation (5G) wireless networks. With the network development, the cell size becomes smaller with increasing BS density to enhance spectrum efficiency and network throughput. However, the ultra-dense deployment of small cells also leads to a high construction and maintenance cost.

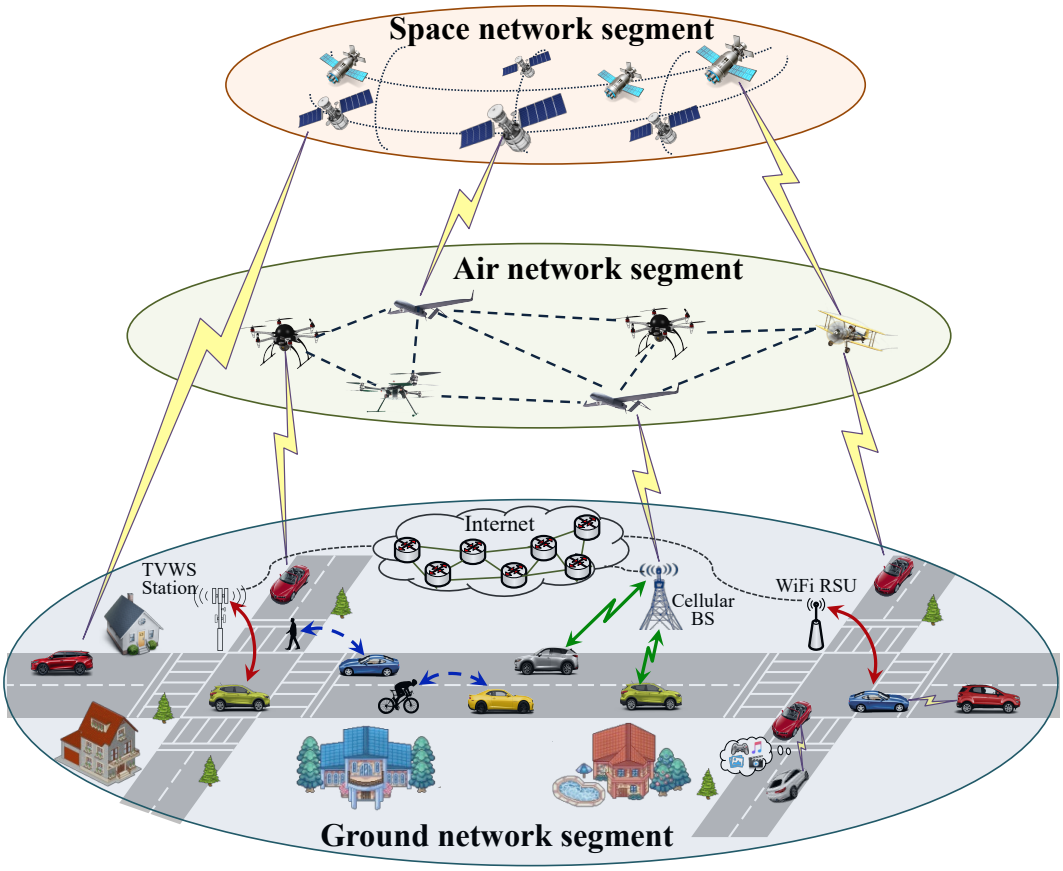


Figure 1.2: An illustration for space-air-ground integrated vehicular networks

2) High-speed fiber links - Ground nodes in the backbone network are interconnected via fiber optic links, which can provide low-cost high-throughput transmissions.

3) High-performance computing and caching - The data centers and servers in ground networks generally have powerful computing capabilities and massive storage for caching. These can ensure the network operation and management and provide a variety of services.

4) Fixed and limited coverage - Basically, the ground infrastructure is fixedly deployed to provide network access. Therefore, ground network coverage is limited and geographically constrained, especially in sparsely populated or remote mountainous areas.

B. Aerial Networks

Aerial networks are formed by UAVs at different altitudes, including the high altitude platform (HAP) which generally operates at an altitude of 17-22 km and the low altitude platform (LAP) which typically works at an altitude of no more than several kilometers [14]. Comparing LAP and HAP networks, HAP networks can provide a wider coverage with longer endurance, while LAP networks are more flexibly deployed and configured with better short-range communication performance [15]. In this thesis, we focus on LAP-UAVs to assist terrestrial networks in content delivery[1]. With the inherent characteristics of flexibility and controllability, UAVs have been considered as an indispensable component and a promising technique in the next generation networks due to the following advantages:

1) Line-of-sight (LoS) communication links - Compared to terrestrial communication links, there exist fewer obstacles between UAVs and ground users due to the high UAV operation altitude. With a higher probability of LoS connections, UAV communication links are generally more reliable with a better communication performance.

2) Flexible deployment - Different from the fixed deployment of terrestrial BSs, UAVs can be dynamically deployed to adapt to the spatially- and temporally-varying ground traffic. Since UAVs can be dispatched to different areas on demand, it is more cost-effective than deploying static BSs to guarantee QoS.

3) Fully-controlled mobility - Leveraging the agility and full controllability, UAVs' positions can be adjusted and optimized in real-time to improve communication link quality and response to potential emergency situations.

Despite the advantages of UAVs, there exist technical issues that greatly affect the UAV communication performance. First of all, UAVs are battery-powered and energy-constrained, where propulsion and directional adjustment consume most of the energy. Thus, energy-efficiency in UAV communications is critical to guarantee long-term network access without service interruption. In addition, extreme weather conditions can also affect the UAV operation and should be considered in UAV communications.

C. Space Networks

Space networks are composed of satellites in different orbits, i.e., geostationary-earth-orbit (GEO), medium-earth-orbit (MEO), and low-earth-orbit (LEO) satellites [16]. Although GEO and MEO satellites can provide a large coverage with a low relative velocity between satellites and terrestrial users, the excessive propagation delay inhibits their applicability

[1]In the rest of this thesis, "UAVs" are used to refer to "LAP-UAVs" for simplicity.

Table 1.1: Comparison in characteristics of terrestrial, aerial, and space network segments

	Terrestrial Networks	Aerial Networks	Space Networks
System deployment	Fixed deployment before use	Flexible deployment on demands	System configuration required with a long lead time
Mobility	Static	30-460 km/h [17]	28,000 km/h at orbits of 200 km (reduce with increased altitude)
Propagation delay	Low	Low	Less than 14 ms [18]
Advantages	High throughput, powerful computing and caching capabilities	Low cost, flexible movement, high LoS link probability	Large coverage, ultra reliability, broadcast/multicast capabilities
Disadvantages	Limited coverage, inflexible deployment, vulnerable to disaster	Energy-constrained, limited capacity, high mobility	Limited capacity, high mobility, non-negligible propagation delay

in most time-sensitive vehicular applications. In recent years, LEO satellite communications have attracted significant attention and are deemed as promising solutions to be incorporated in future network architectures due to the following advantages:

1) Globally seamless coverage - Benefiting from global availability, LEO networks have the potential to capacitate worldwide seamless service coverage in a cost-effective way, especially for users dispersed over wide geographical areas or in inaccessible areas.

2) Low delay for long-distance communications - Due to the low orbit altitude (300 km $\sim$ 1,500 km) compared to GEO (35,786 km) and MEO (7,000 km $\sim$ 25,000 km) satellites, the one-way LEO communication propagation delay is less than 14 ms. Users can experience a round-trip delay of no more than 50 ms, which is comparable to that of terrestrial links [18]. For long-distance communications, satellite communications can achieve even lower delay than terrestrial links due to the small number of hops.

3) Enhanced communication efficiency - The inherent broadcast/multicast nature of satellite communications enables group-based transmissions to enhance communication efficiency. For example, satellite communications can support CAV software update simultaneously for millions of vehicles with negligible communication cost.

4) Ultra network reliability and robustness - Due to the invulnerability to natural disasters, space networks can support ultra-reliable and robust service provisioning.

However, similar to UAVs, the limited energy capacities of LEO satellites also constrain the service duration and affect service functionalities including sensing, transmission, and processing. In addition, the high speed of LEO satellites (up to 28,000 km/h relative to the Earth's surface) results in highly dynamic network topologies, intermittent network connections, and frequent handover.

D. Integration of Space-Air-Ground Networks

As mentioned above, different network segments have their pros and cons in providing services, such as in terms of coverage, transmission delay, throughput, and reliability, as summarized in Table 1.1. Via effective internetworking, the complementary advantages of different network segments can be leveraged to enhance vehicular content delivery service qualities. For instance, satellite communications can supplement terrestrial networks for service provisioning in remote or sparsely populated areas; meanwhile, the complementary properties of satellite links (wide coverage) and a fiber optic backbone (high data rate) can be considered as alternative backbone technologies to wireless backhaul to mitigate the long-distance multihop backhaul. UAVs can provide flexible and reliable connectivity for vehicles in congested areas with dynamic traffic demands to relieve the terrestrial network burden and to boost service capacity. In addition, satellites/UAVs with remote sensing technologies can provide larger-scale monitoring data to assist terrestrial networks for efficient resource management and planning decisions. Therefore, the SAGVN can ensure seamless, robust, and reliable vehicular service provisioning.

1.2.2 Edge Caching-Assisted Vehicular Content Delivery

Despite the tremendous potentials of the SAGVN, the backhaul limitations (e.g., limited capacity and long propagation delay) may degrade the performance of vehicular content delivery. Stemming from the observations in [19], a large portion of mobile multimedia traffic can be attributed to duplicated downloads of a small fraction of popular content files. Furthermore, duplicated content requests can be intensified in small regions with certain events (e.g., concerts or sports games) where people have common interests in hot content. In VNs, location-based applications boost the repetitive download of location-oriented data (e.g., real-time traffic reports, high definition maps, and so forth) [20]. To better utilize the computing and storage capacities of current network infrastructure and modern vehicles, it is possible to cache the popular content closer to the end users. By

enabling direct content delivery from the caching-enabled access points (APs), e.g., Wi-Fi RSUs, TVWS stations, cellular base stations (CBSs), UAVs, and LEO satellites, to the vehicles, content caching on network edge infrastructure can significantly offload backhaul traffic [21, 22]. Besides, since the repeated transmission on backhaul links can be avoided, the content delivery delay is reduced significantly, which is essential in VNs to facilitate efficient content delivery with rapidly changing network topology.

When designing the caching strategies, there are two fundamental building blocks: *content placement (or content caching*[2]*)* and *content delivery*. Content placement mainly concentrates on the decision of content files to be cached and the selection of appropriate caching nodes to store the content files. Basically, the design of the content placement schemes is related to the management of the caching resources with the objective of serving more users' content requests with better content delivery performance. Content delivery, on the other hand, focuses on the dissemination of cached content files from the source to the destination. Content delivery is mainly about the management of the communication resources including the design of routing and forwarding policies, spectrum allocation schemes, and power control strategies. The research issues and required information for content placement and delivery are summarized in Table 1.2.

Extensive research has been devoted to vehicular content delivery in the caching-assisted HetVNet to achieve better delivery performance. The caching nodes in wireless edge networks include user equipment (vehicles and mobile phones), RSUs, CBSs (small, macro, and femto cells), UAVs, and satellites. To determine which content files to cache in the HetVNet, the content popularity should be considered to maximize the cache hit rate, i.e., the probability that requested content files are cached. The content popularity is generally assumed to follow static models (e.g., Zipf model [23]) or dynamic models (e.g., shot noise model (SNM) [24]). In recent years, machine learning-based content popularity prediction methods have also attracted significant attention. Based on the knowledge of content popularity and network information (such as network topology, content request pattern, and user mobility), various caching policies and algorithms have been proposed to enhance caching efficiency. Conventional caching policies such as the least recently used (LRU) and least frequently used (LFU) policies have been widely utilized in existing works [25]. A user preference profile (UPP) based caching policy has been proposed in [26] to utilize user preferences toward different video categories to enhance caching performance. Considering that content popularity varies with time and is not known a prior, a deep reinforcement learning-based caching policy has been proposed in [27] to jointly optimize the content placement and delivery with reduced system cost, decreased content delivery latency, and improved content hit ratio in the vehicular edge network. In VNs, the high vehicle mobil-

[2]In this thesis, we use the terms content caching and content placement interchangeably.

Table 1.2: Summary on content placement and delivery

	Research Issues	Description	Required Information
Content Placement	Where to cache	• Caching nodes: Vehicles/BSs/ UAVs/LEO satellites • Caching node placement • Caching node selection	• (Expected) network topology • User information: Content request pattern and user mobility • Content size • Content popularity • Caching storage capacity
	What to cache	• Reusable information (such as multimedia data) can be cached, while interactive applications, voice calls, or control signals cannot be cached • Determining which content files to cache	
	How to cache	• Deciding files to be cached in different caching nodes • Caching storage allocation	
Content Delivery	Where to obtain the content	• Association between caching nodes and content requesters	• (Expected) network topology • Content request pattern • User mobility • Content size • Network condition: Network capacity, link quality, congestion condition, etc.
	How to deliver the content	• Routing path selection • Communication resource (such as power and bandwidth) allocation	

ity intensifies the design complexity and hampers the efficiency of edge caching schemes in HetVNets. To address the mobility issues, a general framework for mobility-aware caching in content-centric networks (CCNs) has been proposed in [28] with known mobility information. When mobility information is unavailable, mobility prediction algorithms are leveraged to facilitate cooperative caching in vehicular CCNs [29]. With appropriate content files cached in caching nodes, the content delivery strategy should also be investigated to optimize the content delivery performance. In [30], the optimal content delivery is investigated in cache-enabled HetVNets with the stochastic content multicast scheduling to satisfy dynamic user demands. The signal transmission has also been optimized for content delivery to minimize the energy consumption in cache-assisted HetVNets [31].

1.3 Motivation and Contributions

1.3.1 Challenges of Content Caching and Delivery in the SAGVN

In spite of the initial research works mentioned above, efficient content caching and delivery in the SAGVN are still not sufficiently studied. To be specific, the design and implementation of the caching-assisted SAGVN still face some essential challenges:

1) Support high mobility - In the SAGVN, there are various types of mobility introduced by vehicular users, UAVs, and LEO satellites. As a result, the contact duration between a vehicular user and an AP (e.g., CBS, Wi-Fi RSU, UAV, LEO satellite) in the SAGVN is limited. Such limited contact duration is generally insufficient for content delivery, especially for large-size content files, since the volume of data that can be transmitted within one AP's coverage area is limited. Therefore, caching the whole content files in APs is inefficient since the complete downloading of one file requires multiple encounters with APs [32]. The caching scheme design with considering the user mobility and content file size should be investigated to improve caching efficiency.

2) Heterogeneous network characteristics - As introduced in Section 1.2.1, different network segments in the SAGVN have their specific pros and cons in terms of coverage, throughput, propagation delay, etc. The unprecedented heterogeneity in the SAGVN renders the traditional caching schemes inadequate. It is critical to customize the content caching and delivery policies by considering heterogeneous network characteristics to fully unleash their differential merits.

3) Coupled UAV trajectory design and resource allocation - Note that UAVs' ability to address terrestrial traffic variations is capacitated by the fully controllable mobility. Therefore, the UAV trajectory design problem, which optimizes UAVs' flying traces to serve vehicular users, is critical for UAV-assisted service provisioning [33]. On the other hand, the real-time allocation of both caching and communication resources should also be optimized to adapt to the varying user request distribution and improve the UAV content delivery performance in vehicular networks. Therefore, content caching, UAV trajectory, and content delivery are tightly coupled and should be jointly investigated, which has not been well addressed.

4) Energy constraints - Different from the terrestrial infrastructure, which has stable power supply, satellites/UAVs are powered by batteries and/or solar energy. The limited energy capacities of satellites/UAVs constrain the service duration and further affect service functionalities including sensing, transmission, and processing. In addition, service-unrelated energy consumption for satellites (due to intense radiation and space-variant

temperature) and UAVs (due to propulsion and direction adjustment) further deteriorates the service endurance. Therefore, vehicular users in the SAGVN can potentially suffer intermittent connection and service interruption due to satellite/UAV energy depletion. Improving energy efficiency to prolong the service duration of satellites/UAVs is critical for persistent vehicular content delivery service provisioning.

Besides the technical challenges, most countries and international organizations have specified some regulatory rules in terms of the usage of UAVs [34] and satellites [35]. To ensure the legal use of UAVs, all the UAVs should be controlled by UAV operators or governments rather than individual users. According to the International Telecommunication Union (ITU) regulations, the emission energy from satellites should be regulated to prevent harmful interference to the incumbent terrestrial communication systems. International regulations and standards are imperative to ensure the proper operation of the SAGVN system. Nevertheless, in this thesis, we focus on the technical challenges faced by content delivery in the SAGVN.

1.3.2 Approaches and Contributions

In this thesis, we investigate the content caching and delivery schemes in the caching-enabled SAGVN to tackle the above-mentioned challenges. The objective is to develop efficient caching policies in heterogeneous APs and design cooperative content delivery schemes to increase the network capability, improve resource utilization efficiency, and enhance service quality. In specific, we focus on the following three research topics.

- We first investigate the content caching scheme design in terrestrial HetVNets with fixed APs, where CBSs, Wi-Fi RSUs, and TVWS stations can cache popular content files to support vehicular content delivery. Considering the limited network coverage and high vehicle mobility, an on-off model with service interruptions is established to characterize the vehicular content delivery process with intermittent network connections. To resist the impact of unstable network connections, content coding then is utilized to enhance caching efficiency. By jointly considering file characteristics and network conditions, we formulate the content placement problem as an integer linear programming (ILP) problem to minimize the average content retrieving delay. Adopting the idea of the student admission model, we then transform the ILP problem into a many-to-one matching problem between content files and HetVNet APs and propose a stable-matching-based caching scheme to solve it. Simulation results demonstrate that the proposed scheme can achieve near-optimal performances in terms of delivery delay and offloading ratio with a low complexity.

- Focusing on the aerial-ground vehicular network (AGVN), UAV-aided caching is considered to assist vehicular content delivery. To maximize the overall network throughput, a joint caching and trajectory optimization (JCTO) problem is investigated to jointly optimize content caching, content delivery, and UAV trajectories. Considering the inter-coupled optimization variables and limited UAV on-board energy, the JCTO problem is intractable directly and timely. To enable real-time decision-making in highly dynamic vehicular networks, we propose a deep supervised learning (DSL) scheme to solve the JCTO problem. Specifically, a clustering-based two-layered (CBTL) algorithm is first designed to solve the JCTO problem offline. With a given content caching policy, we design a time-based graph decomposition method to jointly optimize content delivery and UAV trajectory, with which the particle swarm optimization algorithm is then leveraged to further optimize the content caching. We then design a DSL architecture of the convolutional neural network (CNN) to make online decisions. The network density and content request distributions with spatial-temporal variations are labeled as channeled images and input to the CNN-based model, and the results obtained from the CBTL algorithm are labeled as model outputs. The CNN-based model can be trained to intelligently learn the mapping function between the input network information and output decisions and make real-time inferences. Extensive trace-driven experiments are conducted to demonstrate the efficiency of CBTL in solving the JCTO problem and the superior learning performance with the CNN-based model.

- To further improve the vehicular content delivery performance in the SAGVN, we investigate caching-assisted cooperative content delivery to minimize the overall content retrieving delay. In particular, vehicular content requests can be cooperatively served by multiple APs in space, aerial, and terrestrial networks. A joint optimization problem of vehicle-to-AP association, bandwidth allocation, and content delivery ratio, referred to as the *ABC* problem, is then formulated. To address the tightly-coupled optimization variables, we propose a load- and mobility-aware *ABC (LMA-ABC)* scheme to solve the joint optimization problem as follows. We first decompose the *ABC* problem to optimize the content delivery ratio by considering the distinct characteristics of different network segments. Then the impact of bandwidth allocation on the achievable delay performance is analyzed, and an effect of diminishing delay performance gain is revealed. Based on the analysis results, the *LMA-ABC* scheme is designed with the consideration of user fairness, load balancing, and vehicle mobility. Simulation results demonstrate that the proposed *LMA-ABC* scheme can significantly reduce the cooperative content delivery delay compared to the benchmark schemes.

1.4 Thesis Outline

The remainder of the thesis is organized as follows: In Chapter 2, we provide a comprehensive background and review of vehicular content delivery in the HetVNet and the state-of-the-art content caching and delivery schemes. In Chapter 3, a coding-based content caching scheme is designed for the terrestrial HetVNet with intermittent network connections, and a matching-based algorithm is proposed to solve the content caching problem with minimized content retrieving delay. In Chapter 4, we formulate the JCTO problem in the AGVN to jointly optimize the content placement, content delivery, and UAVs' trajectories. Then, a CNN-based deep learning algorithm is proposed to solve the problem in a real-time manner. In Chapter 5, the cooperative content delivery in the SAGVN is investigated and the *ABC* problem is formulated. Then, the *LMA-ABC* scheme is proposed to solve the problem with the consideration of user fairness, load balancing, and vehicle mobility. Finally, we conclude the thesis and discuss future works in Chapter 6.

Chapter 2

Literature Review

This chapter aims to introduce the background and related works of caching-assisted vehicular content delivery in the HetVNet, including the vehicular content delivery in the terrestrial HetVNet, AGVN, and SAGVN, as well as the state-of-the-art content caching and delivery techniques.

2.1 HetVNet-Based Vehicular Content Delivery

To alleviate the burden caused by the high vehicular communication demands, HetVNet-based data offloading (or traffic offloading) is an effective approach as it utilizes complementary and revolutionary networking techniques to deliver mobile data originally planned for transmissions over cellular networks. Basically, the HetVNet can be classified into two categories, i.e., a multi-tier network with a single RAT and a multi-tier network with multiple RATs [36]. In single-RAT HetVNet scenarios, the content delivery can be offloaded to smaller cells like pico or femto cells, which also operate on the same cellular band as the macro BSs. The multi-RAT HetVNet, on the other hand, indicates that the cellular macrocells cooperate with other RATs, e.g., Wi-Fi and TVWS. Regarding the capacity constraint in cellular networks, we mainly target multi-RAT HetVNet for vehicular content delivery in this thesis.

2.1.1 Vehicular Content Delivery in the Terrestrial HetVNet

Heterogeneous networks have emerged as one of the most promising network architectures to increase system throughput in wireless networks. Up to now, substantial heterogeneous

offloading trials have been implemented in academic and industrial communities. In this part, we review the state of the arts in data offloading through heterogeneous terrestrial APs. In particular, two candidate techniques, i.e., the Wi-Fi-based and TVWS-based techniques, are discussed in detail. Moreover, offloading strategies developed for content delivery in the HetVNet are demonstrated.

A. Wi-Fi-Based Techniques

Wi-Fi stands for Wireless Fidelity in the research community, which is an IEEE 802.11 standard. As a popular wireless broadband access technology, Wi-Fi operates on the unlicensed spectrum (2.4 GHz and 5 GHz) and offers high data rates with limited coverage. Existing works have shown that Wi-Fi networks can significantly offload cellular traffic in vehicular scenarios with properly designed offloading strategies [37, 38]. Nowadays, Wi-Fi becomes a natural solution to offload cellular traffic due to the built-in Wi-Fi capabilities of modern devices like smartphones. In light of the cellular service performance degradation in overloaded areas, more and more network operators are extending their access networks by deploying Wi-Fi hotspots that are directly managed by them. For example, AT&T has deployed more than 30,000 Wi-Fi hotspots in the US. China Mobile has deployed 4.4 million public Wi-Fi access points throughout China. Similarly, KT Corporation (formerly Korea Telecom) in South Korea owns and operates more than 140,000 Wi-Fi hotspots that are actively used for offloading traffic. Wi-Fi access is attractive because of its following advantages. *1)* Wi-Fi hotspots are widely deployed in many urban areas. Globally, total public Wi-Fi hotspots (including homespots) are forecasted to grow four-fold from 2018 to 2023, from 169 million in 2018 to 628 million by 2023 [39]; *2)* Wi-Fi access is often free of charge or inexpensive. For example, China Mobile offers Wi-Fi services with less than $20 a month for unlimited data usage; *3)* Most current mobile devices, such as smartphones, tablets, laptops, and more and more modern vehicles are equipped with Wi-Fi interfaces; and 4) Currently Wi-Fi technologies (e.g., IEEE 802.11ac/ad) can provide data rates of up to several Gbps.

Depending on the level of integration between Wi-Fi and cellular networks, network operators have three main approaches to offload cellular traffic to Wi-Fi networks [40]. The corresponding characteristics of the three approaches are presented in Table 2.1.

1) Network bypass or unmanaged data offloading approaches - In this case, users' data is automatically directed to Wi-Fi networks when the users locate within the Wi-Fi coverage, completely bypassing the cellular network for data services. However, voice services will remain on the cellular network.

2) Managed data offloading approaches - In this approach, the operators can retain

Table 2.1: Characteristics of different Wi-Fi offloading approaches

Offloading Type	Unmanaged data offloading	Managed data offloading	Integrated data offloading
Advantages	1) No need to deploy any network equipment; 2) Easy to implement.	1) Operators have control over subscribers; 2) Complete integration of cellular and Wi-Fi networks is not required	1) Operators have control over subscribers; 2) Operators do not lose revenue
Disadvantages	1) Operators cannot control the subscribers in Wi-Fi coverage; 2) Operators cannot deliver subscribed content and lose revenue.	Operators cannot deliver subscribed content and lose revenue	Integration between Wi-Fi and cellular systems is required.
Implementation	Placing applications in devices to switch on Wi-Fi interface when detecting Wi-Fi coverage	Placing intelligent session-aware gateway to detect subscribers' Wi-Fi sessions traversing to the Internet	Forming a bridge between cellular and Wi-Fi networks through which data flow can be established.

control of their subscribers. For instance, operators can provide services such as parental control/filtering which cannot be achieved in unmanaged data offloading mode. Another example is that some operators can observe subscribers' browsing habits for targeted marketing reasons. However, the operators in this case still are not allowed to push subscribed content.

3) Integrated data offloading approaches - In this case, operators can fully control their subscribers and deliver the subscribed content when users are within Wi-Fi coverage. In the integrated approaches, a bridge can be established between the Wi-Fi and cellular networks to allow data flow. The Wi-Fi and cellular systems can either be loosely coupled or tightly coupled. In loose coupling architecture, the networks are independent and require no major cooperation between them. The Wi-Fi network is connected indirectly to the cellular core network and service connectivity is provided by roaming between the two networks. In tight coupling architecture, the networks share a common core and the majority of network functions such as vertical handover, resource management, and billing are controlled and managed centrally.

In VNs, there exist many experimental and theoretical studies to evaluate the performance and demonstrate the effectiveness of Wi-Fi-based drive-thru networks. Based on the performance metrics, existing works can be classified into four main subcategories: *delay-oriented, throughput-oriented, continuity-oriented, and offloading efficiency-oriented.*

To evaluate the performance of the Wi-Fi access delay (time required for authentication, IP address assignment, and so on), authors in [38] adopt the Markov chain to study the impact of different factors on the access delay, such as the number of contending Wi-Fi users, wireless channel conditions, and the utilized authentication mechanisms. The accuracy of the theoretical analysis is then verified via MATLAB simulations and experimental testing. In [41], delay-constrained data transmission is investigated in Wi-Fi VNs, where data traffic is optimally distributed over cognitive radio and Wi-Fi interfaces to ensure timely and energy-efficient transmission.

In addition to the delay, throughput is another important metric to evaluate the performance of Wi-Fi-assisted vehicular content delivery. In [42], the throughput performance of the Wi-Fi-based drive-thru Internet is investigated by considering the impact of the access procedure. Particularly, two access strategies, i.e., WPA2-PSK and Hotspot 2.0, are studied to show their impact on the throughput performance. In [43], a handoff scheme in vehicular multi-tier multi-hop mesh networks is proposed for the Wi-Fi-based VNs. Evaluation results show that triple throughput can be achieved with the proposed Wi-Fi-based handoff scheme.

In highly dynamic VNs, it is critical to mitigate communication disruption and maintain service continuity. To achieve this goal, authors in [44] propose a scheme named SWIMMING to ensure seamless and efficient Wi-Fi-based Internet access for moving vehicles, by designing innovative protocols for both uplink and downlink. Particularly, an ACK detection function is designed to eliminate the adverse effect of multiple ACKs and improve channel utilization efficiency. In [45], an optimal scheme named ViFi (V-band Wi-Fi) is proposed to solve the disruptions in connectivity. ViFi utilizes the BSs in close proximity to vehicles to relay packets to reduce the disruption frequency. A two-month long trace-driven simulations show that ViFi can achieve considerable improvement in terms of both TCP performance and VoIP services.

Another important metric for vehicular Wi-Fi communication is how much data can be offloaded to Wi-Fi networks, i.e., offloading efficiency. In [46], a prediction-based offloading scheme named Wiffler is proposed to determine the accessed networks for different applications in HetVNets. Particularly, for delay-tolerant applications, Wiffler leverages a Wi-Fi connectivity prediction model to defer application data on Wi-Fi. For delay-sensitive applications, Wiffler proactively switches to cellular networks to avoid high delay penalties. In [47], a V2V assisted Wi-Fi offloading scheme is proposed to improve the offloading efficiency in drive-thru Internet. Vehicles are associated with different APs and can utilize V2V communication to assist peers' data tasks via their own Wi-Fi resource.

In addition to targeting only one performance metric in a work, there also exist extensive researches focusing on two or more metrics or exploring the trade-off among different

metrics. In [48], two game-theory based Wi-Fi offloading schemes are proposed to offload vehicular traffic through Wi-Fi networks. It is shown that the proposed two offloading mechanisms can improve the average utility of vehicular users, reduce average service delay, and effectively offload cellular traffic. In [49], the cost-effectiveness of a Wi-Fi solution for vehicular Internet access is investigated. By deploying Wi-Fi RSUs at the signalized intersection and studying the impact of traffic signals on Wi-Fi access, the trade-off between cost-effectiveness and the normalized service delay is examined. The delay-cost trade-off is also investigated in [37]. By assuming exponential and Gaussian distributions for vehicle-RSU encountering times, an adaptive algorithm is proposed to design a data downloading strategy to achieve the best delay-cost trade-off.

B. TVWS-Based Techniques

Another potential solution to alleviate the cellular spectrum scarcity problem is to leverage the TVWS band. While analog TV broadcasting became obsolete, the TV spectrum has been significantly under-utilized currently. According to a study in Japan, more than 100 MHz of TVWS spectrum are observed available in about 84.3% of the country's area. Similarly, more than 50% of the TV channels are vacant in US and Hongkong [50]. More available TV spectrum is expected in some developing countries because of fewer TV stations used. Thanks to the analog-to-digital transition, which has been completed or is expected to be completed in the near future in many developed countries, a substantial amount of TV spectrum that was previously occupied by the TV broadcasting system can be released, which allows the unlicensed use of TV spectrum when it is not utilized by licensed users. The unused TV spectrum is referred to as the TVWS band. Due to its low frequency, the TVWS band can provide high penetration capabilities, low path loss, and wide coverage up to several kilometers [51]. Specifically, when operating in the TVWS band, to ensure non-interfering spectrum utilization, vehicles can access the TVWS channels only if they are not being used by incumbent users. In other words, the TVWS network access might be interrupted when the primary users become active.

Compared with the cellular and Wi-Fi radio interfaces which are widely adopted in most modern devices, TVWS technologies have not been widely implemented. However, there have been many standards and research works focusing on unlicensed communications in the TVWS band. To better utilize the TVWS spectrum, regulators around the world have specified regulation frameworks for unlicensed wireless devices operating in the TVWS band [52]. Furthermore, a set of standards have also been proposed and adopted for unlicensed use of TVWS spectrum, including IEEE 802.11af [53], IEEE 802.22 [54], and ETSI reconfigurable radio systems (RRS) [55]. IEEE 802.19.1 standard [56] has also

been published to facilitate the coexistence of heterogeneous networks in the TVWS band. Furthermore, there exist many industrial organizations (such as Carlson RuralConnect, Adaptrum, and 6 Harmonics) that provide devices and systems for TVWS Internet connectivity. Therefore, it can be expected that, just like Wi-Fi, the widespread implementation of TVWS technology will also become a reality in the near future.

Substantial field tests and theoretical analysis have been done to evaluate the achievable performance of the communication over the TVWS band. In [57], field tests as well as indoor evaluation of a multi-hop V2V communication system with distributed and autonomous TVWS channel selection are conducted in Japan. In [58], the performances of 802.11n (in 2.4GHz) and 802.11af (in TVWS band) standards are compared in real environments by means of theoretical analysis and simulations. The results show that the 802.11n outperforms the 802.11af in terms of data rate, but struggles in complex environments and NLOS conditions, where the 802.11af can effectively improve the communication performance. Focusing on TVWS as the offloading technology, a connection-aware balancing algorithm (CABA) is proposed in [59] to exploit multiple radio connections to balance the load and route the traffic through the best possible interface given the network condition. Running a multi-interface AP with TVWS and Wi-Fi in the field tests, the results show that balancing the load between Wi-Fi and TVWS can provide a higher playback quality (up to 15% of the average quality index) in scenarios in which the Wi-Fi is received at a low strength.

Recently there have been more and more research works focusing on exploiting TVWS for offloading in VNs. In [60], the TVWS application scenarios in HetVNets are presented. The authors also propose two TVWS geolocation database based architectures for V2V and V2I communications. The key technical challenges and future research directions toward exploiting TVWS for vehicular communications are also highlighted. Similarly, challenges and opportunities for TVWS access in HetVNets are addressed in [51], with an emphasis on media access control (MAC) layer issues. Numerical results of the DSRC system augmented with a TVWS cognitive module show that TVWS-enabled DSRC vehicular network outperforms the standard DSRC network in terms of network throughput and packet loss ratio. In [61], the TVWS band is exploited for vehicular safety message dissemination in NLOS intersections, where two mechanisms, i.e., a collaborative procedure and a dynamic optimal configuration, are adopted to ensure reliable delivery in all vehicle densities without relying on the infrastructure.

2.1.2 Vehicular Content Delivery in the AGVN

The UAV-assisted AGVN is a promising solution to support vehicular content delivery. For offloading through fixed deployed APs, a very dense AP deployment is required to guarantee a uniform service coverage in the spatial domain. Furthermore, it will inevitably take a long period for hardware equipment upgrade with a high maintenance cost. Furthermore, the fixed infrastructure deployment can hardly keep pace with the explosive increasing and unevenly distributed mobile data. Compared with fixed deployed APs, UAVs have several important advantages including flexible deployment, cost-effectiveness, and better LoS communication links, as mentioned in Section 1.2.1. These benefits make UAV-aided wireless communications a promising integral component of future wireless systems to support diverse applications with orders-of-magnitude capacity improvement.

In recent years, UAVs have attracted increasing attention in industrial applications. For example, *General Atomics and Boeing* are among the notable manufactures providing military UAVs. *Prime Air* and *Project Wing* are delivery systems to offer rapid delivery by using UAVs. *AT&T* and *Verizon* have both conducted trials with LTE BSs mounted on UAVs. Moreover, *Qualcomm* is also planning to deploy UAVs for enabling wide-scale wireless communications in the 5G wireless networks.

There are three typical use cases of UAV-aided wireless communications [15]: *1) UAV-aided ubiquitous coverage* - UAVs can be utilized to assist the existing terrestrial networks to extend network coverage [62, 63]. In this scenario, UAVs are dispatched to cover areas without terrestrial network coverage (due to incomplete infrastructure deployment or infrastructure damage due to natural disasters) or extremely crowded areas; *2) UAV-aided relaying* - UAVs are exploited to relay information between two or more distant users or user groups without reliable direct communication links [64]; and *3) UAV-aided information dissemination* - UAVs are also capable of disseminating (or collecting) delay-tolerant information to (from) a large number of distributed wireless users [65]. One example scenario is that UAVs can intelligently broadcast the map information to a cluster of vehicles to alleviate the traffic burden in terrestrial networks.

A. UAV Deployment and Trajectory Design in the AGVN

The UAVs' mobility control is essential for exploiting the full potential of UAV-aided wireless communications. Therefore, in most current research works, the UAV-aided system is investigated with the optimization of UAV deployment or trajectory design to better serve the mobile users. The UAV deployment design aims to find the optimal locations for UAVs, including flying altitudes, horizontal positions, and spatial density, to achieve

the best performance. In [66], a UAV-assisted HetVNet is investigated with a multi-layer aerial-road vehicular architecture, and a density-aware deployment scheme is proposed to maximize the throughput with an iterative three-dimensional matching resource allocation algorithm. In [62], a three-dimensional UAV-cell deployment problem is investigated with a given number of UAVs being deployed to maximize the user coverage while maintaining UAV-to-user link qualities. Particularly, a per-UAV iterated particle swarm optimization algorithm is proposed to optimize UAV deployments for different UAV numbers.

Different from UAV deployment, the UAV trajectory design focuses on designing the optimal trajectory, following which the UAVs fly to serve users in multiple areas. Recently, UAV trajectory design has attracted more and more research attention due to its cost-effectiveness, since one flying UAV can cover multiple areas without requiring the deployment of multiple UAVs. In [33,67–70], UAVs work as mobile BSs to assist vehicular content delivery in the AGVN. In [67], UAVs are equipped with the cellular-V2X (C-V2X) technology to assist vehicular communications with optimized UAV design and radio resource management. Aiming to serve vehicles that are not covered by terrestrial infrastructure, a deep reinforcement learning (DRL) based approach is proposed in [68] to optimize UAV trajectories with the consideration of vehicular network dynamics. In [33], a multi-UAV trajectory planning and resource allocation (TPRA) problem is studied to maximize the accumulative network throughput while guaranteeing user fairness, UAV power consumption, and link quality constraints. The TPRA problem is decomposed into two sub-problems and solved by the proposed hierarchical DRL-based approach. In [69], to maximize the overall achievable sum rate of V2I users while guaranteeing the reliability of V2V communications, the authors analyze the optimal transmission power of vehicular users, optimize the spectrum sharing and resource allocation by graph-based methods, and control the UAV trajectory control by proposing a Q-learning algorithm. Caching-enabled UAVs are considered in [70] to assist vehicular content delivery in the AGVN. Aiming to maximize the operational utility, the caching decisions, UAV trajectory, and radio resource allocation are jointly optimized while considering the environmental uncertainties and UAVs' energy limitations. In [71], a UAV is considered as a relay between BSs and vehicular users in the AGVN with optimized UAV trajectory and power allocation. In addition, a dynamic NOMA/OMA scheme is proposed where the NOMA and OMA modes are selected by considering the trade-off between the complexity of successive-interference-cancellation (SIC) decoding in NOMA and the achievable sum-rate gain. UAV-aided computing offloading is investigated in [72], where a UAV is deployed to provide mobile edge computing services to a set of ground vehicles. An optimization framework for total utility maximization is developed by jointly optimizing the transmit power of vehicles and the UAV trajectory via a dynamic programming method.

B. UAV-Based AGVN Design with Known UAV Mobility

Although the mobility of UAVs is a critical research problem, not all research works on UAV-assisted communications focus on the UAV deployment or trajectory optimization. There exist many existing works on UAV-based AGVN design that leverage the full controllability of UAVs and assume that the UAV positions/trajectories are known in advance. In [73], UAVs are utilized as store-carry-forward nodes to enhance the availability of connectivity paths among vehicles and reduce the end-to-end packet delivery delay, where UAVs periodically broadcast their mobility information (including location, speed, and travel direction) to facilitate network management. In [74], an anti-jamming UAV relay problem is studied where UAVs relay messages from vehicles to RSUs to improve the communication performance against smart jammers. A hot-booting policy hill climbing-based UAV relay strategy is proposed to achieve the optimal relay policy without requiring information on the vehicular model and the jamming model. In [75], the real-time positions of vehicles and UAVs are assumed to be known by using a global positioning system (GPS). Then, an efficient routing scheme based on a flooding technique is proposed to ensure reliable and robust data delivery in the AGVN, where UAVs and vehicles cooperate in an ad hoc fashion to provide routing paths. UAVs in [76] act as relays to enhance the vehicular communication performance, where the mobility information of UAVs and vehicles are obtained via cooperative information exchange. The relay selection problem is then studied and formulated as a multi-objective optimization problem by jointly considering the state transition probability of communication interruption and the transmission consumption including energy consumption and delay consumption.

2.1.3 Vehicular Content Delivery in the SAGVN

As the SAGVN is still in its infancy, extensive industrial and academic efforts have been devoted to construct satellite constellation systems and provide insights into the convergence of space, aerial, and terrestrial networks.

A. Industrial Efforts and Standardization Activities

In the past decades, several LEO satellite constellation systems were established and applied for global wireless communications. The first global LEO satellite network is the *Iridium* system, mainly focusing on voice and low-rate data services. Other LEO satellite systems, including *Globalstar* and *Orbcomm*, tried to support satellite phone or Internet services to terrestrial users. However, these existing satellite networks did not perform well

in the communication market and eventually went bankrupt due to the high construction cost. Recently, driven by the micro-satellite manufacturing and low-cost launch technologies, there are various initiatives to construct satellite constellations and launch thousands of LEO satellites. For instance, *Starlink* is an LEO satellite system proposed by *SpaceX*, which plans to launch 42,000 satellites into orbits with altitudes of 340 km $\sim$ 1150 km. Currently, more than 1325 Starlink satellites have been launched, and the global service is expected by late 2021 or 2022 [77]. *OneWeb* satellite system is expected to include 648 LEO satellites on the orbits with an altitude of 1,200 km, operating on the Ka and Ku bands. Since 2019, *OneWeb* has launched 110 satellites into the orbit, and the globally commercial usage is expected to begin in 2021 [78]. *TeleSat*, another LEO satellite network, is expected to have 300 satellites on the orbits with an altitude of 1,000 km and provide global service starting 2022 [79].

The standardization of satellite-terrestrial network integration has also been developed to guide the implementation of the SAGVN with high performance. The 3rd generation partnership project (3GPP) has done substantial standardization work on satellite terrestrial network integration. Standards [80–83] have been developed to explore the scenarios, identify service requirements, study the application of new radio (NR), and classify the service application scenarios. ETSI, another standard organization, has also proposed some standards related to the convergence of satellite and terrestrial networks. In [84, 85], the definition and classification of communication scenarios, the role of satellites in disaster management, and the resource requirements for different applications are presented, Considering the scenario of satellite backhauling, [86, 87] address the problem of traffic distribution in the wireless access network and the implementation of the cooperation between satellite networks and 3G femto BSs. In [88], the Satellite Independent Service Access Point (SI-SAP) is proposed and the physical air interfaces for broadband services are regulated.

In addition to industrial satellite constellation construction and the standardization processes, there are also some related projects exploring the solutions for the integrated SAGVN. Project *CoRaSat* (Cognitive Radio for Satellite Communications) aims to apply cognitive radio (CR) technology in satellite communication systems to improve the frequency resource utilization [89]. Project *SANSA* (Shared Access Terrestrial-Satellite Backhaul Network enabled by Smart Antennas) proposes a satellite terrestrial network to mitigate the backhaul pressure [90]. *VITAL* (VIrtualized hybrid satellite-TerrestriAl systems) project introduces software-defined networking (SDN) and network function virtualization (NFV) technologies to enable flexible management of the integrated network [91]. In project *SATNEX IV* (SATellite NEtwork of Experts IV), the utilization of terrestrial communication technologies in space networks is evaluated [92]. *SaT5G* (Satellite and ter-

restrial network for 5G) project aims to bring satellite communications into 5G by defining optimal satellite-based backhaul and traffic offloading solutions [93]. Project SATis5 aims to build a large-scale real-time live end-to-end 5G integrated satellite terrestrial network proof-of-concept testbed [94].

B. Research Activities in the SAGVN

The SAGVN shows great potential in the next-generation network systems. There have been a large number of works focusing on the possible architecture of the integrated SAGVN. In [11], an SDN-based integrated network architecture is proposed to enable flexible, efficient, and global management for the SAGVN. In [4], an SDN-based SAGVN is investigated with a hybrid and hierarchical control architecture, and artificial intelligence (AI)-based engineering solutions are proposed to facilitate efficient network slicing, mobility management, and cooperative content caching and delivery. In [95], the SAGVN is studied to address the issues including network reconfiguration under dynamic space resources constraints, multi-dimensional sensing and context information integration, and real-time and secure vehicular communications. In [96], a framework of SAG integrated moving cell, namely, SAGECELL, is proposed to combine space, aerial, and terrestrial networks in a complementary fashion for matching dynamic varying traffic demands with limited network capacity supplies.

Focusing on the communication resource allocation in the integrated network, a state-based and event-driven system model is proposed to facilitate content delivery in [97]. In [98], a cooperative multi-group multicast-based content delivery strategy is proposed. In particular, beamforming technologies are utilized to improve the network efficiency and to serve the diversified service requirements with limited radio resources. In [99], the ultra-dense LEO satellites are integrated with the terrestrial network to achieve efficient data offloading. Considering the LEO-based backhaul link capacity constraints, the objective of the work is to maximize the sum data rate and the number of accessed users.

To fully utilize the computing capability of the heterogeneous devices in the SAGVN, edge computing enhanced SAGVN is promising to enable ubiquitous data processing and content sharing [100]. Aiming to minimize the task completion time and satellite resource usage, a joint offloading decision and caching placement problem is investigated for vehicles in the remote area. In [101], the joint optimization of radio resource allocation and bidirectional communication and computation task offloading is investigated. The original optimization problem is then decoupled into two sub-problems and solved by the proposed heuristic algorithm. Leveraging the complementary advantages of different network segments in terms of communication and computing capabilities, a bidirectional mission

offloading scheme is developed in [13].

A comprehensive survey of the integrated SAG networks is provided in [102], where topics on cross-layer design, resource management and allocation, system integration, and network performance analysis are discussed. In [103], a comprehensive simulation platform is developed for the integrated SAG network, integrating multiple network protocols, node mobility, and control algorithms. Furthermore, various interfaces are provided to enable functionality extension to facilitate user-defined mobility traces and control algorithms. In view of the complex and dynamic network environment of the integrated SAG network, AI-based techniques are leveraged in [104] to improve the network performance by addressing challenges in network control, spectrum management, energy management, routing and handover management, and security guarantee.

2.2 Caching-Assisted Vehicular Content Delivery

Edge caching is a promising technique to address the backhaul limitations including backhaul congestion, unreliable backhaul links, and long backhaul delay. Caching techniques enable replicating content files in strategically placed caching nodes/servers during off-peak time, and redirecting content requests to the most appropriate servers at peak time. In fact, the caching based content delivery networks (CDNs) have been increasingly deployed in the world. For example, *Akamai Intelligent Platform, Google Global Cache system, Facebook CDN, Amazon CloudFront*, and *ARA CDN* are several representative state-of-the-art caching systems [105]. In this section, the existing works on content placement and delivery strategies are provided.

2.2.1 Content Placement Strategies

The key idea of content placement is to cache content files in nodes close to the requesting users, enabling content download from caching nodes instead of the remote content server. Generally, the content placement problem involves three critical issues: 1) where to cache; 2) what to cache; and 3) how to cache. In the SAGVN, the potential caching nodes could be vehicles and/or the edge infrastructure (e.g., CBSs, Wi-Fi RSUs, TVWS stations, UAVs, and LEO satellites) in different network segments. The selection of caching nodes is affected by multiple factors including but not limited to the energy and storage capacity constraints of caching nodes, the willingness of caching nodes to share their resources, and the contact characteristics between the requesting vehicles and the caching nodes. The problem of what to cache, on the other hand, is largely determined by content file sizes, the storage

capacity of caching nodes, and content popularity distributions. To answer the problem of how to cache, extensive research efforts have been devoted to determining the caching relationship between caching nodes and content files, as well as the management of caching storage resources.

To efficiently utilize the limited caching and communication resources for vehicular content delivery in the caching-assisted HetVNet, caching schemes should be designed specifically for different scenarios. Based on different caching decision-making manners, the caching schemes can be divided into different categories (such as proactive/reactive, centralized/distributed, deterministic/probabilistic) with their corresponding advantages and disadvantages, as summarized in Table 2.2.

There have been extensive research efforts to investigate the content placement problem in terrestrial VNs. Focusing on distributed content storage in V2V scenarios, a dynamic probability caching scheme is introduced by evaluating the community similarity and privacy rating of vehicles in [106]. Then, the caching vehicles are selected based on content popularity to reduce the cache redundancy. In [107], an efficient V2V-based caching strategy is investigated, where each vehicle makes its caching decisions independently by considering the requirements of different types of applications, the crucial features of data, and a set of key attributes of the VNs. In [108], an SDN-based incentive V2V caching scheme is proposed, where a small BS encourages vehicles to cache the popular content by offering them a reward. The single leader multiple followers Stackelberg game is utilized to model the problem and the Stackelberg equilibrium is derived. Targeting only V2I communications, content caching in roadside APs is investigated to maximize content retrievability while considering the limited storage capacities by applying an ILP-based optimization framework in [109] and multi-object auction-based solutions in [110]. Allowing content caching in both vehicles and fixed APs, an edge caching scheme in RSUs is developed in [111] to analyze the vehicular content requests and determine where to obtain the requested content by considering the cooperation between vehicles and RSUs. In [27], assuming that content files can be cached at a macro-cell BS, RSUs, and smart vehicles, the authors propose a cooperative edge caching scheme to jointly optimize the content placement and delivery. The optimization problem is then formulated as a double timescale Markov decision process and solved by the proposed deep deterministic policy gradient (DDPG) based solutions. In addition, a cooperative caching scheme based on mobility prediction is proposed in [112]. The authors adopt a technique called prediction based on partial matching to predict the vehicle trajectories with the aid of fixed RSUs and analyze the expected sojourn time in different hot regions. Then vehicles with a longer sojourn time in hot regions are selected as caching nodes.

Recently content caching in UAVs has attracted wide research interest due to UAVs'

Table 2.2: Content caching solutions summary [1]

Approach	Features	Advantages	Disadvantages
Reactive Caching	Cache content after being requested	Only cache content that is actually requested, which is cost-effective	Additional overhead is required in the initial response
Proactive Caching	Cache content before being requested	Alleviate peak-hour traffic and reduce network latency	Caching performance relies on prediction accuracy
Centralized Caching	Decisions are made by a centralized entity	Achieve better network performance with a global network vision	Single point of failure; Does not scale well
Distributed Caching	Decisions are made distributedly with localized information	Does not require a global processing unit; Easy access to up-to-date local information	May not get the global optimal solutions; High complexity of distributed algorithms and protocols
Deterministic Caching	Content caching is determined based on given information	Better performance in static networks	Not suitable for dynamic situations
Probabilistic Caching	Cache content with certain probabilities	Adaptive to uncertain network status	Hard to obtain the optimal solutions
Non-cooperative Caching	Content caching is decided independently	Low complexity and signaling overhead	May result in inefficient cache utilization
Cooperative Caching	Cache nodes share content with each other	High cache utilization efficiency	High signal overhead by sharing caching status; Additional content retrieving delay
Uncoded Caching	Complete or a part of a file is cached	Easy implementation; Less processing required	Low cache utilization efficiency
Coded Caching	Encode content into packets for caching	Improved reliability; High cache utilization efficiency	Require more computation resources

inherent characteristics of limited wireless backhaul. However, most of the current UAV-aided caching solutions focus on mobile networks with no or low mobility rather than the highly dynamic VNs. However, the ideas behind these solutions are valuable to future related researches in VNs. Therefore, some caching strategies in non-vehicular networks are still analyzed below. In [113], the framework of caching UAV-enabled small cell networks is proposed to alleviate backhaul congestion, reduce energy consumption, and improve the quality of experience (QoE). In [114–116], both the content placement and UAV deployment are optimized. In [114], the problem of proactive deployment of caching-enabled UAVs is

studied to optimize the users' QoE while minimizing the transmit power used by UAVs. In specific, the content placement and the optimal location for each UAV are derived based on the predicted content request distribution, user mobility pattern, and user-UAV associations. In [115], cache-enabled UAVs are adopted to maximize the throughput among IoT devices by optimizing content placement and UAV deployment. In specific, the UAV position is optimized by first determining the best height to maximize the coverage area and then obtaining the optimal 2D position by enumeration search. Then, the caching probability optimization is formulated as a linear problem and solved with Lagrangian function. The joint UAV placement and caching problem is also studied in [116] to maximize the cache hit ratio (CHR) by optimizing the deployment of UAVs, the content caching, and the UAV-user associations. Considering the limited endurance of UAVs, a proactive caching scheme is proposed in [117] to serve ground nodes. An optimization problem is formulated to obtain a trade-off between the file caching cost and file retrieval cost by jointly optimizing the caching policy, the UAV trajectory, and communication scheduling. In [65], the problem of joint caching and resource allocation is investigated for cache-enabled UAVs to jointly optimize user association, spectrum allocation, and content caching with the proposed machine learning framework of the liquid state machine. In [118], a hybrid caching network with UAVs and ground small BSs is developed with probabilistic caching by considering the successful content delivery probability, energy efficiency, and content popularity. Aiming to maximize the spectral efficiency (SE), a hybrid caching strategy is proposed in [119], where a popularity threshold is optimized to divide content into two subsets, popular files that are cached at all the UAVs and less popular files which are cached at only one UAV.

Content caching in satellites was initially proposed based on proxy services, where the content is cached in ground stations with the assistance of satellites. A representative is the cache satellite distribution system (CSDS) [120] in which the proxy caches periodically report to a central ground station about requests received from their clients. The central station then utilizes the satellite broadcast capability to push some Web documents to the participating proxy caches, which caches the documents locally for future local requests. Recently, content placement in satellite networks is primarily considered to be performed with on-board cache [121–125]. In [121], the content placement in LEO satellites is optimized by considering the interactions among distributed satellites for individual content caching decision-making. An exchange-stable matching (ESM) algorithm is proposed to solve the content caching problem based on a many-to-many matching game with externalities. Content caching in satellite-terrestrial networks is considered in [122], where a two-layer caching model is proposed to enable caching in both ground stations and satellites. To minimize the satellite bandwidth consumption, content requests during an aggregation window will be aggregated and then be served by the satellite as the window expires. Uti-

lizing the paradigm of information-centric networking (ICN), in-network content caching is investigated for satellite networks [123]. Considering the specific characteristics of satellite networks, the solutions developed for terrestrial ICNs need to be redesigned, and a novel caching scheme named SatCache is proposed. In [124], a multilayered satellite network is considered, where content caching in LEO and GEO satellites is optimized to realize load balancing. In [125], a CR-based satellite-terrestrial network is considered and the content placement on different satellites is studied to optimize the cache hit rate based on content popularity. Furthermore, the successful download probability is analyzed and two optimal cache space assignment strategies are proposed based on different terrestrial user densities.

2.2.2 Content Delivery Strategies

The design of content delivery strategies aims to effectively and efficiently disseminate requested content files from caching nodes to requesting users. Generally, content delivery solutions can be classified into four different categories [126]: *reverse request path, content announcements, periodic broadcast, and delivery scheduling*. In *reverse request path* schemes, content requests are sent until reaching a caching node that has the required content, and then the requested content is delivered to the requesting node using the reverse path. In *content announcement* schemes, caching nodes announce to their neighbors the content files they have cached, and then vehicles interested in these content files can request directly from these caching nodes. *Periodic broadcast* refers to solutions in which caching nodes periodically and actively broadcast content to the passing vehicles. The *delivery scheduling* schemes usually leverage the expected network topology to schedule a delivery from caching nodes to requesting nodes by considering the expected contact between nodes. In the following, the content delivery strategies are presented in HetVNet scenarios with different network types.

In the terrestrial HetVNet, content files are delivered by V2V communications, V2I communications, or the combination of the two. Focusing only on V2V communications, a cached data transferring scheme is studied in [127] to maintain the survival of the stored data in a designated region. Particularly, to avoid the loss of cached data when caching vehicles leave the region of interests, leaving vehicles push their cached coded data packets to the incoming vehicles it meets via the one-hop V2V link at the entrance/exit of the region. In [128], the authors propose a scheme named ParkCast, in which parked vehicles are grouped into a line cluster with a cluster head assisting file transfer. This scheme involves two types of content delivery: reverse request path and content announcements. The moving vehicles can report their requests and their carrying content files to the cluster head, and then the cluster head decides whether to deliver the requested file to the moving

vehicle and/or selects some parked vehicles to store the content from the moving vehicle. Similarly in [129], parked vehicles are utilized to increase the storage capacity of the content server by using the content files cached in parked vehicles. The authors propose a reverse request path based solution, in which moving vehicles send their interests to other vehicles in a social spot (with parked vehicles), and then the content server sends the content if it is available in the cache. Considering both V2I and V2V communications, the authors propose an in-network caching scheme in VNs in [130], where every vehicle is not only a subscriber to request a file but a cache node to respond to other vehicles' requests. It is assumed that all the vehicles are interested in the same file. Thus the RSU first downloads the file from the server and then broadcasts the file to all the vehicles. Then some vehicles receiving the file can cache this file and help spread it to vehicles that have not received the file through V2V communications. In [131], the authors consider a scenario where vehicles can only download several packets of the entire file which is broadcast by the RSU. The authors propose a coalition formation algorithm to enable cooperative V2V content sharing among different vehicles to complement the missing packets.

When it comes to UAV-aided and satellite-aided content delivery algorithms, most related research works focus on mobile networks with low mobility rather than VNs. However, some of these studies are still discussed in the following because these solutions can provide insight into their applicability for VNs. In [132], an air-ground integrated vehicular architecture is proposed, where high altitude UAVs proactively broadcast popular content to vehicles a prior to requests while the ground RSUs provide services on demand through unicast. In [63], the content files are delivered by ground CBSs or the UAVs if requested. Particularly, the authors propose a cooperative UAV clustering scheme to form UAV clusters to cooperatively deliver the content to ground mobile users to offload traffic from ground cellular networks. A RaptorQ-based content dissemination mechanism is proposed in [133], in which the UAVs are constantly broadcasting content files to ground moving vehicles. Particularly, content files are encoded by using RaptorQ codes to enhance coding performance and energy efficiency. In [134], a centralized UAV-based content delivery scheduling problem is investigated in which a control center is responsible for managing the service delivery. When the control center receives a service request, it will distribute the service request to different regions with different UAVs by considering the distance of delivery, region size, and user's priority. In [135], a cooperative framework is proposed where caching-enabled UAVs and RSUs cooperate to support vehicular content delivery via scheduling and content management.

Focusing on context-aware multimedia content delivery over cooperative satellite-terrestrial networks, authors of [136] point out the potential challenges caused by the inherently different network characteristics, and propose a dynamic spectrum allocation scheme to ably

provide context-aware content files. In [137], the multimedia content delivery over a GEO satellite-terrestrial network is investigated, where multicast multimedia content transmission is optimized via managing the radio resources. In specific, a multicast subgrouping-maximum satisfaction index (MS-MSI) algorithm is proposed, where users are divided into multiple multicast subgroups based on the experienced channel qualities and radio resources are assigned based on subgroup configuration. In [138], layered content delivery is investigated in a content-oriented and satellite-integrated CR network, where content in different layers has different levels of quality representations. A novel state-based and event-driven system model is proposed to analyze the content delivery performance in terms of throughput, energy efficiency, and content quality. In [139], content delivery in high-speed railways over a satellite-terrestrial network is studied. A scheduling and resource allocation algorithm is then proposed to enhance content delivery performance with the prediction of handovers and channel state information.

2.3 Summary

In this chapter, we have surveyed the existing literature for the HetVNet-based vehicular content delivery (in the terrestrial HetVNet, the AGVN, and the SAGVN) and the content caching and delivery strategies in VNs. With the literature review, we desire to achieve a clearer understanding of the limitations and deficiencies in the current studies on this area, and motivate our research works.

Chapter 3

Delay-Minimized Edge Caching in the Terrestrial HetVNet

In this chapter, we investigate content caching in terrestrial HetVNets where Wi-Fi RSUs, TVWS stations, and CBSs are considered to cache content files and provide vehicular content delivery. Particularly, to characterize the intermittent network connection provided by Wi-Fi RSUs and TVWS stations, we establish an on-off model with service interruptions to describe the content delivery process. Content coding then is leveraged to resist the impact of unstable network connections with optimized coding parameters. By jointly considering file characteristics and network conditions, we investigate the content placement in heterogeneous APs to minimize the average content delivery delay, which is formulated as an ILP problem. Adopting the idea of the student admission model, the ILP problem is then transformed into a many-to-one matching problem and solved by our proposed stable-matching-based caching scheme. Simulation results demonstrate that the proposed scheme can achieve near-optimal performances in terms of delivery delay and offloading ratio with low complexity.

3.1 Background and Motivations

To support the tremendous vehicular content delivery with enhanced QoS, the HetVNet can be utilized to exploit the complementary advantages of different RATs and alleviate the access network congestion problem. In addition, edge caching is a promising solution to facilitate content delivery by caching popular content in the HetVNet APs to relieve the

backhaul traffic with a lower delivery delay. Therefore, in this chapter, we focus on edge caching in terrestrial HetVNets to serve the vehicular content requests.

Although caching-assisted content delivery has attracted substantial research interests as mentioned in Chapter 2, the following problems, which are essential in the highly dynamic vehicular networks to provide enhanced and diversified wireless network access for moving vehicles and reduce delivery delay, are insufficiently studied in existing works: 1) in the time-varying and unreliable VNs, content delivery might encounter service interruptions, which significantly affect the caching performance and further the caching policy. However, this inherent vehicular characteristic has not been considered in exiting works on HetVNet caching; 2) most existing works do not take full advantage of the heterogeneous network resources to boost the caching performance gain. Instead, caching content files in multiple types of APs should be considered by taking into account their specific network characteristics including network coverage, network capacity, and AP distributions; and 3) the street layout or vehicle mobility patterns in most existing works are idealized or assumed to follow certain distributions, which is not practical.

In this chapter, we investigate content caching in terrestrial HetVNet APs, i.e., Wi-Fi RSUs, TVWS stations, and CBSs, by taking into account the above-mentioned problems, to provide enhanced and diversified wireless network access for moving vehicles, effectively offload cellular traffic, and reduce delivery delay. Considering the high vehicle mobility and the intermittent network access provided by Wi-Fi and TVWS transmissions, the volume of data that can be transmitted within one coverage area is limited. Therefore, caching the whole content files, especially large files, in the Wi-Fi RSUs or TVWS stations is inefficient since the complete downloading of one file requires multiple encounters with Wi-Fi RSUs or TVWS stations. To improve the caching capability, resist the impact of intermittent network connection, and enhance storage efficiency, coded caching can be applied to encode files into small packets. Each AP only needs to cache the encoded packets with smaller sizes. Recovering the entire file requires downloading a certain number of encoded packets, which may need the cooperation among the APs. Particularly, *Fountain Codes (also known as rateless erasure codes)* [140] are used in this work to encode the files for the following reasons. Firstly, when using a fixed-rate erasure code, a receiver missing a source packet must successfully receive another source packet it has not previously received. This introduces additional overhead when coordinating different APs to serve a moving vehicle. Fountain codes, however, allow receivers to recover the original file by retrieving any subset of encoded packets of size slightly larger than the set of source packets, which is more flexible and reliable with lower communication overhead. Secondly, compared with other codes widely used in distributed data storage systems with $O(K^3)$ complexity, e.g., random linear codes, fountain codes have a superior decoding complexity

of $O(K \ln K)$ [141], where K is the number of source packets to be encoded.

To minimize the content delivery delay for vehicles, we propose a matching-based caching scheme in HetVNets. By modeling the intermittent Wi-Fi/TVWS network connections as on-off service processes, the delivery delay is analyzed by applying partial repeat-after-interruption (PRAI) transmission mode. Based on the delay analysis, the caching placement problem, which is formulated as an ILP problem, is further transformed into a many-to-one preference-based matching problem between the content files and the HetVNet APs. More specifically, by designing the preference lists of content files and HetVNet APs based on file popularity, vehicle mobility, and APs' storage capacities, a student admission (SA) matching-based caching scheme is proposed, which is further solved by leveraging the Gale-Shapley (GS) algorithm [142] to obtain a stable matching. Simulation results show that the proposed scheme can effectively reduce the delivery delay and offload the cellular traffic. The main contributions of this work are listed as follows:

1. By leveraging the interplay between file characteristics and network conditions, the problem of edge caching in multiple types of APs in HetVNets is investigated. Particularly, the dynamics of the content files (e.g., file size and file popularity) and the network connection (e.g., network capacity, AP distribution, and vehicle mobility pattern) are jointly considered in this work. The joint consideration facilitates efficient content caching schemes in the heterogeneous APs to achieve the minimal average delivery delay.

2. Taking into account the inherent time-varying and unreliable characteristics of VNs, we model the intermittent network connections to Wi-Fi RSUs and TVWS stations as on-off service processes with generally distributed on-periods and off-periods. Furthermore, coded caching is leveraged to resist the impact of unstable network connections. The coding parameters are optimized to adapt to the characteristics of different access networks. Then, by applying PRAI transmission mode, the proposed coded caching scheme can achieve a good balance between delivery delay and offloading performance (i.e., the volume of data downloaded without going through backhaul links).

3. The problem of content caching in HetVNets with service interruption is formulated as a many-to-one matching problem and solved by our proposed stable-matching-based caching algorithm. The construction of the two-sided preference lists is multi-objective, considering both the delivery delay and offloading performances. With the carefully designed preferences for content files and the APs, our matching-based caching scheme achieves a good performance with a low time complexity.

4. We carry out extensive experimental results and provide insightful views on the suitability of various caching schemes in different HetVNet scenarios, by comparing multiple performance metrics including delivery delay, offloading ratio, and cache hit rate.

The remainder of this chapter is organized as follows. In Section 3.2, the system model and content delivery scenario in HetVNets are described and the problem formulation is given. In Section 3.3, the detailed derivation of the average content downloading delay from HetVNet APs is analyzed, and the matching-based content placement optimization scheme is described in Section 3.4. Simulation results are carried out in Section 3.5 to demonstrate the performance of the proposed scheme. At last, we conclude this chapter in Section 3.6.

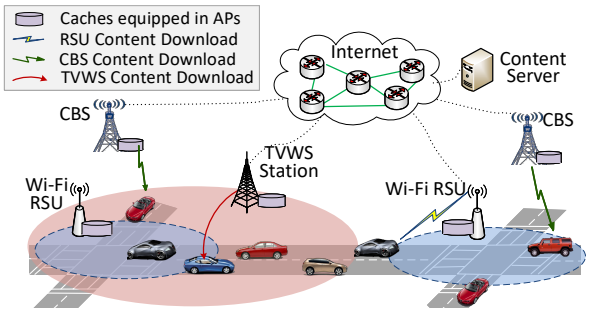
3.2 System Scenario and Problem Formulation

3.2.1 Scenario Description and Assumptions

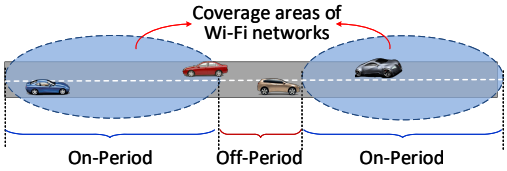
Vehicular users in this work are assumed to be equipped with three radio interfaces for cellular, Wi-Fi, and TVWS communications. Notice that, in addition to Wi-Fi and TVWS based access technologies, there exist many other techniques [143]. Although only Wi-Fi, TVWS, and cellular networks are considered in this work, our methodology and the proposed algorithm are applicable to HetVNets scenarios with more access techniques.

This work studies content caching in HetVNet APs, i.e., Wi-Fi RSUs, TVWS stations, and CBSs, and the communication scenario is depicted in Fig. 3.1(a). Considering urban and sub-urban scenarios where CBSs are densely deployed, we assume that CBSs can provide seamless network connections for vehicles at any time. Content delivery from Wi-Fi RSUs or TVWS stations is available only when vehicles travel through the corresponding coverage areas, as shown in Fig. 3.1(b). The intermittent connections to Wi-Fi/TVWS networks are modeled as on-off processes, which will be introduced in detail in Section 3.2.2. When a vehicle generates content requests, there are two possible cases as shown in Fig. 3.1(c): *1) cache miss* - if the requested files are not cached in the APs, the CBSs can fetch them from the content server via backhaul links and then deliver to the vehicle; and *2) cache hit* - the requested files are cached and the vehicle can download data directly from the APs based on the caching location.

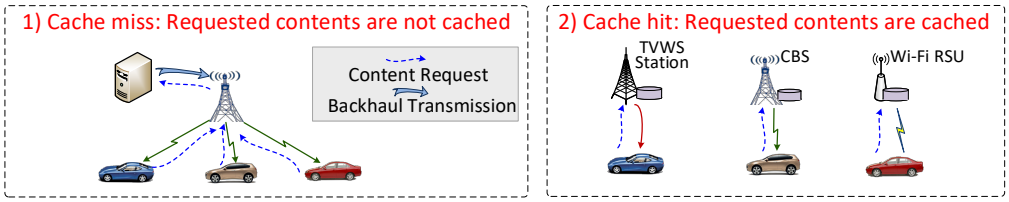
In addition to the CBSs covering the entire target area, there also exist N_T TVWS stations and N_W Wi-Fi RSUs in the scenario. Notations used in this chapter are summarized



(a) Communication scenario with coexistence of cellular, Wi-Fi, and TVWS access networks



(b) An example of on-off service processes of Wi-Fi networks



(c) Content delivery cases: cache miss and cache hit

Figure 3.1: Caching-based content delivery scenario in HetVNet.

in Table 3.1. The TVWS and Wi-Fi coverage radii are denoted by r_T and r_W, respectively. The bandwidth of a Wi-Fi RSU is shared by all vehicles associated to it, i.e., the average transmission data rate of one vehicle equals to $\overline{R}_W = R_W^a/N_W^a$, where R_W^a is the overall achievable aggregate rate and N_W^a is the average number of vehicles associated with one RSU. Likewise, the bandwidths of TVWS stations and CBSs are shared by vehicles associated to the same AP. The average TVWS and CBS transmission data rates are denoted by $\overline{R}_T$ and $\overline{R}_C$, respectively.

Notice that the file popularity distribution has a great impact on the caching performance, including hit ratio, delivery delay, and cellular traffic offloading ratio. Let $\mathcal{F} = \{f_1, \cdots, f_M\}$ be the set of M files and $\mathbf{z}_f = [z_1, \cdots, z_M]$ be the vector representing the sizes of the M content files. All the files in set $\mathcal{F}$ are sorted by descending order based on the file popularity[1]. Thus, f_m is the m-th popular file with a request probability of p_{req}^m. Given the fact that the popularity distribution of the network content items (e.g., YouTube videos) approximately follows Zipf's law, we model the file popularity by the Zipf distribution in this work: for a content file which ranks m, the probability that it is

[1]File popularity can be estimated based on the historical requests and predicted as studied in many existing works (e.g., [20]). Popularity prediction is beyond the scope of this work.

Table 3.1: Summary of Notations

N_T, N_W	Number of Wi-Fi RSUs and TVWS stations, respectively.
r_T, r_W	Coverage radius of Wi-Fi RSUs and TVWS stations, respectively.
$\overline{R}_T, \overline{R}_W, \overline{R}_C$	Average TVWS, Wi-Fi, and CBS transmission rates, respectively.
k_m, α_m	The number of source (or encoded) packets and size of each packet for file f_m, respectively.
K_m	Number of encoded packets required to recover file f_m.
p^m_{req}	Probability that file f_m is requested by vehicles.
p^W_{suc}	Probability that vehicles can successfully download at least one encoded packet from Wi-Fi RSUs.
$p^W_{\max}$	Probability that vehicles can download enough encoded packets from RSUs without wasting time.
$p^W_{\text{on}}(x)$, $p^W_{\text{off}}(x)$	Pmfs of the time length of the on-periods and off-periods for Wi-Fi content downloading.
a^W_m, a^T_m, a^C_m	Indicators showing the caching of file f_m in Wi-Fi RSUs, TVWS stations, and CBSs, respectively.
n^W_m, n^T_m	Number of encoded packets of f_m cached in each Wi-Fi RSU and TVWS station, respectively.
$\overline{D}^W_m, \overline{D}^T_m$, $\overline{D}^C_m, \overline{D}^B_m$	Average download delay of f_m from Wi-Fi, TVWS, CBS, and backhaul delivery, respectively.
C_T, C_W, C_C	Storage capacities of the Wi-Fi RSUs, TVWS stations, and CBSs, respectively.
$\mu^W_{\text{on}}, \mu^W_{\text{off}}$	Average time length in slots for on- and off-periods for Wi-Fi transmission.
σ	Probability that an arbitrary slot is an on-slot.
δ	Probability that on-period continues after an on-slot.
$\mathcal{P}_{\text{files}}(f_m, I)$, $\mathcal{P}_I(I, f_m)$	File f_m's preference over the APs and APs' preference over content files.

requested by vehicles is

$$p^m_{\text{req}} = \frac{1}{m^\xi} \Big/ \left(\sum\nolimits_{k=1}^{M} \frac{1}{k^\xi} \right), \tag{3.1}$$

where the exponent ξ ($\xi \geq 0$) controls the relative popularity of the files, i.e., a larger ξ means that the first few popular files account for the majority of requests.

3.2.2 On-Off Service Model

Recall that Wi-Fi RSUs and TVWS stations provide intermittent network access for drive-thru vehicles. To avoid harmful interference to licensed users, TVWS band can only be accessed by unlicensed users when it is not occupied by incumbent users.[2] Intermittent Wi-Fi and TVWS network services are modeled as on-off processes in this work. For Wi-Fi transmissions, as shown in Fig. 3.1(b), *on-periods* correspond to the time duration when Wi-Fi access is available and *off-periods* appear when the vehicle is not covered by Wi-Fi RSUs. For TVWS transmissions, the off-periods also include the duration when TVWS channels are occupied by incumbent users and not available for secondary access. In this work, we consider a discrete-time system to divide on-periods and off-periods into constant length intervals called slots. Taking Wi-Fi transmission as an example, we define the length of one slot as the time required to transmit one bit of data by Wi-Fi RSUs: $l = 1/\overline{R}_W$. Thus, an on-period with time length of T_{on}^W has T_{on}^W/l slots.

Let $p_{on}^W(x)$ and $p_{off}^W(x)$ be the probability mass functions (pmfs) of the duration of the on-periods and off-periods for the Wi-Fi transmission. Generally, the distributions of the on-off periods are affected by the characteristics of the RSUs (e.g., the deployment density and the coverage radius) and the mobility patterns of the vehicles. The distributions of the on- and off-periods can be obtained by observing the vehicle mobility traces in certain areas, which leads to various distributions in different areas. Alternatively, the distributions can also be assumed to follow geometrical distributions for analysis simplicity. In this work, the on- and off-periods are assumed to be generally distributed, and the scheme proposed in this work can be applied to general cases with any known distributions for the on- and off-periods.

3.2.3 Fountain Coding

In this work, rateless fountain codes are used to encode files cached in TVWS stations and Wi-Fi RSUs due to their good computational efficiency and high flexibility and reliability. In the following, LT (Luby Transform) codes [144], the first proposed fountain codes, are briefly introduced and used in our subsequent discussions and performance evaluation.

[2]To conform to this rule, a database-assisted TVWS network architecture is used according to 802.11af, where a master TVWS device (TVWSD) (i.e., TVWS station in this work) can communicate with the geolocation database to obtain a list of available TVWS channels, while slave TVWSDs (i.e., vehicles in this work) can only access to TVWS channels under the control of the master TVWSD. The TVWSDs need to update the TVWS information subjecting to regulatory constraint, e.g., every 60 seconds according to 802.11af.

When applying LT coding, a source file f_m is divided into k_m source packets $s_1, s_2, \cdots, s_{k_m}$, each of which has a size of $\alpha_m = \frac{z_m}{k_m}$, where z_m is the total size of f_m. Each encoded packet is obtained from the bitwise exclusive-or (XOR) of d randomly and independently chosen source packets, where d is drawn from a degree probability distribution $\Omega(d)$ with $1 \leq d \leq k_m$. In other words, with d obtained from $\Omega(d)$, a vector $(v_1, v_2, \cdots, v_{k_m})$ is generated randomly satisfying $v_i \in \{0, 1\}$ for $i = 1, 2, \cdots, k_m$ and $\sum_{i=1}^{i=k_m} v_i = d$. The encoded packet is $\sum_{i=1}^{i=k_m} v_i s_i$ (bitwise sum modulo 2). From any set of K_m encoded packets (K_m is slightly larger than k_m, which will be explained at the end of this subsection), source file f_m can be decoded with success probability $1-\epsilon$, where ϵ is the decoding failure probability when receiving K_m encoded packets.

Following [144], the degree distribution $\Omega(d)$ follows the *Robust Soliton Distribution*. Let $R \equiv c\sqrt{k_m}\ln(\frac{k_m}{\epsilon})$ for some suitable constant $c > 0$ and $0 < \epsilon \leq 1$. Define

$$\rho(d) = \begin{cases} 1/k_m & \text{for } d = 1 \\ 1/\left[d(d-1)\right] & \text{for } d = 2, \cdots, k_m \end{cases}, \tag{3.2}$$

$$\phi(d) = \begin{cases} R/(k_m d) & \text{for } d = 1, \cdots, \frac{k_m}{R} - 1 \\ R\ln\left(R/\epsilon\right)/k_m & \text{for } d = \frac{k_m}{R} \\ 0 & \text{for } d > \frac{k_m}{R} \end{cases},$$

$$\beta_m = \sum\nolimits_{d=1}^{d=k_m} [\rho(d) + \phi(d)].$$

Then we have $\Omega(d) = [\phi(d) + \rho(d)]/\beta_m$ for $d = 1, \cdots, k_m$. To ensure that the source file can be decoded with a success probability no smaller than $1 - \epsilon$, at least $K_m = k_m\beta_m$ encoded packets should be downloaded. Since $\sum_d \rho(d) = 1$, β_m is always larger than 1. Therefore, the improvement of caching reliability and storage efficiency in Wi-Fi RSUs and TVWS stations are achieved at the expense of total storage space. Considering that data downloading from CBSs is always available, content files cached in the CBSs are stored without coding to avoid unnecessary extra storage occupancy and delivery delay.

3.2.4 Problem Formulation

In this work, content caching in HetVNet APs is investigated to minimize the average content delivery delay. For files with various popularities and data sizes, caching them in different types of APs with diverse coverage ranges, transmission data rates, and availabilities leads to distinct content delivery performances. Therefore, different files are suitable to be cached in different types of HetVNet APs. In this work, we jointly consider the file characteristics and network conditions to facilitate efficient content caching schemes in heterogeneous terrestrial APs to minimize the average delivery delay.

Let a_m^W, a_m^T, and a_m^C indicate the caching of file f_m in Wi-Fi RSUs, TVWS stations, and CBSs, respectively, where

$$a_m^W = \begin{cases} 1, & \text{file } f_m \text{ is cached in Wi-Fi RSUs} \\ 0, & \text{Otherwise} \end{cases},$$

$$a_m^T = \begin{cases} 1, & \text{file } f_m \text{ is cached in TVWS stations} \\ 0, & \text{Otherwise} \end{cases},$$

$$a_m^C = \begin{cases} 1, & \text{file } f_m \text{ is cached in CBSs} \\ 0, & \text{Otherwise} \end{cases}.$$

Notice that, one content file can only be cached in one type of APs in this work for the following reasons: 1) when adopting encoded caching, if a vehicle downloads encoded packets of a file from multiple access networks, the HetVNet APs need to negotiate and keep the same coding parameters to ensure successful content decoding. However, different types of access networks are managed by different service operators, which are competitors in the market and generally do not cooperate or coordinate the caching scheme; and 2) by constraining $a_m^W + a_m^T + a_m^C \leq 1$, the storage efficiency can be improved and more content files can be cached in HetVNet APs to serve more vehicular requests, which facilitates the overall content delivery delay minimization.

The ideal case is that all the files can be cached in the APs to avoid extra backhaul delays, which however is impractical due to limited storage capacities of the APs. Therefore, what kind of content files should be selected for caching and where they should be cached need to be carefully designed to reach an overall low delay. Uncached files can be downloaded from CBSs through backhaul links without coding. Therefore, focusing on the minimization of the average delivery latency for all the files in the library, we formulate our objective function as

$$\min_{\mathbf{A}_T, \mathbf{A}_W, \mathbf{A}_C} \sum\nolimits_{m=1}^{M} p_{req}^m \left(a_m^W \overline{D}_m^W + a_m^T \overline{D}_m^T + a_m^C \overline{D}_m^C + \left(1 - a_m^T - a_m^W - a_m^C\right) \overline{D}_m^B \right) \tag{3.3}$$

$$s.t. \quad \sum\nolimits_{m=1}^{M} a_m^T n_m^T \alpha_m \leq C_T, \tag{3.3a}$$

$$\sum\nolimits_{m=1}^{M} a_m^W n_m^W \alpha_m \leq C_W, \tag{3.3b}$$

$$\sum\nolimits_{m=1}^{M} a_m^C z_m \leq C_C, \tag{3.3c}$$

$$a_m^W + a_m^T + a_m^C \leq 1, \forall m = 1, \cdots, M, \tag{3.3d}$$

$$a_m^W, a_m^T, a_m^C \in \{0, 1\}, \tag{3.3e}$$

where $\mathbf{A}_W = \left[a_1^W, \cdots, a_m^W, \cdots, a_M^W\right]$, $\mathbf{A}_T = \left[a_1^T, \cdots, a_m^T, \cdots, a_M^T\right]$, and $\mathbf{A}_C = \left[a_1^C, \cdots, a_m^C, \cdots, a_M^C\right]$. C_W, C_T, and C_C denote the storage capacities of the Wi-Fi RSUs, TVWS stations, and CBSs, respectively. n_m^T and n_m^W are the numbers of encoded packets of file f_m cached in each TVWS station and Wi-Fi RSU, respectively. $\overline{D}_m^W, \overline{D}_m^T, \overline{D}_m^C$, and $\overline{D}_m^B$ represent the average delays of downloading file f_m from the Wi-Fi, TVWS, CBS, and backhaul transmissions, respectively. Therefore, constraints (3.3a) - (3.3c) indicate that the total size of files cached in the HetVNet APs should not exceed the corresponding maximum storage capacities. Constraint (3.3d) shows that one content file can be cached in at most one type of APs.

3.3 Average Delivery Delay Analysis in HetVNet

To design a caching policy with minimized average content delivery delay, the delay performances of different delivery options (i.e., Wi-Fi, TVWS, CBS, and backhaul transmissions) should be analyzed. Firstly, for files encoded and cached in Wi-Fi RSUs and TVWS stations, the coding parameters are optimized based on the file characteristics and network conditions. Then the PRAI transmission mode is used to deliver the encoded packets and the corresponding average delivery delay is analyzed. For files not cached in Wi-Fi RSUs or TVWS stations, the delays of the CBS and backhaul transmission will also be provided.

3.3.1 Determination of Coding Parameters

Determined by the AP deployment and vehicle mobility patterns, the distributions of the on- and off-periods for Wi-Fi and TVWS transmissions are spatially and temporally variant. For instance, the on-periods in urban scenarios generally sustain longer than in rural areas due to lower vehicle velocity and denser deployment of the APs. Targeting only on the urban scenarios, the distributions of the on-off periods vary in rush hours and in off-peak hours by virtue of different vehicle densities and velocities. The information of these distributions, however, can be gathered by monitoring vehicle mobility traces over a certain area. Generally, the distributions of the on-off periods might change over a day, but regularity can be observed for the same time period (e.g., rush hours) in different days. Without loss of generality, this work assumes that the characteristics of the on-off service processes are known with previous observations. In the following, content download from Wi-Fi RSUs is taken as an example to illustrate the impact of the on-off model on the determination of coding parameters.

Basically, the time length of the on-periods determines the volume of data that can be transmitted within one coverage area. In our coding-based caching scheme, the size of one encoded packet is determined based on the distribution of on-periods to ensure that most vehicles can successfully download at least one packet when traveling through the RSUs' coverage areas. To ensure that vehicles driving through a Wi-Fi coverage area have a probability of at least p_{suc}^W to successfully download at least one packet, we determine the coding parameters k_m^W and α_m^W based on

$$\Pr(T_{on}^W \geq \frac{\alpha_m^W}{\overline{R}_W}) \geq p_{suc}^W \quad \Rightarrow \quad \sum_{T_{on}^W = \alpha_m^W / \overline{R}_W}^{\infty} p_{on}^W(T_{on}^W) \geq p_{suc}^W, \tag{3.4}$$

where T_{on}^W is the length of the on-period.

With any known distribution of the on-periods, we can obtain the upper bound for the value of α_m^W from (3.4), which is denoted by $\alpha_m^{\max}$. Considering that the encoding and decoding complexity of LT codes increase with the value of k_m^W [141], we choose the smallest possible value of k_m^W (largest possible value of α_m^W) as follows.

$$k_m^W = \lceil z_m / \alpha_m^{\max} \rceil, \quad \alpha_m^W = z_m / k_m^W. \tag{3.5}$$

Since vehicles spend different amount of time within different coverage areas, the volume of data downloaded by the vehicles within each RSU varies from one another. Given that fountain codes can generate unlimited number of encoded packets for each file, the number of packets cached in each RSU should be carefully designed. On one hand, a small number of cached packets gives rise to large delivery delays for vehicles spending long time in the coverage area, since they have to wait after downloading all the cached packets in the RSU. On the other hand, caching storage is wasted if too many packets are cached in each RSU while the vehicles can never download so much data within one coverage area.

To achieve a good trade-off between delivery delay and storage efficiency, the number of encoded packets cached in each RSU can be determined based on service requirements. For instance, to ensure that at least $p_{\max}^W \times 100\%$ of the vehicles can download enough number of packets for f_m within one RSU without wasting time, each RSU should cache at least n_m^W packets:

$$\Pr(0 \leq T_{on}^W \leq \frac{n_m^W \alpha_m^W}{\overline{R}_W}) \geq p_{\max}^W \quad \Rightarrow \quad \sum_{0}^{\frac{n_m^W \alpha_m^W}{\overline{R}_W}} p_{on}^W(T_{on}^W) \geq p_{\max}^W. \tag{3.6}$$

With known pmf for on-periods, we can easily obtain the values of α_m^W, k_m^W, K_m^W, and n_m^W. Note that small files (with $k_m^W = 1$ and $\alpha_m^W = z_m$) can be cached without coding to avoid

Algorithm 1: Determination of Coding Parameters in Wi-Fi Transmission

$\mathcal{F}$: Set of all content files. α_m: Size of one encoded packet.
z_m: Size of file f_m. k_m^W: Number of source packets.
K_m^W: Number of encoded packets required to recover file f_m.
n_m^W: Number of encoded packets cached in each Wi-Fi RSU.

```
begin
    for f_m ∈ F do
        Calculate the coding parameters α_m^W and k_m^W based on Eqs. (3.4)∼(3.5).
        if k_m^W = 1 and α_m^W = z_m then
            File f_m has a small file size and can be cached in Wi-Fi RSUs without
            coding.
            n_m^W = 1, K_m^W = 1.
        else
            Obtain the value of n_m^W based on Eq. (3.6) and calculate the value of K_m^W
            based on analysis in Section 3.2.3.
            n_m^W = min{n_m^W, K_m^W}.
        end
    end
    Output: α_m^W, k_m^W, n_m^W, and K_m^W for any f_m ∈ F.
end
```

extra storage occupancy and delivery delay. In addition, when the value of n_m^W calculated from Eq. (3.6) is larger than K_m, then $n_m^W = K_m^W$ since K_m^W encoded packets are sufficient to recover f_m. The detailed procedure of determining the coding parameters is given in **Algorithm 1**. Similar analysis can be applied to coded caching parameter design in TVWS stations.

3.3.2 Effective Service Time

First we define the terms *service time* and *effective service time (EST)*. Service time of a packet is the time required for transmission without interruption. By defining the length of one slot as the time required to transmit one bit by Wi-Fi RSUs, the service time of f_m in Wi-Fi transmission is equal to z_m slots[3]. On the other hand, the EST of delivering content file f_m is defined as the time period between the slot when the file request is generated

[3]Similarly, for TVWS transmission, we can define the length of one slot as the time required to transmit one bit by TVWS station, and the service time of file f_m in TVWS transmission is equal to z_m slots.

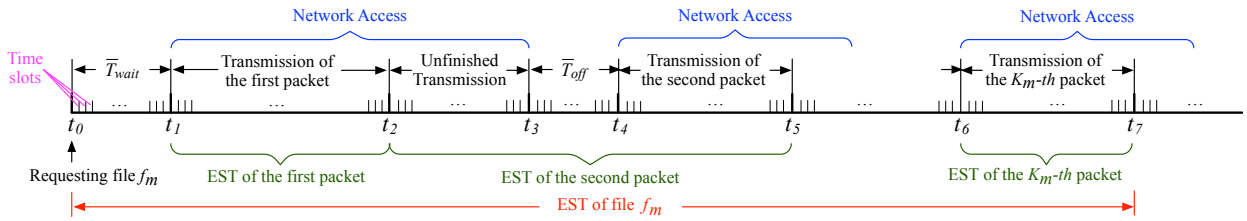


Figure 3.2: Effective service time illustration for PRAI transmission mode.

and the end of the slot when transmission of the K_m-th packet is finished. For the Wi-Fi and TVWS content delivery, the EST includes the periods when content downloading is available and the time duration when the service is interrupted. In the following, the EST of content delivery is analyzed by taking the Wi-Fi content delivery as an example.

Considering that content files are encoded using LT codes and then cached in Wi-Fi RSUs, continuous-after-interruption (CAI) transmission mode is not suitable since the encoded data packets cached in different RSUs are different. Thus, PRAI transmission mode is adopted in this work. When delivering an encoded packet, if it is not finished before service interruption, this packet will be dropped and a new encoded packet of the same content file needs to be transmitted when the vehicle encounters another available Wi-Fi coverage in PRAI.

Taking the illustration in Fig. 3.2 as an example, a vehicle requests file f_m with size z_m at time t_0 and content delivery starts as soon as the network access is available at time t_1. Therefore, if the vehicle is not covered by a Wi-Fi RSU when generating the file request, it has to wait for $t_1 - t_0$ to get served; otherwise, $t_1 = t_0$. After successfully downloading the first packet of f_m, the transmission of the second packet is interrupted when the vehicle leaves the Wi-Fi coverage area. When entering the coverage area of another Wi-Fi RSU, the second packet (not necessarily the same as the unfinished one) needs to be re-transmitted. Thus, the EST of the second packet is $t_5 - t_2$. Correspondingly, the EST of file f_m is $t_7 - t_0$, which includes the waiting period $t_0 \sim t_1$ and the ESTs of K_m packets.

3.3.3 Average Delay of Wi-Fi and TVWS Delivery

Let μ_{on}^W and μ_{off}^W denote the average time length in slots for on- and off-periods. The probability that an arbitrary slot is an on-slot, denoted by σ, can be expressed as

$$\sigma = \mu_{on}^W / \left(\mu_{on}^W + \mu_{off}^W\right). \tag{3.7}$$

For a randomly generated file request, if the vehicle is out of the Wi-Fi service range, it has to wait for a certain time to get served. Denoting by T_{off}^W the length of the off-period in slots with mean μ_{off}^W, the average waiting time slots can be obtained by:

$$\begin{aligned}\overline{T}_{wait}^W &= E\left\{(1-\sigma)\cdot\sum_{x=1}^{T_{off}^W}\frac{1}{T_{off}^W}x\right\} \\ &= (1-\sigma)\cdot E\left\{\frac{1}{T_{off}^W}\cdot\frac{T_{off}^W(T_{off}^W+1)}{2}\right\} \\ &= \frac{(1-\sigma)}{2}(\mu_{off}^W+1).\end{aligned} \tag{3.8}$$

Denote by $\mathbb{A}$ the event that an on-period continues after an on-slot. Let δ denote the probability that event $\mathbb{A}$ happens and T_{on}^W denote the duration of on-periods with mean value μ_{on}^W. We have:

$$\begin{aligned}\delta = \Pr(\mathbb{A}) &= \sum_{x=0}^{\infty}\Pr(\mathbb{A}|T_{on}^W=x)\times\Pr(T_{on}^W=x) \\ &= \sum_{x=0}^{\infty}\frac{x-1}{x}\times p_{on}^W(x) = 1-E\left\{\frac{1}{T_{on}^W}\right\},\end{aligned} \tag{3.9}$$

which can be easily calculated with known $p_{on}^W(x)$.

To obtain the EST of the files, we first calculate the EST of one encoded packet. Referring to the repeat-after-interruption mode in [145], let $s_{n,\ell}^W(x)$ denote the probability that the remaining EST of a packet with size n bits equals x slots given that the remaining service time is ℓ slots and that the slot preceding the remaining EST is an on-slot. Thus $s_{n,\ell}^W(x)=0$ for $x<\ell$ and

$$s_{n,\ell}^W(x) = \delta s_{n,\ell-1}^W(x-1) + (1-\delta)\sum\nolimits_{j=1}^{\infty}p_{off}^W(j)s_{n,n-1}^W(x-j-1), \tag{3.10}$$

which is obtained based on the on-off state of the first slot of the remaining EST.

Let $S_{n,\ell}^W(z)$ and $P_{off}^W(z)$ be the probability generating functions (pgfs) of $s_{n,\ell}^W(x)$ and $p_{off}^W(x)$, respectively, i.e.,

$$P_{off}^W(z) = \sum\nolimits_{x=0}^{\infty}p_{off}^W(x)z^x, \quad S_{n,\ell}^W(z) = \sum\nolimits_{x=0}^{\infty}s_{n,\ell}^W(x)z^x. \tag{3.11}$$

Thus, we have the following lemma:

Lemma 1. *The average EST (in slots) of delivering a packet with size n bits through Wi-Fi transmission, denoted by $\overline{T}_n^W$, is*

$$\overline{T}_n^W = \frac{\delta}{1-\delta}\left(1+(1-\delta)\mu_{off}^W\right)\left(\frac{1}{\delta^n}-1\right). \tag{3.12}$$

Proof of Lemma 1. According to Eqs. (3.10) and (3.11), the pgf of $s_{n,\ell}^W(x)$ is

$$\begin{aligned}
S_{n,\ell}^W(z) &= \sum_{x=1}^{\infty}\delta s_{n,\ell-1}^W(x-1)z^x + \sum_{x=1}^{\infty}(1-\delta)\sum_{j=1}^{\infty}p_{off}^W(j)s_{n,n-1}^W(x-j-1)z^x \\
&= \delta z\sum_{x=0}^{\infty}s_{n,\ell-1}^W(x)z^x + (1-\delta)z\sum_{x=1}^{\infty}\sum_{j=1}^{\infty}p_{off}^W(j)z^j s_{n,n-1}^W(x-j-1)z^{x-j-1} \\
&= \delta z S_{n,\ell-1}^W(z) + (1-\delta)zP_{off}^W(z)S_{n,n-1}^W(z).
\end{aligned}$$

For notational simplicity, let $\zeta = (1-\delta)zP_{off}^W(z)$. Thus we have

$$S_{n,\ell}^W(z) = \delta z S_{n,\ell-1}^W(z) + \zeta S_{n,n-1}^W(z).$$

By substituting different values for ℓ, we have:

- $\boldsymbol{\ell = n}$**:** $S_{n,n}^W(z) = \delta z S_{n,n-1}^W(z) + \zeta S_{n,n-1}^W(z) \quad\Rightarrow\quad S_{n,n}^W(z) = (\delta z+\zeta)\,S_{n,n-1}^W(z)$;
- $\boldsymbol{\ell = n-1}$**:** $S_{n,n-1}^W(z) = \delta z S_{n,n-2}^W(z) + \zeta S_{n,n-1}^W(z) \quad\Rightarrow\quad S_{n,n-1}^W(z) = \frac{\delta z}{1-\zeta}S_{n,n-2}^W(z)$;
- $\boldsymbol{\ell = n-2}$**:** $S_{n,n-2}^W(z) = \delta z S_{n,n-3}^W(z) + \zeta S_{n,n-1}^W(z) \quad\Rightarrow\quad S_{n,n-1}^W(z) = \frac{(\delta z)^2}{1-\zeta-\zeta\delta z}S_{n,n-3}^W(z)$;
- $\boldsymbol{\ell = n-3}$**:** $S_{n,n-3}^W(z) = \delta z S_{n,n-4}^W(z) + \zeta S_{n,n-1}^W(z)$

$$\Rightarrow\quad S_{n,n-1}^W(z) = \frac{(\delta z)^3}{1-\zeta-\zeta\delta z-\zeta(\delta z)^2}S_{n,n-4}^W(z);$$

...

By deductive proof, we have

$$\begin{aligned}
S_{n,n-1}^W(z) &= \frac{(\delta z)^{n-1}}{1-\zeta\sum_{j=0}^{n-2}(\delta z)^j}S_{n,0}^W(z) = \frac{(\delta z)^{n-1}(1-\delta z)}{1-\delta z-\zeta[1-(\delta z)^{n-1}]}, \\
S_{n,n}^W(z) &= (\delta z+\zeta)\,\frac{(\delta z)^{n-1}(1-\delta z)}{1-\delta z-\zeta[1-(\delta z)^{n-1}]} \\
&= \frac{\left(\delta z+(1-\delta)zP_{off}^W(z)\right)(\delta z)^{n-1}(1-\delta z)}{1-\delta z-(1-\delta)zP_{off}^W(z)\left[1-(\delta z)^{n-1}\right]}.
\end{aligned}$$

Notice that $S^W_{n,0}(z) = 1$ because if there are no more bits to send, the downloading process ends in the current slot with probability 1. Based on the pgf's moment-generating property, the average EST $\overline{T}^W_n$ (in slots) of transmitting a packet with size n bits can be obtained by

$$\begin{aligned}\overline{T}^W_n &= \left.\frac{dS^W_{n,n}(z)}{dz}\right|_{z=1} = \left(\frac{\delta}{1-\delta} + \delta\mu^W_{off}\right)\left(\frac{1}{\delta^n} - 1\right) \\ &= \frac{\delta}{1-\delta}\left(1 + (1-\delta)\mu^W_{off}\right)\left(\frac{1}{\delta^n} - 1\right),\end{aligned}$$

which concludes the proof. □

Next we analyze the delivery delay of content file f_m with K^W_m packets, each of which is of size α^W_m. After waiting for $\overline{T}^W_{wait}$ slots, the following slot is an on-slot which can serve one unit of data. Then, the EST of transmitting the remaining $\alpha^W_m - 1$ units of the first packet can be obtained by replacing n by $\alpha^W_m - 1$ following **Lemma 1**. After transmitting the first packet, the remaining $K^W_m - 1$ packets' delivery is preceded by an on-slot as the last slot of each packet's service is clearly an on-slot. Therefore, each of the remaining $K^W_m - 1$ packets has an EST of $\overline{T}^W_{\alpha_m}$ slots. Thus, the average EST (i.e., delivery delay) of file f_m, denoted by $\overline{D}^W_m$, is expressed as:

$$\overline{D}^W_m = (\overline{T}^W_{wait} + 1 + \overline{T}^W_{\alpha^W_m - 1} + (K^W_m - 1)\overline{T}^W_{\alpha^W_m}) \times l. \tag{3.13}$$

As shown in the equation above, the EST of a content file is determined by the on-off distribution (affected by AP distributions and vehicle mobility patterns), network capacity, and the content file size. The same analysis procedure can be applied when calculating $\overline{D}^{TV}_m$, the average delay of downloading file f_m from TVWS stations, with different distributions of the on-off periods and network capacities.

3.3.4 Delivery Delay of Cellular Downloading

Recall that CBSs can provide seamless coverage for driving vehicles and uncoded caching scheme is applied for CBS caching. When a file is not completely delivered within the range of one CBS, a CAI transmission mode can be applied when a vehicle travels through multiple CBSs. With uncoded caching and CAI transmission mode, no re-transmission is required and the average delay of downloading a file f_m from CBSs can be expressed as

$$\overline{D}^C_m = z_m/\overline{R}_C. \tag{3.14}$$

For content files not cached in the HetVNet APs, they should be served by the CBS through backhaul links. Without loss of generality, CBSs are connected to the core network with wired backhaul links. Referring to [146], one hop can be assumed for the wired backhauls. Thus, the average delay and for transmitting file f_m with size z_m through wired backhaul links is:

$$\overline{D}_m^B = \overline{D}_m^C + \left(\left(1 + 1.28\frac{\lambda_b}{\lambda_g}\right)\kappa\right) \cdot (a + bz_m), \tag{3.15}$$

where the second term is the backhaul transmission delay, λ_b and λ_g are the densities of CBSs and gateways, respectively. a, b, and κ are constants that reflect the processing capability of the nodes[4].

3.4 Matching-Based Content Caching Scheme

In this section, the content placement is optimized to minimize the overall content delivery delay based on the optimized coding parameters and the average delivery delay analysis given in Section 3.3. Note that (3.3) is an ILP problem, and the optimal solution can be obtained by using ILP algorithms, e.g., branch and bound (B&B) algorithm. Considering the high time complexity of the B&B algorithm, a more efficient way to solve this problem is needed.

Construct a weighted bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{I}, \mathcal{E})$, where $\mathcal{V}$ is the set of content files, $\mathcal{I}$ is the set of HetVNet APs, and $\mathcal{E}$ is the set of edges connecting vertices in $\mathcal{V}$ and $\mathcal{I}$. Each edge has a weight which can be defined related to the content delivery delay as analyzed in Section 3.3. Therefore, the content placement problem is actually a weighted bipartite b-matching problem, i.e., to find a subgraph $\mathcal{A} \subset \mathcal{G}$ to minimize the overall delay such that each vertex in $\mathcal{A}$ has at most b edges. Specifically, each file has $b = 1$, but the values of b for the APs are unknown for two reasons: 1) HetVNet APs have different storage capacities; and 2) caching different files in the same AP requires different storage sizes. Therefore, b-matching algorithms are not suitable for the content placement optimization. In this work, we adopt the idea of the SA model and apply GS-based stable matching algorithm to allocate content files to the HetVNet APs.

[4]The exact values of the above parameters can be obtained through fitting using real measurements, which goes beyond the scope of this work.

3.4.1 Complexity Analysis

As stated above, the optimal solution to the caching placement problem can be found by applying the B&B algorithm. Although the B&B algorithm is guaranteed to find the optimal solution, its worst-case time complexity is as high as that of brute-force exhaustive search. In this work, considering that each file has four possible states (i.e., cached in Wi-Fi RSUs, TVWS stations, CBSs, or uncached), the worst-case complexity of the B&B algorithm is $\mathcal{O}(4^M)$, where M is the total number of content files. Although the practical searching times is not as large as 4^M in most cases, and the average complexity of the B&B algorithm can be reduced to be polynomial under some conditions, there is no complexity guarantee and the effectiveness of B&B is still limited by the potential exponential growth of the execution time as a function of problem size.

SA model solves the matching between students and colleges which have limited quotas, based on two-sided preferences. By adopting the SA model in this work, we can map content files to be students and the HetVNet APs to be colleges. The content placement problem can then be formulated as a many-to-one matching problem between content files and the HetVNet APs. That is, one file can only be cached in one type of APs, while one type of APs can cache multiple files up to its quota (i.e., storage capacity). Then, the GS algorithm can be leveraged to solve this matching problem with a much lower time complexity, which is $\mathcal{O}(4 \times M)$.

3.4.2 Preference Lists

The matching between files and the caching APs is processed based on the two-sided preferences, the construction of which can significantly affect the matching results and further the caching performance gain. Basically, the two-sided preference lists should be defined highly related to, but not always exactly the same as the optimization objective. In this work, a multi-objective construction of the preference lists is considered, by using two different metrics when designing the preference lists for content files and the APs.

Recall that our optimization objective is to minimize the overall delivery delay for all files. Thus, the preferences of content files over the HetVNet APs can be measured by the average delivery delay. Specifically, file f_m's preference over the APs is expressed as

$$\mathcal{P}_{files}(f_m, I) = \overline{D}_m^I, \tag{3.16}$$

where I refers to different ways to download file f_m, i.e., Wi-Fi RSUs, TVWS stations, and CBS transmissions. In other words, $\mathcal{P}_{files}(f_m, \text{Wi-Fi}) = \overline{D}_m^W$, $\mathcal{P}_{files}(f_m, \text{TVWS}) = \overline{D}_m^T$, and

$\mathcal{P}_{files}(f_m, \text{CBS}) = \overline{D}_m^C$. Basically, it is preferred that a content file is cached in the type of APs leading to the lowest delivery delay. Thus, by sorting the elements in $\mathcal{P}_{files}(f_m, I)$ in ascending order, the first type of APs in f_m's preference list is the most preferred APs for caching f_m.

When designing preference lists for the APs, however, we do not prioritize the content files based on the delivery delay. Since the content delivery delay is largely dependent on file size, using delay as a metric leads to an intuitive result that all the APs prefer to cache small files, regardless of the file popularity. To address this issue, in this work, we define a new metric to rank the files based on file popularity and the volume of data that can be offloaded from the backhaul traffic. In other words, APs prefer to cache files that have a higher request probability with a larger requested data size. By caching this kind of files, the APs can leverage their storage capacities more efficiently and offload more traffic with lower delivery latency. Thus, the APs ' preferences over file f_m are measured by

$$\mathcal{P}_I(I, f_m) = p_{\text{req}}^m \cdot \alpha_m^I \cdot K_m^I, \tag{3.17}$$

where the definition of I is the same as in (3.16). Thus, by sorting the elements in $\mathcal{P}_I(I, f_m)$ in descending order, the first file in each type of APs' preference list is file f_m leading to the maximum average offloading data size.

3.4.3 Matching-Based Content Placement Policy

In this subsection, we illustrate the caching placement scheme in HetVNets with on-off service model. The details are summarized in **Algorithm 2**. Firstly, for every content file f_m, the average delay performance is analyzed for all the HetVNet APs, based on which the preference lists are constructed as discussed in (3.16) and (3.17). After that, the GS algorithm is exploited to solve the SA-based many-to-one matching problem between the content files and the APs. The matching process can be described as follows:

Step 1: Each content file proposes to its current most favorite caching APs and then removes this type of APs from its preference list.

Step 2: Each type of APs check all the received proposals from the files, including both the new proposals and those accepted in former iterations, and then accept the most preferred files within the storage capacity constraint and reject the rest.

Step 3: For all the rejected files, go to **Step 1**. The matching process terminates when all the files are successfully cached or all the APs' storage capacities are occupied.

As a summary, **Algorithm 3** shows the overall process of our proposed matching-based content caching optimization scheme with the on-off service model.

Algorithm 2: Matching-Based Caching Optimization Algorithm

$\mathcal{F}, \mathcal{F}_u$: Sets of all the content files and unmatched content files, respectively.
z_m: Size of content file f_m. $\mathcal{P}_{files}(f_m, I)$: Preference lists of content files.
$\mathcal{P}_I(I, f_m)$: Preference lists of HetVNet APs.
C_T, C_W, C_C: Storage capacities of Wi-Fi RSUs, TVWS stations, and CBSs.
a_m^W, a_m^T, a_m^C: Indicators showing the caching of file f_m in Wi-Fi RSUs, TVWS stations, and CBS, respectively.

begin
  Initialize $\mathcal{F}_u = \mathcal{F}$.
  repeat
    for $f_m \in \mathcal{F}_u$ **do**
      Propose to the first type of APs I in its preference list $\mathcal{P}_{files}(f_m, I)$.
      Set $a_m^I = 1$ ($a_m^I \in \{a_m^W, a_m^T, a_m^C\}$) and remove I from $\mathcal{P}_{files}(f_m, I)$.
    end
    for $I \in$ *Wi-Fi RSUs, TVWS stations, CBSs* **do**
      $S_{I,req}^m = z_m$ if I is CBS; otherwise, $S_{I,req}^m = \alpha_m^I \cdot n_m^I$.
      if $\sum_{m \in \mathcal{F}} (a_m^I \cdot S_{I,req}^m) \leq C_I$ **then**
        I keeps all the proposing files and removes accepted files from $\mathcal{F}_u$.
      else
        I keeps the most preferred files under storage capacity constraint and rejects the rest;
        Remove these accepted files from $\mathcal{F}_u$.
        For the rejected files, set $a_m^I = 0$ and add them into $\mathcal{F}_u$.
      end
      $C_{I,remain} = C_I - \sum_{m \in \mathcal{F}} (a_m^I \cdot S_{I,req}^m)$
    end
  until $\mathcal{F}_u = \emptyset$ *or* $C_{I,remain} \leq \min_{f_m \in \mathcal{F}_u} S_{I,req}^m, \forall I$;
  Output: a_m^W, a_m^T, and a_m^C for any $f_m \in \mathcal{F}$.
end

3.5 Performance Evaluation

A. Simulation Setting

We conduct simulations based on the real scenario of University of Waterloo campus. The campus map is shown in Fig. 3.3a and the main roads are drawn in Fig. 3.3b. Vehicles in the target region can always access to the CBS, which is marked as the pink triangle in

Algorithm 3: Overall Process of the Proposed Matching-Based Caching Scheme

begin

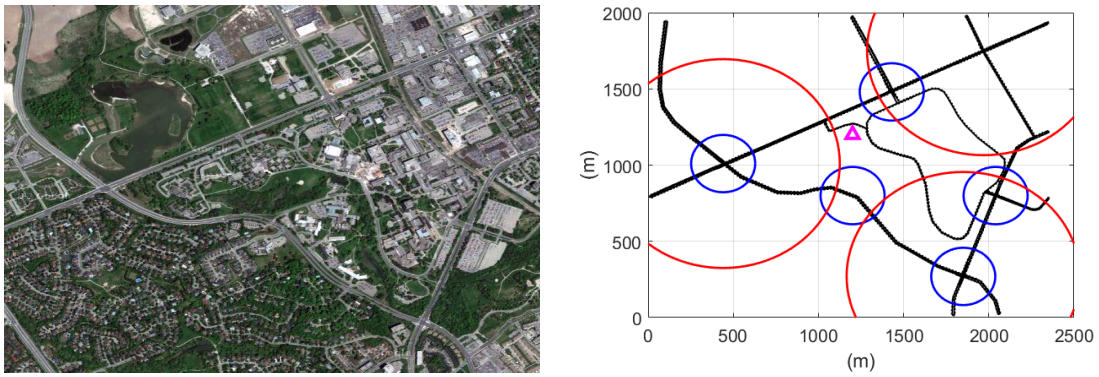
Step 1: Determine the coding parameters according to **Algorithm 1**.

Step 2: Analyze the content delivery delay of the coded caching scheme in HetVNets with service interruption based on **Sections 3.3.3 and 3.3.4**.

Step 3: Construct multi-objective two-sided preference lists based on **Section 3.4.2**.

Step 4: Optimize content placement according to the matching-based algorithm in **Algorithm 2**.

end



(a) Map (b) AP deployment

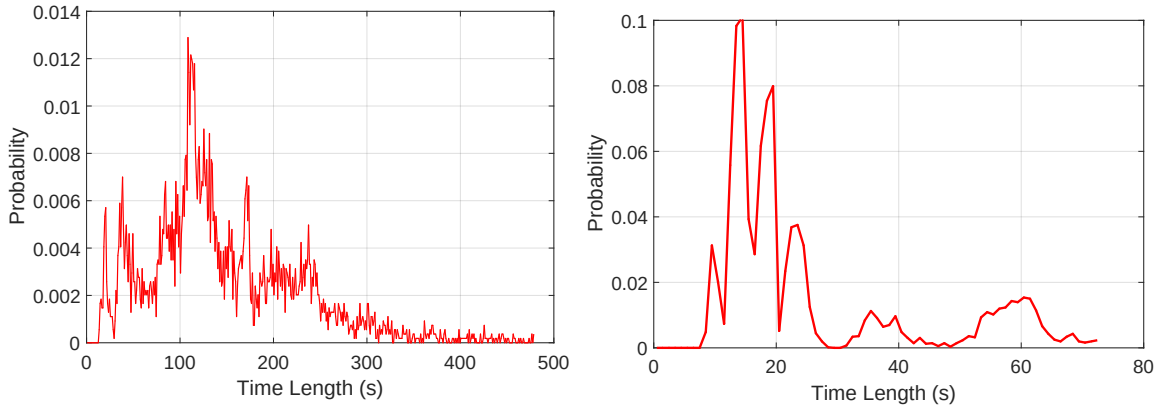
Figure 3.3: Simulation settings.

Fig. 3.3b. Blue circles in Fig. 3.3b represent the coverage areas of five Wi-Fi RSUs, and the red circles are the coverage areas of three TVWS stations. To simulate realistic vehicle traffic, we use VISSIM simulation tool to generate the traffic of 200 vehicles in the campus scenario. The content file size is within the range of [0 MB, 1000 MB]. The file popularity follows Zipf distribution with exponent $\xi = 0.7$. The default values of main simulation parameters are listed in Table 3.2 unless otherwise specified.

Recall that we assume known distributions of the on-off periods in this work. Considering that the time length of the on-off periods might not follow any well-known distributions (e.g., exponential distribution and normal distribution) in practice and hence a general distribution is assumed for the on-off service models. In our simulation, based on the mobility

Table 3.2: Simulation Parameters

$[r_W, r_T]$: Coverage radii of Wi-Fi RSUs and TVWS stations	[150, 600] m
$[R^a_W, R^a_T, R^a_C]$: Aggregate rates of a Wi-Fi RSU, TVWS station, and CBS	[65, 54, 128] Mbps
c and ϵ: Constant in Robust Soliton Distribution and decoding failure probability	0.1 and 0.05
p^W_{suc} and p^T_{suc}: Probabilities that vehicles can download at least one packet from Wi-Fi RSUs and TVWS stations	0.99
$p^W_{\max}$ and $p^T_{\max}$: Probability that vehicles can download enough packets from Wi-Fi RSUs and TVWS stations without waiting	0.9
$[C_W, C_T, C_C]$: Storage capacities of a Wi-Fi RSU, a TVWS station, and a CBS	[10, 10, 20] GB



(a) On-periods

(b) Off-periods

Figure 3.4: Distributions of on-off periods for TVWS transmission.

traces generated by VISSIM and the deployment of the APs, we can easily obtain the time vehicles spend in each kind of APs, thus distributions of the on-off periods can be acquired. For instance, our simulation uses Matlab to process the vehicle trace data, and the distributions of the on-off service periods for TVWS transmission are shown in Fig. 3.4. Note that the time length of the on- and off-periods is affected by factors including but not limited to the distance between the TVWS stations, the road layout, and the

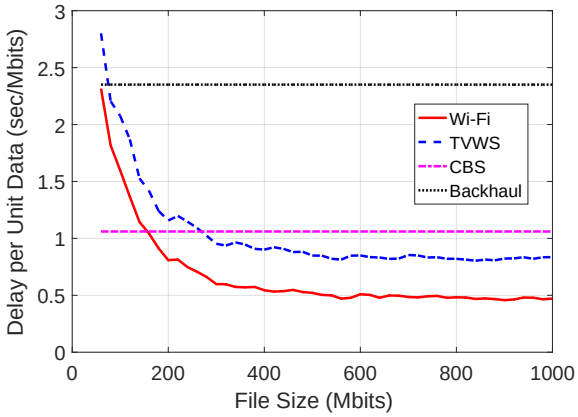


Figure 3.5: Average delay per unit data vs. file size

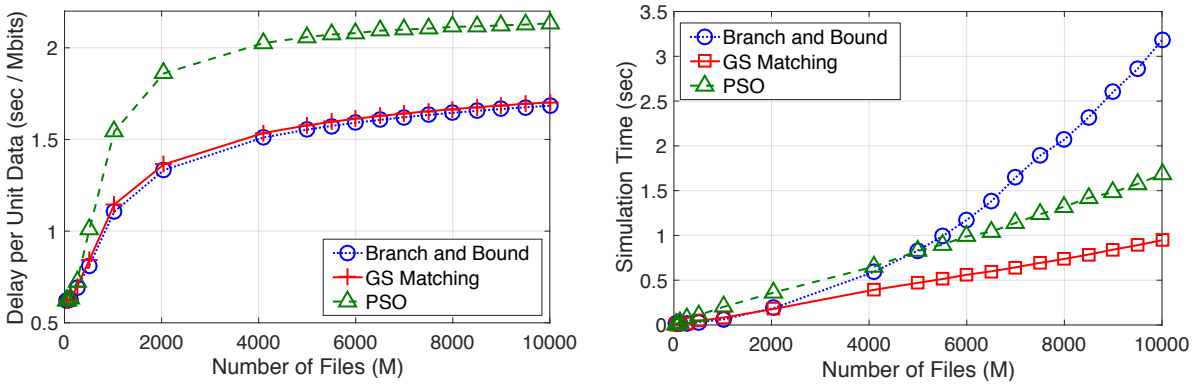
average vehicle velocity. Therefore, the distributions vary in different target regions with various AP deployments. Simulations in this work leverage the on-off TVWS service time distributions in Fig. 3.4, and the on-off Wi-Fi service time distribution can be obtained in a similar way. However, the caching scheme proposed in this work can be applied to scenarios with any other well-known distributions for the on-off periods.

To evaluate the delay performance, we monitor all the vehicles in the target region, which generate content requests based on the file popularity distribution. Then a vehicle is randomly chosen at a random time instant and its data downloading performance is observed. All the following simulation results are averaged over 1000 trials.

B. Impact of File Size

Fig. 3.5 shows the impact of file size on the average delay performance[5] of downloading files from Wi-Fi RSUs, TVWS stations, CBS, or backhaul transmission. Since the average transmission rates of the CBS and backhaul delivery are mainly determined by the number of vehicles sharing the spectrum and the deployment of CBSs, the average delays of these two kinds of transmissions keep unchanged with increasing file size, as shown in Fig. 3.5. The average delays per unit data for Wi-Fi and TVWS transmissions decline with a larger file size, and the former has a better delay performance than the latter. For small-size files, Wi-Fi and TVWS transmissions have poor delay performance. The reason is that,

[5]Given that overall content delivery delay is significantly affected by file sizes, average delay per unit data (sec/bit), which is the ratio of the overall delay defined in (3.3) over the total size of the requested data, is adopted in the simulation to better illustrate the content delivery delay performance.



(a) Average delay per unit data

(b) Time complexity comparison

Figure 3.6: Delay and complexity performance comparison between B&B, PSO, and GS matching algorithms.

when compared to the average waiting time, the required service time for small files plays a minor part in the EST in our on-off service models, therefore leading to a large delay per unit data. When file size grows, an increasing part of the EST comes from the service time rather than the time wasted in waiting for service. Finally, the average delay per unit data converges to a constant value which is determined by the average Wi-Fi/TVWS transmission rate and the ratio of the average time length of on-periods over that of off-periods. Therefore, it is advisable that small files (e.g., texts or pictures) are cached in the CBS without coding, while large files such as movies and high definition maps should be cached in Wi-Fi RSUs or TVWS stations with coding to achieve performance improvement.

C. Tradeoff between Delay Performance and Complexity

Fig. 3.6 shows the impact of the number of content files on the achievable delay performance and complexity of the algorithms. In addition to the B&B and our matching-based algorithm, one evolutionary algorithm, the particle swarm optimization (PSO) algorithm is also included in the performance comparison to further illustrate the effectiveness and efficiency of our proposed scheme. Intuitively, with more content files in the network, the delivery delay per unit data and the time complexities of all the three algorithms increase. As shown in Fig. 3.6a, the delivery delay increases because more files need to be fetched from backhaul transmission due to limited storage capacities. The delay performance of the B&B algorithm outperforms that of the GS matching-based algorithm, while the per-

formance gap is insignificant. On the other hand, compared to the B&B algorithm which has exponentially increased complexity over the network size, the GS-based matching algorithm is significantly less time-consuming, which can be seen in Fig. 3.6b, especially when the system scales. The PSO algorithm, as shown in the figure, achieves a larger delivery delay with a longer simulation time when compared with the proposed matching-based algorithm. The delay performance of the PSO algorithm can be further improved, while the corresponding simulation time will also increase significantly. Therefore, the proposed algorithm is a favorable choice to reduce the time complexity with modest delay performance loss, especially in complex or heterogeneous networks with a large number of files.

D. Coded Caching vs. Uncoded Caching

Recall that content files are encoded and cached in the Wi-Fi RSUs and TVWS stations. If the transmission of an encoded packet is interrupted, a PRAI transmission mode is adopted as explained in Section 3.3.2. In contrast, uncoded caching scheme indicates that files are cached entirely in APs and the CAI transmission mode can be applied when the vehicle travels through multiple APs. With uncoded caching and CAI transmission mode, one may naively believe that the delivery delay can be reduced since no re-transmission is required. However, uncoded caching leads to a low storage efficiency, which further affects the overall delay and offloading performance. Therefore, in this part, we compare performances of the coded caching and uncoded caching schemes to dispel any wishful thinking.

Similar to the analysis in Section 3.3.3, the EST of transmitting a file with size z_m by using the CAI transmission mode is analyzed as follows (taking Wi-Fi transmission as an example). Given that the slot preceding the effective service time is an on-slot, the probability that the EST of a packet of length n bits equals ℓ slots is:

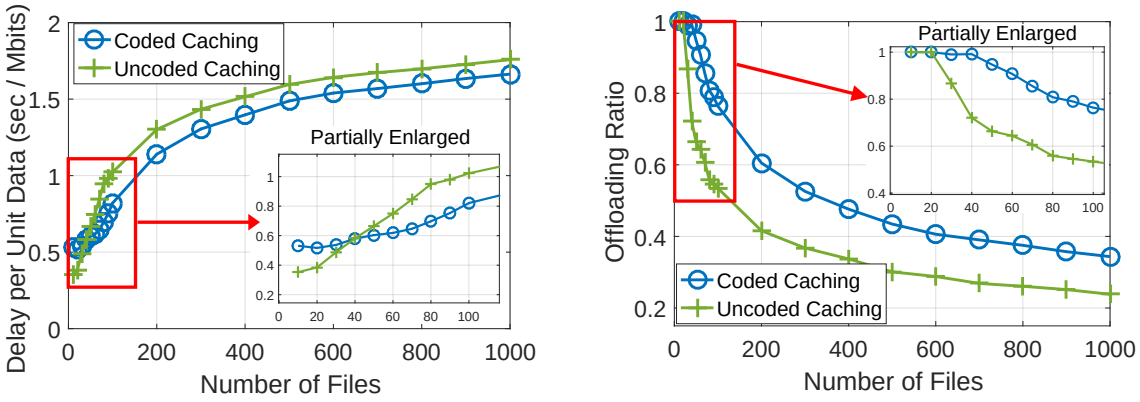
$$s_n^{\mathrm{W,CAI}}(\ell) = \delta s_{n-1}^{\mathrm{W,CAI}}(\ell-1) + (1-\delta)\sum\nolimits_{j=1}^{\infty} p_{off}^{W}(j)\, s_{n-1}^{\mathrm{W,CAI}}(\ell-j-1), \tag{3.18}$$

The corresponding pgf of $s_n^{\mathrm{W,CAI}}(\ell)$ is:

$$\begin{aligned} & S_n^{CAI}(z) = \left[\delta z + (1-\delta) z P_{off}^{W}(z)\right] S_{n-1}^{CAI}(z) \\ \Rightarrow \quad & S_n^{CAI}(z) = \left[\delta z + (1-\delta) z P_{off}^{W}(z)\right]^n \end{aligned} \tag{3.19}$$

Therefore, the EST of transmitting a file with size z_m in CAI mode is:

$$\begin{aligned} \overline{D}_m^{\mathrm{W,\ CAI}} &= \left(\overline{T}_{wait}^{\mathrm{W,\ CAI}} + 1 + \left. \frac{dS_{\alpha_m-1}^{CAI}(z)}{dz} \right|_{z=1} \right) \times l \\ &= \left(\frac{(1-\sigma)\mu_{off}^{W}}{2} + 1 + (z_m - 1)\left[1 + (1-\delta)\mu_{off}^{W}\right] \right) \times l. \end{aligned} \tag{3.20}$$



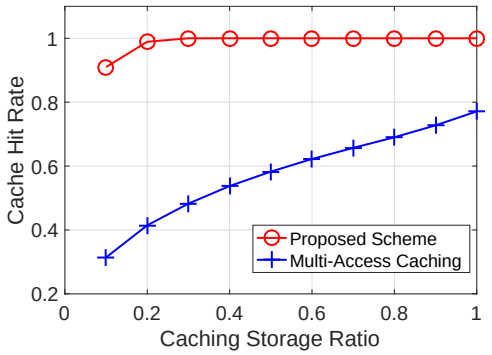
(a) Average delay per unit data

(b) Offloading performance

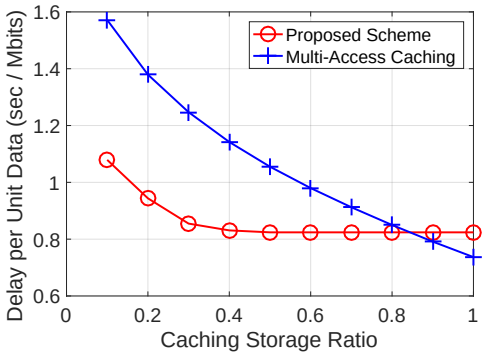
Figure 3.7: Delay and offloading performance comparison for coded caching and uncoded caching schemes

In Fig. 3.7, we compare the delay and offloading performances of coded and uncoded content caching schemes. As shown in Fig. 3.7a, the average delay per unit data for both caching schemes increases with more content files, because more files need to be retrieved by backhaul transmissions due to limited storage capacities of the HetVNet APs. It is worth noting that when the number of files is large enough (≥ 50), coded caching scheme outperforms the uncoded scheme in terms of overall average delay, since the former can cache more content files in the APs due to higher storage efficiency.

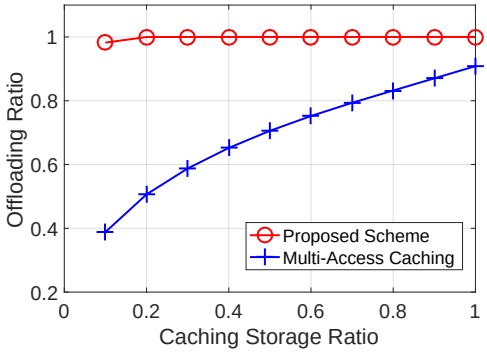
Define offloading ratio as the ratio of the data volume downloaded without going through backhaul links over the overall requested data volume. As shown in Fig. 3.7b, the offloading ratio performances of the coded and uncoded caching schemes are identical when there are less than twenty files, since the HetVNet APs can successfully cache all the files. With increasing number of files, a smaller portion of files can be cached in the APs for uncoded caching scheme, leading to higher probability of backhaul downloading and lower offloading ratio. On the other hand, despite the decline of offloading ratio, the coded caching scheme has significant advantage in terms of offloading performance when compared with the uncoded scheme. Stemming from the above observations, uncoded caching is preferred in scenarios with small number of content files, while coded caching scheme is more suitable when network scales to achieve better delay and offloading performances.



(a) Cache hit rate



(b) Average delay per unit data



(c) Offloading performance

Figure 3.8: Cache hit rate, delay, and offloading performance comparison between the proposed scheme and multi-access-based caching scheme.

E. Single-Access-Based Caching vs. Multi-Access-Based Caching

In the proposed caching scheme, one file is allowed to be cached in only one type of APs to improve caching storage efficiency without requiring the cooperation among different access networks. Intuitively, for the cached files, the delivery delay can be further reduced if they are stored in all the APs such that the vehicles can get served within any access network coverage. However, whether the overall delay performance can be improved remains unknown. In this part, we compare the proposed caching scheme with the case where the files are encoded and cached in all the APs, named as the "Multi-Access Caching", to reveal insights on the suitability of these two types of caching schemes in different scenarios. Notice that, in "Multi-Access Caching" scheme, the files are encoded with the same coding parameters (i.e., in **Algorithm 1**, let $\alpha_m = \min\{\alpha_m^W, \alpha_m^T\}$ and $k_m = z_m/\alpha_m$, then n_m^I and

K_m^I (I refers to Wi-Fi RSUs, TVWS stations, or CBSs) can be calculated accordingly).

In Fig. 3.8, we provide the performance comparison between the proposed scheme and the "Multi-Access Caching" scheme with different caching storage ratio[6]. With increasing storage capacities of the HetVNet APs, both caching schemes achieve better cache hit rate, delivery delay, and offloading performances. As shown in Figs. 3.8a and 3.8c, the proposed scheme can achieve a high cache hit rate and offloading ratio. On the other hand, when adopting the "Multi-Access Caching" scheme, much fewer content files can be cached in the HetVNet APs for backhaul offloading. Although the cached files can be downloaded faster in the "Multi-Access Caching" scheme, the overall delivery delay performance is unsatisfactory due to the substantial backhaul transmission when the caching storage capacity is limited. Nevertheless, the "Multi-Access Caching" scheme outperforms the proposed scheme in terms of overall delivery delay when the storage capacity is large enough, e.g., when the cache size of each AP is no less than 0.9 of the total size of all the files as shown in Fig. 3.8b. To summarize, the "Multi-Access Caching" scheme is a favourable choice when the storage capacity is sufficiently large, while the proposed scheme works well in general cases with limited caching resources.

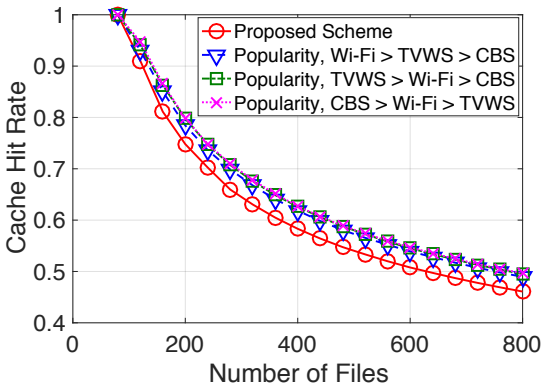
F. Performance Comparison with Popularity-Based Caching schemes

Then, we compare our proposed caching scheme with the popularity-based caching schemes. In particular, the popularity-based schemes prioritize and cache the files based only on file popularity. As shown in Fig. 3.9, we use 'Popularity' to denote the popularity-based caching schemes. In addition, 'Wi-Fi > TVWS > CBS' denotes that content files are cached in Wi-Fi RSUs with the highest priority, i.e., the most popular content files are first cached in Wi-Fi RSUs until reaching the caching capacity, then in the TVWS stations, and the CBSs have the lowest priority. 'TVWS > Wi-Fi > CBS' and 'CBS > Wi-Fi > TVWS' are defined in a similar way. In addition to the average delay and offloading performance, the cache hit rate[7] is also considered to further compare the performance of the proposed algorithm and the popularity-based algorithms.

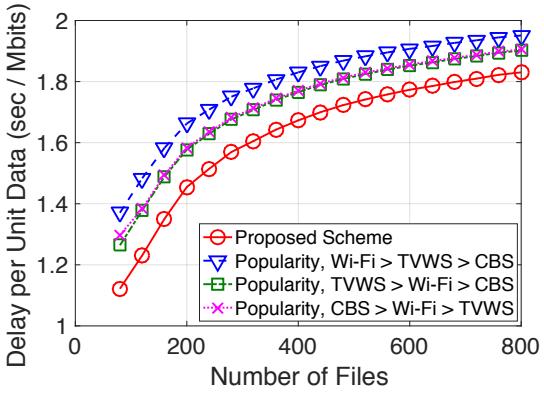
As shown in Fig. 3.9, with a small number of content files, all the files can be cached in the APs, thus the cache hit rate and offloading ratio are both equal to 1 for all the algorithms. With increasing number of files, more files need to be retrieved from backhaul links, thereby leading to a lower cache hit rate, longer delivery delay, and lower offloading

[6]All the APs are assumed to have the same storage capacity, and the caching storage ratio is the ratio of the storage capacity of one AP over the total size of all the content files.

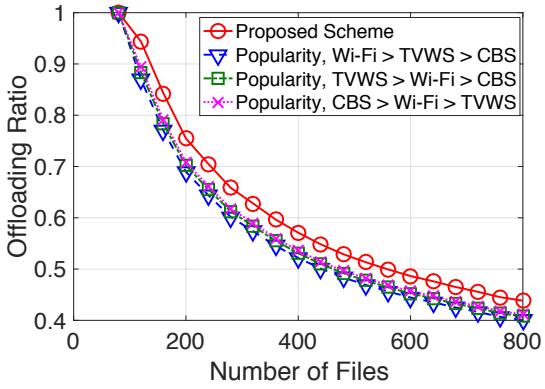
[7]The cache hit rate is defined as the ratio of the number of cache hit in all the APs' caches to the overall number of vehicular content requests.



(a) Cache hit rate



(b) Average delay per unit data



(c) Offloading Performance

Figure 3.9: Cache hit rate, delay, and offloading performance comparison between the proposed scheme and popularity-based caching schemes.

ratio for all the algorithms. As shown in Fig. 3.9a, the popularity-based schemes have a higher cache hit rate than the proposed scheme since the former ones only cache the most popular files. On the other hand, in addition to the file popularity, the proposed scheme also takes file size, vehicle mobility, and network characteristics into consideration to ably cache different types of files in the APs. Therefore, the proposed scheme presents better delay and offloading performances as shown in Figs. 3.9b and 3.9c.

3.6 Summary

In this chapter, we have investigated content caching in HetVNet APs to provide enhanced and diversified wireless network access for moving vehicles and reduce delivery delay, by considering the impact of factors including file popularity, vehicle mobility, network service interruption, and storage capacities of the APs. Specifically, we have proposed a matching-based scheme with multi-objective two-sided preference lists to optimize the content placement problem. Simulation results have validated the effectiveness of the proposed content caching scheme, which can further provide an insight into the optimization of content sharing in different network conditions.

Chapter 4

Optimal UAV Caching and Trajectory Design in the AGVN

In this chapter, we investigate the UAV-aided edge caching to assist terrestrial vehicular networks in delivering high-bandwidth content files. Aiming at maximizing the overall network throughput, we formulate a JCTO problem to make decisions on content placement, content delivery, and UAV trajectory simultaneously. As the decisions interact with each other and the UAV energy is limited, the formulated JCTO problem is intractable directly and timely. To this end, we propose a deep supervised learning scheme to enable intelligent edge for real-time decision-making in the highly dynamic vehicular networks. In specific, we first propose a CBTL algorithm to solve the JCTO problem offline. With a given content placement strategy, we devise a time-based graph decomposition method to jointly optimize the content delivery and trajectory design, with which we then leverage the particle swarm optimization (PSO) algorithm to further optimize the content placement. We then design a deep supervised learning architecture of the CNN to make fast decisions online. The network density and content request distribution with spatio-temporal dimensions are labeled as channeled images and input to the CNN-based model, and the results achieved by the CBTL algorithm are labeled as model outputs. With the CNN-based model, a function which maps the input network information to the output decision can be intelligently learnt to make timely inferences and facilitate online decisions. We conduct extensive trace-driven experiments, and our results demonstrate both the efficiency of CBTL in solving the JCTO problem and the superior learning performance with the CNN-based model.

4.1 Background and Motivations

To accommodate ever-increasing vehicular traffic demands especially in the future driverless era, the Internet of Vehicles (IoV) is envisioned to be vigorously robust and deliver high-bandwidth content files limitlessly [10]. As the vehicular network resource is highly dynamic while the terrestrial network resources (4G/5G) are fixed and rigid, it is a challenging task to guarantee satisfactory network performance anywhere at any time, such as at urban busy roads during rush hours. UAV caching is a promising paradigm to assist the terrestrial network. By proactively caching popular and repetitively requested content files with large size (such as HD map and the video streaming of a football match or a concert), UAV caching can significantly alleviate the traffic burden of the terrestrial network. Particularly, the caching-enabled UAVs are physically free from the backhaul limitation, which makes the implementation of UAV communications more feasible considering the high agility of UAVs. With fully controllable mobility and high altitude, UAVs can support fast reconfiguration with high LoS probability for UAV-to-vehicle (U2V) wireless links. Besides, on-demand UAV communications can be dispatched when the terrestrial network is overloaded, the manner of which is flexible and cost-effective.

Significant research efforts have been put on the UAV-assisted communications. Despite the extensive existing works mentioned in Chapter 2, there are still various technical challenges associated with UAV-assisted caching in VNs. First, most existing works consider UAV caching in networks with low/no user mobility, which cannot be directly applied to the vehicular scenarios. The high vehicle mobility causes spatio-temporal variation in vehicle density and content request distribution, and further affects the UAV caching performance. Second, the joint decision of content placement, UAV trajectory, and content delivery in the AGVN has not been well addressed. The joint optimization is essential to improve the UAV content delivery performance as the three decisions interact with each other. Third, as the vehicular network condition varies significantly with uncertainties, online decisions should be constantly made to keep pace with the dynamic vehicular environments, posing real-time requirements to the optimization solution.

In this chapter, we focus on the joint design of UAV caching (including content placement and delivery) and UAV trajectory in urban vehicular networks that have substantial content demands. Our objective is to find the optimal solution to the joint optimization problem in real time to maximize the overall network throughput under the UAVs' energy constraints. By partitioning the target area into small regular areas and representing each area by a point, we construct a topology graph to find the optimal paths of the UAVs, where the edge weights are affected by the content placement and delivery scheme. As the formulated JCTO problem is intractable directly and timely due to the coupling of vari-

ables, we propose a learning-based scheme named *LB-JCTO* to enable edge intelligence (EI) and make real-time decisions in the highly dynamic vehicular environments.

LB-JCTO is an offline optimization and learning for online decision framework, in which deep supervised learning is conducted to facilitate online decisions under the supervision of offline optimized targets. Particularly, in the first stage of *LB-JCTO*, we propose a CBTL algorithm to solve the JCTO problem. Given a content placement strategy, we jointly optimize UAV trajectory and content delivery by a time-based graph decomposition method, where a directed graph is constructed in accordance with the spatial-temporal variant vehicle densities and content requests. Resource constrained shortest path (RCSP) algorithms [147] are then used to find the optimal path, representing the optimal content delivery and trajectory solution. Based on the achieved optimal results, we then leverage the particle swarm optimization (PSO) algorithm to optimize content placement, further enhancing the network throughput. Although the CBTL algorithm can achieve a satisfactory performance, the algorithm complexity still cannot meet the real-time requirements for online decisions. To this end, in the second stage of *LB-JCTO*, we adopt a deep learning architecture of CNN [148] to conduct supervised learning at the intelligent edge. Particularly, the spatio-temporal variant network density and content requests are labeled as channeled images and input to the learning model, and the optimized solutions obtained by the CBTL algorithm are labeled as model outputs. With the well-trained CNN model, a function that maps the input network information to the output decision can be learnt to enable timely decision-making.

The merits of *LB-JCTO* are three-fold: 1) at the offline optimization stage, we can use a complicated algorithm with a relatively high computation complexity to obtain the optimized results, which are good learning targets; 2) for offline training, as EI-based *LB-JCTO* is trained to learn from a good target, its performance can be well guaranteed; and 3) at the online stage, with the well-trained CNN-based model, *LB-JCTO* simply runs a matching function in the UAVs for the up-to-date collected information, which can output high-quality and real-time decisions. To evaluate the performance of *LB-JCTO*, we conduct extensive trace-driven experiments based on DiDi Chuxing GPS Dataset [149]. Performance results show that: 1) the offline optimization algorithm CBTL can achieve near-optimal performance and outperform the benchmark scheme significantly; and 2) guided by the CBTL algorithm, the CNN-based deep learning approach can also achieve superior performance, and more importantly, it can react to the network input rapidly. We highlight our major contributions in this work as follows.

- We study the joint design of UAV caching and trajectory in vehicular networks, which is of significant importance for future CAVs to deliver high-bandwidth content files

robustly. Specifically, we formulate the JCTO problem to investigate the interplay between the caching scheme and UAV trajectory design, where the analysis and derivation of UAV energy consumption, achievable throughput of UAV-based moving cells and terrestrial networks are respectively presented.

- To solve the JCTO problem in real time, we propose a learning-based scheme named *LB-JCTO* to make online inferences in response to the dynamic vehicular networks. Particularly, in the *LB-JCTO* scheme, the CBTL algorithm is devised to optimize the JCTO problem offline, and then a CNN-based learning scheme is designed to facilitate online decisions.
- Extensive trace-driven experiments are carried out, and results demonstrate that the CBTL algorithm can efficiently solve the joint optimization problem, and the CNN-based learning model can well emulate the capability of CBTL while satisfying the real-time requirements.

We organize the remainder of this chapter as follows. System model and problem formulation are given in Section 4.2. Section 4.3 presents the proposed *LB-JCTO* scheme, which includes the CBTL-based offline optimization as presented in Section 4.4, and the offline model training and online decision as given in Section 4.5. Trace-driven experimental results are carried out in Section 4.6, followed by the conclusion in Section 4.7.

4.2 System Scenario and Problem Formulation

4.2.1 Scenario Description

We investigate UAV-assisted edge caching in the AGVN by utilizing UAVs to cache content files and then serve the ground vehicles. Without loss of generality, we focus on a rectangle area covered by a single CBS, as shown in Fig. 4.1. A set $\mathcal{K}$ of K rotary-wing UAVs equipped with caching storage units are dispatched into the system to act as moving BSs to serve the ground vehicles along with the CBS. Let $\mathcal{F} = \{f_1, f_2, \cdots, f_F\}$ be the set of F files requested in the scenario. The size of content files is assumed to be the same[1] and denoted by $\varsigma_f, \forall f_f \in \mathcal{F}$. The caching storage capacity of UAV k is C_k, i.e., UAV k can cache no more than C_k content files. In addition, the UAV flies horizontally at a constant altitude of H to achieve a lower level of energy consumption [150].

[1]In reality, for files with different sizes, the analysis can be easily extended by dividing each file into chunks of equal size.

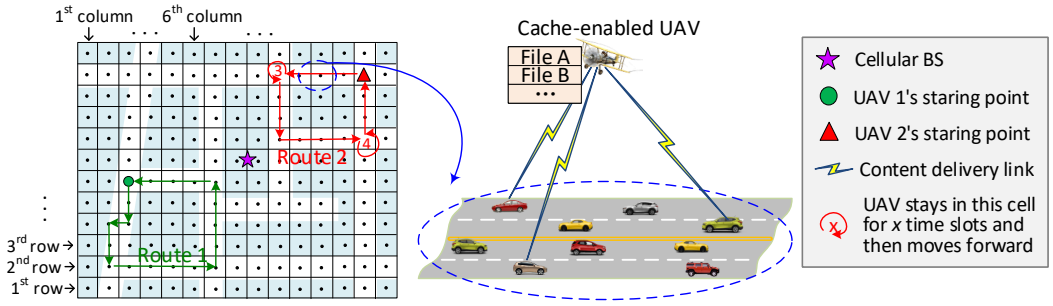


Figure 4.1: Overview of UAV-aided edge caching in vehicular networks.

The ground vehicular networks are highly dynamic with spatio-temporal variance for vehicle densities and content request distributions. Constrained by the street-layout, vehicle densities on the roads (white areas in Fig. 4.1) are generally larger than those on the other areas, so as the content request probabilities. Basically, vehicle density can be estimated based on position information collected from vehicles equipped with GPS devices. Content popularity can be modeled as Zipf distributions according to the analysis for many real datasets [151]. With Zipf-based content popularity, the request distributions still vary spatially and temporally due to different user preferences. Although there exist many works modeling and predicting a user's preference by using learning methods, the statistical modeling of the preference of a certain user set based on real-world datasets has not been well investigated [152]. Inspired by the model in [152], we model the file preference in each grid at time slot t as follows:

- A grid v and a content file f_f are respectively associated with feature values $\phi_{v,t}$ and ϑ_f, where $\phi_{v,t}, \vartheta_f \in [0,1]$[2].
- The probability that file f_f is requested by users in grid v at time t is:

$$r_{v,t,f} = p_f \frac{g(\phi_{v,t}, \vartheta_f)}{\sum_{v' \in \mathcal{V}} g(\phi_{v',t}, \vartheta_f)}, \tag{4.1}$$

where

$$p_f = \frac{1/f^{\xi}}{\sum_{m \in \mathcal{F}} 1/m^{\xi}}, \quad g(\phi_{v,t}, \vartheta_f) = (1 - |\phi_{v,t} - \vartheta_f|)^{\frac{1}{\alpha^3} - 1} \tag{4.2}$$

[2]Features of a location may include weather characteristics, historical park, and upcoming famous events; a content file is associated with features such as file type and size, metadata, keywords or tags. To simplify the model, we use a random value to represent the features of a grid/file, but this basic model can be easily extended to multi-dimensional feature vectors.

are the popularity of content f_f and the kernel function representing the correlation between grid v and file f_f[3].

Basically, $r_{v_1,t_1,f} \neq r_{v_2,t_2,f}$ for $v_1 \neq v_2$ or $t_1 \neq t_2$ due to its spatio-temporal variance, despite that vehicles may have similar preferences over some popular content files.

Aiming at maximizing the overall network throughput, the UAVs should fly to different locations to keep pace with the network dynamics. To better illustrate the time-varying locations of the UAVs, we partition the target area into small square grids with side length w. Each grid square is represented by its central point and denoted by $(i, j),\ i \in [1, N_{\text{row}}], j \in [1, N_{\text{col}}]$. This means that the grid locates in the i-th row and j-th column, and N_{row} and N_{col} are the numbers of rows and columns in the target area. We can then construct a topology graph $G = (\mathcal{V}, \mathcal{E})$ representing the whole map, where $\mathcal{V}$ is the set of central points of the grid squares and $\mathcal{E}$ contains the edges connecting the central points. For two points $u = (i, j)$ and $v = (m, n)$, $(u, v) \in \mathcal{E}$ if and only if $u, v \in \mathcal{V}, |i - m| \leq 1, |j - n| \leq 1$[4].

The UAV endurance is equally discretized into T_U time slots, each with a time length of Δ_t. UAVs can fly along the points and edges in graph G, where a UAV can move from point u to v in two continuous time slots only if $(u, v) \in \mathcal{E}$. Notice that the starting and ending points of a trajectory are the same, i.e., the location where the UAV is dispatched and collected for battery charging. The starting and ending points for multiple UAVs can be different. As shown in Fig. 4.1, two UAVs fly along different trajectories, which are respectively marked in red and green, with different starting points. Along the flying trajectory, UAVs can keep changing locations at different time slots (Route 1) or choose to hover above a certain location for multiple time slots (Route 2).

To facilitate analysis, we define necessary notation and symbols below.

1) $\mathbf{D}_{N_{\text{row}} \times N_{\text{col}} \times T_U}$ is a three-dimensional array showing the spatial- and temporal-varying vehicle densities. The (i, j, t)-th entry $d_{i,j,t}$ (or $d_{v,t}$ if $v = (i, j)$) is the average number of vehicles in the grid represented by point (i, j) ($(i, j) \in \mathcal{V}$) at time slot t ($t \leq T_u$).

2) $\mathbf{R}_{N_{\text{row}} \times N_{\text{col}} \times T_U \times F}$ is a four-dimensional array representing the spatial- and temporal-varying content request distributions. The (i, j, t, f)-th entry $r_{i,j,t,f}$ (or $r_{v,t,f}$ if $v = (i, j)$) is the request probability of file f_f in the grid represented by point (i, j) at time slot t.

[3]The grids and files are correlated due to the underlying correlation between location and content features.

[4]In the remainder of this chapter, v and (i, j) are used interchangeably to represent a grid square.

3) $\mathbf{X}_{K\times T_U} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_K]$ is the flying trajectories of K UAVs. $\mathbf{x}_k = [x_{k,1}, x_{k,2}, \cdots, x_{T_U}]$ is the k-th UAV's trajectory within its endurance time, where $x_{k,t}$ is its location at time slot t. Thus we have $(x_{k,t}, x_{k,t+1}) \in \mathcal{E}$ and $x_{k,1} = x_{k,T_U} = v_{0,k}$, where $v_{0,k}$ is the starting point where UAV k is released.

4) $\mathbf{A}_{K\times F} = [\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_K]$ is an indicator matrix showing the caching of content files in the UAVs, where $\mathbf{a}_k = [a_{k,1}, a_{k,2}, \cdots a_{k,F}]$ represents the k-th UAV's caching status. $a_{k,f} = 1$ if content file f_f is cached in UAV k; otherwise, $a_{k,f} = 0$.

5) $\mathbf{S}_{K\times T_U} = [\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_K]$ denotes the content delivery decisions of UAVs along their trajectories, where $\mathbf{s}_k = [s_{k,1}, s_{k,2}, \cdots s_{k,T_U}]$. $s_{k,t} = 1$ if UAV k chooses to serve the vehicular requests at time t, whereas $s_{k,t} = 0$ means that UAV k flies without content delivery at time t.

4.2.2 Communication and UAV Energy Consumption Models

In this part, we introduce the models for U2V and CBS-to-vehicle (C2V) communications.

A. U2V Communications

In this work, the caching-enabled UAVs work in the Wi-Fi spectrum with a constant transmission power, which is denoted by P_U. Notice that NLoS links can severely degrade the communication performance and cannot support efficient U2V content delivery. Therefore, the UAVs' coverage radius, denoted by r_{UAV}, can be defined by constraining the LoS probability and free-space pathloss [153]. With a fixed flying altitude H, we have

$$r_{\text{UAV}} = \min\left\{\frac{H}{\tan\left(a_1 - \frac{1}{a_2}\ln\left(\frac{1-\xi_{LoS}}{a_1\xi_{LoS}}\right)\right)}, \sqrt{\left(\frac{c\gamma_{\max}}{4\pi f_c}\right)^2 - H^2}\right\}, \tag{4.3}$$

where a_1 and a_2 are constant values determined by environment. f_c, c, ξ_{LoS} and $\gamma_{\max}$ respectively represent the carrier frequency, light speed, the LoS probability requirement, and U2V free space pathloss threshold. When partitioning the target area into grid squares, we can set the side length of each square as $w = \sqrt{2}r_{\text{UAV}}$. With this setting, when a UAV hovers above a grid square, its coverage area can approximately equal the square area.

According to IEEE 802.11 standard, the Wi-Fi coverage area can be divided into zone areas based on the achieved signal-to-noise ratio (SNR) levels, and the data rates are calculated based on the Wi-Fi modulation and coding schemes [42]. Therefore, the coverage

area of a Wi-Fi-based UAV can be divided into L zones. The j-th zone has a distinct annulus area with width l_j and a data rate of R_j. Thus, a vehicle within a grid area can achieve a mean throughput of:

$$\overline{R}_{UAV} = \rho \left(\frac{\sum_{j=1}^{j=L} \left(R_j \left[(l_j + l_{j-1})^2 - l_{j-1}^2 \right] \right)}{\left(\sum_{j=1}^{j=L} l_j \right)^2} \right), \tag{4.4}$$

where $l_0 = 0$ and ρ is Wi-Fi throughput efficiency factor, which characterizes the overhead of protocol negotiations and packet headers.

The Wi-Fi channel is shared by associated vehicles under a contention-based mechanism. With N vehicles sharing the Wi-Fi channel in a grid area, each vehicle achieves a data rate of $\overline{R}_{UAV}/N$. When the associated vehicles have a throughput requirement of R_{req}, the number of vehicles that can be simultaneously served by a UAV should be no more than $N_{U,\max}$, where

$$N_{U,\max} = \left\lfloor \frac{\overline{R}_{UAV}}{R_{req}} \right\rfloor. \tag{4.5}$$

B. C2V Communications

In this work, channel inversion power control is adopted for the CBS, where transmit power is allocated based on the channel conditions to ensure equal average SNR for all the associated vehicles. The C2V channel gain is considered as $h_{i,C} = \varrho_{i,C} d_{i,C}^{-\alpha}$, where $\varrho_{i,C}$ is the channel fading and follows an exponential distribution with unit mean, $d_{i,C}$ is the distance between the vehicle and the CBS, and α is the pathloss exponent. Let B_C be the total available cellular bandwidth, which is equally shared by vehicles using C2V communications. $P_{C,\max}$ is the maximum available transmission power of the CBS. Given that the C2V channel gains keep changing due to vehicle mobility, it is costly to perform real-time power allocation based on every vehicle's instantaneous channel condition. Thus, the average C2V pathloss is used for power allocation for vehicles in the same grid. The average pathloss is dependent on the average distance to the CBS, which can be calculated referring to *Lemma 1* in [154]. When there are no UAVs in the system, the overall cellular network throughput at time slot t is

$$R_C(t) = \frac{B_C}{\sum_{v \in \mathcal{V}} d_{v,t}} \sum\nolimits_{v \in \mathcal{V}} d_{v,t} \log \left(1 + \frac{P_{v,C}(t) h_{v,C}}{\frac{B_C \sigma^2}{\sum_{v \in \mathcal{V}} d_{v,t}}} \right), \tag{4.6}$$

where $P_{v,C}(t)$ and $h_{v,C}$ are the transmission power and average channel gain from the CBS to the vehicles in v ($v \in \mathcal{V}$), and σ^2 is the noise power density. Thus, we have

$$\begin{aligned} &\sum_{v\in\mathcal{V}} d_{v,t} P_{v,C}(t) = P_{C,\max}, \\ &P_{v_1,C}(t) h_{v_1,C} = P_{v_2,C}(t) h_{v_2,C},\ \forall v_1, v_2 \in \mathcal{V}. \end{aligned} \tag{4.7}$$

Assuming that K UAVs have caching status $\mathbf{A}$, delivery decision $\mathbf{S}$, and trajectories $\mathbf{X}$, the set of positions of the K UAVs at time slot t is $\mathcal{V}_t = \{x_{1,t}, x_{2,t}, \cdots, x_{K,t}\}$. In grid $x_{k,t}$, the average number of vehicles that need to be served by the CBS is derived as:

$$n_{C,k,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} = (1 - s_{k,t}) d_{x_{k,t},t} + s_{k,t} d_{x_{k,t},t} \left(1 - \sum_{f_f \in \mathbf{a_k}} r_{x_{k,t},t,f}\right). \tag{4.8}$$

The cellular bandwidth allocated to each associated vehicle, denoted by $B_{C,t}^{\mathbf{A},\mathbf{X},\mathbf{S}}$, is

$$B_{C,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} = \frac{B_C}{\sum_{k=1}^{K} n_{C,k,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} + \sum_{u\in\mathcal{V}, u\notin\mathcal{V}_t} d_{u,t}}. \tag{4.9}$$

The average throughput achieved by the CBS (denoted by $R_{C,t}^{\mathbf{A},\mathbf{X},\mathbf{S}}$) and UAV k (denoted by $R_{U,k,t}^{\mathbf{A},\mathbf{X},\mathbf{S}}$) can be respectively expressed as:

$$\begin{aligned} R_{C,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} =& B_{C,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} \Bigg(\sum_{u\in\mathcal{V}, u\notin\mathcal{V}_t} d_{u,t} \log\left(1 + \frac{P_{u,C}(t) h_{u,C}}{B_{C,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} \sigma^2}\right) \\ &+ \sum_{k=1}^{K} n_{C,k,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} \log\left(1 + \frac{P_{x_{k,t},C}(t) h_{x_{k,t},C}}{B_{C,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} \sigma^2}\right)\Bigg), \\ R_{U,k,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} =& \overline{R}_{UAV} \cdot \varepsilon\left(d_{x_{k,t},t} \cdot s_{k,t} \cdot \sum_{f_f \in \mathbf{a_k}} r_{x_{k,t},t,f}\right), \end{aligned} \tag{4.10}$$

where $\varepsilon(x)$ is a unit step function, $\varepsilon(x) = 1$ if $x > 0$; otherwise, $\varepsilon(x) = 0$. Then we can derive the overall system throughput as:

$$R(\mathbf{A}, \mathbf{X}, \mathbf{S}) = \sum_{t=1}^{T_U} \left(R_{C,t}^{\mathbf{A},\mathbf{X},\mathbf{S}} + \sum_{k=1}^{K} R_{U,k,t}^{\mathbf{A},\mathbf{X},\mathbf{S}}\right). \tag{4.11}$$

C. UAV Energy Consumption Models

The energy consumption of the caching-enabled UAVs mainly includes two parts: the propulsion energy required to support its movement and the communication energy for content delivery. Notice that the computing energy consumption in UAVs is not considered here since the computation-intensive tasks are carried out in the UAV control center instead of UAVs, which will be discussed with more details in Section 4.3.

Propulsion Energy: According to [155] and [154], for a rotary-wing UAV with speed V, the propulsion power consumption and the propulsion energy consumption during one time slot can be respectively expressed as:

$$P(V) = P_0\left(1+\frac{3V^3}{U^2}\right) + P_1\left(\left(1+\frac{V^4}{4v_r^4}\right)^{\frac{1}{2}} - \frac{V^2}{2v_r^2}\right)^{\frac{1}{2}} + \frac{1}{2}AV^3, \tag{4.12}$$
$$E_p(x) = \frac{x}{V}P(V) + \max\left\{\Delta_t - \frac{x}{V}, 0\right\}\cdot(P_0+P_1),$$

where $x \in \{0, w, \sqrt{2}w\}$ is the flying distance within one time slot determined by the UAV trajectory planning, P_0, P_1, U, v_r, and A are constant parameters related to the UAV's weight, wing area, air density, etc.

Communication Energy: When UAV k flies above grid v at time slot t, the probability that there are n vehicles requesting file f_f (with request probability $r_{v,t,f}$) is:

$$\Pr(f_f, d_{v,t}, n) = \begin{pmatrix} d_{v,t} \\ n \end{pmatrix} r_{v,t,f}^n (1-r_{v,t,f})^{d_{v,t}-n}. \tag{4.13}$$

Let $\mathbf{a}_k$ be the caching status of UAV k and $\mathcal{F}_k = \{f_{k,1}, f_{k,2}, \cdots, f_{k,m}\}$ be the set of m content files satisfying $a_{k,f} = 1,\ \forall f_f \in \mathcal{F}_k$. Assuming that requests for different files are independent, the probability that there are n requests for files in $\mathcal{F}_k$ is

$$\begin{aligned}\Pr(\mathcal{F}_k, d_{v,t}, n) = &\sum_{n_1=0}^{n} \Pr(f_{k,1}, d_{v,t}, n_1) \sum_{n_2=0}^{n-n_1} \Pr(f_{k,2}, d_{v,t}, n_2)\cdots \\ &\sum_{n_{m-1}=0}^{n-\sum_{j=1}^{m-2} n_j} \Pr(f_{k,m-1}, d_{v,t}, n_{m-1}) \Pr(f_{k,m}, d_{v,t}, n-\sum_{j=1}^{m-1} n_j).\end{aligned} \tag{4.14}$$

Note that one extreme case is that every vehicle requests all the cached files, in which case there are $m \cdot d_{v,t}$ requests for files in $\mathcal{F}_k$. For $n > m \cdot d_{v,t}$, we have $\Pr(\mathcal{F}_k, d_{v,t}, n) = 0$. Therefore, given UAV k's caching status $\mathbf{a}_k$ and its position $x_{k,t}$ at time t, the average communication energy consumption to serve the requesting vehicles is:

$$E_c(\mathbf{a}_k, x_{k,t}) = \sum_{n=1}^{m \cdot d_{v,t}} \Pr\left(\mathcal{F}_k, d_{x_{k,t},t}, n\right) P_U \min\left\{\Delta_t, \frac{n \cdot \varsigma_f}{\overline{R}_{\text{UAV}}}\right\}. \tag{4.15}$$

4.2.3 Problem Formulation

To maximize the overall network throughput in the UAV-assisted edge caching system under UAV energy constraints, $\mathbf{A}$, $\mathbf{S}$, and $\mathbf{X}$ should be jointly optimized since they interact with each other. Based on the network throughput and UAV energy consumption analysis given in Section 4.2.2, the JCTO problem can be formulated as:

$$(\text{JCTO}): \max_{\mathbf{A},\mathbf{X},\mathbf{S}} \quad R(\mathbf{A}, \mathbf{X}, \mathbf{S}) \tag{4.16}$$

$$s.t. \quad \sum_{f_f \in \mathcal{F}} a_{k,f} \leq C_k, \quad \forall f_f \in \mathcal{F}, \forall k \in \mathcal{K}, \tag{4.16a}$$

$$\sum_{t=1}^{T_U} E_p(\|x_{k,t} - x_{k,t-1}\|) + E_c(\mathbf{a}_k, x_{k,t}) \cdot s_{k,t} \leq E_{k,\max}, \tag{4.16b}$$

$$x_{k,1} = x_{k,T_U} = v_{0,k}, \quad (x_{k,t-1}, x_{k,t}) \in \mathcal{E}, \tag{4.16c}$$

$$a_{k,f} = \{0, 1\}, \quad s_{k,t} = \{0, 1\}, \tag{4.16d}$$

where $E_{k,\max}$ is the overall on-board energy of the k-th UAV. Constraint (4.16a) restricts the maximum number of files cached in each UAV and constraint (4.16b) regulates the maximum allowable UAV energy consumption. Constraint (4.16c) represents that UAVs can only fly along the edges in the graph and need to finally return to the starting points. The JCTO problem in (4.16) is non-convex and intractable since $\mathbf{A}, \mathbf{X}$, and $\mathbf{S}$ should be jointly optimized for all the K UAVs under spatio-temporal network variations and the energy constraints of UAVs.

4.3 Design of *LB-JCTO*

In this work, we propose a learning-based framework named *LB-JCTO* to solve the JCTO problem in (4.16). As shown in Fig. 4.2, the *LB-JCTO* scheme includes two major stages: 1) offline optimization; and 2) offline model training and online decision.

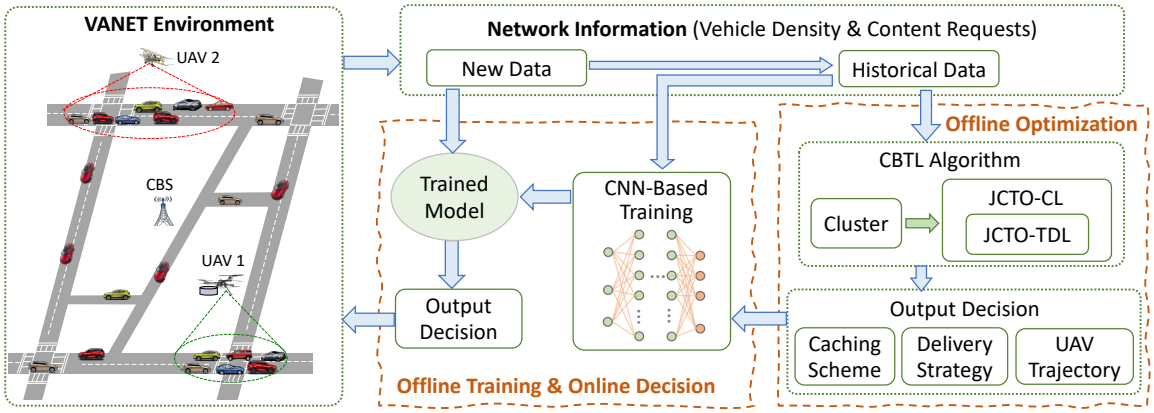


Figure 4.2: Working diagram of the proposed *LB-JCTO* scheme.

Offline Optimization: In this stage, network information (including vehicle density and content request distribution) is obtained either from collected historical data or from predictions. Then we propose a CBTL algorithm to effectively solve the JCTO problem and achieve near-optimal joint solutions.

Offline Model Training and Online Decision: We leverage a CNN-based deep learning scheme to learn from the CBTL algorithm and make fast decisions. The CBTL algorithm works as a labeler (or supervisor) for the CNN-based learning model. In specific, the network information and the solution obtained by the CBTL algorithm work together as labeled data, based on which a function that maps the input network information to the output decisions can be learnt with the CNN-based model. After well-trained, the learning model can be utilized to output the corresponding JCTO decisions rapidly to react to new network information. In the meantime, the new network information can be collected and used to further train and update the learning model.

Notice that in the *LB-JCTO scheme*, network information collection, CBTL-based offline optimization, and CNN-based offline model training can be conducted at the UAV control center or edge server with powerful computing and processing capabilities. Then the well-trained learning model can be transferred and implemented on the UAVs to perform model inference locally and make fast response to the dynamic network information.

4.3.1 Offline Optimization

In the offline optimization stage, the proposed CBTL algorithm first groups vehicles into $K+1$ clusters, each served by a UAV or the CBS. The clustering process ensures that

each UAV only needs to fly within a certain area rather than traveling a long distance. This helps save the limited on-board energy and prevent potential collisions among UAVs. When clustering the vehicles, a new metric combining three different types of similarities is considered in this work. More specifically, cellular performance similarity, physical location similarity, and content preference similarity are considered to ensure that vehicles in the same cluster have similar C2V channel conditions, physical locations, and content interests.

After the vehicle clustering, the JCTO problem needs to be solved for the UAV for each cluster. Since the JCTO problem is non-convex and difficult to solve, the proposed CBTL algorithm adopts a vertical decomposition that leads to the following two-layered structure of the problem:

- *Caching-Layer (CL) Optimization:* The CL optimization problem can be reformulated as:

$$\text{(JCTO-CL)}: \max_{\mathbf{A}} \quad \sum\nolimits_{t=1}^{T} R(\mathbf{A}, \mathbf{X}, \mathbf{S}) \tag{4.17}$$

$$s.t. \quad \text{Constraint (4.16a).} \tag{4.17a}$$

- *Trajectory-and-Delivery-Layer (TDL) Optimization:* The TDL optimization problem can be reformulated as:

$$\text{(JCTO-TDL)}: \max_{\mathbf{X},\mathbf{S}} \quad \sum\nolimits_{t=1}^{T} R(\mathbf{A}, \mathbf{X}, \mathbf{S}) \tag{4.18}$$

$$s.t. \quad \text{Constraints (4.16b-4.16d).} \tag{4.18a}$$

In this work, the JCTO-CL problem is solved by leveraging the PSO algorithm, and a time-based graph decomposition method is devised to solve the JCTO-TDL problem, which will be elaborated in Sections 4.4.3 and 4.4.4. Notice that when solving the JCTO-CL problem, the quality of a caching policy (i.e., the achievable $R(\mathbf{A}, \mathbf{X}, \mathbf{S})$) is determined by the optimal achievable performance with the JCTO-TDL problem. In other words, the JCTO-TDL optimization is embedded within the JCTO-CL problem in our CBTL-based offline optimization algorithm.

4.3.2 Offline Model Training and Online Decision

Given network information, the CBTL-based offline optimization algorithm can achieve satisfied throughput performance. However, in practice it might be difficult to precisely

predict the vehicle mobility and content request distributions within each grid. For example, a vehicle collision can easily change the vehicle density in the accident location as well as in the surrounding grids. When the network condition changes unexpectedly after dispatching the UAVs, the achievable system throughput might decrease if the UAVs keep moving along the pre-designed trajectories. Under such circumstances, re-calculating a good trajectory is critical to guarantee efficient UAV service provision. Since UAVs are generally energy-constrained and have limited computing power, our proposed CBTL algorithm is not suitable to be implemented in UAVs to continuously update the trajectories and delivery decisions in real time. Therefore, in this work, we design a CNN-based deep learning model that is trained offline under the supervision of the CBTL-based algorithm. Then UAVs can obtain the trained model from the edge server and perform real-time model inference locally [156].

CNN is an effective image processing algorithm and has been widely applied in many fields [157, 158]. With superior learning ability in image understanding, CNN is suitable for our problem for the following reasons. First, the convolutional layers can effectively capture the local dependencies and extract important features, e.g., network features like the road layout or dense areas with frequent requests. Second, CNN also introduces pooling mechanisms to reduce data dimension while preserving dominant features. With these two characteristics, the CNN-based model is not only good at learning features but also scalable to large-scale problems.

In the offline training of the CNN-based model, supervised learning is conducted to learn the matching function from the input data to the output decisions. Specifically, the input network information is labeled as channeled images and normalized before fed into the learning model. The optimized solutions obtained by the CBTL algorithm are labeled as model outputs to provide supervision to the learning model. The detailed learning model structure will be introduced in Section 4.5.

4.4 CBTL-Based Offline Optimization

4.4.1 Determining the Number of UAVs

With UAV-assisted communications, the network throughput can be improved and thus each vehicle's QoS is enhanced. The optimal number of UAVs dispatched into the system depends on users' service satisfaction requirements. In general, users prefer more UAVs to achieve higher performance satisfaction levels. The service providers, on the other hand, tend to use the least resource to achieve the best gain and focus more on cost

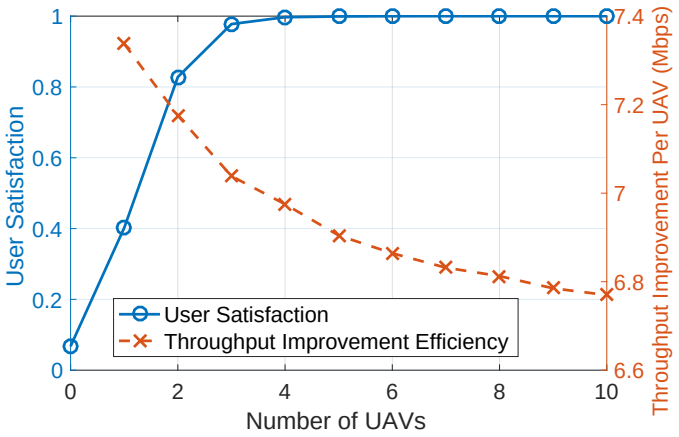


Figure 4.3: Impact of K on user satisfaction level and throughput improvement efficiency. $H = 35$ m, $P_{C,\max} = 43$ dBm, $B_C = 20$ MHz, $P_U = 28$ dBm.

efficiency, e.g., the performance enhancement introduced by each UAV. Fig. 4.3 shows the impact of the number of UAVs on users' satisfaction level (approximated by using sigmoid function [159]) and throughput improvement by each UAV. It can be seen that, with more UAVs launched into the system, the average user satisfaction level increases while the throughput improvement efficiency decreases. Thus, the optimal K varies in different scenarios with diverse user satisfaction requirements, the number of UAVs a provider has, and/or the deployment cost efficiency constraint.

In our UAV-based edge caching system, we aim to minimize the number of UAVs required to satisfy the vehicles' throughput requirement of R_{req}. Recall that channel inversion power control is adopted for the CBS and the bandwidth is equally allocated to all associated vehicles. All vehicles served by the CBS have the same received signal strength γ_C, which can be calculated based on (4.7):

$$\gamma_C = P_{C,\max} \Big/ \left(\sum_{v \in \mathcal{V}} \frac{d_{v,t}}{h_{v,C}} \right). \tag{4.19}$$

When guaranteeing the vehicles' throughput requirement, the CBS can serve more vehicles if it serves close vehicles rather than remote vehicles, since the close ones have better C2V links and require smaller transmit power. For this reason, we sort the grids in descending order based on C2V channel power gain. Let v_i be the i-th closest grid to the CBS. When the CBS serves users in the first N_C nearest grids, the achievable throughput

per vehicle should satisfy

$$\frac{B_C}{\sum_{i=1}^{N_C} d_{v_i,t}} \log\left(1 + \frac{P_{C,\max} \sum_{i=1}^{N_C} d_{v_i,t}}{B_C \sigma^2 \sum_{i=1}^{N_C} \frac{d_{v_i,t}}{h_{v_i,C}}}\right) \geq R_{req}. \tag{4.20}$$

The left side of Eq. (4.20) decreases monotonously with N_C. Let $N_{C,\max}$ denote the maximum value of N_C that satisfies Eq. (4.20). Although the closed-form expression of the optimal N_C cannot be derived, $N_{C,\max}$ can be determined by using approaches like bisection method. Therefore, the minimum number of UAVs required to ensure vehicle throughput requirement is expressed as:

$$K_{\min} = \left\lceil \frac{\sum_{v\in\mathcal{V}} d_{v,t} - \sum_{i=1}^{N_{C,\max}} d_{v_i,t}}{N_{U,\max}} \right\rceil. \tag{4.21}$$

4.4.2 Vehicle Clustering

With K UAVs dispatched into the system, vehicles can be clustered into $K+1$ groups to be served by the UAVs and the CBS. In this work, a widely used clustering method, K-means clustering [160], is adopted by considering the following similarities:

1) Cellular Performance Similarity - When a UAV is dispatched into the system, the throughput gain increases if it serves vehicles with poor cellular performance [161]. Thus, the cellular throughput similarity is an important factor to be considered. Vehicles with good C2V channels prefer to be in the same cluster and served by the CBS, while other vehicles need to be served by UAVs. When assigned the same cellular bandwidth B_0 and transmit power P_0, vehicles in grid v achieve a throughput of $R_{C,v} = B_0 \log(1+P_0 h_{v,C}/\sigma^2)$. The similarity between vehicles in grids v and u is evaluated by

$$\text{sim}_{u,v,1} = \min\left\{\frac{R_{C,v}}{R_{C,u}}, \frac{R_{C,u}}{R_{C,v}}\right\} \in [0,1]. \tag{4.22}$$

2) Physical Location Similarity - Basically, the grid squares in the same cluster served by a UAV should be in proximity to avoid extra UAV propulsion energy consumption. The physical location similarity among grids is evaluated by

$$\text{sim}_{u,v,2} = 1 - \frac{\text{dist}_{u,v}}{\max_{u,v\in\mathcal{V}} \text{dist}_{u,v}} \in [0,1], \tag{4.23}$$

where $\text{dist}_{u,v}$ is the average distance between grids u and v.

Algorithm 4: K-Means-Based Vehicle Clustering

Let $\overline{d}_v = \frac{1}{T_U}\sum_{t=1}^{T_U} d_{v,t}$, $\overline{\mathbf{r}}_v = \frac{1}{T_U}\sum_{t=1}^{T_U} \mathbf{r}_{v,t}$, $\alpha_1, \alpha_2, \alpha_3 = 1$.

Step 1: Centroid Initialization: The first cluster centroid v_C^0 is the grid where the CBS is located. Besides, randomly choose K grids, denoted by $v_C^1, \cdots, v_C^K$, as the centroids for the remaining K clusters.

Step 2: Grid Clustering: $K+1$ clusters, denoted by $\mathcal{C}_0, \mathcal{C}_1, \cdots, \mathcal{C}_K$, are created by associating every grid with the centroid with maximum similarity based on (4.25).

Step 3: Centroid Update: Update the $K+1$ cluster centroids:

$$v_C^0 = v_C^0, \quad v_C^k = \frac{1}{\sum_{v\in\mathcal{C}_k}\overline{d}_v}\sum_{v\in\mathcal{C}_k}\overline{d}_v v,$$

$$R_{C,v_C^k} = \frac{1}{\sum_{v\in\mathcal{C}_k}\overline{d}_v}\sum_{v\in\mathcal{C}_k}\overline{d}_v R_{C,v}, \ \mathbf{r}_{v_C^k} = \frac{1}{\sum_{v\in\mathcal{C}_k}\overline{d}_v}\sum_{v\in\mathcal{C}_k}\overline{d}_v\overline{\mathbf{r}}_v.$$

Step 4: Repeat **Steps 2-3** until converging.

Step 5: Repeat **Steps 1-4** and choose the best for multiple runs.

3) Content Preference Similarity - To improve the utilization efficiency for the limited UAV caching resources, vehicles with similar content interests should be grouped and served by the same UAV. Utilizing cosine similarity to evaluate the file preference similarity [152], we express the average interest similarity between grids u and v during T_U time slots as:

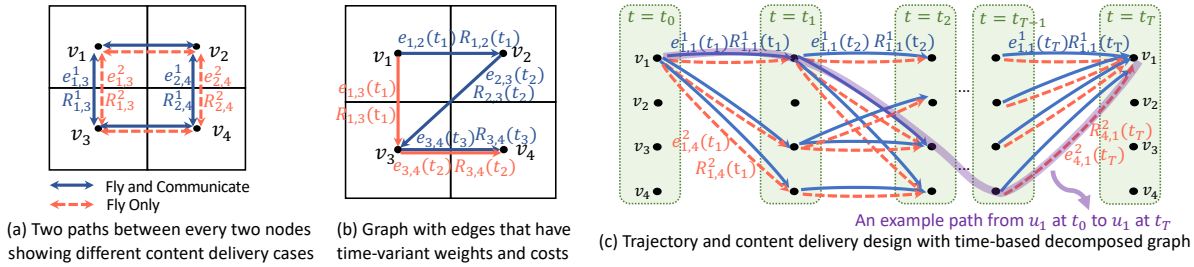
$$\text{sim}_{u,v,3} = \frac{1}{T_U}\sum_{t=1}^{T_U}\frac{\mathbf{r}_{v,t}\cdot\mathbf{r}_{u,t}}{\|\mathbf{r}_{v,t}\|\cdot\|\mathbf{r}_{u,t}\|}, \tag{4.24}$$

where $\mathbf{r}_{v,t} = [r_{v,t,f_1}, \cdots, r_{v,t,f_F}]$ is the request distribution in grid v at time t and $\text{sim}_{u,v,3} \in [0,1]$.

Taking the above-mentioned three metrics into account, the overall similarity between grids u and v can be evaluated by

$$\text{sim}_{u,v}^{all} = \text{sim}_{u,v,1}^{\alpha_1}\cdot\text{sim}_{u,v,2}^{\alpha_2}\cdot\text{sim}_{u,v,3}^{\alpha_3}, \tag{4.25}$$

where parameters α_1, α_2 and α_3 control the relative importance of the three metrics. For example, with $\alpha_1 > 1$, we give a higher priority to the cellular performance similarity, while $\alpha < 1$ indicates that we put more emphasis on the other two metrics. Targeting at maximizing $\text{sim}_{u,v}^{all}$ within a cluster, the K-means based clustering algorithm can be summarized as given in **Algorithm 4.**



(a) Two paths between every two nodes showing different content delivery cases

(b) Graph with edges that have time-variant weights and costs

(c) Trajectory and content delivery design with time-based decomposed graph

Figure 4.4: A simple example of trajectory and content delivery design with time-based graph decomposition. ($R^1_{i,j}(t)$ and $R^2_{i,j}(t)$ denote the achievable throughput when flying from v_i to v_j at time t with and without content delivery, $e^1_{i,j}(t)$ and $e^2_{i,j}(t)$ are the corresponding energy consumption.)

4.4.3 JCTO-TDL Optimization in the CBTL Algorithm

After vehicle clustering, the CBTL algorithm is applied to solve the JCTO problem for each UAV based on the vertical problem decomposition as mentioned in Section 4.3.1. In this subsection, the JCTO-TDL problem is addressed by the proposed time-based graph decomposition method.

Given the caching policy and the current position of a UAV, when the UAV flies to any feasible position in the next time slot, the achievable network throughput and the corresponding energy consumption can be calculated based on Section 4.2.2. The JCTO-TDL sub-problem, which aims to find the optimal content delivery and UAV trajectory to maximize the network throughput under the energy constraint, is similar to the RCSP problem, which has been investigated in [147]. However, the RCSP algorithm cannot be directly applied to the JCTO-TDL problem for the following reasons: *1)* in each time slot, $s_{k,t}$ can be either 0 or 1, which corresponds to two edges between the adjacent grids with different weights (i.e., achievable network throughput) and costs (i.e., energy consumption), as shown in Fig. 4.4a; and *2)* due to the time-variant $\mathbf{D}$ and $\mathbf{R}$, the weights and costs of an edge change when visited by UAVs at different time slots. Fig. 4.4b gives two examples of trajectories from v_1 to v_4, which are respectively marked in red and blue. The throughput and energy consumption vary when the UAV flies from v_3 to v_4 at different times.

To address the above-mentioned issues, we propose a time-based decomposition method to expand graph $\mathcal{G}$ into a directed graph, as shown in Fig. 4.4c. The edge exists between v_i at time t_1 and v_j at time t_2 only if $t_2 = t_1 + \Delta_t$ and $(v_i, v_j) \in \mathcal{V}$. An example path is given as the purple curve in Fig. 4.4c, which represents a possible UAV trajectory and content delivery decision in each step from source (v_1 at time t_0) to destination (v_1 at time

Algorithm 5: JCTO-TDL Optimization in CBTL Algorithm

v_k: the starting and ending point of UAV k.
Step 1: For any $(u, v) \in \mathcal{E}$ and $t \in [1, T_U]$, calculate the weights (throughput) and costs (energy consumption).
Step 2: Use shortest path (SP) algorithms (e.g., Dijkstra's algorithm) to find the path with smallest cost.
Let E_{t_0,t_T} denote the sum cost from source to v_k at time slot t_T.
Find t_T such that $E_{t_0,t_T} \leq E_{k,all}$ and $E_{t_0,t_{T+1}} > E_{k,all}$. Let $t_T^{\max} = t_T$.
Step 3: Use SP algorithms to find the path with largest cost.
Find t_T such that $E_{t_0,t_T} \leq E_{k,all}$ and $E_{t_0,t_{T+1}} > E_{k,all}$. Let $t_T^{\min} = t_T$.
for $t_T = [t_T^{\min}, t_T^{\max}]$ **do**
    Construct time-based decomposed graph as shown in Fig. 4.4(c).
    Apply RCSP algorithms to find the optimal path in the graph. Record the best path which leads to the maximum achievable network throughput.
end
Output the recorded best path.

t_T[5]). To this end, the JCTO-TDL problem is equivalent to finding the optimal path in the expanded graph to maximize sum weights under the cost constraint. With given source and destination, RCSP algorithms can be leveraged to find the optimal path. Given that the UAV's energy consumption varies with different trajectories and delivery decisions, it is difficult to determine the destination t_k when the UAV exhausts its energy. To address this issue, we execute an energy-constrained line-search on t_k to find the optimal JCTO-TDL solution, as described in **Algorithm 5**.

4.4.4 JCTO-CL Optimization in the CBTL Algorithm

The JCTO-TDL optimization provides the optimal $\mathbf{X}$, $\mathbf{S}$, and the corresponding $R(\mathbf{A}, \mathbf{X}, \mathbf{S})$ with given caching policy. In this subsection, the content placement is optimized to further improve the network throughput. However, conventional linear programming approaches cannot solve the JCTO-CL problem because we are not able to provide a closed-form expression for $R(\mathbf{A}, \mathbf{X}, \mathbf{S})$. Compared to the exhaustive searching scheme with exorbitant time complexity, heuristic algorithms, especially the evolutionary heuristics, are considered

[5]The UAV returns to the UAV center at time $t_T \leq T_U$. From time slot t_{T+1} to T_U, the UAV stays in the UAV center without content delivery and charges its battery for the next flight.

as better alternative choices to approach the optima [162]. More specifically, we utilize the PSO algorithm in this work due to its low computational cost and fast convergence [163].

When applying the PSO algorithm to solve the JCTO-CL problem, we first generate a group of particles, each of which has a position indicating a potential caching scheme. Then the fitness values (achievable network throughput) of these particles are calculated based on the analysis in Section 4.4.3. Based on the particles' positions and fitness values, there exist a local optimal position ($\varpi^{\ell}_{local}(t)$) for each particle ℓ and a global optimal position ($\varpi_{global}(t)$) for the entire particle swarm at the t-th iteration. Then, at iteration $t+1$, the position $\varpi^{\ell}(t+1)$ and velocity $\nu^{\ell}(t+1)$ of particle ℓ are updated as:

$$\begin{aligned}\nu^{\ell}(t+1) =& \phi\nu^{\ell}(t) + c_1\phi_1(\varpi^{\ell}_{\text{local}}(t) - \varpi^{\ell}(t)) + c_2\phi_2(\varpi_{\text{global}}(t) - \varpi^{\ell}(t)), \\ \varpi^{\ell}(t+1) =& \varpi^{\ell}(t) + \nu^{\ell}(t+1),\end{aligned} \tag{4.26}$$

where ϕ determines convergence speed, c_1 and c_2 are local and global learning coefficients, and ϕ_1 and ϕ_2 are positive random variables. The iteration terminates when a termination criterion (e.g., reaching the maximum iterations or minimum error criteria) is met.

To this end, with given ground vehicle densities and content request distributions, the JCTO problem can be solved effectively by our proposed CBTL algorithm.

4.5 CNN-Based Learning for Online Decision

Based on the offline optimized solutions provided by the CBTL algorithm, a CNN-based deep supervised learning scheme is designed in this section to make real-time decisions under dynamic network conditions.

4.5.1 Image-Like Input Data

As stated in Section IV, the JCTO problem is investigated with dynamic network information (e.g., $\mathbf{D}$ and $\mathbf{R}$). In this part, we adopt an image-based method to present the spatio-temporal network dynamics as images to facilitate the learning scheme.

Vehicle density $\mathbf{D}$ is a three-dimensional array, which consists of T_U two-dimensional matrices. Each two-dimensional matrix has $N_{\text{row}} \times N_{\text{col}}$ elements and can be viewed as a channel of an image. In this way, each pixel in the image corresponds to one element in the matrix. The input vehicle density can then be considered as an image with T_U channels, which differs from traditional images which commonly have three channels, i.e., RGB.

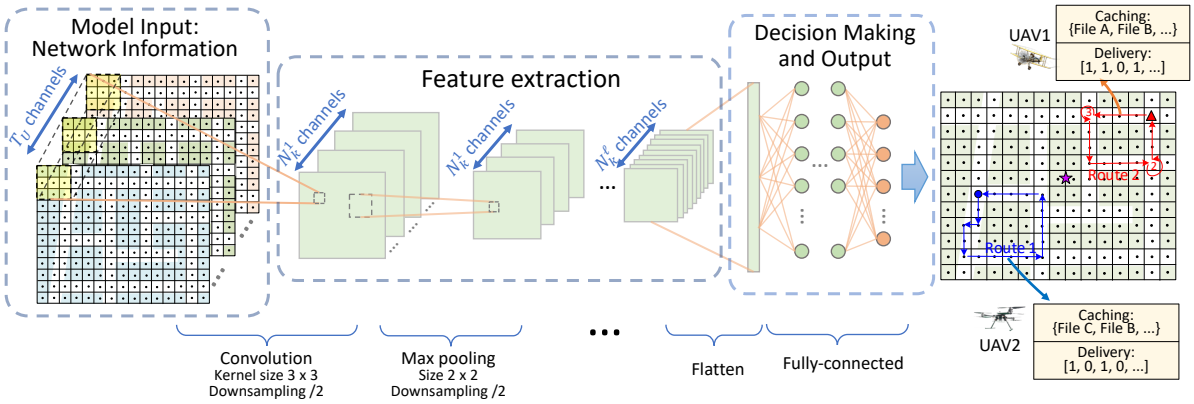


Figure 4.5: Structure of the CNN-based deep supervised learning model.

Content request distribution $\mathbf{R}$, on the other hand, is a four-dimensional array. To make the input dimension consistent without losing useful information about the request distribution, we use $\phi_{v,t}$ (as discussed in Section 4.2.1, $v = (i, j), i \in [1, N_{\text{row}}], j \in [1, N_{\text{col}}]$) to represent the content request distribution in grid v at time t. Then the three-dimensional array $\mathbf{\Phi}_{N_{\text{row}} \times N_{\text{col}} \times T_U}$, with the (i, j, t)-th entry being $\phi_{i,j,t}$, can also be treated as an image with T_U channels.

Notice that $\mathbf{D}$ and $\mathbf{\Phi}$ are of different scales. Considering that neural networks are sensitive to the scaling and distribution of their inputs, proper normalization is critical for convergence [164]. In our CNN-based learning model, the input data is normalized before fed into the learning model by using the min-max normalization as follows:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}. \tag{4.27}$$

With normalized range of the input data, the impact of $\mathbf{D}$ and $\mathbf{\Phi}$ are re-scaled to approximately the same level to facilitate the learning process.

4.5.2 CNN-Based Model Training

Fig. 4.5 shows the structure of our CNN-based learning model with three main parts, i.e., model input, feature extraction, and decision making and output.

1) Model input contains the image-like network information arrays with spatio-temporal characteristics, as explained in Section 4.5.1. Thus, the l-th input data can be written as:

$$\mathrm{IN}_l = [\mathbf{d}_1, \cdots, \mathbf{d}_t, \cdots, \mathbf{d}_{T_U}, \boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_t, \cdots, \boldsymbol{\phi}_{T_U}], \tag{4.28}$$

where $\mathbf{d}_t$ and $\boldsymbol{\phi}_t$ are $N_{\text{row}} \times N_{\text{col}}$ matrices with (i,j)-th entry being $d_{i,j,t}$ and $\phi_{i,j,t}$, respectively.

2) The extraction of the network features is accomplished by the combination of convolutional and pooling layers, which is the core part of the CNN model. In the ℓ-th convolutional layer, there are N_k^ℓ kernels (or filters) of size $W_k^\ell \times H_k^\ell$ with stride s_k^ℓ. For instance, as shown in the 1st convolutional layer in Fig. 4.5, there are N_k^1 kernels of size 3×3 with stride 2, then the input network information of size $2N_{\text{row}} \times N_{\text{col}} \times T_U$ is fed into the 1st layer to convolve with the N_k^1 kernels, producing an output of size $\left(\lfloor \frac{2N_{\text{row}}-3}{2} \rfloor + 1\right) \times \left(\lfloor \frac{N_{\text{col}}-3}{2} \rfloor + 1\right) \times N_k^1$. This output is then added by biases and activated by an activation function (such as ReLU, Sigmoid, and Softmax), which introduces nonlinearity into the system to improve the model's learning capability. After the convolutional layer, a pooling layer is introduced to downsample the convolved features by summarizing the presence of features, by using either a max-pooling function or an average-pooling function. For example, in the second layer (first pooling layer) in Fig. 4.5, we use max-pooling with size 2×2 and stride 2 to downsample the convolved features and the output is of size $\left(\left\lfloor \frac{\lfloor \frac{2N_{\text{row}}-3}{2} \rfloor - 1}{2} \right\rfloor + 1\right) \times \left(\left\lfloor \frac{\lfloor \frac{N_{\text{col}}-3}{2} \rfloor - 1}{2} \right\rfloor + 1\right) \times N_k^1$. With max-pooling layers, the data dimension can be reduced whereas dominant features can be preserved, and the level of distortion invariance can be improved. Basically, the CNN has more than one convolutional layer. With added layers, the CNN can capture not only some straightforward features (e.g., detection of road layout), but also sophisticated features (e.g., recognizing needy areas with intensive requests or undesired C2V links) of the model input.

3) In the decision making part, the extracted features are first flattened and concatenated into a column vector. Then some fully-connected layers are added to map the input data to the decision output, by learning the combinations of the extracted network features. Activation functions can be added after each fully-connected layer to introduce non-linearity and improve learning capability. Over a series of training epochs, a possibly nonlinear function between the input and output can be learnt with the CNN-based model. Notice that the output of the CNN-based model is represented by a column with length $\sum_k C_k + K \cdot (T_U + T_U \cdot 2)$, which includes:

- $\sum_k C_k$ numbers in the output show the caching status in the UAVs, with each number in range $[1, F]$ showing the index of the files being cached;
- $K \cdot T_U$ numbers in the output represent the K UAVs' delivery decisions, with each number in $\{0, 1\}$;

- $K \cdot T_U \cdot 2$ numbers in the output describe the trajectories of the K UAVs in T_U time slots, with each tuple of two numbers $(i, j) \in [1, N_{\text{row}}] \times [1, N_{\text{col}}]$ showing the location of a UAV.

4.6 Performance Evaluation

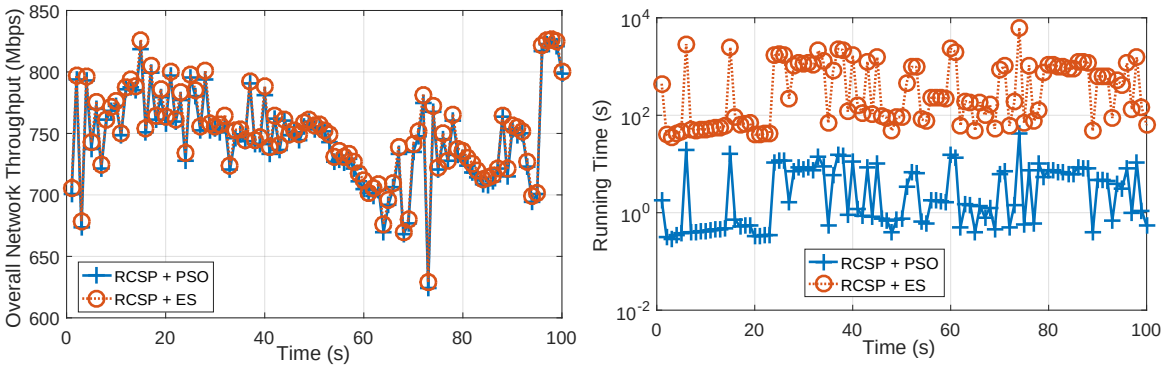
4.6.1 Experiment Settings

In this section, we perform extensive trace-driven simulations to evaluate the proposed *LB-JCTO* scheme. We adopt the Didi Chuxing GAIA Initiative dataset, which includes taxi GPS traces within the second ring road in Xi'an [149]. The dataset logs key attributes of vehicular mobility including vehicle positions, vehicle ID, and corresponding timestamps. The traffic data is aggregated every 2-4 seconds from 1 October 2016 to 31 October 2016 (31 days). Specifically, we focus on a 2000 m $\times$ 2000 m square area within longitude range $(108.9169, 108.9300)$ and latitude range $(34.2290, 34.2466)$. For UAV communications, the zone parameters and corresponding data rates are calculated as shown in [154]. The default values of main parameters related to the UAVs are: $K = 2$, $H = 30$ m, $V = 15$ m/s, $E_{k,\max} = 50$ KJ, $\xi_{LoS} = 0.99$, and $\gamma_{\max} = 37.5$ dB. For cellular communications, the default parameters are: $\alpha = 3$, $B_C = 20$ MHz, $P_{C,\max} = 20$ W, and $\sigma^2 = 10^{-15}$ W/Hz. The Zipf exponent ξ is set to 0.7 and time slot length Δ_t is set to 5s unless otherwise specified. For learning model training, we implement a learning model with 11 layers detailed as follows. The model has three convolutional layers with channel sizes of 16, 32, and 64, respectively. The kernel size and strides for each convolutional layer are (3, 3) and (2, 2), respectively. After each convolutional layer, a max-pooling layer is added with pool size (2, 2). Four fully-connected layers are then added with 1024, 1024, 512, and 256 neurons, respectively, followed by one output layer. ReLU activation function is added after each layer to introduce non-linearity and improve learning capability.

4.6.2 Evaluation of CBTL-Based Offline Optimization

The following benchmark schemes are used for performance comparison to evaluate the performance of the offline optimization CBTL algorithm.

- *Exhaustive Search (ES) Algorithm:* An ES method is used in JCTO-CL to optimize the content placement decision.



(a) Network throughput

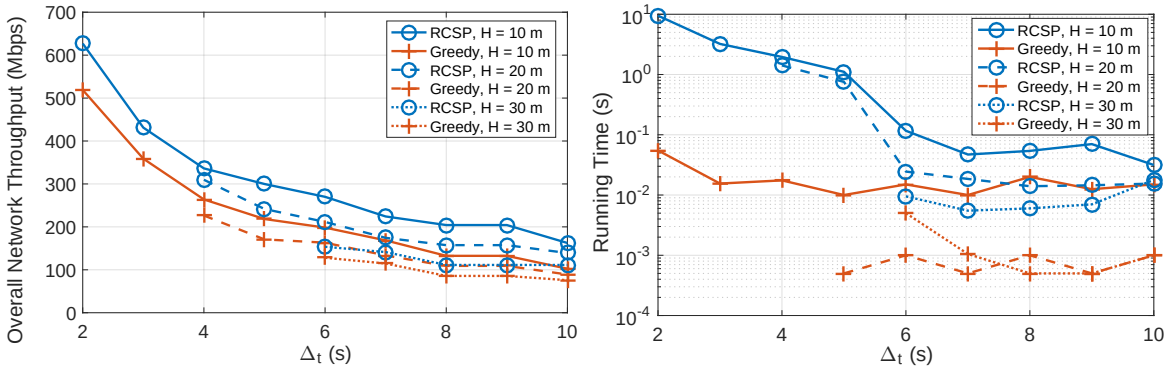
(b) Simulation running time

Figure 4.6: Comparison between PSO- and ES-based algorithms.

- *Greedy Algorithm:* UAVs fly and deliver content greedily in JCTO-TDL optimization, where the UAVs always visit nearby locations with the best throughput performance in each step. The UAVs return to the starting points when the remaining energy is barely sufficient for returning.

Fig. 4.6 shows the network throughput and execution time with the PSO- and ES-based algorithms to solve the JCTO-CL problem. As can be seen in Fig. 4.6a, applying the PSO-based method in our CBTL algorithm achieves almost the same throughput performance as applying the ES method. However, the PSO-based method is much less time-consuming than the ES algorithm, as shown in Fig. 4.6b. Therefore, instead of learning from the optimal ES algorithm, the CNN-based model in this work utilizes the CBTL algorithm as the learning supervisor to save substantial time in data labelling such that more data can be used to train the model, which is more energy-efficient.

Fig. 4.7 shows the network throughput and the corresponding execution time of applying RCSP-based and greedy-based algorithms with different values of Δ_t and H. A small Δ_t means a fine-grained CBTL optimization in time domain is conducted, whereas a large Δ_t (e.g., Δ_t equals the UAV endurance time and $T_U = 1$) is more related to the case with UAV deployment instead of trajectory design. Besides, with the simulation setting in Fig. 4.7, a lower UAV flying height corresponds to a smaller coverage area and indicates a refined division of the target area in spatial domain. It can be seen that, the network throughput and execution time both increase with smaller Δ_t and H, which can be attributed to the more sophisticated design in our CBTL algorithm. Focusing on the network throughput, we can conclude from Fig. 4.7a that applying RCSP-based method in our CBTL algorithm

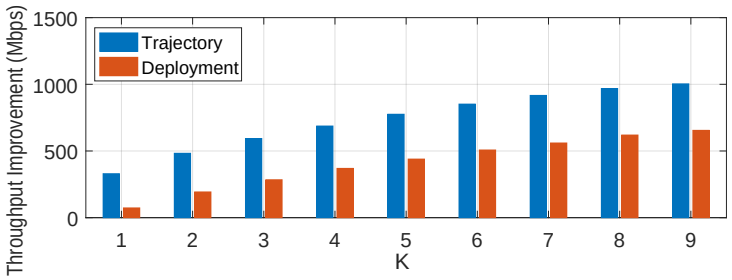


(a) Network throughput (b) Simulation running time

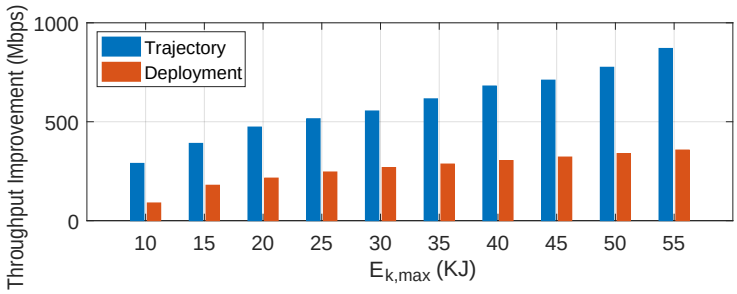
Figure 4.7: Comparison between RCSP- and greedy-based algorithms.

achieves a better performance. However, it has higher time complexity than the greedy algorithm, especially in the fine-grained optimization cases with small Δ_t and H, as shown in Fig. 4.7b. Concluding from Figs. 4.6-4.7, the proposed CBTL optimization algorithm can achieve near-optimal network throughput performance with time complexity slightly higher than the greedy-based algorithm.

Fig. 4.8 shows the impact of the number of UAVs K and the available UAV on-board energy $E_{k,\max}$ on the achievable network throughput. To further demonstrate the effectiveness of the proposed scheme, we also compare the case of UAV deployment. When UAVs are fixedly deployed, the optimal positions for UAVs are selected based on the network conditions at the first slot and then UAVs hover in those positions until energy depletion. As shown in Fig. 4.8a, the network throughput improvement increases with more UAVs dispatched into the system. However, throughput improvement introduced by each UAV diminishes with increasing K. The UAV trajectory design case outperforms the UAV deployment case, since the former is able to capture the network dynamics to ensure effective content delivery. In addition, as shown in Fig. 4.8b, a larger $E_{k,\max}$ leads to a higher network throughput for both UAV trajectory design and deployment cases since the UAVs can stay in the system longer. Notice that the throughput performance gap between the UAV trajectory and deployment cases increases with $E_{k,\max}$, since UAV trajectory design scheme is adaptive to the network variance and enables effective utilization of the energy to provide delivery services.



(a) Impact of number of UAVs



(b) Impact of available UAV on-board energy

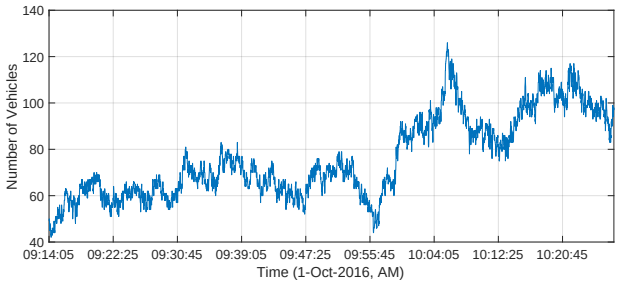
Figure 4.8: Throughput performance vs. K and $E_{k,\max}$.

4.6.3 Evaluation of EI-Based CNN Learning Model

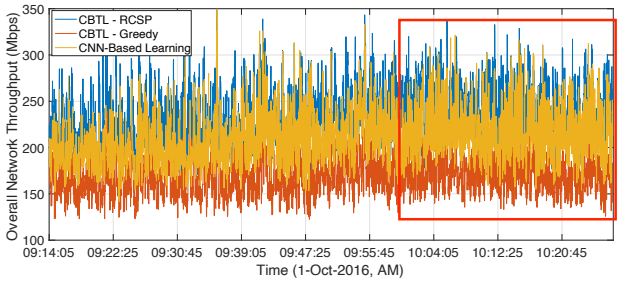
In this subsection, the achievable performance with the CNN-based learning model is evaluated. More specifically, we have trained multiple models by using different sets of trace data for performance comparison. For instance, "Model: [20, 50]", "Model: [50, 80]", "Model: $\geq$ 80", and "Model: General" are used to represent the models which are trained by using data where the number of vehicles in the target scenario is between 20 and 50, between 50 and 80, no less than 80, and unconstrained, respectively. For all the experiments, we adopt the trace data that is never used in the model training process to test the performance.

Fig. 4.9 shows the network throughput and execution time with the CNN-based learning model by using vehicle trace data during 9:15 AM $\sim$ 10:30 AM on 1 October 2016. The real-time number of vehicles in the target area is depicted in Fig. 4.9a. When using the general model to make online decisions, although the achieved network throughput outperforms that of the greedy-based algorithm, it is not as satisfactory as the RCSP-based

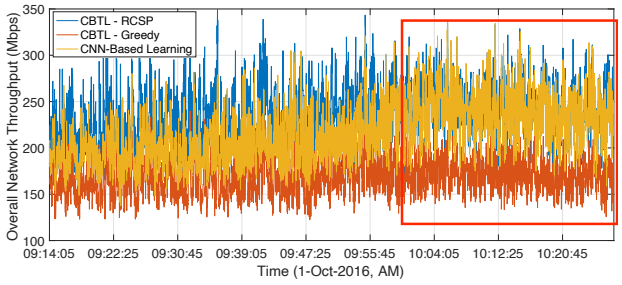
(a) Vehicle density



(b) Network throughput with "Model: General"



(c) Network throughput with "Model: ≥ 80"



(d) Simulation running time with "Model: ≥ 80"

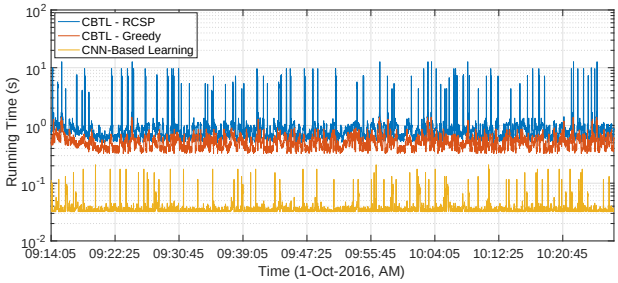


Figure 4.9: Performance for CNN-based online decision model.

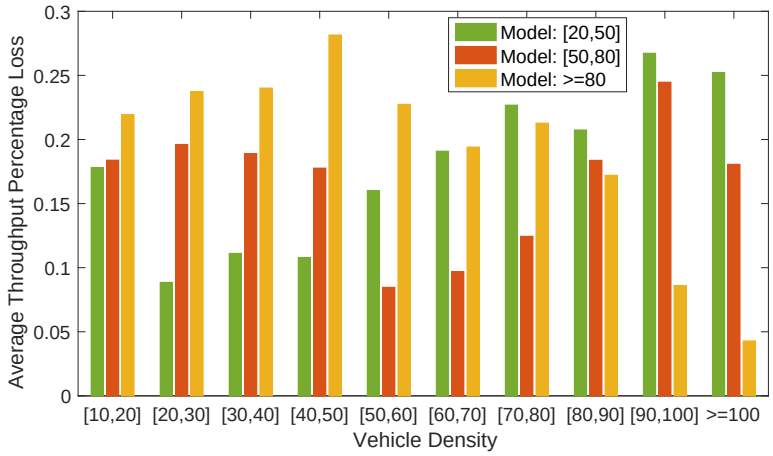


Figure 4.10: Throughput performance with density-related CNN models.

offline CBTL algorithm. Therefore, it is not always the best option to train one model to incorporate all the features in different network conditions where the vehicle density varies from 40 to 120. To make comparison, in Fig. 4.9c, the "Model: ≥ 80" is used to make the JCTO decisions. Although the throughput performance is far from ideal in the beginning, the CNN-based learning model provides almost the same network throughput as the RCSP-based optimization algorithm when the vehicle density increases over 80 (shown within the red rectangle). More importantly, as shown in Fig. 4.9d, the CNN-based learning model takes much less time when compared with the CBTL-based offline optimization algorithms. Therefore, a well-trained CNN-based model can be utilized to make online decisions with a satisfactory network throughput performance and low-complexity. However, how to select and refine the models to apply to different network conditions requires further investigation, in order to achieve the best learning performance.

Fig. 4.10 shows the achievable performance with different models under different density conditions. Specifically, average throughput percentage loss is used as a metric to evaluate the performance[6]. As shown in Fig. 4.10, "Model: [20, 50]" achieves the best performance (i.e., with the lowest throughput percentage losses) when applied to scenarios with vehicle densities falling in range [20, 50], but its performance is unsatisfactory in other cases. Similar results can be found for "Model: [50, 80]" and "Model: ≥ 80". Therefore, training multiple density-specified models and applying them in corresponding scenarios is a decent

[6]Throughput percentage loss is defined $(\hat{R}-\tilde{R})/\hat{R}$, where $\hat{R}$ and $\tilde{R}$ are the achievable network throughput of the CBTL-based offline optimization algorithm and the CNN-based learning model, respectively.

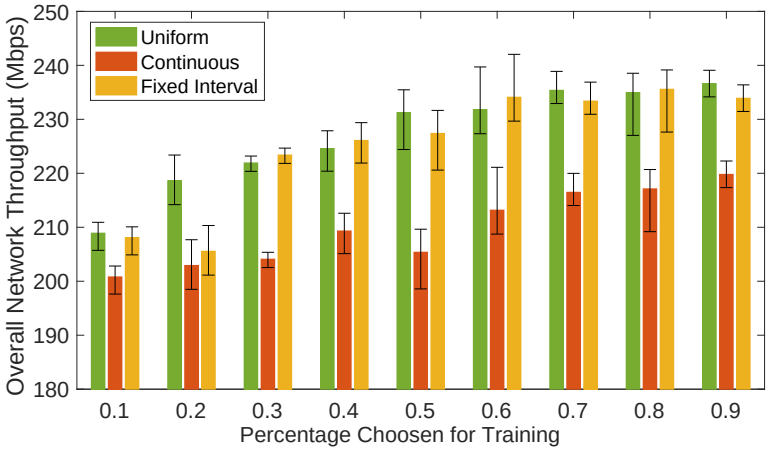


Figure 4.11: Network throughput with different methods of training data selection.

method to enhance model performance. However, a fine-grained model training can inflict significant training and storage cost. Besides, if the model granularity is too fine, the performance will be impacted since less training data is available. Therefore, the optimal number of learning models should be determined based on the throughput requirements, computing and storage capacities of devices, data availability, and so on.

Fig. 4.11 shows the impact of training data on the performance of the CNN-based models. The X axis represents the cases where $0.1 \sim 0.9$ of the available data is selected for model training. "Uniform" and "Continuous" indicate that the training data is selected uniformly and continuously from the available dataset, respectively. In "Fixed Interval" case, the training data is selected at fixed intervals, e.g., choosing the first two out of every ten pieces of data. As shown in Fig. 4.11, more data used for model training generally leads to a better throughput performance, since more information about the network features and regularities can be learnt with the CNN-based model. Among the three different methods of training data selection, the "Continuous" case achieves the worst performance since the training data is highly temporally correlated and it cannot learn all the network dynamics in time domain. On the other hand, the "Uniform" and "Fixed Interval" behave well since they are able to capture the network variance in different time intervals. Thus, to obtain a well-performed learning model, one should gather and select enough training data to learn as many potential features as possible.

4.7 Summary

In this chapter, we have investigated the joint design of UAV caching and trajectory in highly dynamic vehicular networks. As the formulated JCTO problem is non-convex and difficult to solve in a timely manner, we have proposed *LB-JCTO* to offline optimize the JCTO problem and train a learning model at the edge to make online decisions. Particularly, in the offline stage, the CBTL algorithm has been devised to solve the JCTO problem. Then a CNN-based deep supervised learning model is trained to learn the CBTL algorithm, which can be used in the online stage to make fast decisions. Extensive trace-driven experiments have been carried out to demonstrate that the CBTL algorithm can efficiently solve the JCTO problem, and the CNN-based learning model can well emulate the capability of CBTL while satisfying the real-time requirements.

Chapter 5

Load- and Mobility-Aware Cooperative Content Delivery in the SAGVN

In this chapter, we investigate cooperative content delivery in the SAGVN, where vehicular content requests can be simultaneously served by multiple APs in space, aerial, and terrestrial networks. In specific, a joint optimization problem of vehicle-to-AP association, bandwidth allocation, and content delivery ratio, referred to as the *ABC* problem, is formulated to minimize the overall content delivery delay while satisfying vehicular QoS requirements. To address the tightly-coupled optimization variables, we propose a load- and mobility-aware *ABC* (*LMA-ABC*) scheme to solve the joint optimization problem as follows. We first decompose the *ABC* problem to optimize the content delivery ratio. Then the impact of bandwidth allocation on the achievable delay performance is analyzed, and an effect of diminishing delay performance gain is revealed. Based on the analysis results, the *LMA-ABC* scheme is designed with the consideration of user fairness, load balancing, and vehicle mobility. Simulation results demonstrate that the proposed *LMA-ABC* scheme can significantly reduce the cooperative content delivery delay comparing to the benchmark schemes.

5.1 Background and Motivations

To support multifarious vehicular services with differentiated QoS requirements, the SAGVN is envisioned as a promising solution to provide global network connectivity, enhanced net-

work flexibility, and improved network reliability. In the SAGVN, CAVs should be able to access their requested data with minimal latency. To achieve this goal, different network segments in the SAGVN need to cooperatively provision vehicular content requests by leveraging heterogeneous network resources ingeniously. With cooperative content delivery, CAVs can be served by multiple APs (including the terrestrial BSs, aerial UAVs, and space satellites) simultaneously to enhance the QoS. However, achieving efficient cooperative delivery is a daunting task in the SAGVN, since significant research issues arise, including 1) vehicle-to-AP association; 2) wireless communication resource allocation; and 3) content delivery ratio optimization (i.e., determining the content delivery ratio at different APs), which are crucial to content delivery performance yet hard to be addressed due to their intercoupling relationships.

In the literature, there exist substantial studies on user association and resource allocation. In [165], the network-wide user association problem is studied in heterogeneous cellular networks, where a low-complexity algorithm is proposed to find a near-optimal solution. In [166], joint user association and resource allocation is investigated and solved distributively by employing decomposition methods. In [167], a problem of joint user association, channel allocation, antenna selection, and power control is studied. Recently, reinforcement-learning-based methods have attracted growing research attention to solving user association and resource allocation problems [168, 169]. However, these studies focus on single-AP association, where each user can be associated with at most one AP each time. This significantly constrains the communication performance including throughput and reliability [170]. In this work, we focus on a more complicated cooperative content delivery scenario with multi-AP association to enhance the QoS of vehicular content delivery, where the existing solutions cannot be applied. Furthermore, distinct characteristics of different network segments should be considered when making decisions on the vehicle-to-AP association, bandwidth allocation, and delivery ratio optimization, which are not considered in the above-mentioned existing works.

In this chapter, we focus on cooperative content delivery in the SAGVN, where LEO satellites, UAVs, and ground BSs cooperatively serve the content requests from CAVs. In specific, an ABC problem is formulated to jointly optimize the vehicle-to-AP association, bandwidth allocation, and content delivery ratio, with the objective of minimizing the overall content delivery delay while satisfying vehicular QoS requirements. The joint optimization problem is intractable due to the following reasons. First, the vehicle-to-AP association is tricky in the complicated SAGVN scenario due to the unprecedented heterogeneity and large network scale. Second, the wireless communication resource allocation should be optimized for all the service requests to guarantee the overall network performance, with the user mobility and load balancing taken into consideration. Third, the

content delivery ratio optimization should consider differentiated network characteristics including network capacity, propagation delay, traffic load, and so on. Furthermore, these decisions are tightly coupled and involve both integer and continuous variables. To tackle these challenges, we propose an *LMA-ABC* scheme to solve the joint optimization problem as follows. We first decompose the *ABC* problem to optimize the content delivery ratio by considering differentiated network characteristics including network capacity and propagation delay. For the bandwidth resource allocation, a diminishing gain effect is revealed, i.e., with more bandwidth resources allocated to the same user, the performance gain in terms of content delivery delay becomes marginal. Based on the delivery ratio optimization and the diminishing gain effect, the *LMA-ABC* scheme is designed to solve the joint optimization problem with the consideration of user fairness, load balancing, and vehicle mobility. The main contributions of this work are summarized as follows:

- We study the cooperative vehicular content delivery in the SAGVN to enable ingenious cooperation among different network segments. Specifically, we formulate the *ABC* problem, which jointly optimizes user association, spectrum resource allocation, and content delivery ratio, to reduce the overall content delivery delay, which is of paramount importance for CAV services.
- To efficiently utilize the heterogeneous network resources, we propose an *LMA-ABC* scheme to solve the *ABC* problem, where the impact of different variables on the achievable delay performance is analyzed by problem decomposition. By leveraging the interplay between vehicle mobility and heterogeneous network characteristics (e.g., network capacity and propagation delay), the *LMA-ABC* scheme can effectively solve the joint optimization problem to achieve user fairness, load balancing, and vehicle mobility.
- Extensive experimental results are conducted, and results show that the proposed *LMA-ABC* scheme can significantly reduce the overall content delivery delay comparing to the benchmark schemes.

The remainder of this chapter is organized as follows. Section 5.2 shows the system model and problem formulation. Section 5.3 provides the problem analysis to reveal the impact of different variables on the achievable content delivery delay. The proposed *LMA-ABC* scheme design is demonstrated in Section 5.4. Performance evaluation is carried out in Section 5.5 to demonstrate the performance of the proposed scheme, followed by the conclusions and future works in Section 5.6.

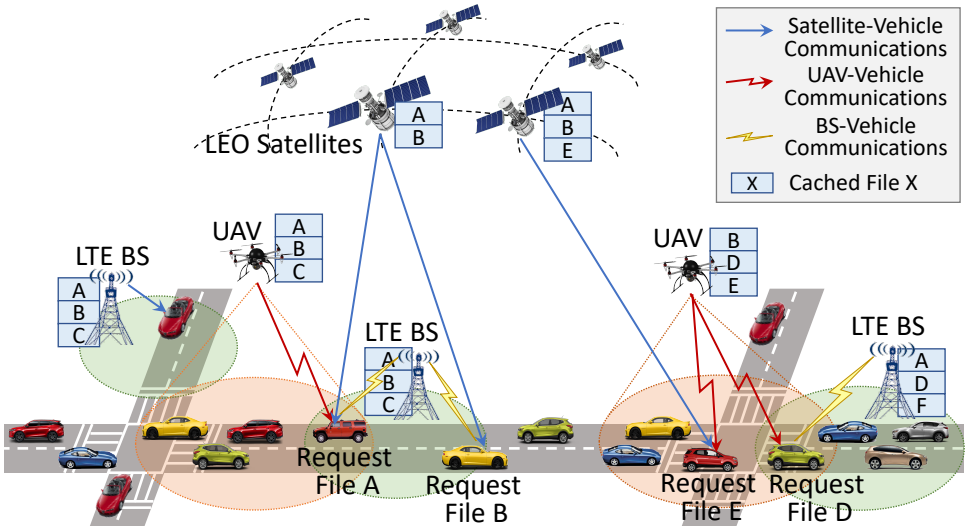


Figure 5.1: Illustration of cooperative content delivery in SAGVN

5.2 System Model and Problem Formulation

5.2.1 Scenario Description and Assumptions

We consider an SAGVN scenario with N_S LEO satellites, N_U UAVs, N_B LTE BSs, and N_V vehicles, as shown in Fig. 5.1. Let $\mathcal{SAT} = \{s_1, \cdots, s_{N_S}\}$, $\mathcal{UAV} = \{u_1, \cdots, u_{N_U}\}$, $\mathcal{BS} = \{b_1, \cdots, b_{N_B}\}$, and $\mathcal{V} = \{v_1, \cdots, v_{N_V}\}$ denote the set of LEO satellites, UAVs, BSs, and vehicles, respectively. $\mathcal{AP}_{all} = \mathcal{SAT} \cup \mathcal{UAV} \cup \mathcal{BS}$ denotes the set of all the APs. Each vehicular user is equipped with three radio interfaces for LTE, UAV, and satellite communications, respectively. Notice that satellites, UAVs, and BSs use orthogonal spectrum frequencies for communications to avoid co-channel interference. The available spectrum bandwidth for AP ap is B_{ap}, $ap \in \mathcal{AP}_{all}$. In this work, we adopt the control architecture proposed in [4], where the BSs, UAVs, and satellites are controlled by a centralized controller to perform cooperative content delivery management.

In this work, ground BSs, UAVs, and satellites are caching-enabled, i.e., some content files have already been cached in the APs to serve the vehicles. Therefore, the unstable and capacity-limited UAV backhaul and the satellite feeder links can be avoided. Considering that satellites can cover users all around the world, caching commonly popular files (e.g., pop music, hot movies, global news, etc.) on satellites can achieve a good caching performance gain. On the other hand, files cached in UAVs can be determined based on the flying trajectories and user request profiles [171], and the BSs can cache files based on the

BS-vehicle contact duration, file request pattern, and caching storage capacities [32, 172]. Denote by $\mathcal{F}$ the file library containing all the F content files, and the size of file f is denoted by ς_f, $\forall f \in \mathcal{F}$. Denote by $c_{ap,f}$ the caching indicator, $c_{ap,f} = 1$ if file f is cached in AP ap, otherwise $c_{ap,f} = 0$, $\forall ap \in \mathcal{AP}_{all}, \forall f \in \mathcal{F}$.

5.2.2 Communication Model

In the SAGVN, vehicular content requests can be served by different types of APs simultaneously. When vehicle v requests content file f, the satellites, UAVs, and BSs that have cached file f can act as the candidate APs. Let $\delta_{v,f}$ denote the content request indicator, $\delta_{v,f} = 1$ when vehicle v requests for content f, otherwise $\delta_{v,f} = 0$. Without loss of generality, each vehicle requests at most one content file at each time [114]. Considering the mobility of vehicles, UAVs, and LEO satellites, we denote the remaining contact time between vehicle v and AP ap by $T^{\text{rem}}_{ap,v}$.[1] When vehicle v requests file f, the set of candidate APs (i.e., the APs that have cached f and are available for v) is

$$\mathcal{AP}_{v,f} = \left\{ ap \;\middle|\; ap \in \mathcal{AP}_{all}, \delta_{v,f} \cdot c_{ap,f} = 1, T^{\text{rem}}_{ap,v} > 0 \right\}. \tag{5.1}$$

Let $a_{ap,v}$ be the association indicator, $a_{ap,v} = 1$ when vehicle v is associated with AP ap, and $a_{ap,v} = 0$ otherwise. When vehicle v is within the coverage of multiple APs of the same type (e.g., multiple LTE BSs/UAVs/satellites), it can connect to at most one AP from the same network segment each time, i.e.,

$$\sum_{s_i \in \mathcal{SAT}} a_{s_i,v} \leq 1, \quad \sum_{u_j \in \mathcal{UAV}} a_{u_j,v} \leq 1, \quad \sum_{b_k \in \mathcal{BS}} a_{b_k,v} \leq 1. \tag{5.2}$$

Let $\varsigma_{ap,v,f}$ be the size of file f delivered from ap ($ap \in \mathcal{AP}_{v,f}$) to vehicle v ($v \in \mathcal{V}$). Thus we have

$$0 \leq \varsigma_{ap,v,f} \leq \varsigma_f, \quad \sum_{AP \in \mathcal{AP}_{v,f}} \varsigma_{ap,v,f} = \varsigma_f. \tag{5.3}$$

[1] Vehicles can upload their locations and planned trajectories to the centralized controller, based on which $T^{\text{rem}}_{ap,v}$ can be calculated with the fixed deployment of LTE BSs and the trackable locations of UAVs and satellites.

A. BS-to-Vehicle (B2V) Communications

For a B2V communication link between BS b_k and vehicle v, the achievable signal-to-noise ratio (SNR) is derived as:

$$\Gamma_{b_k,v} = \frac{P_{b_k,v} d_{b_k,v}^{-\alpha} h_{b_k,v}}{\sigma^2}, \tag{5.4}$$

where $P_{b_k,v}, d_{b_k,v}, h_{b_k,v}, \alpha$, and σ^2 are the transmit power of BS b_k, the distance between b_k and vehicle v, the channel fading (following Rayleigh fading) from b_k to v, the pathloss exponent, and the Gaussian noise power. With an allocated spectrum bandwidth of $B_{b_k,v}$ from BS b_k, the achievable data rate of B2V communication is $R_{b_k,v} = B_{b_k,v} \log_2(1 + \Gamma_{b_k,v})$.

B. UAV-to-Vehicle (U2V) Communications

The achievable SNR of U2V communications is

$$\Gamma_{u_j,v} = \frac{P_{u_j,v} PL_{u_j,v} h_{u_j,v}}{\sigma^2}, \tag{5.5}$$

where $P_{u_j,v}$ is the transmit power of UAV u_j, $PL_{u_j,v}$ is the path loss from u_j to vehicle v consisting of LoS and NLOS components [171], and $h_{u_j,v}$ is the Rayleigh channel fading. With an allocated bandwidth of $B_{u_j,v}$ from UAV u_j, the achievable U2V data rate is $R_{u_j,v} = B_{u_j,v} \log_2(1 + \Gamma_{u_j,v})$.

C. Satellite-to-Vehicle (S2V) Communications

The achievable SNR of an S2V communication link between satellite s_i and vehicle v is

$$\Gamma_{s_i,v} = \frac{P_{s_i,v} d_{s_i,v}^{-\alpha} h_{s_i,v}}{\sigma^2}, \tag{5.6}$$

where $P_{s_i,v}$ is the transmit power of satellite s_i, $d_{s_i,v}$ is distance between s_i and vehicle v, and $h_{s_i,v}$ is the channel fading. For S2V communications, the LoS signal is a strong dominant component. Therefore, the S2V wireless channels are considered as Rician fading channels [173], with the probability density function of the channel fading as

$$f(h) = \frac{K+1}{\Omega} \exp\left\{-K - \frac{(K+1)h}{\Omega}\right\} I_0\left(2\sqrt{\frac{K(K+1)h}{\Omega}}\right), \tag{5.7}$$

where K is the ratio between the power in the LoS path and the power in the other scattered paths, Ω is the total power of the LOS and scattering signals, and $I_0(\cdot)$ is the modified Bessel function of the first kind with zero order [173]. With an allocated bandwidth of $B_{s_i,v}$ from satellite s_i, the achievable S2V data rate is $R_{s_i,v} = B_{s_i,v}\log_2(1+\Gamma_{s_i,v})$.

For B2V and U2V content delivery, the propagation delay is negligible. However, the propagation delay for satellite communications is not negligible and should be considered in S2V transmissions. Due to the trackability of the satellites, the position and the elevation angle of satellites are available at any time. For a given observation point (e.g., vehicle v), the altitude and elevation angle of LEO satellite s_i can be observed and denoted by h_{s_i} and ε_{s_i}, respectively. Thus, the S2V communication distance $d_{s_i,v}$ can be calculated as follows:

$$\begin{aligned} & r_e^2 + d_{s_i,v}^2 - 2r_e d_{s_i,v}\cos\left(\varepsilon_{s_i}+\frac{\pi}{2}\right) = (r_e+h_{s_i})^2 \\ \Rightarrow \quad & d_{s_i,v} = \sqrt{h_{s_i}^2 + 2h_{s_i}r_e + r_e^2\sin^2\varepsilon_{s_i}} - r_e\sin\varepsilon_{s_i}, \end{aligned} \tag{5.8}$$

where r_e is the earth radius. Thus, the propagation delay from s_i to v is $D_{s_i,v}^{\text{prop}} = d_{s_i,v}/c$, where c is the speed of light.

5.2.3 Problem Formulation

Aiming to minimize the overall content retrieving delay for all content requests, we investigate the ABC problem to jointly optimize 1) vehicle-to-AP association $\boldsymbol{a} = \{a_{ap,v}\}$, 2) bandwidth allocation $\boldsymbol{b} = \{B_{ap,v}\}$, and 3) content delivery ratio $\boldsymbol{\varsigma} = \{\varsigma_{ap,v,f}\}$. Notice that, when associating to multiple APs for cooperative content delivery, the overall delivery delay is determined by the worst-case link, i.e., the link with the longest delivery delay. Therefore, with cooperative delivery decisions $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{\varsigma}$, the expected delay of delivering file f to vehicle v is

$$\begin{aligned} D_{v,f} = \max\Bigg\{ & \sum_{s_i\in\mathcal{SAT}} a_{s_i,v}\left(\frac{\varsigma_{s_i,v,f}}{R_{s_i,v}} + D_{s_i,v}^{\text{prop}}\right), \sum_{u_j\in\mathcal{UAV}} \frac{a_{u_j,v}\varsigma_{u_j,v,f}}{R_{u_j,v}}, \\ & \sum_{b_k\in\mathcal{BS}} \frac{a_{b_k,v}\varsigma_{b_k,v,f}}{R_{b_k,v}} \Bigg\} + \left(1 - \max_{ap\in\mathcal{AP}_{v,f}}\{a_{ap,v}\}\right) D_{\max}. \end{aligned} \tag{5.9}$$

The last term in (5.9) indicates that, when vehicle v is not associated with any APs, the content delivery fails and the corresponding content delivery delay is $D_{\max}$, which is

a sufficiently large number to penalize the unsuccessful content delivery. To this end, the ABC problem can be formulated as:

$$\min_{\boldsymbol{a},\boldsymbol{b},\boldsymbol{\varsigma}} \quad \sum_{v\in\mathcal{V}}\sum_{f\in\mathcal{F}} \delta_{v,f} D_{v,f} \tag{5.10}$$

$$s.t. \quad a_{ap,v} \in \{0,1\}, \quad \varsigma_{ap,v,f} \in [0,\varsigma_f], \tag{5.10a}$$

$$\sum_{s_i\in\mathcal{SAT}} a_{s_i,v} \le 1, \sum_{u_j\in\mathcal{UAV}} a_{u_j,v} \le 1, \tag{5.10b}$$

$$\sum_{b_k\in\mathcal{BS}} a_{b_k,v} \le 1, \sum_{ap\in\mathcal{AP}_{v,f}} \varsigma_{ap,v,f} = \varsigma_f, \tag{5.10c}$$

$$0 \le B_{ap,v} \le B_{ap}, \quad \sum_{v\in\mathcal{V}} B_{ap,v} \le B_{ap}, \tag{5.10d}$$

$$a_{ap,v} \le \mathbb{1}_{T^{\text{rem}}_{ap,v}>0}\ c_{ap,f}\ \delta_{v,f}, \tag{5.10e}$$

$$a_{ap,v}\Gamma_{ap,v} \ge a_{ap,v}\Gamma_{th}, \quad \forall ap \in \mathcal{AP}_{all}, \forall v \in \mathcal{V}, \tag{5.10f}$$

where Γ_{th} is the minimum SNR requirement for correct data detection at the receiving vehicle. $\mathbb{1}_{condition}$ is an indicator, where $\mathbb{1}_{condition} = 1$ if the *condition* is true, and $\mathbb{1}_{condition} = 0$ otherwise. Constraint (5.10d) means that the allocated bandwidth cannot exceed the total available bandwidth resources. Constraints (5.10e) and (5.10f) indicate that vehicle v can be associated with an AP only when the AP belongs to the candidate set $\mathcal{AP}_{v,f}$ and $\Gamma_{ap,v} \ge \Gamma_{th}$.

5.3 Problem Analysis based on Decomposition

The optimization problem (5.10) is non-convex and intractable since it involves both integer and continuous variables, and decision variables $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{\varsigma}$ are tightly coupled. To solve this problem, we first decompose the optimization problem to analyze the impact of content delivery ratio and bandwidth allocation on the achievable content delivery delay performance in this section.

5.3.1 Optimization of $\boldsymbol{\varsigma}$ with Known $\boldsymbol{a}$ and $\boldsymbol{b}$

When given the AP-vehicle association and the corresponding bandwidth allocation, each vehicle v can easily calculate the achievable data rate from different APs. The associated

satellite, UAV, and BS, and the corresponding achievable data rates for vehicle v can be respectively expressed as:

$$\begin{aligned} s_i^* &= \{s_i \in \mathcal{SAT} | a_{s_i,v} = 1\}, \quad R_{s_i^*,v} = B_{s_i^*,v} \log_2(1 + \Gamma_{s_i^*,v}), \\ u_j^* &= \{u_j \in \mathcal{UAV} | a_{u_j,v} = 1\}, \quad R_{u_j^*,v} = B_{u_j^*,v} \log_2(1 + \Gamma_{u_j^*,v}), \\ b_k^* &= \{b_k \in \mathcal{BS} | a_{b_k,v} = 1\}, \quad R_{b_k^*,v} = B_{b_k^*,v} \log_2(1 + \Gamma_{b_k^*,v}). \end{aligned} \tag{5.11}$$

If vehicle v is not associated with any satellites, UAVs, or BSs, then $s_i^* = \emptyset$, $u_j^* = \emptyset$, or $b_k^* = \emptyset$. Notice that, for the case with $s_i^* = u_j^* = b_k^* = \emptyset$ or $R_{s_i^*,v} + R_{u_j^*,v} + R_{b_k^*,v} = 0$, the optimal delivery ratio is $\varsigma_{s_i^*,v} = \varsigma_{u_j^*,v} = \varsigma_{b_k^*,v} = 0$, and the corresponding content delivery delay is $D_{v,f} = D_{\max}$. For notational simplicity, we use $\mathbb{1}_{s_i^*}$, $\mathbb{1}_{u_j^*}$, and $\mathbb{1}_{b_k^*}$ to represent $\mathbb{1}_{s_i^* \neq \emptyset}$, $\mathbb{1}_{u_j^* \neq \emptyset}$, and $\mathbb{1}_{b_k^* \neq \emptyset}$, respectively. The optimal content delivery ratio for vehicle v can be derived as given in the following lemma, considering the non-trivial case where $\mathbb{1}_{s_i^*} + \mathbb{1}_{u_j^*} + \mathbb{1}_{b_k^*} \geq 1$.

Lemma 2. *With known $\boldsymbol{a}$, $\boldsymbol{b}$, and the corresponding achievable data rates, the optimal content delivery ratio is:*

$$\varsigma_{s_i^*,v,f} = \frac{\mathbb{1}_{s_i^*} R_{s_i^*,v} \max\left\{\varsigma_f - D_{s_i^*,v}^{prop}\left(\mathbb{1}_{u_j^*} R_{u_j^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v}\right), 0\right\}}{R_{s_i^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v}},$$

$$\varsigma_{b_k^*,v,f} = \frac{\mathbb{1}_{b_k^*} R_{b_k^*,v}\left[\varsigma_f + \mathbb{1}_{s_i^*} \mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + R_{b_k^*,v}} > D_{s_i^*,v}^{prop}} D_{s_i^*,v}^{prop} R_{s_i^*,v}\right]}{\mathbb{1}_{s_i^*} \mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + R_{b_k^*,v}} > D_{s_i^*,v}^{prop}} R_{s_i^*,v} + R_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v}},$$

$$\varsigma_{u_j^*,v,f} = \frac{\mathbb{1}_{u_j^*} R_{u_j^*,v}\left[\varsigma_f + \mathbb{1}_{s_i^*} \mathbb{1}_{\frac{\varsigma_f}{R_{u_j^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v}} > D_{s_i^*,v}^{prop}} D_{s_i^*,v}^{prop} R_{s_i^*,v}\right]}{\mathbb{1}_{s_i^*} \mathbb{1}_{\frac{\varsigma_f}{R_{u_j^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v}} > D_{s_i^*,v}^{prop}} R_{s_i^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v} + R_{u_j^*,v}}.$$

Proof of Lemma 2. Recall that the overall delay performance is determined by the worst-case link. Considering the unnegligible satellite propagation delay, in the following, the vehicle's content delivery ratio optimization can be divided into two cases based on whether satellites participate in the cooperative content delivery.

Case 1: $\mathbb{1}_{s_i^*} = 0$ or $\frac{\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v}} \leq D_{s_i^*,v}^{\mathbf{prop}}$:

When $s_i^* = \emptyset$ or the content delivery can be accomplished by the BS and/or the UAV with a delay shorter than the satellite propagation delay $D_{s_i^*,v}^{\text{prop}}$, the satellite does not participate

in the content delivery, and the optimal solution is

$$\begin{cases} \varsigma_{u_j^*,v,f} = \mathbb{1}_{u_j^*}\varsigma_f, \quad \varsigma_{b_k^*,v,f} = \mathbb{1}_{b_k^*}\varsigma_f, & \text{if } \mathbb{1}_{u_j^*} + \mathbb{1}_{b_k^*} = 1, \\ \frac{\varsigma_{u_j^*,v,f}}{R_{u_j^*,v}} = \frac{\varsigma_{b_k^*,v,f}}{R_{b_k^*,v}}, & \text{if } \mathbb{1}_{u_j^*} + \mathbb{1}_{b_k^*} = 2, \end{cases}$$

$$\Rightarrow \quad \begin{cases} \varsigma_{u_j^*,v,f} = \frac{\mathbb{1}_{u_j^*} R_{u_j^*,v}\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v}}, \\ \varsigma_{b_k^*,v,f} = \frac{\mathbb{1}_{b_k^*} R_{b_k^*,v}\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v}}, \\ \varsigma_{s_i^*,v,f} = 0. \end{cases}$$

Case 2: $\mathbb{1}_{s_i^*} = 1$ **and** $\frac{\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v}} > D_{s_i^*,v}^{\mathbf{prop}}$**:**

When the content delivery accomplished by the BS and/or the UAV has a delay longer than $D_{s_i^*,v}^{\text{prop}}$, the satellite can participate in content delivery to further reduce the delay. In this case, we have

$$\begin{cases} \varsigma_{s_i^*,v,f} = \varsigma_f, \quad \varsigma_{u_j^*,v,f} = \varsigma_{b_k^*,v,f} = 0, & \text{if } \mathbb{1}_{u_j^*} + \mathbb{1}_{b_k^*} = 0, \\ \frac{\varsigma_{s_i^*,v,f}}{R_{s_i^*,v}} + D_{s_i^*,v}^{\text{prop}} = \mathbb{1}_{u_j^*}\frac{\varsigma_{u_j^*,v,f}}{R_{u_j^*,v}} + \mathbb{1}_{b_k^*}\frac{\varsigma_{b_k^*,v,f}}{R_{b_k^*,v}}, & \text{if } \mathbb{1}_{u_j^*} + \mathbb{1}_{b_k^*} = 1, \\ \frac{\varsigma_{s_i^*,v,f}}{R_{s_i^*,v}} + D_{s_i^*,v}^{\text{prop}} = \frac{\varsigma_{u_j^*,v,f}}{R_{u_j^*,v}} = \frac{\varsigma_{b_k^*,v,f}}{R_{b_k^*,v}}, & \text{if } \mathbb{1}_{u_j^*} + \mathbb{1}_{b_k^*} = 2, \end{cases}$$

$$\Rightarrow \quad \begin{cases} \varsigma_{s_i^*,v,f} = \frac{R_{s_i^*,v}\varsigma_f - D_{s_i^*,v}^{\text{prop}} R_{s_i^*,v}\left(\mathbb{1}_{u_j^*} R_{u_j^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v}\right)}{R_{s_i^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v}}, \\ \varsigma_{b_k^*,v,f} = \mathbb{1}_{b_k^*}\frac{R_{b_k^*,v}\cdot\varsigma_f + D_{s_i^*,v}^{\text{prop}} R_{s_i^*,v} R_{b_k^*,v}}{R_{s_i^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v}}, \\ \varsigma_{u_j^*,v,f} = \mathbb{1}_{u_j^*}\frac{R_{u_j^*,v}\cdot\varsigma_f + D_{s_i^*,v}^{\text{prop}} R_{s_i^*,v} R_{u_j^*,v}}{R_{s_i^*,v} + \mathbb{1}_{b_k^*} R_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v}}. \end{cases}$$

Combining the above two cases, we can derive the optimal content delivery ratio as expressed in Lemma 2. □

5.3.2 Delay Performance Gain with Bandwidth Allocation

In this part, we investigate the impact of $\boldsymbol{b}$ on the achievable content delivery delay, when given $\boldsymbol{a}$.

Lemma 3 (**Diminishing Gain Effect**). *For bandwidth allocation in each AP (i.e., BS, UAV, or satellite), with more bandwidth resources allocated to the same vehicular user, the delay performance gain (i.e., delay decrement) diminishes.*

Proof of Lemma 3. Let $\boldsymbol{b}_0$ be the initial bandwidth allocation scheme. Then the optimal content delivery ratio and the corresponding overall content delivery delay performance can be calculated based on Lemma 2 and (5.9). Taking B2V communications as an example, for vehicle v which is connected with BS b_k^* for the given $\boldsymbol{a}$ and $\boldsymbol{b}_0$, its content retrieving delay is

$$D_{v,f}(\boldsymbol{a},\boldsymbol{b}_0)=\frac{\varsigma_{b_k^*,v,f}}{R_{b_k^*,v}}=\frac{\varsigma_f+\mathbb{1}_{s_i^*}\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*}R_{u_j^*,v}+R_{b_k^*,v}}>D^{\text{prop}}_{s_i^*,v}}D^{\text{prop}}_{s_i^*,v}R_{s_i^*,v}}{\mathbb{1}_{s_i^*}\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*}R_{u_j^*,v}+R_{b_k^*,v}}>D^{\text{prop}}_{s_i^*,v}}R_{s_i^*,v}+B^0_{b_k^*,v}\gamma_{b_k^*,v}+\mathbb{1}_{u_j^*}R_{u_j^*,v}},\tag{5.12}$$

where $B^0_{b_k^*,v}$ is the bandwidth allocated from BS b_k^* to vehicle v with the given $\boldsymbol{b}_0$, and $\gamma_{b_k^*,v}=\log_2(1+\Gamma_{b_k^*,v})$ is the spectrum efficiency of the B2V communication.

For a new bandwidth allocation decision $\boldsymbol{b}'$, in which BS b_k^* allocates an extra bandwidth of $\Delta B_{b_k^*,v}$ to vehicle v (the other allocation decisions keep the same with $\boldsymbol{b}_0$), the delay of delivering content f to vehicle v is

$$D_{v,f}(\boldsymbol{a},\boldsymbol{b}')=\frac{\varsigma_f+\mathbb{1}_{s_i^*}\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*}R_{u_j^*,v}+R_{b_k^*,v}}>D^{\text{prop}}_{s_i^*,v}}D^{\text{prop}}_{s_i^*,v}R_{s_i^*,v}}{\mathbb{1}_{s_i^*}\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*}R_{u_j^*,v}+R_{b_k^*,v}}>D^{\text{prop}}_{s_i^*,v}}R_{s_i^*,v}+(B^0_{b_k^*,v}+\Delta B_{b_k^*,v})\gamma_{b_k^*,v}+\mathbb{1}_{u_j^*}R_{u_j^*,v}}.\tag{5.13}$$

When the value of $\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*}R_{u_j^*,v}+R_{b_k^*,v}}>D^{\text{prop}}_{s_i^*,v}}$ keeps unchanged for decisions $\boldsymbol{b}$ and $\boldsymbol{b}'$, for notational simplicity, let

$$\mu=\varsigma_f+\mathbb{1}_{s_i^*}\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*}R_{u_j^*,v}+R_{b_k^*,v}}>D^{\text{prop}}_{s_i^*,v}}D^{\text{prop}}_{s_i^*,v}R_{s_i^*,v},$$
$$\nu=\mathbb{1}_{s_i^*}\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*}R_{u_j^*,v}+R_{b_k^*,v}}>D^{\text{prop}}_{s_i^*,v}}R_{s_i^*,v}+B^0_{b_k^*,v}\gamma_{b_k^*,v}+\mathbb{1}_{u_j^*}R_{u_j^*,v},$$

then the delay performance gain (i.e., delay decrement) is

$$\Delta D_{v,f}(\Delta B_{b_k^*,v})=D_{v,f}(\boldsymbol{a},\boldsymbol{b}_0)-D_{v,f}(\boldsymbol{a},\boldsymbol{b}')=\frac{\mu\gamma_{b_k^*,v}\Delta B_{b_k^*,v}}{\nu(\nu+\Delta B_{b_k^*,v}\gamma_{b_k^*,v})}.\tag{5.14}$$

If the value of $\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + R_{b_k^*,v}} > D^{\text{prop}}_{s_i^*,v}}$ changes, i.e., $\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + R_{b_k^*,v}} > D^{\text{prop}}_{s_i^*,v}} = 1$ for $\boldsymbol{b}$ and $\mathbb{1}_{\frac{\varsigma_f}{\mathbb{1}_{u_j^*} R_{u_j^*,v} + R_{b_k^*,v}} > D^{\text{prop}}_{s_i^*,v}} = 0$ for $\boldsymbol{b}'$, then the delay performance gain is

$$\begin{aligned}
\Delta D_{v,f}(\Delta B_{b_k^*,v}) &= \frac{\varsigma_f + \mathbb{1}_{s_i^*} D^{\text{prop}}_{s_i^*,v} R_{s_i^*,v}}{\mathbb{1}_{s_i^*} R_{s_i^*,v} + B^0_{b_k^*,v} \gamma_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v}} - \frac{\varsigma_f}{(B^0_{b_k^*,v} + \Delta B_{b_k^*,v}) \gamma_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v}} \\
&= \frac{\Delta B_{b_k^*,v} \gamma_{b_k^*,v} (\varsigma_f + \mathbb{1}_{s_i^*} D^{\text{prop}}_{s_i^*,v} R_{s_i^*,v}) - \mathbb{1}_{s_i^*} R_{s_i^*,v} \left[\varsigma_f - D^{\text{prop}}_{s_i^*,v} (B^0_{b_k^*,v} \gamma_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v}) \right]}{\left[\mathbb{1}_{s_i^*} R_{s_i^*,v} + B^0_{b_k^*,v} \gamma_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v} \right] \left[(B^0_{b_k^*,v} + \Delta B_{b_k^*,v}) \gamma_{b_k^*,v} + \mathbb{1}_{u_j^*} R_{u_j^*,v} \right]}.
\end{aligned} \tag{5.15}$$

For (5.14) and (5.15), their second derivatives of $\Delta B_{b_k^*,v}$ are both negative, which means the delay performance gain is a concave function of $\Delta B_{b_k^*,v}$. In other words, when BS b_k^* allocates more bandwidth to vehicle v, the delay performance gain diminishes. Similarly, when considering bandwidth allocation from each UAV/satellite to a vehicle, the diminishing gain effect also exists given that the others' allocation decisions are fixed, which can conclude the proof. □

The diminishing gain effect naturally promotes user fairness in our scheme design. This is consistent with the resource allocation in real systems, where improving the well-served users' performance generally has a lower priority than allocating resources to users with unsatisfactory performance.

5.4 LMA-ABC Scheme for Cooperative Content Delivery

Based on the analysis presented in Section 5.3, we propose an *LMA-ABC* scheme to solve the *ABC* problem by taking user fairness, load balancing, and vehicle mobility into consideration.

5.4.1 Posterior Association Determination

In the joint optimization problem, the vehicle-to-AP association $\boldsymbol{a}$ can significantly affect the optimization of $\boldsymbol{\varsigma}$ and $\boldsymbol{b}$, as analyzed in Sections 5.3.1 and 5.3.2. However, it is worth noting that, the optimization results of $\boldsymbol{\varsigma}$ and $\boldsymbol{b}$ can also affect the determination of $\boldsymbol{a}$:

- For a given $\boldsymbol{a}$ with $a_{ap,v} = 1$, if the bandwidth allocation decision is $B_{ap,v} = 0$ to minimize the overall delay, the association should be adjusted to avoid unnecessary association cost;

- Based on Lemma 2, there exists a special case with $a_{s_i^*,v} = 1, \varsigma_{s_i^*,v,f} = 0$ due to the long satellite propagation delay. In this case, the corresponding $a_{s_i^*,v}$ (and also $B_{s_i^*,v}$) should be adjusted to 0 to avoid undesirable resource waste.

Therefore, in the proposed scheme, to avoid inappropriate association decisions, we perform posterior association determination to decide the vehicle-to-AP association based on the optimization results of $\boldsymbol{b}$ and $\boldsymbol{\varsigma}$, i.e.,

$$a_{ap,v} = \mathbb{1}_{B_{ap,v}>0} \cdot \mathbb{1}_{\varsigma_{ap,v,f}>0}, \quad \forall ap \in \mathcal{AP}_{v,f}, \forall v \in \mathcal{V}. \tag{5.16}$$

5.4.2 Bandwidth Allocation with Diminishing Gain Effect

According to the diminishing gain property in Lemma 3, allocating a large amount of bandwidth resources to the same user is undesirable. Considering that the bandwidth resources for each AP consist of multiple sub-channels, in the proposed *LMA-ABC* scheme, we implement bandwidth allocation in the units of sub-channels. In specific, we propose a greedy-based bandwidth allocation scheme considering the diminishing gain effect to improve the content delivery performance, detailed as follows:

- **Step 1:** According to the analysis in Lemma 3, the achievable delay performance gain for an AP's bandwidth allocation decision is affected by the other APs' decisions. In other words, making bandwidth allocation decisions for one AP can greatly impact the subsequent bandwidth allocation. To avoid prioritizing the APs (i.e., with different bandwidth allocation order), where different AP priorities might affect the final delay performance, in this work, all the resources are centrally managed by the central controller.

- **Step 2:** Based on vehicles' content requests ($\delta_{v,f}$), the cached content availability ($c_{ap,f}$), and link quality ($\Gamma_{ap,v}$), we can construct a connected graph between the APs and the vehicles, where a v-to-ap connection is feasible if and only if $c_{ap,f}\ \delta_{ap,f}\ \mathbb{1}_{T^{\text{rem}}_{ap,v}>0} = 1$ and $\Gamma_{ap,v} \geq \Gamma_{th}$.

- **Step 3:** For each feasible v-to-ap connection, when a sub-channel is allocated, the corresponding $\boldsymbol{\varsigma}$ can be obtained from Lemma 2 and the delay performance gain can

Algorithm 6: Procedure of the *LMA-ABC* Scheme

Initialization:
$a_{ap,v} = 0$, $B_{ap,v} = 0$, and $\varsigma_{ap,v,f} = 0$, $\forall ap \in \mathcal{AP}_{all}, \forall v \in \mathcal{V}$.
Phase 1: Construct a connected graph $\mathcal{G}$ between the APs and the vehicles according to **Step 2** in Section 5.4.2.
Phase 2:
for *each sub-channel with bandwidth* ΔB_{ap}, $\forall ap \in \mathcal{AP}_{all}$ **do**
  for $v \in \mathcal{V}$ *and v-to-ap connection is feasible in* $\mathcal{G}$ **do**
    Calculate the optimal $\boldsymbol{\varsigma}$ based on Lemma 2.
    if *v has not been associated with any AP that has the same type with ap* **then**
      Calculate the delay performance gain $\Delta D^{all}_{v,f,ap}$ based on (5.17).
    else
      Calculate the delay performance gain $\Delta D_{v,f}(\Delta B_{ap,v})$ based on (5.14).
    end
  end
end
Select the sub-channel-vehicle association with the largest delay performance gain.
Denote the selected AP and vehicle by ap^* and v^*, respectively.
Let $a_{ap^*,v^*} = 1$ and remove connections between v^* and the other APs that have the same type with ap^* in $\mathcal{G}$.
Go to ***Phase 2*** and repeat until bandwidth resource depletes.
Phase 3: Calculate the optimal content delivery ratio $\boldsymbol{\varsigma}$ based on the final results of $\boldsymbol{a}$ and $\boldsymbol{b}$ according to Lemma 2.
Output: $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{\varsigma}$.

be calculated based on (5.14), with the other APs' bandwidth allocation decisions unchanged. Therefore, the sub-channel-vehicle association can be selected with the best delay performance gain.

- **Step 4:** Recall that one vehicle can be associated with at most one AP of the same type (i.e., at most one BS, one UAV, and one satellite). Therefore, after performing **Step 3**, the connected graph should be updated by removing all the connections between the selected vehicle and the other APs of the same type. Repeat **Step 3** and **Step 4** until bandwidth resource depletion.

5.4.3 LMA-ABC Scheme Design

The bandwidth allocation with considering diminishing gain effect can effectively guarantee user fairness and enhance delivery performance, by preventing an AP from allocating all the resources to a single user. However, the potential AP overloading problems might still occur. As shown in (5.14), the delay decrements are affected by not only the other APs' bandwidth allocation decisions, but also the channel quality (represented by the spectrum efficiency $\gamma_{b_k,v}$). Taking B2V communications as an example, if there exists a BS b_k which is the best choice with the largest $\gamma_{b_k,v}$ for all the vehicles, then getting a sub-channel from BS b_k always has a higher delay performance gain than getting a sub-channel from other BSs. This may result in load imbalance among APs, where all the vehicles are associated with BS b_k while the resources of other BSs are wasted. Another type of improper association might happen when associating vehicles to the APs with a small $T^{\text{rem}}_{ap,v}$. With vehicle mobility, the association becomes invalid shortly. For vehicles located in multiple APs' coverage overlapping area, improper association can lead to unnecessary handover, which degrades the network performance. Furthermore, it may also lead to frequent re-execution of the *LMA-ABC* scheme and consume substantial computational resources. These problems are caused by the short-sighted gain calculation which focuses only on the performance gain achieved by each sub-channel allocation.

To address these issues, we propose a far-sighted gain design in the *LMA-ABC* scheme by considering the potential future gain. When calculating the delay performance gain for a potential v-to-ap connection in **Step 3** in Section 5.4.2, if v has not been associated with any APs that have the same type with ap, we design a load- and mobility-aware gain as follows:

$$\Delta D^{all}_{v,f,ap} = \Delta D_{v,f}(\Delta B_{ap}) + \lambda \Delta D_{v,f}\left(\frac{B_{ap}}{\sum_v a_{ap,v}}\right) + \beta T^{\text{rem}}_{ap,v}, \tag{5.17}$$

where ΔB_{ap} is the bandwidth of a sub-channel allocated from ap, $\Delta D_{v,f}(\cdot)$ is the delay performance gain as defined in (5.14), T^{rem}_{ap} is the remaining contact time between v and ap, and $\lambda, \beta \in [0, 1]$ are constants. The second and third terms in the right part of (5.17) refer to the potential future gain that could be obtained when associating with ap, and the values of λ and β control the relative importance of the current gain and the future potential gain. With small λ and β, vehicles prefer to associate with the APs with good link quality; while for large λ and β, vehicles are more likely to associate with APs that are less crowded or have a long communication remaining time to achieve load-balancing and avoid unnecessary handover overhead. The detailed procedure of the proposed *LMA-ABC* scheme for cooperative content delivery can be found in Algorithm 6.

Table 5.1: Simulation Parameters

Number of LTE BSs	10
Number of UAVs	3
Number of LEO satellites	2
Altitude of satellites	781 km
Transmission power of LTE BSs $P_{b_k,v}$	28 dBm
Transmission power of UAVs	23 dBm
Transmission power of LEO satellites	10 dBW [174]
Satellite transmission antenna gain	20 dBi [174]
Satellite teceiving antenna gain	30 ddBi [174]
Pathloss exponents of B2V communications	3.5
Pathloss exponents of U2V communications	2.8
Pathloss exponents of S2V communications	2.5
Available bandwidth for each LTE BS B_{b_k}	100 MHz
Available bandwidth for each UAVB_{u_j}	100 MHz
Available bandwidth for each LEO satellite B_{s_i}	500 MHz
Size of content files ς_f	[0 Mb, 100 Mb]
SNR threshold Γ_{th}	5 dB

5.5 Performance Evaluation

In this section, we conduct simulations to evaluate the performance of the proposed *LMA-ABC* scheme. The simulations are carried out based on the real scenario of University of Waterloo campus, where the VISSIM simulation tool is used to emulate the vehicle traffic in the campus scenario. The simulation parameters are summarized in Table 5.1.

Given that the content delivery delay is dominated by file sizes and the number of requests, average delay per unit data (sec/Mbits) is used as a performance metric in our simulation for fairness consideration. All the simulation results presented in this section are averaged over 100 random testing trials. To further evaluate the effectiveness of the *LMA-ABC* scheme, the following benchmark schemes are considered for performance comparison:

- Best SNR association (BSA): All the vehicles are associated with the APs with the best SNR. The optimization of $\boldsymbol{b}$ and $\boldsymbol{\varsigma}$ keeps the same as the *LMA-ABC* scheme.

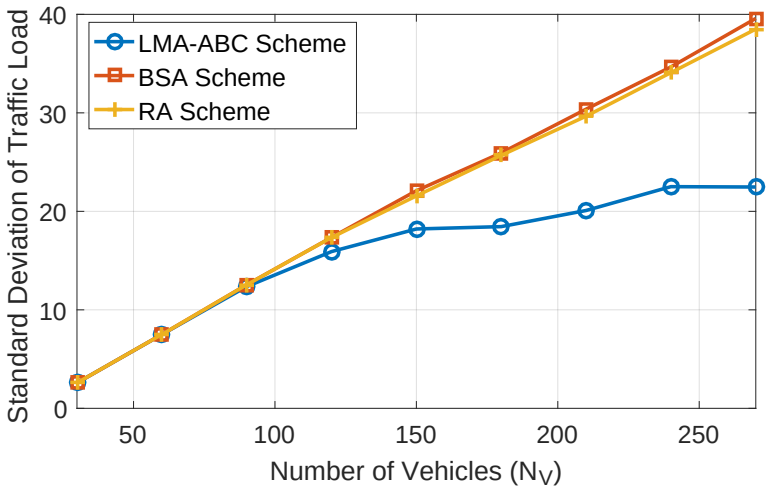


Figure 5.2: Traffic load balancing performances of schemes with different association methods

- Random association (RA): All the vehicles are randomly associated with the APs. The optimization of $\boldsymbol{b}$ and $\boldsymbol{\varsigma}$ keeps the same as the *LMA-ABC* scheme.
- Equal bandwidth allocation (EBA): Use Algorithm 6 to determine $\boldsymbol{a}$. Bandwidth resources of each AP are equally allocated to all the associated vehicles, and $\boldsymbol{\varsigma}$ is calculated based on Lemma 2.
- Equal throughput bandwidth allocation (ETBA): Use Algorithm 6 to determine $\boldsymbol{a}$. Bandwidth resources of each AP are allocated such that all the associated vehicles have the same throughput, and $\boldsymbol{\varsigma}$ is calculated based on Lemma 2.

In Fig. 5.2, we compare the traffic load balancing performance with different association schemes. Without loss of generality, we use the standard deviation (STD) of the number of vehicles associated to different APs to evaluate the traffic load balancing performance. Generally, a smaller traffic load STD indicates a more balanced traffic load distribution. As shown in Fig. 5.2, since vehicles are unevenly distributed in the scenario, with more vehicles in the scenario, the traffic load STD increases. However, the proposed *LMA-ABC* scheme can always achieve a better load balancing performance when compared to the BSA and RA schemes, and the performance gap enlarges with increasing N_V. For the BSA scheme, the network load unbalancing problem exacerbates with more vehicles in the scenario, especially in some intersection areas where a large number of vehicles are associated to the same AP. The traffic load STD of the RA scheme is slightly smaller

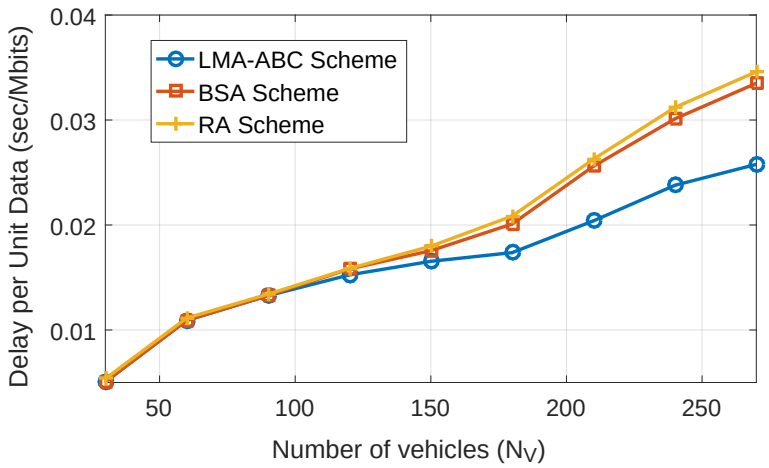


Figure 5.3: Content Delivery delay performance of schemes with different association methods

than that of the BSA scheme, but the difference is negligible. On the other hand, the *LMA-ABC* scheme achieves a significantly smaller traffic load STD since it considers the potential future gain to alleviate network load imbalance.

Figure 5.3 shows the delay performance of the schemes with different association methods. We can observe that for all the schemes, the delivery delay increases with the number of vehicles, which is reasonable since fewer resources are available for each content delivery. With a small N_V, vehicles are sparsely distributed and network resources are sufficient for content deliveries, and thus, all the schemes can achieve a good delay performance. In overloaded networks with a large N_V, the delay performance for BSA and RA schemes degrades significantly. This is caused by the traffic load unbalancing problem as explained in Fig. 5.2. On the other hand, the *LMA-ABC* scheme can outperform the other two schemes and achieve the lowest delay since it considers channel quality, load balancing, and vehicle mobility to guarantee balanced user association with good content delivery performance.

Figure 5.4 shows the delay performance of the schemes with different bandwidth allocation methods. When N_V increases, the delivery delay of the *LMA-ABC* scheme increases, which is consistent with the results in Fig. 5.3. However, the delivery delay of the EBA and ETBA schemes shows a decreasing trend. For the EBA and ETBA schemes, the traffic load and the diminishing gain effect are not taken into account for bandwidth allocation. Therefore, some vehicles with unsatisfactory channel conditions or a large content size are not allocated sufficient bandwidth resources to support the content delivery. In light-

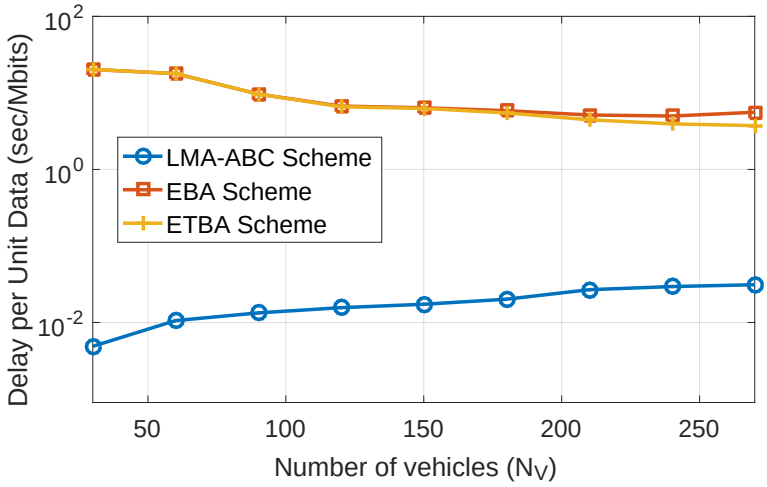


Figure 5.4: Content Delivery delay performance of schemes with different bandwidth allocation methods

loaded networks, the undesirable bandwidth allocation has an non-negligible impact on the overall delay performance. With a large N_V, although the overall delivery delay increases, the average delay per unit data decreases since the impact of the undesirable bandwidth allocation diminishes. Furthermore, the *LMA-ABC* scheme significantly outperforms the other two schemes since it considers the diminishing gain effect and load balancing to guarantee superior delay performance.

To investigate the impact of λ on the performance of the *LMA-ABC* scheme, we first demonstrate the traffic load performance with different values of λ. Focusing on the scenario with 300 vehicles, the number of vehicles associated with each AP and the traffic load STD is plotted in Fig. 5.5. From the figure, we can observe that:

- When λ is small, the impact of the future potential gain is negligible when calculating the delay performance gain of each sub-channel allocation. Therefore, all the vehicles prefer to associate with the APs with the best SNR, leading to a poor load balancing performance with a large traffic load STD. Furthermore, due to the diminishing gain effect, each AP prefers to serve multiple vehicles instead of allocating all its resources to the same vehicles. Therefore, in this case, the total number of vehicle-to-AP associations is large.

- As λ increases, the future potential gain can also affect the calculation of delay performance gain. As shown in the figure, the traffic balancing performance is significantly

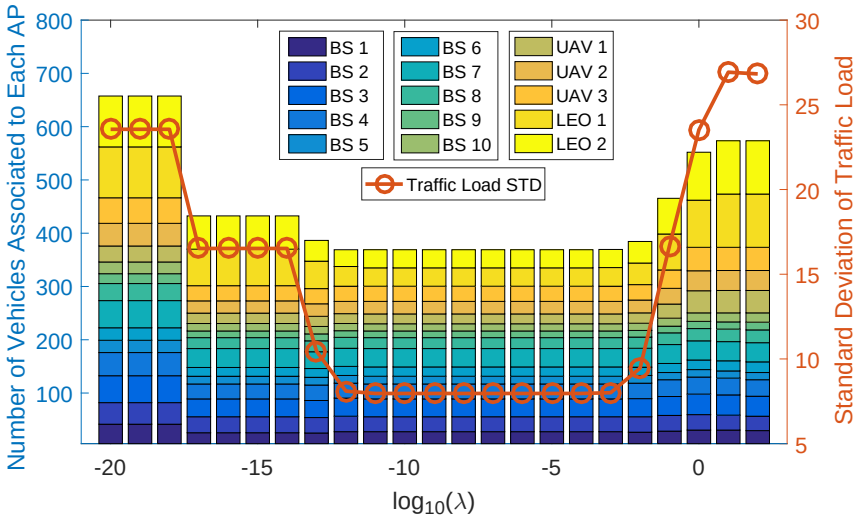


Figure 5.5: Traffic load balancing performance of the *LMA-ABC* scheme with different λ

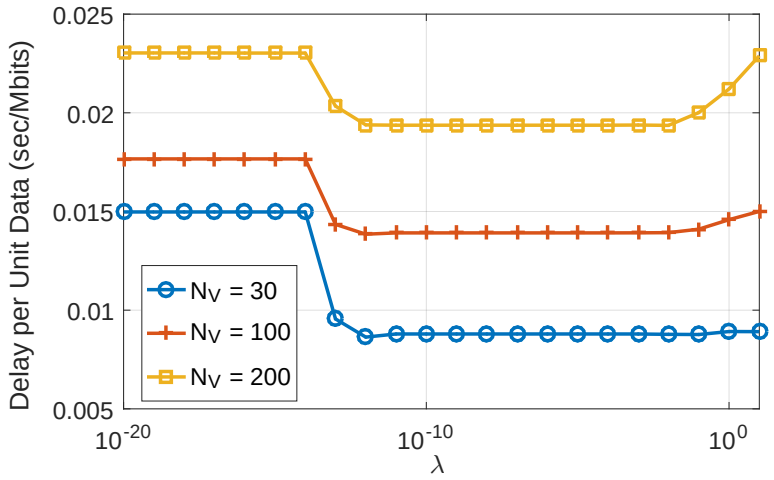


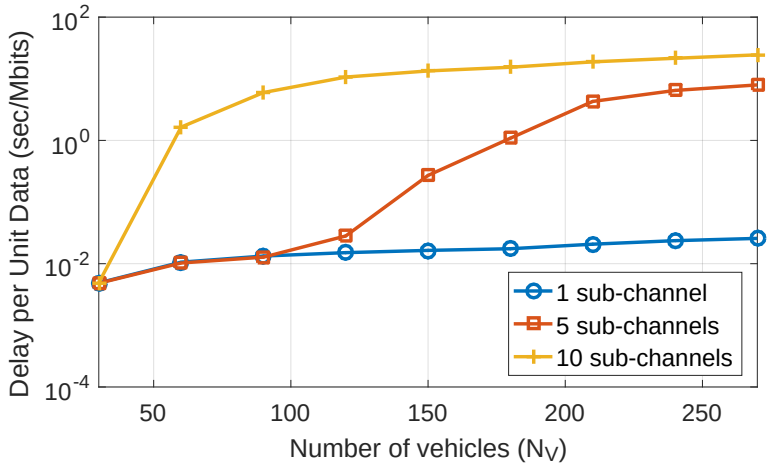
Figure 5.6: Impact of λ on the delay performance of the *LMA-ABC* scheme

enhanced with decreasing STD. In this case, vehicles are generally associated with less crowded APs with good channel conditions, and the performance gain is limited for a vehicle to associate with an already congested AP. For instance, for a vehicle connecting to a BS with a good content delivery performance, it does not need the cooperation from a congested UAV or LEO satellite, and thus leading to a decreased total number of vehicle-to-AP associations.

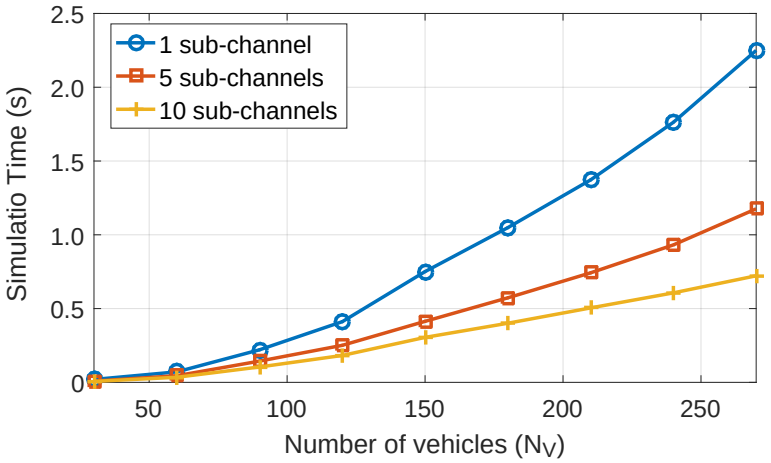
- When the value of λ keeps increasing to a sufficiently large number, the importance of the future potential gain is overemphasized. Therefore, vehicles always prefer less crowded APs regardless of the channel conditions. In this case, some vehicles are connected to APs with a poor achievable SNR and require more cooperative APs to enhance the content delivery delay performance, leading to an increasing number of vehicle-to-AP associations. Due to the different coverage radii of the APs (e.g., LEO satellites' coverage is much larger than that of an LTE BS), the number of vehicles associated with each AP varies significantly with a large traffic load STD.

Figure 5.6 shows the impact of λ on the achievable delay performance of the *LMA-ABC* scheme. When λ increases, the delay performance first decreases due to the benefit from load-balancing. When λ keeps increasing, the load balancing is overemphasized and impact of channel conditions is underestimated, leading to a delay performance degradation, which is consistent with the results in Fig. 5.5. We can also see from the figure that when λ becomes too large, the value of λ has a larger impact on the delay performance for the case with more vehicles, since the network resources are insufficient and the improper allocation can greatly affect the overall delay performance. Therefore, the value of λ should be carefully optimized to achieve a good trade-off between the load-balancing and channel conditions.

In Fig. 5.7, we compare the delay and simulation time performances of the *LMA-ABC* scheme with different bandwidth allocation granularities. As mentioned in Section 5.4, the bandwidth allocation is performed in the units of sub-channels. Therefore, the bandwidth allocation granularity greatly determines the delay performance and the decision-making complexity. Generally, with coarser bandwidth allocation granularities, e.g., allocating 5 or 10 sub-channels at a time, the delay performance degrades as shown in Fig. 5.7a, but the corresponding computational complexity also decreases as can be seen in Fig. 5.7b. It is worth noting that in scenarios with a small number of vehicles, the network resources are sufficient to support the vehicular content requests. In this case, we can moderately adjust the allocation granularities to achieve a good delay performance with a low complexity. For example, as shown in Fig. 5.7a, when the vehicle number is below 90, allocating 5 sub-channels at a time can achieve the same delay performance as the case which allocates one sub-channel at a time, but with a lower delay. Therefore, the bandwidth allocation granularity should be carefully designed in different scenarios to balance between the requirements on delay performance and time complexity.



(a) Delay Performance



(b) Simulation time

Figure 5.7: Delay and simulation time of the *LMA-ABC* scheme with different bandwidth allocation granularities

5.6 Summary

In this chapter, we have investigated the cooperative content delivery in the SAGVN to minimize the overall content delivery delay. As the formulated *ABC* problem is intractable due to the tightly coupled continuous and integer variables, we have proposed an *LMA-*

ABC scheme to jointly optimize the vehicle-to-AP association, bandwidth allocation, and content delivery ratio. With the analysis on the content delivery ratio optimization and the diminishing gain effect for bandwidth allocation, the *LMA-ABC* scheme can effectively solve the *ABC* problem by considering user fairness, load balancing, and vehicle mobility. Simulation results demonstrate that the proposed *LMA-ABC* scheme can significantly reduce the cooperative content delivery delay compared to the benchmark schemes.

Chapter 6

Conclusions and Future Works

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

6.1 Main Research Contributions

In this thesis, we have investigated caching-assisted vehicular content delivery in the HetVNet. In specific, three content caching and delivery schemes have been proposed for different HetVNet scenarios, i.e., the many-to-one matching based content placement scheme in the terrestrial HetVNet; the *LB-JCTO* scheme which jointly optimizes UAV content caching, UAV content delivery, and UAV trajectory design in the AGVN; and the *LMA-ABC* scheme which jointly addresses the user association, bandwidth allocation, and content delivery ratio optimization in the SAGVN. Heterogeneous network characteristics, UAV energy consumption, vehicle mobility patterns, and content file properties are considered by the proposed schemes. The main contributions of this thesis are summarized as follows.

1. The general framework of caching-assisted HetVNet has been proposed, where heterogeneous network segments can cooperate to enhance the vehicular content delivery performance. In specific, the impact of factors including content popularity, vehicle mobility, network service disruptions, and APs' caching capacity constraints on the achievable content delivery delay performance has been theoretically analyzed. To resist the impact of intermittent network connections, content coding is leveraged with optimized coding parameters to encode content files into packets, and the

PRAI transmission mode is applied to strike a good balance between the delay performance and the offloading ratio. Based on the analysis, a matching-based scheme with multi-objective two-sided preference lists has been proposed to optimize the content placement in heterogeneous APs. This work provides a theoretical basis for future studies related to content caching in heterogeneous networks such as the SAGVN.

2. We have investigated the joint optimization of content caching, content delivery, and UAV trajectory design in the AGVN. To find the optimal solution in real time to maximize the overall network throughput under the UAVs' energy constraints, we have proposed an *LB-JCTO* scheme. *LB-JCTO* is an offline optimization and learning for online decision framework, in which a CNN-based learning model is trained to facilitate online decisions under the supervision of offline optimized targets obtained by the CBTL algorithm. The problem formulation of JCTO and the optimization process of CBTL in this work can provide a theoretical basis for future studies related to caching-enabled UAV systems. In addition, we believe the principle of offline optimization and learning for online decisions can also be valuable for other complicated resource management in future heterogeneous networks.

3. We have proposed an *LMA-ABC* scheme for effective cooperative content delivery in the SAGVN, which jointly optimizes the user association, spectrum resource allocation, and content delivery ratio. In specific, the *LMA-ABC* scheme aims to reduce the overall content delivery delay by taking user fairness, load balancing, and vehicle mobility into account. With the proposed scheme, heterogeneous network resources in the SAGVN can be efficiently exploited to fully unleash their differential merits, and the overall content delivery delay can be significantly reduced, which is of capital importance for CAV services. Besides, the problem formulation of *ABC* and the optimization process of the *LMA-ABC* can provide a theoretical basis for future research on multi-connectivity-based SAGVNs.

6.2 Future Works

Towards enhancing the QoS of vehicular applications in the SAGVN, there still exist many open research issues. For the future research, there are some interesting and promising research directions listed as follows:

1. **SDN-based control architecture in the SAGVN** - To enable flexible, reliable, and scalable resource management in the SAGVN, an SDN-based hybrid and hierarchical control architecture has been proposed in [4]. In this control architecture,

the placement of SDN controllers is critical due to its significant impact on network performance such as communication latency, load balance of controllers, network availability, and energy consumption. The SDN controller placement problem (SCPP) involves the determination of the optimal number of SDN controllers, the best placement locations, and the division of the control domains. Basically, the SCPP optimization should consider factors such as the capacity of controllers, the network traffic loads, and control latency requirements. Therefore, the problem of how to strategically place the SDN controllers in the SAGVN should be carefully investigated to guarantee reliable network control with low signaling overhead and short control latency.

2. **Service-oriented multi-dimensional resource orchestration** - Basically, different vehicular services have differentiated requirements on communication, computing, and caching resources, each of which can be provided by different SAGVN network infrastructure (e.g., terrestrial BSs, UAVs, satellites). To achieve the optimal network performance and increase resource utilization efficiency, a service-oriented resource orchestration scheme should be developed in the SAGVN by considering the following factors. 1) The space, air, and terrestrial network segments have distinctive characteristics in terms of communication delay, throughput, coverage, computing capability, jitter, and so on; 2) Multiple types of mobility introduced by the vehicles, UAVs, and LEO satellites lead to dynamic network resource availability, and the spatial-temporal variations in communication link conditions affect the required resources to satisfy the QoS requirements; and 3) Due to the differentiated QoS requirements, the priorities for various vehicular applications to access the network resources from different network segments are different. Therefore, service-oriented multi-dimensional resource orchestration solutions are imperative but challenging.

3. **Artificial intelligence (AI)-based network management** - In the SAGVN, modeling the dynamic and complex network system is painstaking, if not impossible. Furthermore, traditional model- and optimization-based resource management approaches are inadequate due to the high complexity. Thus, innovative AI-based engineering solutions are necessary to make high-quality and real-time decisions to keep pace with the dynamic environment. Particularly, distributed AI can be adopted to enable intelligent decision-making at different granular levels through parallel training processes. When designing the distributed AI-based network management schemes, the appropriate splitting of data and model, communication overhead of model updating, and the computing capability and energy constraints of different network nodes should be taken into consideration. Some recently proposed distributed AI paradigms, e.g., federated learning and split learning, can also be considered.

References

[1] J. Yao, T. Han, and N. Ansari, "On mobile edge caching," *IEEE Commun. Surveys Tut.*, vol. 21, no. 3, pp. 2525–2553, Mar. 2019.

[2] F. Lyu, N. Cheng, H. Zhu, H. Zhou, W. Xu, M. Li, and X. Shen, "Towards rear-end collision avoidance: Adaptive beaconing for connected vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 2, pp. 1248–1263, Feb. 2021.

[3] A. Bazzi, C. Campolo, B. M. Masini, A. Molinaro, A. Zanella, and A. O. Berthet, "Enhancing cooperative driving in IEEE 802.11 vehicular networks through full-duplex radios," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2402–2416, Apr. 2018.

[4] H. Wu, J. Chen, C. Zhou, W. Shi, N. Cheng, W. Xu, W. Zhuang, and X. Shen, "Resource management in space-air-ground integrated vehicular networks: SDN control and AI algorithm design," *IEEE Wireless Commun.*, vol. 27, no. 6, pp. 52–60, Dec. 2020.

[5] International Data Corporation (IDC). A new IDC forecast shows how vehicles will gradually incorporate the technologies that lead to autonomy. (Accessed: Feb. 2021). [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=prUS46887020&utm_medium=rss_feed&utm_source=Alert&utm_campaign=rss_syndication

[6] J. Wang, J. Liu, and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Commun. Surveys Tut.*, vol. 21, no. 2, pp. 1243–1274, Dec. 2018.

[7] Z. MacHardy, A. Khan, K. Obana, and S. Iwashina, "V2X access technologies: Regulation, research, and remaining challenges," *IEEE Commun. Surveys Tut.*, vol. 20, no. 3, pp. 1858–1877, Feb. 2018.

[8] C. Xu and Z. Zhou, "Vehicular content delivery: A big data perspective," *IEEE Wireless Commun.*, vol. 25, no. 1, pp. 90–97, Feb. 2018.

[9] Cisco. Cisco visual networking index: Global mobile data traffic forecast update, 2017–2022. (Accessed: Feb. 2021). [Online]. Available: https://s3.amazonaws.com/media.mediapost.com/uploads/CiscoForecast.pdf

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

[11] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. Shen, "Software defined space-air-ground integrated vehicular networks: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 101–109, Jul. 2017.

[12] T. Hong, W. Zhao, R. Liu, and M. Kadoch, "Space-air-ground IoT network and related key technologies," *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 96–104, Apr. 2020.

[13] S. Zhou, G. Wang, S. Zhang, Z. Niu, and X. Shen, "Bidirectional mission offloading for agile space-air-ground integrated networks," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 38–45, Apr. 2019.

[14] J. Qiu, D. Grace, G. Ding, M. D. Zakaria, and Q. Wu, "Air-ground heterogeneous networks for 5G and beyond via integrating high and low altitude platforms," *IEEE Wireless Commun.*, vol. 26, no. 6, pp. 140–148, Dec. 2019.

[15] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May 2016.

[16] Y. Su, Y. Liu, Y. Zhou, J. Yuan, H. Cao, and J. Shi, "Broadband LEO satellite communications: Architectures and key technologies," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 55–61, Apr. 2019.

[17] H. Nawaz, H. M. Ali, and A. A. Laghari, "UAV communication networks issues: A review," *Archives of Computational Methods in Engineering*, pp. 1–21, Mar. 2020.

[18] B. Di, L. Song, Y. Li, and H. V. Poor, "Ultra-dense LEO: Integration of satellite access networks into 5G and beyond," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 62–69, Apr. 2019.

[19] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 131–139, Feb. 2014.

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

[21] T. H. Luan, L. X. Cai, J. Chen, X. Shen, and F. Bai, "Engineering a distributed infrastructure for large-scale cost-effective content dissemination over urban vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 3, pp. 1419–1435, Mar. 2014.

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

[23] L. Wang, H. Wu, Z. Han, P. Zhang, and H. V. Poor, "Multi-hop cooperative caching in social IoT using matching theory," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2127–2145, Apr. 2018.

[24] S. Traverso, M. Ahmed, M. Garetto, P. Giaccone, E. Leonardi, and S. Niccolini, "Temporal locality in today's content caching: why it matters and how to model it," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 5, pp. 5–12, Nov. 2013.

[25] A. Ioannou and S. Weber, "A survey of caching policies and forwarding mechanisms in information-centric networking," *IEEE Commun. Surveys Tut.*, vol. 18, no. 4, pp. 2847–2886, May 2016.

[26] H. Ahlehagh and S. Dey, "Video-aware scheduling and caching in the radio access network," *IEEE/ACM IEEE/ACM Trans. Netw.*, vol. 22, no. 5, pp. 1444–1462, Jan. 2014.

[27] G. Qiao, S. Leng, S. Maharjan, Y. Zhang, and N. Ansari, "Deep reinforcement learning for cooperative content caching in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 247–257, Oct. 2019.

[28] R. Wang, X. Peng, J. Zhang, and K. B. Letaief, "Mobility-aware caching for content-centric wireless networks: Modeling and methodology," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 77–83, Aug. 2016.

[29] L. Yao, A. Chen, J. Deng, J. Wang, and G. Wu, "A cooperative caching scheme based on mobility prediction in vehicular content centric networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5435–5444, Dec. 2017.

[30] B. Zhou, Y. Cui, and M. Tao, "Stochastic content-centric multicast scheduling for cache-enabled heterogeneous cellular networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6284–6297, Jun. 2016.

[31] F. Guo, H. Zhang, X. Li, H. Ji, and V. C. M. Leung, "Joint optimization of caching and association in energy-harvesting-powered small-cell networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6469–6480, Jul. 2018.

[32] H. Wu, J. Chen, W. Xu, N. Cheng, W. Shi, L. Wang, and X. Shen, "Delay-minimized edge caching in heterogeneous vehicular networks: A matching-based approach," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6409–6424, Oct. 2020.

[33] W. Shi, J. Li, H. Wu, C. Zhou, N. Cheng, and X. Shen, "Drone-cell trajectory planning and resource allocation for highly mobile networks: A hierarchical DRL approach," *IEEE Internet Things J.*, DOI: 10.1109/JIOT.2020.3020067, Aug. 2020.

[34] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tut.*, vol. 18, no. 2, pp. 1123–1152, Nov. 2015.

[35] T. Xia, M. M. Wang, and X. You, "Satellite machine-type communication for maritime internet of things: An interference perspective," *IEEE Access*, vol. 7, pp. 76 404–76 415, May 2019.

[36] T. Alruhaili, G. Aldabbagh, F. Bouabdallah, N. Dimitriou, and M. Win, "Performance evaluation for Wi-Fi offloading schemes in LTE networks," *Int. J. Comput. & Inform. Sci.*, vol. 12, no. 1, pp. 121–131, Sept. 2016.

[37] N. Wang and J. Wu, "Opportunistic WiFi offloading in a vehicular environment: Waiting or downloading now?" in *Proc. IEEE INFOCOM 2016*, San Francisco, CA, USA, Apr. 2016.

[38] W. Xu, H. A. Omar, W. Zhuang, and X. Shen, "Delay analysis of in-vehicle internet access via on-road WiFi access points," *IEEE Access*, vol. 5, pp. 2736–2746, Feb. 2017.

[39] Cisco. Cisco annual internet report (2018–2023). (Accessed: Feb. 2021). [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf

[40] A. Aijaz, A. H. Aghvami, and M. Amani, "A survey on mobile data offloading: Technical and business perspectives," *IEEE Wireless Commun.*, vol. 20, no. 2, pp. 104–112, Apr. 2013.

[41] S.-S. Tzeng and Y.-J. Lin, "Delay-constrained data transmission with minimal energy consumption in cognitive radio/WiFi vehicular networks," *Wireless Personal Commun.*, vol. 107, no. 4, pp. 1777–1797, Apr. 2019.

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

[43] A. Giannoulis, M. Fiore, and E. W. Knightly, "Supporting vehicular mobility in urban multi-hop wireless networks," in *Proc. 6th Int. Conf. Mobile Syst. Appl. Services*, Breckenridge, CO, USA, Jun. 2008.

[44] P. Lv, X. Wang, X. Xue, and M. Xu, "SWIMMING: Seamless and efficient WiFi-based internet access from moving vehicles," *IEEE Trans. Mobile Comput.*, vol. 14, no. 5, pp. 1085–1097, May 2015.

[45] A. Balasubramanian, R. Mahajan, A. Venkataramani, B. N. Levine, and J. Zahorjan, "Interactive WiFi connectivity for moving vehicles," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 4, pp. 427–438, Oct. 2008.

[46] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting mobile 3G using WiFi: Measurement, system design, and implementation," in *Proc. ACM MobiSys*, San Francisco, CA, USA, Jun. 2010.

[47] W. Xu, H. Wu, J. Chen, W. Shi, H. Zhou, N. Cheng, and X. Shen, "ViFi: Vehicle-to-vehicle assisted traffic offloading via roadside WiFi networks," in *Proc. IEEE GLOBECOM 2018*, Abu Dhabi, UAE, Dec. 2018.

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

[49] N. Lu, N. Cheng, N. Zhang, X. Shen, J. W. Mark, and F. Bai, "Wi-Fi hotspot at signalized intersection: Cost-effectiveness for vehicular internet access," *IEEE Trans. Veh. Technol.*, vol. 65, no. 5, pp. 3506–3518, May 2016.

[50] W. Zhang, J. Yang, G. Zhang, L. Yang, and C. K. Yeo, "TV white space and its applications in future wireless networks and communications: A survey," *IET Commun.*, vol. 12, no. 20, pp. 2521–2532, Dec. 2018.

[51] Y. Han, E. Ekici, H. Kremo, and O. Altintas, "Vehicular networking in the TV white space band: Challenges, opportunities, and a media access control layer of access issues," *IEEE Veh. Technol. Mag.*, vol. 12, no. 2, pp. 52–59, Jun. 2017.

[52] H. Harada, "White space communication systems: An overview of regulation, standardization and trial," *IEICE Trans. Commun.*, vol. E97.B, no. 2, pp. 261–274, Feb. 2014.

[53] IEEE Std 802.11af-2013, "Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: Amendment 5: Television white spaces operation," IEEE Standard 802.11, Feb. 2014.

[54] IEEE Computer Society, "IEEE standard for information technology telecommunications - part 22: Cognitive wireless ran medium access control (MAC) and physical layer (PHY) specifications: Policies and procedures for operation in the TV bands," IEEE Standard 802.22b, Oct. 2015.

[55] ETSI, "White space devices (WSD) wireless access systems operating in the 470 MHz to 790 MHz TV broadcast band," ETSI EN 301 598, Feb. 2014.

[56] IEEE Computer Society, "IEEE standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements - part 19: TV white space coexistence methods," IEEE Std 802.19.1, Jun. 2014.

[57] O. Altintas, Y. Ihara, H. Kremo, H. Tanaka, M. Ohtake, T. Fujii, C. Yoshimura, K. Ando, K. Tsukamoto, M. Tsuru, and Y. Oie, "Field tests and indoor emulation of distributed autonomous multi-hop vehicle-to-vehicle communications over TV white space," in *Proc. ACM 18th Annu. Int. Conf. on Mobile Comput. and Netw.*, Istanbul, Turkey, Aug. 2012, pp. 439–442.

[58] L. Bedogni, M. D. Felice, F. Malabocchia, and L. Bononi, "Cognitive modulation and coding scheme adaptation for 802.11n and 802.11af networks," in *Proc. IEEE GLOBECOM 2014 Workshop*, Austin, TX, USA, Dec. 2014.

[59] L. Bedogni, A. Trotta, M. D. Felice, Y. Gao, X. Zhang, Q. Zhang, F. Malabocchia, and L. Bononi, "Dynamic adaptive video streaming on heterogeneous TVWS and Wi-Fi networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 6, pp. 3253–3266, Aug. 2017.

[60] H. Zhou, N. Zhang, Y. Bi, Q. Yu, X. Shen, D. Shan, and F. Bai, "TV white space enabled connected vehicle networks: Challenges and solutions," *IEEE Netw.*, vol. 31, no. 3, pp. 6–13, May 2017.

[61] J.-H. Lim, K. Naito, J.-H. Yun, and M. Gerla, "Reliable safety message dissemination in NLOS intersections using TV white spectrum," *IEEE Trans. Mobile Comput.*, vol. 17, no. 1, pp. 169–182, Jan. 2018.

[62] W. Shi, J. Li, W. Xu, H. Zhou, N. Zhang, S. Zhang, and X. Shen, "Multiple drone-cell deployment analyses and optimization in drone assisted radio access networks," *IEEE Access*, vol. 6, pp. 12 518–12 529, Feb. 2018.

[63] H. Wu, X. Tao, N. Zhang, and X. Shen, "Cooperative UAV cluster-assisted terrestrial cellular networks for ubiquitous coverage," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2045–2058, Sept. 2018.

[64] L. Zhu, J. Zhang, Z. Xiao, X. Cao, X.-G. Xia, and R. Schober, "Millimeter-wave full-duplex UAV relay: Joint positioning, beamforming, and power control," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 9, pp. 2057–2073, Jun. 2020.

[65] M. Chen, W. Saad, and C. Yin, "Liquid state machine learning for resource and cache management in LTE-U unmanned aerial vehicle (UAV) networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1504–1517, Jan. 2019.

[66] B. Wang, R. Zhang, C. Chen, X. Cheng, and Y. Jin, "Density-aware deployment with multi-layer UAV-V2X communication networks," *IET Commun.*, vol. 14, no. 16, pp. 2709–2715, Jul. 2020.

[67] S. Mignardi, C. Buratti, A. Bazzi, and R. Verdone, "Trajectories and resource management of flying base stations for C-V2X," *IEEE Sensors J.*, vol. 19, no. 4, p. 811, Feb. 2019.

[68] M. S. Shokry, D. Ebrahimi, C. Assi, S. Sharafeddine, and A. Ghrayeb, "Leveraging UAVs for coverage in cell-free vehicular networks: A deep reinforcement learning approach," *IEEE Trans. Mobile Comput.*, Apr. 2020.

[69] L. Deng, G. Wu, J. Fu, Y. Zhang, and Y. Yang, "Joint resource allocation and trajectory control for UAV-enabled vehicular communications," *IEEE Access*, vol. 7, pp. 132 806–132 815, Sept. 2019.

[70] A. Al-Hilo, M. Samir, C. Assi, S. Sharafeddine, and D. Ebrahimi, "UAV-assisted content delivery in intelligent transportation systems-joint trajectory planning and cache management," *IEEE Trans. Intell. Transp. Syst.*, Sept. 2020.

[71] O. Abbasi, H. Yanikomeroglu, A. Ebrahimi, and N. M. Yamchi, "Trajectory design and power allocation for drone-assisted NR-V2X network with dynamic NOMA/OMA," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7153–7168, Jul. 2020.

[72] L. Zhang, Z. Zhao, Q. Wu, H. Zhao, H. Xu, and X. Wu, "Energy-aware dynamic resource allocation in UAV assisted mobile edge computing over social internet of vehicles," *IEEE Access*, vol. 6, pp. 56 700–56 715, Oct. 2018.

[73] W. Fawaz, R. Atallah, C. Assi, and M. Khabbaz, "Unmanned aerial vehicles as store-carry-forward nodes for vehicular networks," *IEEE Access*, vol. 5, pp. 23 710–23 718, Oct. 2017.

[74] L. Xiao, X. Lu, D. Xu, Y. Tang, L. Wang, and W. Zhuang, "Uav relay in VANETs against smart jamming with reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4087–4097, Jan. 2018.

[75] O. S. Oubbati, N. Chaib, A. Lakas, P. Lorenz, and A. Rachedi, "UAV-assisted supporting services connectivity in urban VANETs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3944–3951, Feb. 2019.

[76] Y. He, D. Zhai, Y. Jiang, and R. Zhang, "Relay selection for UAV-assisted urban vehicular ad hoc networks," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1379–1383, APR. 2020.

[77] Starlink. (Accessed: Feb. 2021). [Online]. Available: https://en.wikipedia.org/wiki/Starlink

[78] OneWeb Home Page. (Accessed: Feb. 2021). [Online]. Available: https://www.oneweb.world/launches

[79] Telesat. (Accessed: Feb. 2021). [Online]. Available: https://www.telesat.com/leo-satellites/

[80] 3GPP, "Study on scenarios and requirements for next generation access technologies, version 15.0.0, release 15," Standard TR 38.913, Jul. 2018.

[81] ——, "Service requirements for the 5G system; stage 1, version 17.2.0, release 17," Standard TS 22.261, 3GPP, Mar. 2020.

[82] ——, "Study on new radio (NR) to support non-terrestrial networks, version 15.2.0, release 15," Standard TR 38.811, 3GPP, Oct. 2019.

[83] ——, "Study on using satellite access in 5G; stage 1. version 16.0.0. release 16," Standard TR 22.822, 3GPP, Jul. 2018.

[84] ETSI, Satellite Earth Stations and Systems (SES), "Combined satellite and terrestrial networks scenarios, document v1.1.1," ETSI TR 103 124, Jul. 2013.

[85] ——, "Overview of present satellite emergency communications resources, v1.2.2," ETSI TR 102 641, Aug. 2013.

[86] ——, "Multi-link routing scheme in hybrid access network with heterogeneous links, v1.1.1," ETSI TR 103 351, Jul. 2017.

[87] ETSI, Broadband Radio Access Networks (BRAN), "Broadband wireless access and backhauling for remote rural communities. v1.1.1," ETSI TR 103 293, Jul. 2015.

[88] ETSI, Satellite Earth Stations and Systems (SES), "Broadband satellite multimedia (BSM); common air interface specification; satellite independent service access point (SISAP) interface: Primitives. v1.2.1," ETSI TS 102 357, May 2015.

[89] K. Liolis, G. Schlueter, J. Krause, F. Zimmer, L. Combelles, J. Grotz, S. Chatzinotas, B. Evans, A. Guidotti, D. Tarchi *et al.*, "Cognitive radio scenarios for satellite communications: The CoRaSat approach," in *Proc. Future Netw. & Mobile Summit*. Lisboa, Portugal: IEEE, Oct. 2013.

[90] SANSA Home Page. (Accessed: Feb. 2021). [Online]. Available: https://sansa-h2020.eu/

[91] VIrtualized hybrid satellite-TerrestriAl systems for resilient and flexible future networks. (Accessed: Feb. 2021). [Online]. Available: https://cordis.europa.eu/project/id/644843

[92] SATNEX IV Home Page. (Accessed: Feb. 2021). [Online]. Available: https://artes.esa.int/projects/satnex-iv

[93] Sat 5G Home Page. (Accessed: Feb. 2021). [Online]. Available: http://sat5g-project.eu/

[94] SATis5 Home Page. (Accessed: Feb. 2021). [Online]. Available: https://satis5.eurescom.eu/

[95] Z. Niu, X. S. Shen, Q. Zhang, and Y. Tang, "Space-air-ground integrated vehicular network for connected and automated vehicles: Challenges and solutions," *Intell. and Converged Netw.*, vol. 1, no. 2, pp. 142–169, Dec. 2020.

[96] Z. Zhou, J. Feng, C. Zhang, Z. Chang, Y. Zhang, and K. M. S. Huq, "SAGECELL: Software-defined space-air-ground integrated moving cells," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 92–99, Aug. 2018.

[97] G. Gür and S. Kafiloğlu, "Layered content delivery over satellite integrated cognitive radio networks," *IEEE Wireless Commun. Lett.*, vol. 6, no. 3, pp. 390–393, Apr. 2017.

[98] X. Zhu, C. Jiang, L. Yin, L. Kuang, N. Ge, and J. Lu, "Cooperative multi-group multicast transmission in integrated terrestrial–satellite networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 5, pp. 981–992, May 2018.

[99] B. Di, H. Zhang, L. Song, Y. Li, and G. Y. Li, "Ultra-dense LEO: Integrating terrestrial-satellite networks into 5G and beyond for data offloading," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 47–62, Dec. 2018.

[100] S. Yu, X. Gong, Q. Shi, X. Wang, and X. Chen, "EC-SAGINs: Edge computing-enhanced space-air-ground integrated networks for internet of vehicles," *IEEE Internet Things J.*, DOI: 10.1109/JIOT.2021.3052542, Jan. 2021.

[101] G. Wang, S. Zhou, and Z. Niu, "Radio resource allocation for bidirectional offloading in space-air-ground integrated vehicular network," *J. of Commun. and Inf. Netw.*, vol. 4, no. 4, pp. 24–31, Dec. 2019.

[102] J. Liu, Y. Shi, Z. M. Fadlullah, and N. Kato, "Space-air-ground integrated network: A survey," *IEEE Commun. Surveys Tut.*, vol. 20, no. 4, pp. 2714–2741, May 2018.

[103] N. Cheng, W. Quan, W. Shi, H. Wu, Q. Ye, H. Zhou, W. Zhuang, X. Shen, and B. Bai, "A comprehensive simulation platform for space-air-ground integrated network," *IEEE Wireless Commun.*, vol. 27, no. 1, pp. 178–185, Feb. 2020.

[104] N. Kato, Z. M. Fadlullah, F. Tang, B. Mao, S. Tani, A. Okamura, and J. Liu, "Optimizing space-air-ground integrated networks by artificial intelligence," *IEEE Wireless Commun.*, vol. 26, no. 4, pp. 140–147, Aug. 2019.

[105] G. S. Paschos, G. Iosifidis, M. Tao, D. Towsley, and G. Caire, "The role of caching in future communication systems and networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1111–1125, Jun. 2018.

[106] W. Zhao, Y. Qin, D. Gao, C. H. Foh, and H.-C. Chao, "An efficient cache strategy in information centric networking vehicle-to-vehicle scenario," *IEEE Access*, vol. 5, pp. 12 657–12 667, Jun. 2017.

[107] D. D. Van, Q. Ai, Q. Liu, and D.-T. Huynh, "Efficient caching strategy in content-centric networking for vehicular ad-hoc network applications," *IET Intell. Transp. Syst.*, vol. 12, no. 7, pp. 703–711, Aug. 2018.

[108] A. Alioua, S. Simoud, S. Bourema, M. Khelifi, and S.-M. Senouci, "A stackelberg game approach for incentive V2V caching in software-defined 5G-enabled VANET," in *Proc. IEEE Symp. on Comput. and Commun. (ISCC).* Rennes, France: IEEE, Jul. 2020.

[109] G. Mauri, M. Gerla, F. Bruno, M. Cesana, and G. Verticale, "Optimal content prefetching in NDN vehicle-to-infrastructure scenario," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2513–2525, Mar. 2017.

[110] Z. Hu, Z. Zheng, T. Wang, L. Song, and X. Li, "Roadside unit caching: Auction-based storage allocation for multiple content providers," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6321–6334, Oct. 2017.

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

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

[113] N. Zhao, F. R. Yu, L. Fan, Y. Chen, J. Tang, A. Nallanathan, and V. C. Leung, "Caching unmanned aerial vehicle-enabled small-cell networks: Employing energy-efficient methods that store and retrieve popular content," *IEEE Veh. Technol. Mag.*, vol. 14, no. 1, pp. 71–79, Jan. 2019.

[114] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1046–1061, May 2017.

[115] B. Jiang, J. Yang, H. Xu, H. Song, and G. Zheng, "Multimedia data throughput maximization in internet-of-things system based on optimization of cache-enabled UAV," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3525–3532, Apr. 2019.

[116] E. Lakiotakis, P. Sermpezis, and X. Dimitropoulos, "Joint optimization of UAV placement and caching under battery constraints in UAV-aided small-cell networks," in *Proc. the ACM SIGCOMM 2019 Workshop on Mobile AirGround Edge Computing, Syst., Netw, and Appl.*, Beijing, China, Aug. 2019.

[117] X. Xu, Y. Zeng, Y. L. Guan, and R. Zhang, "Overcoming endurance issue: UAV-enabled communications with proactive caching," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1231–1244, Jun. 2018.

[118] A. A. Khuwaja, Y. Zhu, G. Zheng, Y. Chen, and W. Liu, "Performance analysis of hybrid UAV networks for probabilistic content caching," *IEEE Syst. J.*, DOI: 10.1109/JSYST.2020.3013786, Aug. 2020.

[119] F. Zhou, N. Wang, G. Luo, L. Fan, and W. Chen, "Edge caching in multi-UAV-enabled radio access networks: 3D modeling and spectral efficiency optimization," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 6, pp. 329–341, Apr. 2020.

[120] A. Armon and H. Levy, "Cache satellite distribution systems: Modeling analysis, and efficient operation," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 2, pp. 218–228, Feb. 2004.

[121] S. Liu, X. Hu, Y. Wang, G. Cui, and W. Wang, "Distributed caching based on matching game in LEO satellite constellation networks," *IEEE Commun. Lett.*, vol. 22, no. 2, pp. 300–303, Nov. 2017.

[122] H. Wu, J. Li, H. Lu, and P. Hong, "A two-layer caching model for content delivery services in satellite-terrestrial networks," in *Proc. IEEE global commun. conference (GLOBECOM).* Washington, DC, USA: IEEE, Dec. 2016.

[123] S. D'Oro, L. Galluccio, G. Morabito, and S. Palazzo, "SatCache: a profile-aware caching strategy for information-centric satellite networks," *Trans. on Emerging Telecommun. Technol.*, vol. 25, no. 4, pp. 436–444, Apr. 2014.

[124] E. Wang, H. Li, and S. Zhang, "Load balancing based on cache resource allocation in satellite networks," *IEEE Access*, vol. 7, pp. 56 864–56 879, Apr. 2019.

[125] E. Wang, X. Lin, and S. Zhang, "Content placement based on utility function for satellite networks," *IEEE Access*, vol. 7, pp. 163 150–163 159, Nov. 2019.

[126] F. A. Silva, A. Boukerche, T. R. M. B. Silva, L. B. Ruiz, E. Cerqueira, and A. A. F. Loureiro, "Vehicular networks: A new challenge for content-delivery-based applications," *ACM Computing Surveys (CSUR)*, vol. 49, no. 1, Sept. 2016.

[127] B. Hu, L. Fang, X. Cheng, and L. Yang, "Vehicle-to-vehicle distributed storage in vehicular networks," in *Proc. IEEE ICC 2018*, Kansas City, MO, USA, May 2018.

[128] N. Liu, M. Liu, G. Chen, and J. Gao, "The sharing at roadside: Vehicular content distribution using parked vehicles," in *Proc. INFOCOM 2012*, Orlando, FL, USA, Mar 2012.

[129] z. Su, Y. Hui, and S. Guo, "D2D-based content delivery with parked vehicles in vehicular social networks," *IEEE Wireless Commun. Lett.*, vol. 23, no. 4, pp. 90–95, Aug. 2016.

[130] H. Tian, Y. Otsuka, M. Mohri, Y. Shiraishi, and M. Morii, "Leveraging in-network caching in vehicular network for content distribution," *International Journal of Distributed Sensor Networks*, vol. 12, no. 6, Jun. 2016.

[131] T. Wang, L. Song, Z. Han, and B. Jiao, "Dynamic popular content distribution in vehicular networks using coalition formation games," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 538–547, Jul. 2013.

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

[133] S. Ortiz, C. T. Calafate, J.-C. Cano, P. Manzoni, and C. K. Toh, "A UAV-based content delivery architecture for rural areas and future smart cities," *IEEE Internet Comput.*, vol. 23, no. 1, pp. 29–36, Dec. 2018.

[134] H. Zhang, S. Wei, W. Yu, G. Chen, D. Shen, and K. Pham, "Scheduling methods for unmanned aerial vehicle based delivery systems," in *2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC)*, Colorado Springs, CO, USA, Oct. 2014.

[135] A. Al-Hilo, M. Samir, C. Assi, S. Sharafeddine, and D. Ebrahimi, "Cooperative content delivery in UAV-RSU assisted vehicular networks," in *Proc. ACM MobiCom Workshop on Drone Assisted Wireless Communications for 5G and Beyond*, London United Kingdom, 2020.

[136] Y. Kawamoto, Z. M. Fadlullah, H. Nishiyama, N. Kato, and M. Toyoshima, "Prospects and challenges of context-aware multimedia content delivery in cooperative satellite and terrestrial networks," *IEEE Commun. Mag.*, vol. 52, no. 6, pp. 55–61, Jun. 2014.

[137] G. Araniti, I. Bisio, M. De Sanctis, A. Orsino, and J. Cosmas, "Multimedia content delivery for emerging 5G-satellite networks," *IEEE Trans. Broadcast*, vol. 62, no. 1, pp. 10–23, Jan. 2016.

[138] G. Gür and S. Kafiloğlu, "Layered content delivery over satellite integrated cognitive radio," *IEEE Wireless Commun. Lett.*, vol. 6, no. 3, pp. 390–393, Apr. 2017.

[139] X. Wang, H. Liy, W. Yao, T. Lany, and Q. Wu, "Content delivery for high-speed railway via integrated terrestrial-satellite networks," in *Proc. IEEE WCNC*. Seoul, Korea (South): IEEE, Jun. 2020.

[140] L. Wang, H. Yang, X. Qi, J. Xu, and K. Wu, "iCast: Fine-grained wireless video streaming over internet of intelligent vehicles," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 111–123, Feb. 2019.

[141] Y. Lin, B. Liang, and B. Li, "Data persistence in large-scale sensor networks with decentralized fountain codes," in *Proc. IEEE INFOCOM 2007*, Barcelona, Spain, May 2007.

[142] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *Amer. Math. Monthly*, vol. 69, no. 1, pp. 9–15, Jan. 1962.

[143] O. Kaiwartya, A. H. Abdullah, Y. Cao, A. Altameem, M. Prasad, C.-T. Lin, and X. Liu, "Internet of vehicles: Motivation, layered architecture, network model, challenges, and future aspects," *IEEE Access*, vol. 4, pp. 5356–5373, Sept. 2016.

[144] M. Luby, "LT codes," in *Proc. IEEE FOCS 2002*, Vancouver, BC, Canada, Nov. 2002, pp. 271–280.

[145] D. Fiems, B. Steyaert, and H. Bruneel, "Discrete-time queues with generally distributed service times and renewal-type server interruptions," *Performance Evaluation*, vol. 55, no. 3-4, pp. 277–298, Feb. 2004.

[146] G. Zhang, T. Q. S. Quek, M. Kountouris, A. Huang, and H. Shan, "Fundamentals of heterogeneous backhaul design—analysis and optimization," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 876–889, Feb. 2016.

[147] G. Y. Handler and I. Zang, "A dual algorithm for the constrained shortest path problem," *Networks*, vol. 10, no. 4, pp. 293–309, 1980.

[148] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[149] Didi Chuxing GAIA Initiative. (Accessed: Feb. 2021). [Online]. Available: https://gaia.didichuxing.com

[150] F. Ono, H. Ochiai, and R. Miura, "A wireless relay network based on unmanned aircraft system with rate optimization," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7699–7708, Sept. 2016.

[151] E. Baştuğ, M. Bennis, E. Zeydan, M. A. Kader, I. A. Karatepe, A. S. Er, and M. Debbah, "Big data meets telcos: A proactive caching perspective," *J. Commun. Netw.*, vol. 17, no. 6, pp. 549–557, Dec. 2015.

[152] B. Chen and C. Yang, "Caching policy for cache-enabled D2D communications by learning user preference," *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6586–6601, Dec. 2018.

[153] W. Shi, J. Li, N. Cheng, F. Lyu, S. Zhang, H. Zhou, and X. Shen, "Multi-drone 3-D trajectory planning and scheduling in drone-assisted radio access networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 8145–8158, Aug. 2019.

[154] H. Wu, J. Chen, F. Lyu, L. Wang, and X. Shen, "Joint caching and trajectory design for cache-enabled UAV in vehicular networks," in *Proc. IEEE WCSP 2019*, Xi'an, China, Oct. 2019, pp. 1–6.

[155] Y. Zeng, J. Xu, and R. Zhang, "Energy minimization for wireless communication with rotary-wing UAV," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2329–2345, Mar. 2019.

[156] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proc. IEEE*, vol. 107, no. 8, pp. 1738–1762, Aug. 2019.

[157] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Advances in neural information processing systems*, Lake Tahoe, USA, Dec. 2012, pp. 1097–1105.

[158] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proc. Twenty-ninth AAAI conference on artificial intelligence*, Austin Texas, USA, Jan. 2015, pp. 2267–2273.

[159] E. B. Rodrigues, F. R. M. Lima, T. F. Maciel, and F. R. P. Cavalcanti, "Maximization of user satisfaction in OFDMA systems using utility-based resource allocation," *Wirel. Commun. Mob. Comput.*, vol. 16, no. 4, pp. 376–392, Mar. 2016.

[160] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[161] F. Lyu, P. Yang, W. Shi, H. Wu, W. Wu, N. Cheng, and X. Shen, "Online UAV scheduling towards throughput QoS guarantee for dynamic IoVs," in *Proc. IEEE ICC 2019*, Shanghai, China, May 2019, pp. 1–6.

[162] L. Liu, Y. Song, H. Zhang, H. Ma, and A. V. Vasilakos, "Physarum optimization: A biology-inspired algorithm for the steiner tree problem in networks," *IEEE Trans. Comput.*, vol. 64, no. 3, pp. 818–831, Mar. 2015.

[163] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, vol. 6, no. 1, pp. 58–73, Feb. 2002.

[164] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of ICML*, vol. 37, Jul. 2015, pp. 448–456.

[165] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. G. Andrews, "User association for load balancing in heterogeneous cellular networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2706–2716, Apr. 2013.

[166] Q. Han, B. Yang, G. Miao, C. Chen, X. Wang, and X. Guan, "Backhaul-aware user association and resource allocation for energy-constrained hetnets," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 580–593, Mar. 2016.

[167] A. Khalili, S. Akhlaghi, H. Tabassum, and D. W. K. Ng, "Joint user association and resource allocation in the uplink of heterogeneous networks," *IEEE Wireless Commun. Lett.*, vol. 9, no. 6, pp. 804–808, Jun. 2020.

[168] Z. Li, C. Wang, and C.-J. Jiang, "User association for load balancing in vehicular networks: An online reinforcement learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2217–2228, Aug. 2017.

[169] C. Chaieb, Z. Mlika, F. Abdelkefi, and W. Ajib, "On the optimization of user association and resource allocation in hetnets with mm-wave base stations," *IEEE Syst. J.*, vol. 14, no. 3, pp. 3957–3967, Sept. 2020.

[170] A. Wolf, P. Schulz, M. Dörpinghaus, J. C. S. Santos Filho, and G. Fettweis, "How reliable and capable is multi-connectivity?" *IEEE Trans. Commun.*, vol. 67, no. 2, pp. 1506–1520, Feb. 2019.

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

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

[173] J. Du, C. Jiang, J. Wang, Y. Ren, S. Yu, and Z. Han, "Resource allocation in space multiaccess systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 2, pp. 598–618, Apr. 2017.

[174] Assembly, ITU Radiocommunication, "Satellite system characteristics to be considered in frequency sharing analyses within the fixed-satellite service," *ITU-R S.1328*, Sept. 2002.