Design of Resource Management and Energy-Efficient Task Scheduling Algorithms for Fog-Empowered Vehicular Networks

Submitted in partial fulfilment of the requirements of the degree of

Doctor of Philosophy

by

THANEDAR MD. ASIF

(Roll No. 720089)

Under the supervision of

Prof. Sanjaya Kumar Panda

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL
WARANGAL - 506004, TELANGANA, INDIA
March, 2024

Dedicated to

My Grandmother, My Parents, My Wife and My Children

Thank You for Your Love and Support

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL
WARANGAL - 506004, TELANGANA, INDIA

THESIS APPROVAL SHEET FOR Ph. D.

This dissertation work entitled, **Design of Resource Management and Energy-Efficient Task Scheduling Algorithms for Fog-Empowered Vehicular Networks**, by **Mr. Thanedar Md. Asif (Roll No. 720089)** is approved for the degree of **Doctor of Philosophy** at the National Institute of Technology Warangal.

Examiners

Supervisor
Prof. Sanjaya Kumar Panda
Dept. of Computer Science and Engg.
NIT Warangal, India

Chairman
Prof. R. Padmavathy
Dept. of Computer Science and Engg.
NIT Warangal, India

Date:

Place:

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL
WARANGAL - 506004, TELANGANA, INDIA

CERTIFICATE

This is to certify that the thesis entitled, **Design of Resource Management and Energy-Efficient Task Scheduling Algorithms for Fog-Empowered Vehicular Networks**, submitted in partial fulfilment of requirements for the award of the degree of **Doctor of Philosophy** to the National Institute of Technology Warangal, is a bonafide research work done by **Mr. Thanedar Md. Asif (Roll No. 720089)** under my supervision. The contents of the thesis have not been submitted elsewhere for the award of any degree.

Prof. Sanjaya Kumar Panda
Assistant Professor and Supervisor
Department of CSE
NIT Warangal, India

Place: NIT Warangal
Date:

DECLARATION

This is to certify that the work presented in the thesis entitled “**Design of Resource Management and Energy-Efficient Task Scheduling Algorithms for Fog-Empowered Vehicular Networks**” is a bonafide work done by me under the supervision of Dr. Sanjaya Kumar Panda and was not submitted elsewhere for the award of any degree.

I declare that this written submission represents my ideas in my own words, and where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all academic honesty and integrity principles and have not misrepresented, fabricated, or falsified any idea/date/fact/source in my submission. I understand that any violation of the above will cause disciplinary action by the institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Mr. Thanedar Md. Asif
(Roll No. 720089)

Date:

Acknowledgements

First and foremost, I extend my heartfelt thanks to my supervisor, Prof. Sanjaya Kumar Panda, whose invaluable insights, encouragement, and unwavering support played a pivotal role in shaping this research. His guidance has been instrumental in every step of the process, and I am genuinely grateful for the knowledge and wisdom he shared. I would like to express my gratitude to the Doctoral Scrutiny Committee (DSC) members, Prof. R. Padmavathy, Prof. S. Ravichandra, Prof. Rashmi Ranjan Rout and Prof. L. Anjaneyulu, for their constructive comments and scholarly input. I would like to thank my DSC external members, Prof. D. Srinivasacharya and Prof. Ch. Venkaiah, for their invaluable suggestions during the comprehensive examination.

I am indebted to Prof. Bidyadhar Subudhi, Director, Prof. A. Sarath Babu, Dean (Academic), Prof. V. T. Somasekhar, Dean (Research and Consultancy), Prof. D. Srinivasacharya, Dean (Student Welfare) and Prof. N. V. Umamahesh, Registrar In-Charge, for their support, providing me with all necessary facilities during my candidature. I would also like to acknowledge the assistance of the Institute fellowship, whose financial support made this research possible. Their investment in my academic pursuits is deeply appreciated. I am immensely thankful to Prof. P. Radha Krishna, Prof. S. Ravichandra, and Prof. R. Padmavathy, Heads of the Department of Computer Science and Engineering (CSE) at NIT Warangal, India, for providing adequate facilities during my stay in the department. I would like to thank all the faculty members and staff of the Department of CSE for their help and support.

I would like to thank the anonymous examiners (i.e., indian and foreign) for their valuable comments, future research directions and timely submission of the report.

Special thanks go to my family for their unwavering support and understanding. I thank my brothers Mr. Arif, Mr. Iliyas, Mr. Ghouse and Mr. Nasir, who have always motivated and encouraged me to achieve good things. I thank my parents, who have given me all kinds of support throughout this Ph. D. period. I thank my wife for backing me in every endeavour. Their love and encouragement sustained me during the challenging phases of

this academic endeavour. I am grateful for the sacrifices you made to ensure my success.

I am indebted to all my seniors, friends, and juniors, Dr. V. Satish Reddy, Dr. Manoj Somesula, Dr. Ch. Rajesh, Dr. Uma Maheshwar Sharma, Mr. Sanjib Kumar Raul, Mr. Ram Pavan Medipalli, Mr. B. Rajashekar, Mr. V. Dilip Kumar, Mr. G. Punnam Chandar, Mrs. G. Mounika Rani, Mrs. A. Aswini Devi, Mr. A. Srinivas, Mrs. D. Sudha, Mr. Hemraj Singh, Mr. A Venkatesh, Mr. Jayanth Babu and Mr. Saurabh Gupta who provided valuable insights and encouragement, making the academic journey more enjoyable.

Place: NIT Warangal

Date: **(Thanedar Md. Asif)**

Abstract

The delay-sensitive applications, such as self-driving, intelligent transportation, navigation, and augmented reality assistance, can be evolved in vehicular ad-hoc networks (VANETs) using one of the leading paradigms, fog computing (FC). The demand for these vehicular services has increased due to the emergence of fifth-generation (5G) technology. By blending FC and 5G technologies, the service quality can be improved in intelligent transportation system (ITS) of smart cities. Intelligent vehicles are connected to the roadside infrastructure, such as high power nodes (HPNs) and roadside units (RSUs), also called fog nodes (FNs), to obtain on-demand services. These FNs possess finite resources and can provide services to limited vehicles. However, when vehicles reach the network spike in demand, the FNs become impuissant in furnishing services in the existing solutions. As a result, there is a significant reduction in the network service capability and throughput and an increase in the FNs' energy consumption. Therefore, we propose resource management algorithms such as dynamic resource management (DRM), efficient resource orchestration (ERO) and energy-efficient resource allocation (EERA) to harmonize the resource blocks (RBs) allocation among FNs. Then, to coordinate the allocation of RBs among FNs, the allocated RBs of vehicles in the overlap coverage regions are reduced. This reduction is done by migrating RBs between pairs of FNs to offload upstream services. The problem of reducing RBs among FNs is formulated as integer linear programming (ILP), and its NP-hardness is determined by reducing it from the seminar assignment problem.

The proposed algorithm, DRM, considers the set of vehicles in overlapped coverage regions of FNs and communicates with those corresponding FNs. Then, it migrates the RBs of the set of vehicles between pairs of FNs to minimize the allocated RBs. As a result, the network's service capability is enhanced. The proposed algorithm, ERO, maximizes the network's throughput by partitioning the FN coverage region into restricted and non-restricted coverage regions. Then, it coordinates the allocation of RBs among FNs by reducing RBs for vehicles in the non-restricted coverage regions. A minimum priority queue is constructed using the occupied capacity of FNs to perform optimal migration between pairs of FNs. However, as the vehicles that reach the network grow, FNs' energy consumption increases. Consequently, FNs become futile in delivering services. Therefore, to handle this issue, we present an EERA algorithm to harmonize RB allocation among FNs to reduce the energy utilization of FNs. The proposed algorithm, EERA, relocates the assigned RBs of vehicles in overlap coverage regions amid pairs of FNs, such that the allocated RBs of FNs and energy consumption of FNs are minimized.

In ITS, FNs (i.e., HPNs and RSUs) are operated with batteries. FNs are deployed such that the coverage region of each FN intersects with the neighbouring FN(s) to provide services in remote areas where consistent power sources are unavailable. Vehicles in such regions offload delay-sensitive tasks into FNs to get services. However, when the number of vehicles arriving into the network grows over peak hours, the energy dissipation of FNs for processing tasks increases. Consequently, energy-limited FNs become ineffective in delivering services without efficient task scheduling. Therefore, we present reinforcement learning (RL) based energy-efficient and delay-aware (EEDA) task scheduling among FNs in the intersecting regions to reduce the energy dissipation of FNs. The RL agent is trained for different vehicle arrival rates to schedule tasks in a suitable FN of the intersecting areas.

The proposed algorithms, DRM, ERO and EERA, are simulated extensively in terms of service capability, serviceability, availability, throughput, energy consumption of FNs and resource utilization. In addition, the simulation results are analogized with benchmark algorithms, such as dynamic resource orchestration (DRO), signal aware (SA), DRO+SA, adaptive resource balance (ARB), minimum cost flow (MCF) and random order (RO), as per their applicability. Similarly, the EEDA algorithm is evaluated by considering FN energy usage, FN response time, and vehicles' sojourn time in intersecting regions to meet task delay constraints. The simulation outcomes are compared with the priority-aware semi-greedy (PSG), earliest deadline first (EDF), and first come, first serve (FCFS).

Keywords: Availability, Deadline, Delay-Sensitive Tasks, Energy Consumption, Fog Computing, Fog-Empowered Vehicular Ad-hoc Networks, Intelligent Transportation Systems, Resource Blocks Allocation, Resource Management, Resource Utilization, Serviceability, Service Capability, Service Migration, Task Scheduling, Throughput, Vehicular Ad-hoc Networks and Vehicular Networks.

Contents

List of Tables

List of Figures

List of Abbreviations and Notations

List of Abbreviations

Acronym	Abbreviation
3GPP	Third Generation Partnership Project
5G	Fifth Generation
ACC	Adaptive Cruise Control
ARB	Adaptive Resource Balancing
CACC	Cooperative Adaptive Cruise Control
CASD	Context-Awareness Safety Driving
CCH	Control Channel
CIs	Confidence Internvals
CNN	Convolution Neural Network
CPU	Central Processing Unit
C-V2X	Cellular Vehicle-to-everything
DATO	Delay-Aware Task Offloading
DDTP	Deadline-Aware Dynamic Task Placement
DRL	Deep Reinforcement Learning
DRM	Dynamic Resource Management
DRO	Dynamic Resource Orchestration
DSRC	Dedicated Short-Range Communication
EDF	Earliest Deadline First
EEDA	Energy-Efficient and Delay-Aware

EERA	Energy-Efficient Resource Allocation
EPAS	Energy and Priority-Aware Scheduling
ERO	Efficient Resource Orchestration
FC	Fog Computing
FCFS	First Come First Serve
FNs	Fog Nodes
FRL	Fuzzy-Based Reinforcement Learning
FVNETs	Fog-Empowered Vehicular Ad-Hoc NETworks
GAP	General Assignment Problem
GLOSA	Green Light Optimal Speed Advisory
GODA	Greedy Offloading Decision Algorithm
HPNs	High Power Nodes
I2I	Infrastructure-to-Infrastructure
I2V	Infrastructure-to-Vehicle
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
ILP	Integer Linear Programming
IoT	Internet of Things
IoV	Internet of Vehicles
ITS	Intelligent Transportation System
LTE-V	Long Term Evolution-V
MAC	Medium Access Control
MCF	Minimum Cost Flow
MIMO	Multiple-Input-Multiple-Output
NFS	Nearest Fastest Set
NNF	Nearest Neighbour Forwarder
OBUs	On-Board Units
OD-SARSA	Offloading Decision-Based State-Action-Reward-State-Action
OFDMA	Orthogonal Frequency Division Multiple Access

PEARL	Protocol for Energy-Efficient Adaptive Scheduling using Reinforcement Learning
PSG	Priority-Aware Semi-Greedy
QoS	Quality of Service
RA	Resource Allocation
RAM	Random Access Memory
RBs	Resource Blocks
RL	Reinforcement Learning
RO	Random Order
RSUs	Road-Side Units
SA	Signal Aware
SAINT	Self Adaptive Interactive Navigation Toll
SAP	Seminar Assignment Problem
SCH	Service CHannel
SINR	Signal-to-Interference-plus-Noise Ratio
UCB	Upper Confidence Bound
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VANETs	Vehicular Ad-Hoc Networks
VFC	Vehicular Fog Computing
WAVE	Wireless Access for the Vehicular Environment

List of Notations

Notation	**Description**
T	Total serviceable time of the network
t_q	Time slot t_q, $1 \leq q \leq T$
δ_s	Time slot duration

$\mathcal{G}$	Number of FNs
R	Communication range of an FN
$\mathbb{N}$	A set of natural numbers
$\mathcal{P}_t(i)$	A set of vehicles served by i^{th} FN at time slot t
λ, μ	Mean arrival and departure rate of vehicles, respectively
b_k^i	Total RBs allocated to k^{th} vehicle from i^{th} FN
B	Bandwidth of an FN
A	A city area in which FVNET is fixed
r^k	Requested data rate of k^{th} vehicle
$\psi(ik)$	SINR of k^{th} vehicle from i^{th} FN
$\mathcal{P}_t^{in}(i)$	A set of vehicles reaching to the i^{th} FN at time slot t
$\mathcal{P}_t^{out}(i)$	A set of vehicles leaving from the i^{th} FN at time slot t
$\mathbb{E}$	Expected value
L_i	i^{th} FN capacity (in RBs)
$O_t(i)$	Current allocated RBs for the i^{th} FN at time slot t
$A_t(i)$	Available/Remaining RBs of i^{th} FN at time slot t
$Sc_t(i)$	Service capability of i^{th} FN at time slot t
Sc_t	Service capability of the network at time slot t
$\mathcal{P}_t(ij)$	A set of vehicles in overlapped coverage area between pairs of FNs i and j at time slot t
$\mathcal{Z}$	A set of overlapped coverage areas in FVNETs
$p_t^*(ij)$	An optimal set of vehicles for migrating RBs between pairs of FNs i and j at time slot t
$\mathcal{P}_t^*(i)$	An optimal set of vehicles served by i^{th} FN after RBs migration at time slot t
$G(V, E)$	A graph of FVNET with a set of vertices V and a set of edges E
$\mathcal{P}_t^r(i)$	A set of vehicles in restricted region of i^{th} FN at time slot t
$\mathcal{P}_t^{nr}(ij)$	A set of vehicles in non-restricted region of i^{th} FN and j^{th} FN at time slot t

$\mathcal{T}_t(ik)$	Achievable rate of k^{th} vehicle from i^{th} FN at time slot t
D_{ki}	Euclidean distance between i^{th} FN and k^{th} vehicle
$\rho_t(ik)$	Transmission power of i^{th} FN when connecting with k^{th} vehicle at time slot t
N_k, I_k	Gaussian noise and co-channel inference of k^{th} vehicle, respectively
$\alpha_t(ik)$	An allocated time fraction of k^{th} vehicle from i^{th} FN at time slot t
$\tau_t(ik)$	k^{th} vehicle throughput from i^{th} FN at time slot t
β	Pathloss exponent
γ	Scaling co-efficient
$V_t(ij)$	Set of vehicles in the intersecting region $(i, j) \in \mathcal{Z}$ at time slot t
c_k	CPU cycles (in MIPS) required by task k
F_i	CPU cycles of FN i
h_k	Data size (in bytes) of task k
$T_t(ij)$	Set of tasks in the intersection part $(i, j) \in \mathcal{Z}$ at time slot t
d_k	Deadline of task k
$\mathcal{S}_{ij}^k$	Sojourn time of the vehicle ϑ_k in the intersecting region $(i, j) \in \mathcal{Z}$
I	The location from where the vehicle enters into the intersecting region
L	Length of the intersecting coverage parts of FVNETs
v_k	Velocity of vehicle $\vartheta_k \in V_t(ij)$
$\delta_{t_2}^c(\vartheta_k)$	Covered distance of vehicle ϑ_k in an intersecting region at time slot t_2
χ_{ki}	Transmission rate of FN i for the task k
T_{ki}^{tr}	Transmission time of task k from FN i
T_{ki}^{e}	Execution time of task k from FN i
T_{ki}^{r}	Response time of FN i for the task k
E_{ki}^{e}	FN i energy consumption in executing task k
E_{ki}^{tr}	FN i energy consumption in transmitting data h_k
$\mathcal{E}$	The energy coefficient depending on the chip architecture
$E_{ki}(t)$	Energy consumed by FN i in serving vehicle ϑ_k at time slot t

$E_{ij}(t)$ Energy consumed by FNs i and j in serving all the tasks in an intersecting region $(i,j) \in \mathcal{Z}$ at time slot t

$E(t)$ Energy consumed in FVNETs at time slot t

Chapter 1

Introduction

The emergence of fifth-generation (5G) technology promotes ubiquitous services through the Internet [1–4]. At the same time, the instantaneous transition of the Internet of Things (IoT) and vehicular technology are attaining tremendous favour in vehicular ad-hoc networks (VANETs) [5–7]. Integrating these technologies initiates a smart city's intelligent transportation system (ITS) to share and transmit information among smart vehicles and the cloud [8, 9]. There is increasing demand for vehicular services and applications, such as virtual reality-based driving assistance, remote intelligent control and vehicular video streaming, which demands high computational capabilities [1, 10–12]. Thus, due to finite resource constraints (i.e., battery, storage, computation, etc.), the vehicle offloads these computation-intensive and delay-sensitive tasks to the cloud server for cloud computing [13]. Nevertheless, the services from the cloud suffer from network congestion and transmission delay [11, 12, 14, 15]. Therefore, a contemporary computing paradigm, fog computing (FC), is incorporated in vehicular networks, emerging as fog-empowered vehicular ad-hoc networks (FVNETs) [16–20]. FC is a distributed architecture consisting of fog nodes (FNs) near the end user and provides services by bringing cloud services to the networks' edge [21–23]. FNs are devices, such as gateways, routers, and other electronic devices, capable of computing, storing and communicating with other FNs and the cloud. As a result, computation-intensive and delay-sensitive tasks are offloaded to FNs for meeting deadlines, low latency and elastic computation [1, 10].

The architecture of FVNETs is shown in Figure 1.1. In ITS, each smart vehicle is fitted with onboard units (OBUs) for storage and processing resources. Roadside infrastructures, such as high power nodes (HPNs) and roadside units (RSUs), are employed along the road to furnish vehicular services. These HPNs and RSUs are considered as FNs [24]. The central supervisor is connected to the cloud over the Internet and manages the HPNs. HPNs have more resource capabilities than RSUs and can coordinate with RSUs nearby. In addition, intelligent vehicles with OBUs seek connection with FNs using various device cooperative schemes, such as signal-aware (SA), content-aware, and capacity-aware. Using SA, the contact between the vehicles and the FNs depends on the signal-to-interference-plus-noise ratio (SINR) quality [25]. The connection using a content-aware scheme between the vehicles and the FNs depends on the user's favourite contents [26]. On the other hand, the connection in the capacity-aware scheme is based on the availability of FNs resources [27]. Further, the 5G technology's key feature, cellular vehicle-to-everything (C-V2X), is used for connection between vehicles and FNs, which supports vehicle-to-infrastructure (V2I) and infrastructure-to-vehicle (I2V) communications [28].

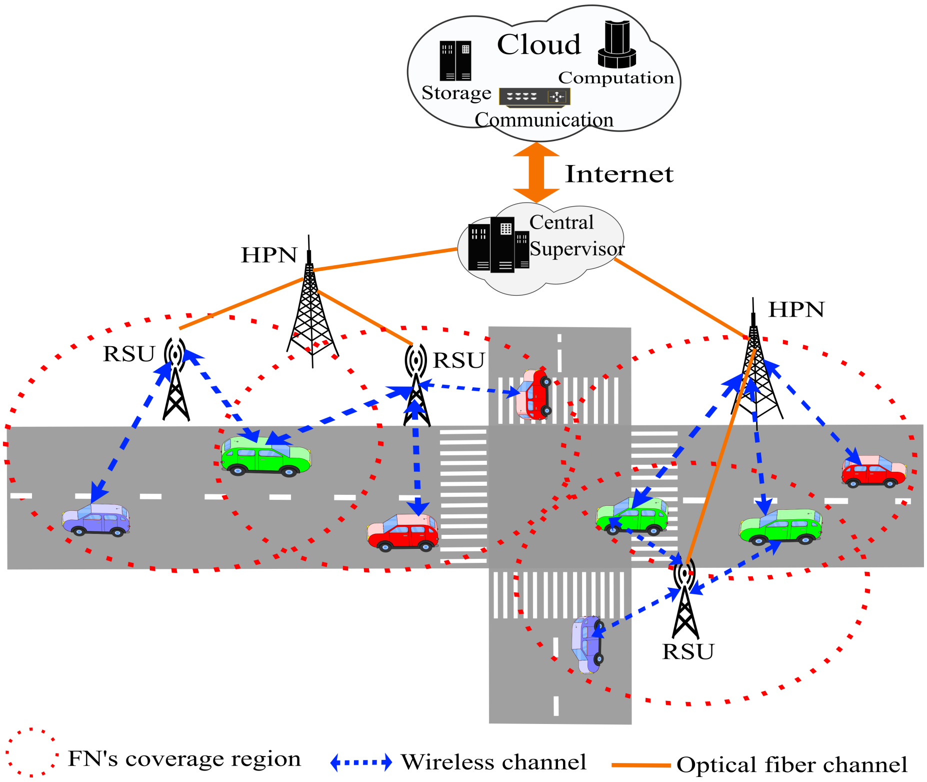


Figure 1.1: Architecture of FVNETs.

The V2I, I2V and vehicle-to-vehicle (V2V) communications are realized using institute of electrical and electronics engineers (IEEE) 802.11p protocol-based dedicated short-range communication (DSRC) technology [29–31]. Therefore, intelligent vehicles can communicate with other vehicles and FNs through OBUs. For this, the spectrum allotted to DSRC in the 5.9 GHz frequency band is split into a single control channel (CCH) and several service channels (SCH) [31, 32]. However, DSRC meets many challenges in terms of high bandwidth, low latency and the network coverage need of vehicle-to-everything (V2X) applications due to the format of its physical and medium access control (MAC) layer [33]. These challenges of DSRC and evolution of long-term evolution (LTE)-V in third generation partnership project (3GPP) release 14 [34] inspired the researchers to examine C-V2X communications. However, the network must allocate orthogonal frequency division multiple access (OFDMA) resource blocks (RBs) with power assignments before providing services to the users. OFDMA is the modulation method for the LTE in downlink. An RB is the smallest time-frequency unit in an OFDMA system, and a smart vehicle must be assigned with RBs before data transmission [35]. In this thesis, serviceable time is a fixed length time interval T in which the network provides services to the vehicles arriving. The serviceable time of the network is partitioned to allow FNs to serve vehicles at different time slots. Figure 1.2 shows the partitioning of the total available time divided into a number of equal time slots. In each time slot t_q, $1 \leq q \leq T$, a set of FNs is active to serve vehicles simultaneously [25].

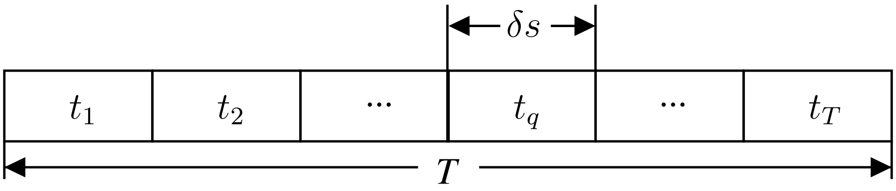


Figure 1.2: FN communication channel time.

As the number of vehicles rises during peak hours, the FNs become inefficient in serving the vehicles due to finite resource restraints. Consequently, there is a reduction in the network service capability and throughput. Therefore, managing FN resources is a key measure to improve the quality of services (QoS) in FVNETs [36]. Moreover, efficient resource engagement to maximize service capability and throughput is the primary challenge for FVNETs. Note that the service capability indicates the availability of the network to serve the vehicles with their desired data rates [36]. Due to limited resources, the FNs become impuissant in serving vehicles. Consequently, vehicles arriving at the network connect the FNs with insufficient resources, reducing the network's throughput. To manage this issue, we present resource management algorithms for coordinating RB allocation and offloading upstream services among FNs to maximize the network service capability and throughput.

The increase in arrived vehicles in the intersecting region rapidly generates compute-intensive and delay-sensitive tasks in the network, causing increases in the processing of tasks at FNs [37]. It may lead to high energy consumption by FNs and influence the end-user experience without proper management. In contrast, efficient energy utilization in energy-limited FVNETs enhances the network's throughput and lifetime. Therefore, we present energy-efficient task scheduling in intersecting regions of energy-limited FVNETs to mitigate the energy consumption of FNs and improve the network's throughput by satisfying the delay constraints of the tasks.

The rest of this chapter is organized as follows. The motivation behind this work is presented in Section 1.1. The overview of the contributions of the thesis is discussed in Section 1.2. Section 1.3 presents the organization of this thesis.

1.1 Motivation and Objectives

The existing literature needs more attention to address challenges in FVNETs when the coverage region of a FN intersects with neighbouring FNs. This gap in research inspired us to concentrate on implementing effective resource management and delay-sensitive task scheduling in FVNETs where the FN coverage region intersects with neighbouring FNs' coverage. Without loss of generality, this thesis considers the deployment of FVNETs covering the city area such that the FNs' coverage region intersects with neighbouring FNs' coverage to provide services. In these circumstances, managing efficient resource utilization to improve the QoS is the primary challenge due to limited resource constraints in FVNETs. As a result, this management of resources is a crucial measure to enhance the service capability and throughput in FVNETs [36]. Besides, this thesis considers that the FNs are powered by rechargeable batteries, which require periodic recharging due to the lack of a permanent power source. Human intervention or solar energy is required to revive FNs when their batteries are depleted to ensure uninterrupted services. This reviving FNs to deliver services and ensure energy preservation till the successive recharge cycle is challenging when the FNs cover highway segments of remote areas (i.e., forests or hill terrain) where consistent power sources are unavailable [38]. In this circumstance, consider that the FVNETs cover remote areas' highway segments such that the FNs' coverage region intersects with neighbouring FNs' coverage. Subsequently, the dealy-sensitive tasks generated by the vehicles in the intersecting regions are offloaded into FNs (i.e., RSUs or HPNs) for computation. However, HPN acts as a central node that decides the scheduling of tasks among FNs of intersecting regions, including itself. It is worth mentioning that the RSUs are exclusively operated to execute the allotted tasks.

In this thesis, the following objectives are formulated concerning the above-mentioned motivation.

Objective 1: Managing allocated RBs to maximize the service capability in FVNETs by considering vehicles in overlapped coverage regions.

Objective 2: Orchestrating allocated RBs to maximize the throughput in FVNETs by considering vehicles in overlap and non-overlap coverage regions.

Objective 3: Managing allocated RBs to minimize the energy consumption of FNs in FVNETs by considering vehicles in overlap and non-overlap coverage regions.

Objective 4: Handling the delay-sensitive tasks among FNs in overlap coverage areas of FVNETs to reduce FNs' energy consumption while meeting task deadlines.

1.2 Overview of the Contributions of the Thesis

In this section, an overview of the chapter-wise contributions of the thesis is presented. Each subsection presents a summary of the contributions of the corresponding chapter.

1.2.1 A Dynamic Resource Management Algorithm for Maximizing Service Capability in FVNETs

Dynamic resource management (DRM), a polynomial time algorithm, is presented for efficient resource allocation among FNs to improve their service capability and resource utilization efficiency. In this work, we consider the set of vehicles in overlapped coverage regions of two or more FNs and served by those FNs. The allocated RBs of vehicles in the overlapped coverage regions are migrated between pairs of FNs. This migration of RBs reduces the allocated RBs of vehicles in overlapped coverage regions to maximize the service capability. We simulate the proposed algorithm by taking 10 to 50 FNs and 300 to 2100 vehicles at an arrival rate of 10 vehicles/s. We compare the simulation results with dynamic resource orchestration (DRO) [24], SA [25], DRO + SA and random order (RO) in terms of service capability, serviceability, availability, throughput, and resource utilization efficiency. The simulation outcomes show that the proposed algorithm reduces occupied RBs among FNs by migrating RBs of the set of vehicles and achieves better service capability, serviceability, availability, throughput and resource utilization efficiency than other migration algorithms, such as DRO, SA, DRO + SA and RO. The major contributions of this work are listed as follows.

1. We consider allocating RBs to newly arrived vehicles by migrating RBs between pairs of FNs without affecting the services. The RBs of vehicles are migrated to minimize overall allocated RBs.

2. The optimal RBs migration problem in FVNETs is formulated into integer linear programming (ILP) by considering the variables that impact FNs' resource constraints and the network's service capability.

3. We propose a polynomial-time DRM algorithm for optimal migration of RBs between pairs of FNs to minimize occupied RBs among FNs and maximize the service capability and resource utilization efficiency of the network by migrating RBs of a set of vehicles.

4. We present extensive simulations to show that the DRM algorithm can achieve better

performance than the four existing algorithms in terms of service capability, serviceability, availability, throughput, and resource utilization efficiency of the network.

1.2.2 An Efficient Resource Orchestration Algorithm for Enhancing Throughput in FVNETs

This work presents an efficient resource orchestration (ERO) algorithm for coordinating RB allocation and offloading upstream services among FNs to maximize the network throughput. ERO algorithm partitions the FNs' coverage region into restricted and non-restricted coverage areas. The restricted coverage area is a coverage region that does not overlap with neighbouring FNs' coverage. Similarly, FN's coverage region overlaps with neighbouring FN's coverage regions, called non-restricted coverage areas. The maximizing throughput problem is formulated to reduce assigned RBs of vehicles in the non-restricted coverage areas. Hence, the assigned RBs of vehicles in the non-restricted coverage areas are migrated between pairs of FNs to reduce allotted RBs. Further, a minimum priority queue is constructed based on the occupied capacities of FNs to perform optimal RB migration. We simulate the proposed algorithm, ERO, considering the vehicle arrival rate as 5 to 10 vehicles/s and 150 to 3000 vehicles with 10 to 50 FNs in FVNETs. We consider the influence of the rise in the number of vehicles on the network's throughput, serviceability, availability and service capability for three scenarios: the mean arrival rate is greater than the mean departure rate, the mean arrival rate is equal to the mean departure rate and the mean arrival rate is less than the mean departure rate. The simulation outcomes are compared with RO, adaptive resource balance (ARB) [39], SA [25], and DRO [24] in terms of throughput, serviceability, availability and service capability of the network. The simulation results show that the ERO performs better than existing algorithms in terms of throughput, serviceability, availability, and service capability. The novel contributions of this work are listed in the following points.

1. The throughput is maximized by migrating allotted RBs of vehicles in non-restricted coverage regions such that the allotted RBs of these vehicles are minimized among pairs of FNs.

2. We formulated the RBs migration problem in FVNETs to an ILP by scrutinizing the variables influencing the network throughput and FNs resource constraints.

3. We propose an ERO algorithm, a polynomial time algorithm, which constructs the minimum priority queue for optimal RBs migration between pairs of FNs to augment the network's throughput.

4. The ERO algorithm synchronizes the RBs allocation for offloading upstream services such that throughput is maximized by partitioning the coverage of FNs into restricted and non-restricted coverage regions.

5. We present the simulation results showing that the ERO algorithm outperforms the existing algorithms regarding throughput, serviceability, availability, and service capability. The results are obtained by considering the influence of the rise in the arrival of vehicles to the network.

1.2.3 An Energy-Efficient Resource Allocation Algorithm for Managing On-Demand Services in FVNETs

We present a resource allocation algorithm, energy-efficient resource allocation (EERA), for offloading upstream services by coordinating RBs allocation among FNs, such that the energy usage of FNs in the downlink is diminished, and the network resource utilization efficiency is maximized. The EERA algorithm builds the B+ tree using occupied RBs of FNs to reduce the allocated RBs of vehicles in the overlap coverage areas of FNs. In addition, the allocated RBs of vehicles are minimized by relocating RBs between pairs of FNs. Further, this reduction of allocated RBs minimizes FNs' energy usage in furnishing vehicles' services. We simulate the proposed algorithm considering the vehicle arrival and departure rates as 10 and 5 vehicles/s, respectively, with a range of 300 to 2100 vehicles and 10 to 50 FNs in FVNETs. The simulation results are compared with RO, minimum cost flow (MCF) [40], DRO and SA in terms of the percentage of RBs occupied, the energy consumption of FNs and the resource utilization efficiency. The outcome of simulations shows that the suggested EERA algorithm surpasses other existing algorithms when analogized with them. The major contributions of this work are listed below.

1. We consider coordinating RBs of vehicles in the overlap coverage areas of FNs in FVNETs while meeting the vehicle's desired requirements by relocating RBs among pairs of FNs.

2. The optimal migration of RBs in FVNETs is framed as ILP by assessing the resource parameters of FNs.

3. We propose an EERA algorithm using a B+ tree for optimal RBs migration among FNs for offloading upstream services, such that the energy utilization of FNs in the downlink is reduced and resource usage efficiency is maximized.

4. We demonstrate the simulations that depict the growing number of vehicles arriving on the energy utilization of FNs and the resource usage efficiency of the network. Also, the influence of increasing the number of FNs on the energy utilization of FNs shows the applicability of the proposed algorithm over the existing algorithms.

1.2.4 An Energy-Efficient and Delay-Aware Task Scheduling in Energy-Limited FVNETs using Q-Learning

We present an RL-based energy-efficient and delay-aware (EEDA) task scheduling algorithm in intersecting regions of energy-limited FVNETs. This algorithm mitigates the energy consumption of FNs while discharging and improving the network's throughput by satisfying the delay constraints of the tasks. EEDA is a greedy-based RL algorithm that provides a sub-optimal solution to task scheduling in the intersecting regions of FVNETs. It uses the Q-learning approach to train FNs for different vehicle arrival rates of traffic scenarios. It chooses the FN, which consumes minimum energy to schedule tasks in each time slot while meeting the deadline of tasks. Further, the selection of FNs for scheduling tasks depends on the sojourn time of the vehicle in the intersecting region, the task deadline, the response time from the FN and the average energy usage of the FN. We simulated the proposed algorithm for the mean arrival rate of vehicles ranging from 2 to 6 per second in each intersecting region to train the FNs for free-flow traffic of 100 seconds duration. Subsequently, simulated results compared with benchmark algorithms, such as priority-aware semi-greedy (PSG) [41], earliest deadline first (EDF) [42], first come first serve (FCFS) [43], and RO [44] in terms of average energy usage, network throughput, total transmitted data, completed service request and total service time of FN. The simulation results show that the RL-based EEDA algorithm performs better than benchmark algorithms in minimizing the average energy consumption of FN and enhancing the network throughput. The main contributions of this work can be summarized as follows.

1. This work investigates the scheduling of time-critical tasks among FNs in the intersecting regions to reduce the average energy usage of FN while considering the task's deadline, FN's energy usage and vehicle's sojourn time in the intersecting region.

2. The scheduling of tasks among FNs is transformed into an ILP by evaluating the task's deadline, response time from the FN and the residing time of the vehicle in the intersecting region.

3. Since the ILP of EEDA is NP-hard, we design a greedy-based task scheduling in the intersecting regions of FVNETs using the Q-learning-based RL technique to mitigate the energy consumption of FN and satisfy the delay constraints of vehicles.

4. Performance evaluation has been carried out by considering the impact of the increase in the number of vehicles in the intersecting regions for various vehicle arrival rates in the network concerning the energy usage of FN and throughput of the network.

1.3 Organization of the Thesis

The main focus of this thesis is to design resource management and energy-efficient task scheduling algorithms for FVNETs. The thesis comprises seven chapters, namely an introduction, a literature review, four contributions and a conclusion. The content of these chapters is briefly described as follows.

Chapter 1: This chapter presents the FVNETs architecture, motivation and thesis objectives. It also provides an overview of the major contributions and a thesis outline.

Chapter 2: This chapter presents the preliminaries in FVNETs and the literature on resource management algorithms categorized based on their objectives. It also presents the literature on optimizing energy consumption in FVNETs.

Chapter 3: This chapter presents the preliminaries of FVNETs, common terminologies, and the various performance metrics used to compare the proposed algorithms' performance with the existing ones.

Chapter 4: This chapter presents the DRM algorithm for migrating RBs between pairs of FNs such that the service capability of FVNETs is maximized.

Chapter 5: This chapter describes the orchestration of RBs in FVNETs to maximize the throughput. It introduces the ERO algorithm using binary heap to migrate RBs of vehicles in the non-restricted coverage regions.

Chapter 6: This chapter presents the EERA algorithm to minimize the energy consumption of FNs in FVNETs. The EERA performs RBs migration among pairs of FN by constructing a B+ tree of FNs' occupied capacities.

Chapter 7: This chapter introduces the EEDA task scheduling algorithm using Q-Learing in energy-limited FVNETs. This algorithm reduces FNs' energy consumption and improves the network's throughput by satisfying delay constraints of the tasks in overlapped coverage regions of FVNETs.

Chapter 8: This chapter summarizes the thesis and discusses the future scopes for the extension of the proposed works.

Chapter 2

Literature Review

This chapter presents the literature review on challenges in resource management and task scheduling in FVNETs using reinforcement learning and fuzzy logic.

2.1 Literature Review

Kansal et al. [45] have presented a systematic review of FC and its advantages in applications such as autonomous vehicles, smart traffic control, the Internet of Vehicles (IoV), and many more. Moreover, vehicles fitted with computational capacities produce massive amounts of data that can be transferred to the FN for optimized processing [46]. FVNETs provide vehicular services and applications for driving safety, traffic efficiency and comfort in transport [47]. However, the service delivery to the requested vehicles is challenging due to interim connections in FVNETs [48]. The FVNETs deployed in a highway segment such that the FN's coverage region intersects with neighbouring FNs and has its distinctive challenges described below.

1. *Limited energy:* The roadside infrastructures or FNs (i.e., RSUs and HPNs) are deployed along the highway segments of remote areas (i.e., forests or hill terrain) where consistent power sources are unavailable. FNs require periodic recharging when powered by large batteries. Reviving the FNs when batteries are depleting using renewable energy or human intervention to ensure continuous services is challenging until the next recharge cycle [38]. However, ensuring energy preservation and

potential consumption of preserved energy until the next recharge cycle is critical to improving network life.

2. *Limited computation:* FNs have limited energy and computational capabilities, such as processing and memory. The smart vehicles are equipped with devices such as OBUs, sensors, processors, memory, global positioning systems and wireless technology. These devices enhance the computational capability of vehicles to process the tasks offloaded by FNs or other vehicles. However, the vehicles offload computation-intensive and delay-sensitive tasks to FNs based on the vehicles' available resources.

3. *Mobility and location of vehicles:* The vehicles are considered mobile nodes in FVNETs. The mobility of the vehicles in FVNETs is constrained by the road network. Since the FVNETs cover the highway segment, the vehicles maintain a steady velocity with high mobility [49]. This steady mobility of vehicles leads to accurately predicting the location of vehicles in the FN coverage area.

4. *Vehicle density:* Vehicle density in FVNETs depends on the vehicles' speed, direction and mean arrival and departure rates. The vehicles' mean arrival and mean departure rates follow a Poisson distribution. Vehicles' interarrival time follows an exponential distribution. Further, a free flow discrete time traffic model is considered for uninterrupted and homogenous vehicular traffic over a fixed length of overlapping coverage regions [38, 50].

5. *Interim connections:* The moving vehicles in overlap coverage regions of FVNETs establish interim connections with stationary FNs for vehicular services. The connection between vehicles and FNs is temporary due to the high mobility of vehicles on highways. Further, it is difficult to furnish seamless services for the entire highway due to the installation cost of stationary FNs [51].

6. *Network topology:* A dynamic hierarchical topology organizes FNs, vehicles and the cloud, as shown in Figure 1.1. The vehicles are connected to a FN using star topology.

However, there are rapid changes in the star topology due to interim connections between FNs and vehicles [52].

2.1.1 Resource Management in FVNETs

The improper management of offloaded data to the FNs leads to a significant increase in energy consumption and subsequently impacts the throughput. Therefore, many researchers are paying attention to improving the performance of FVNETs. Saad et al. [53] have presented an overview of the load-balancing self-optimization models in 5G mobile networks. The load balancing model moves the surplus traffic of heavily loaded cells to their neighbour cells to make a uniform load among the cells. They have listed the technical challenges and suggested solutions for mobile networks. Patil et al. [54] have compared intelligent computing technologies such as cloud and edge to accommodate the demands of delay-sentient applications in VANETs based on different performance metrics.

The 5G feature, C-V2X, provides two types of communications for different applications in FVNETs, namely infrastructure-based communication (i.e., V2I/I2V) and V2V communication [55], as shown in Figure 2.1. In FVNETs, roadside infrastructures, such as RSUs, are considered as FNs [24]. V2V communication plays a vital role in furnishing safety applications. V2I/I2V communication plays a role in collecting real-time data at RSUs for traffic management, smart navigation, logistics and remote intelligent control. These V2I/I2V communications furnish essential information for condition and location-based utilities, such as safe distance warnings, speed restriction information, accident warnings and traffic jam warnings [56].

Facilitating a highly reliable 5G network for realizing V2X communication services with low latency is challenging because V2X communication services are foremost for driver safety applications. Husain et al. [57] have outlined the current efforts within 3GPP aimed at enhancing V2X communications and the role of the 5G network in ensuring expected capacity, lower latency, throughput and high reliability in V2X services. High mobility of vehicles in VANETs leads to uncertainty in channels. Hence, Asim et al. [55] have integrated non-orthogonal multiple access in V2X communications to enhance traffic

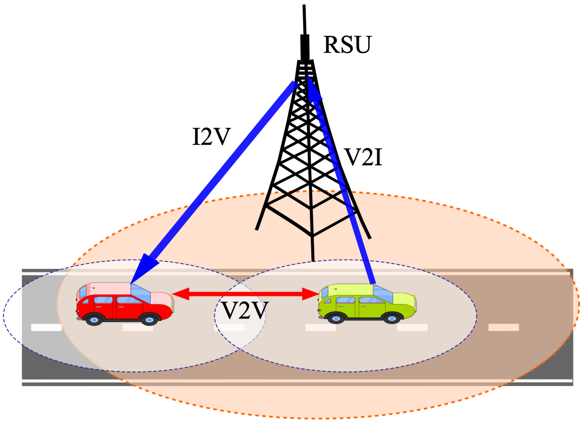


Figure 2.1: Communications in FVNETs.

efficiency, reliability and management in 5G transport systems.

Mehrdad et al. [58] have presented two-way cooperative communications in inter-vehicular networks for communication, namely vehicle-assisted V2V communication and access point-assisted V2V communication. In vehicle-assisted V2V transmission, the source and relay transmission channel is adapted as cascaded Nakagami. In access point-assisted, the transmission channel between the source and the access point is adapted as Nakagami. Tuan et al. [59] have explored the I2V and infrastructure-to-infrastructure (I2I) cooperative communications in an energy-limited vehicular network for driving assistance and efficient traffic management services in ITS. Cooperative techniques, such as relay selection and multiple-input-multiple-output (MIMO), are applicable for enhancing performance and mitigating the energy consumption of FNs. Chapters 4-7 of this thesis examine infrastructure-based communications between FNs (i.e., RSUs or HPNs) and vehicles for data transmission services in FVNETs. I2V communication is used for downlink transmission from FNs to vehicles, and V2I communication is used for uplink transmission from vehicles to FNs. Additionally, these chapters leverage I2I cooperative communication to coordinate FNs within FVNETs.

Information dissemination in VANETs, where the wireless channel is shared by vehicles within the communication range of each other, is affected by the node interference phenomenon. This node interference in the network limits the network's capacity due to the near-linear spatial distribution of vehicles. Therefore, the performance and condition of the network in disseminating information are affected by the range and rate of transmission of information [60]. Wang et al. [61] have presented a system architecture to assist the RSUs in data dissemination in V2I communication. They exploited the benefits of V2V communication among vehicles for sharing data. The author proposed a clustering method where vehicles join and leave based on their velocities. Rmaiyan et al. [62] have used vehicles as intermediate nodes for store-and-forward to transfer data from a fixed source node to a destination node. They addressed the communication problem between two nodes that were far from their communication range using vehicles as relays. Wang et al. [63] have proposed a store-carry-forward scheme to reduce the transmission time of vehicles in dark areas. The dark areas are the coverage areas not covered by the range of stationary RSUs in VANETs. Hence, their proposed scheme uses the vehicles in both directions as relays for transmitting data to the target vehicles in the dark areas.

The dynamic topology arising from the rapid mobility of vehicles and uncertain channel conditions within vehicular networks presents a challenge for data transmission, especially for large-size data. Hence, Chen et al. [64] have proposed a strategy for cooperative communication to transmit large-size data by exploiting V2I, V2V and the mobility of vehicles in the network. They provided the relationship between infrastructure, vehicles, vehicular density and V2V and V2I communications transmission rates. The study in [65] provides the cooperation between centralized and decentralized data transmission in VANETs. In centralized data transmission, RSUs disseminate the data using I2V and V2V cooperative communications. RSUs coordinate with each other by transferring undelivered requests. In decentralized data transmission, vehicles moving in opposite directions are used for data sharing via V2V communication. In [29, 66], the authors have focused on data propagation for seamless communication between RSU and OBU in VANETs. However, data propagation tends to fail due to the mobility of vehicles. In this thesis, infrastructure-based (i.e., I2V and V2I) cooperative communications are used for data dissemination between FNs and vehicles in Chapters 4-7. On the other hand, I2I communication is used in these chapters for data dissemination among FNs.

The existing resource management algorithms proposed by the researchers [24–27, 39, 67–79] can be segregated considering objectives like service capability, throughput, QoS, serviceability, spectral efficiency, etc.

2.1.1.1 Service Capability

The DRO scheme in [24] coordinates the resource allocation amidst pairs of FNs in fog-enabled connected vehicle networks. The DRO scheme maximizes the network's service capability by reducing vehicles' RBs amid pairs of FNs. The problem of RBs reduction is formulated as a maximum weight matching problem to obtain the set of edges as a solution for optimal RBs shifting among FNs. As a result, there is a lowering in throughput, serviceability, availability, and service capability as vehicles arriving at the network increase using this scheme. However, the DRO scheme enhances the throughput, serviceability, availability, and service capability when integrated with the SA algorithm [25].

2.1.1.2 Throughput

A downlink sum-rate optimization (DSRO) scheme in [67] grants the resource allotment using the Hungarian method among remote radio heads (RRHs) in fog radio access networks (F-RANs). The DSRO scheme enhances the throughput and serviceability of the network. However, the authors do not consider the network's service capability and availability. A joint user association technique in [25] uses graph colouring and resource partitioning to reduce the co-tier interference among nodes in two-tier heterogeneous networks. This approach improves user throughput. However, serviceability is reduced due to an imbalance among FNs when there is an increase in vehicles connecting the network. Finally, a joint user association and user scheduling approach in [68] addresses load balancing using a network-wide utility maximization problem over downlink in heterogeneous networks. The nonconvex throughput is obtained using a concave function with user scheduling. In [68], user association and user scheduling (UA-US) are implemented using the distributed convex optimization approach.

2.1.1.3 Quality of Service

The semi-Markov decision process (SMDP) in [69] facilitates video streaming in VANETs. Their proposed resource allocation (RA) scheme provides better QoS by increasing the network's bandwidth. However, it leads to an imbalance amid FNs. Various device cooperative approaches, such as SA in [25], capacity aware in [27], and content-aware in [26], provide better network utilization and serviceability for traditional networks. However, these cooperative techniques use resource scheduling, leading to an inequitable imbalance amid FNs, impacting network utilization efficiency and serviceability. The fuzzy-based systems for resource coordination in [70] are proposed to decide on running applications on fog or cloud layers in VANETs backed by a software-defined network. It improves network utilization efficiency. Priority queue, fuzzy and analytical hierarchy process, a priority-based scheduling algorithm, is proposed in [71] to optimize the latency issues in mobile fog computing. A fair and efficient multi-resource allocation scheme is proposed in [72] to maximize the resource utilization of F-RANs while satisfying economic properties. Furthermore, Adnan et al. [73] have proposed a priority-based scheduling algorithm to address QoS concerns in software-defined vehicular networks. The priority-based scheduling mechanism improves QoS by categorizing the traffic flow into safety and non-safety queues. Several researchers in [74–76] have dedicated their efforts to addressing network load balancing issues in pursuit of achieving an optimal resource allocation. These load-balancing challenges become particularly prominent as the number of vehicles entering the network rises significantly.

2.1.1.4 Serviceability and Spectral Efficiency

The ARB in [39] is proposed for addressing imbalance among RRHs. The ARB scheme uses the backpressure algorithm and the Hungarian method to select user equipment and maximize the serviceability of the F-RANs. However, the authors do not assess other performance measures, such as resource utilization efficiency and service capability, which disturb the network performance as the number of vehicles entering the network rises. The methods in the literature [25,67,69] provide data optimization but need to furnish resources effectively due to the inexact results. Various researchers have been focused on spectral efficiency, a performance metric for network evaluation [77]. However, the optimization techniques proposed in [78] and game theoretical models [79] can be used to enhance this metric.

2.1.2 Energy-Efficiency in Vehicular Networks

Kansal et al. [45] have presented a systematic review of FC and its advantages in autonomous vehicles, smart traffic control, IoV, and many more applications. In traditional vehicular networks, there are various restrictions, such as latency, location cognition, and real-time responses, for delay-sensitive applications. They are addressed by integrating FC with vehicular networks, referred to as vehicular fog computing (VFC) [16]. Moreover, vehicles endowed with computational capacities produce a range of data that can be transferred to the FN for optimized processing [46]. However, energy consumption, response time and throughput have become necessary to enhance the efficiency of the network. Many researchers are dedicating attention to these research directions to improve the performance of FVNETs. Hammad et al. [80] have presented a greedy scheduling algorithm called the nearest fastest set (NFS) scheduler. The NFS schedules faster vehicles among those close to the RSUs. Zhang et al. [81] have offered an offline scheduling algorithm for switching on or off RSU. This algorithm minimizes the energy usage cost by multiple RSUs while providing services to the vehicles.

Abdulla et al. [40] have proposed online and offline scheduling algorithms for constant bit rates. More specifically, online algorithms based on the greedy approach have been offered to schedule energy-efficient vehicles within the RSU range. An offline scheduling algorithm is presented to reduce downlink energy communication costs while providing services to vehicles' requests within the coverage of RSUs. Atoui et al. [82] have offered offline and online scheduling for energy harvesting in RSUs to maximize the number of vehicles serving. Satish et al. [32] have presented an energy-efficient nearest neighbour forwarder (NNF) augmented by the MCF algorithm. The MCF-NNF schedules a relay vehicle near RSU to deliver the remaining data to a vehicle in the uncovered region of RSU. Task scheduling approach in [83] minimizes the offloading energy usage of IoT devices in IoT-fog-cloud architecture. This approach ensures the maximum tolerable delay for tasks considering multi-user multi-fog node scenarios. Jang et al. [84] have proposed an optimal task-offloading strategy for vehicular edge computing for preserving energy in energy-limited vehicles. This offloading strategy significantly reduces the energy consumed by vehicles when offloading tasks to edge nodes.

Asim et al. [55] have proposed a non-orthogonal multiple access multicasting-based energy-efficient power allocation scheme for V2X communications. Their proposed scheme enhances the energy efficiency of RSUs and QoS of vehicles by transforming vehicles' power allocation from the RSUs problem into concave-convex fractional programming through convex approximation. Using a normalized min-max algorithm, Amir et al. [85] have proposed two algorithms for downlink traffic scheduling in vehicular networks. The first algorithm is a greedy algorithm for selecting RSU, and the second algorithm assigns the minimum energy time slot. Nazeri et al. [86] have proposed an evolutionary-based energy-aware scheduling in FC to reduce energy consumption. Pejman et al. [87] have proposed an energy-aware task scheduling to minimize energy consumption in FC using the dynamic voltage and frequency scaling method. However, they generated task sequences using an evolutionary algorithm. Chapters 6 and 7 present the energy-efficient resource allocation and task scheduling algorithms in FVNETs, respectively. Specifically, Chapter 6 presents the efficient RA algorithm to minimize the energy consumption of FNs. Chapter 7 offers the energy-efficient task scheduling algorithm in FVNETs to reduce the energy consumption of FNs and enhance the network throughput.

2.1.3 Task Scheduling in FVNETs

Edge computing and FC are identical in processing the data on computing edge nodes near the end user. However, the rationality between edge computing and FC is in using computing resources from different types of devices. Edge computing was initiated by IBM and Nokia Siemens Network in 2013, whereas FC was introduced by Cisco in 2012 [88]. Edge computing focuses on the devices from local networks or mobile and embedded devices that generate data. In contrast, FC focuses on nodes situated at the network's edge so that the data can be distributed among edge nodes called FNs for processing [88].

Vehicular services require high computational capability for processing. Hence, the vehicles offload these tasks to FNs for computational resources and immediate response in FC [10]. Moreover, the IoV provides connected vehicles for vehicular applications. As a result, a massive amount of IoV data will be generated to process vehicular services. Hence, to meet the computational needs of IoV data, the vehicles are used as computational infrastructures in FC called VFC [89]. Arnav et al. [90] have presented the vehicular congestion identification system using the IoV for vehicular networks. This system is feasible and performs accurately in various scenarios. Feng et al. [46] have proposed an autonomous vehicular edge framework using ant colony optimization to increase the computational capability of vehicles in a dynamic vehicular environment. This framework addresses job assignments by scheduling the jobs using the idle resources of vehicles.

Jiang et al. [91] have proposed a delay-aware task offloading (DATO) scheme for scheduling heterogeneous deadline-sensitive tasks in fog networks. The DATO is a basic offloading model that minimizes overall task disutility by capturing the critical features of the fog network. PSG algorithm is presented by Azizi et al. [41] to schedule diverse IoT tasks in a fog environment. This algorithm tries to minimize tasks' deadline violation time by considering tasks that did not meet their deadline. It also optimizes the energy usage of FNs and maximizes the system's active time while fulfilling the QoS necessities of tasks. Deadline-aware dynamic task placement (DDTP) process to schedule the tasks in appropriate FNs in fog networks is proposed by Sarkar et al. [92]. DDTP is a federated framework containing many fog clusters, and it chooses the suitable FN to place the tasks such that communication time and task completion time are reduced while meeting deadline constraints. Fair scheduling is proposed by Mithun et al. [93] for offloaded tasks in FC. This scheduling strategy increases the number of tasks meeting deadline constraints while retaining network stability.

Mainak et al. [94] have proposed a delay-dependent priority-aware task offloading approach to schedule IoT tasks among suitable FNs. This approach uses the multilevel-feedback queue to distinguish the priority of the tasks, organizes the tasks based on the resource vacancy in FNs, and dispatches delay from IoT devices to diminish the offloading time while satisfying the deadline. Tang et al. [95] have proposed a greedy offloading decision algorithm (GODA) to optimize the latency in VFC. GODA is a greedy scheduling strategy for deciding task offloading among vehicles in VFC, considering the vehicles in the FN's vicinity as mobile FNs. Zhu et al. [96] have suggested Fog Following Me

(Folo), a resolution to offload tasks among mobile FNs. Folo achieves optimal latency and quality-balanced task assignment in VFC by considering restrictions on fog capacity, quality loss, and latency. An application offloading strategy is presented in [97] using swarm optimization for a fog environment. The accelerated particle swarm optimization approach detects suitable FN for real-time tasks based on QoS parameters such as resource usage and computation cost to reduce the overall delay in computation time and average cost. In this thesis, FC architecture is used in vehicular networks. Chapter 7 presents the delay-aware task scheduling in energy-limited FVNETs to reduce the energy usage of FNs in overlap coverage regions of FVNETs.

2.1.3.1 Reinforcement Learning for Vehicular Networks

Recently, many RL-based algorithms were proposed for resource allocation [98, 99], task offloading [10, 38, 100] and relay selection [101] in vehicular networks. For efficient resource allocation in VFC, Seung et al. [98] have proposed an RL-based algorithm considering the vehicles' movement and parking status from the smart environment. Further, the authors combined the RL algorithm with a heuristic algorithm to allocate fog resources and minimize latency. Dong et al. [99] have presented an RL technique to learn RA in a wireless network where the user's mobility is high. They trained the RL agent by increasing the mobility of each user sequentially so that the RL agent could learn accurately. RL's upper confidence bound (UCB) can be used for making sequential decisions under uncertain situations as it quickly convergences [102, 103].

Peng et al. [10] have proposed UCB learning-based task offloading to reduce energy consumption using resources of idle vehicles in vehicular collaborative edge computing. Q-learning-based protocol for energy-efficient adaptive scheduling using RL (PEARL) is presented in [38] to schedule tasks in vehicular networks. The authors assume that the FNs are equipped with energy-limited rechargeable batteries. Hence, PEARL tries to preserve the FNs' energy during discharge and enhance the total completed request and throughput per vehicle. An offloading decision-based state-action-reward-state-action (OD-SARSA) is proposed for time-critical task offloading and resource allotment by Taha et al. [100]. OD-SARSA is a reinforcement learning technique for addressing resource management problems in mobile edge computing to mitigate system cost, processing delay and energy usage. An RL method is proposed to select the relay for broadcasting the task in VANETs, which uses random topology and dynamic behaviour in [101]. They used neural network classification to select forwarding nodes.

The resource management affects the performance of the fog-enabled networks. Hence, Wei et al. [104] have proposed a joint optimization approach for RA in fog-enabled networks using deep reinforcement learning (DRL). They used an actor-centric RL framework to make decisions and reduce latency. e-Divert [105], a multi-agent DRL technique, is proposed using a convolution neural network (CNN) to enhance cooperation among vehicles and competition among them and charging stations to improve energy efficiency and data transmission such that energy consumption is reduced. Li et al. [106] have proposed a DRL-based method to adjust the user's transmit power to share a similar spectrum with another user. This approach trains users to transmit their data with the required QoS in vehicular networks. He et al. [107] have proposed a DRL-based integrated framework for vehicular networks. This framework coordinates the caching and computing resources for the vehicles in the network to enhance the performance of next-generation vehicular networks. A task scheduling algorithm using Q-Learning is presented in Chapter 7 to reduce the energy consumption of FNs in FVNETs. The proposed algorithm schedules the delay-sensitive tasks among FNs of an overlap coverage region in FVNETs to reduce the energy consumption of FNs and maximize the network throughput.

2.1.3.2 Fuzzy Logic for Vehicular Networks

Fuzzy logic is used in various domains, such as image processing, artificial intelligence, control systems, medical diagnosis and natural language processing. Fuzzy logic was introduced by Zadeh in 1965, and it is used to make decisions from uncertain information [108]. Fuzzy logic has three stages: 1. Fuzzification 2. Fuzzy inference engine, and 3. Defuzzification. Figure 2.2 shows the association among these stages in the fuzzy logic system.

Fuzzification takes the crisp values (i.e., input parameters), such as the vehicle's velocity and distance from FNs, and maps to fuzzy values using membership functions. The fuzzy inference engine maps the fuzzy values obtained from fuzzification into fuzzy values

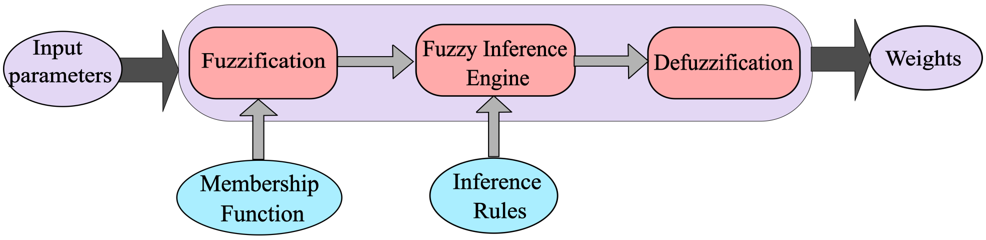


Figure 2.2: Fuzzy logic system.

using predefined inference (i.e., IF-THEN) rules. These inference rules are defined from numerical data. For instance, **IF** $velocity$ is very high and $distance$ is very close **THEN** mark the $vehicle$ to high. Then, the defuzzification maps the fuzzy values obtained from the inference engine to crisp values based on output membership functions.

Sathish et al. [108] have presented the fuzzy-based reinforcement learning (FRL) technique for task offloading in vehicular fog networks. FRL schedules the vehicles to offload IoT tasks to mobile FNs. This greedy-based technique mitigates the energy consumption of FNs to improve the efficiency of the network. An intelligent localization method is proposed by Lina et al. [109] to determine the location of a vehicle in the networks using fuzzy logic. The vehicles are assigned the weights using fuzzy logic to locate the vehicle location accurately. A fuzzy logic-based multi-hop intelligent broadcast protocol [110] is proposed to decide whether to rebroadcast. It uses the user's coverage, connectivity, and mobility to make decisions on broadcasting using fuzzy logic. Multi-hop broadcast is a mechanism for safety applications in vehicular networks, in which vehicles send beacon notices to neighbouring vehicles and identify the multi-hop neighbours based on mobility, distance and connectivity.

2.1.4 Applications of Cooperative Connections in FVNETs

A survey on cooperative vehicular networking presented by Ahmed et al. [111] gives an overview of scheduling, routing, and security in vehicular infrastructure. Cooperative connections/communications, such as I2V/V2I and V2V, assist in ITS for smart navigation, efficient traffic management and remote intelligent control. The applications of cooperative communications are discussed as follows.

2.1.4.1 Smart Navigation

The navigation system directs the vehicles towards their destination via optimal navigation paths. Thus, the navigation system eases transportation through the network-wide optimal paths. Therefore, Jeong et al. [112] have proposed a self-adaptive interactive navigation tool (SAINT) for cloud-based vehicular networks. SAINT provides the interaction between the cloud and the vehicles by collecting travel paths and vehicle navigation experiences. Subsequently, the cloud uses the collected information for better navigation guidance for other vehicles. Evolved SAINT+ [113] is proposed to reduce the vehicle delivery time for emergency services. It also optimizes the navigation routes of vehicles affected by the accident areas. A virtual path reservation strategy is used to guarantee fast delivery of emergency services. Further, the vehicles near the affected accident areas are navigated using the congestion contribution matrix and protection zones.

2.1.4.2 Traffic Management

Platooning is a road train formed by a group of vehicles without coupling between them. A short distance is maintained between vehicles using V2V communication in platooning. Van et al. [114] focused on highway platooning to enhance the fuel efficiency of trucks. Green light optimal speed advisory (GLOSA) [115] is an information advisory on traffic signals. GLOSA predict when the signal will turn green, and the driver can adjust the speed to avoid congestion coming ahead. The authors in [115] also introduced the signal guru, a software service that enables GLOSA to predict traffic signals by leveraging mobile phones.

2.1.4.3 Self Driving Cars

Shen et al. [116] have proposed a context-awareness safety driving (CASD) framework for safe driving in vehicular networks. CASD leverages the V2V cooperative communication to share various information and provides vehicles with a safety action plan for three situations: line-of-sight unsafe, non-life-of-sight unsafe and the safe situation. In line-of-sight unsafe conditions, the vehicles take over the driving, ensuring less risk when driver action fails. Stop-and-go traffic oscillation in VANETs causes uncomfortable driving and additional energy consumption. Therefore, adaptive cruise control (ACC) is widely used to deal with traffic oscillations by leveraging V2V. Cooperative adaptive cruise control (CACC) is an extension of ACC to maintain steady speed differences between vehicles in platoons and guarantee V2V communication for safety [117].

2.2 Summary

This chapter discusses the challenges of FVNETs. Then, it discusses the advantages of incorporating FC in VANETs for vehicular services and various task scheduling approaches in FVNETs. Different RL approaches and fuzzy logic for FVNETs are presented for energy efficiency and task scheduling in FVNETs. Further, various cooperative communications and ways of data transmission in FVNETs using these cooperative communications are discussed. This chapter also presents the applications of cooperative communications in FVNETs, resource management, and other challenges associated with resource management in FVNETs.

Chapter 3

Preliminaries

This chapter presents the preliminaries in FVNETs and common terminologies, followed by the performance metrics, such as service capability, serviceability, availability, and throughput, used to evaluate the performance of proposed algorithms and compare them to the existing algorithms.

3.1 Preliminaries in FVNETs and Common Terminologies

The primary elements of FVNETs are vehicles and FNs. The connection between vehicles and FNs is established using wireless access for the vehicular environment (WAVE) protocol stack of DSRC technology. DSRC is a prominent technology for utilizing distributed channel access based on the IEEE 802.11p protocol [118]. Other components of FVNETs are OBUs and FNs. Here, the applications are hosted by either FNs or OBUs, but the FNs act as service providers to the vehicles. Therefore, the devices that consume the services are users, and the devices that host the applications are providers.

3.1.1 Dedicated Short-Range Communication

DSRC is a core part of the WAVE standard, installed in many countries since its release and used for V2V communication [55]. According to the United States Department of Federal

Communication Commission, 75 MHz licensed spectrum at 5.9 GHz frequency band is assigned for DSRC [119]. The allocated spectrum is split into six SCH and one CCH, each channel having a capacity of 10 MHz, as shown in Figure 3.1. SCH channels dispatch the public safety messages such as collision warnings and driver safety messages. In addition, SCH channels are also used for video and audio message dispatching. Similarly, the CCH channel is used to dispatch control notices.

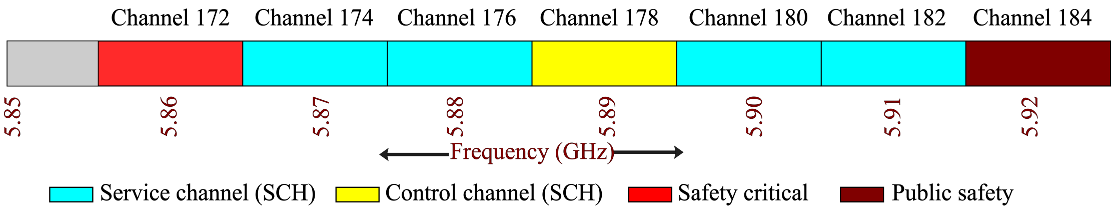


Figure 3.1: DSRC allocated spectrum.

3.1.2 Resource Block

A resource block (RB) is the smallest time-frequency unit in an OFDMA system. The network must allocate RBs with power allocation before data transmission [35]. The LTE channel bandwidth can be 1.4, 3, 5, 10, 15, or 20 MHz according to 3GPP specifications. Further, all available spectrum is divided into RBs. Then, the number of available RBs depends on the LTE channel bandwidth, shown in Table 3.1 [34, 120].

Table 3.1: Number of available RBs according to channel bandwidth

Channel bandwidth (MHz)	1.4	3	5	10	15	20
Number of RBs	6	15	25	50	75	100

3.1.3 Onboard Unit

The major operations of onboard unit (OBU) are ad-hoc routing, message transfer, wireless radio access, and Internet protocol (IP) mobility [121]. Vehicles fitted with OBUs can establish connections with other vehicles or FNs. Further, the advances in dual-radio OBUs enable full-duplex communication in vehicles. As a result, the vehicles can dispatch and collect messages concurrently through identical channels. In this dual-radio OBU, the first radio is tuned to SCH, publicizing public safety notices, such as V2V crash-dodging notifications and driver safety messages. The second radio switches among CCH and SCH in uniform intermissions [122].

3.1.4 Fog Nodes

Fog nodes (FNs) are the network's edge nodes in the FC. This thesis considers roadside infrastructures (i.e., RSU and HPNs) as FNs [24]. These FNs are situated along the road and use wireless communication technologies to furnish vehicular services to vehicles. FNs are coordinated through the central supervisor and connected to the cloud through the Internet.

3.2 Performance Metrics

The effectiveness of the proposed algorithms compared to existing algorithms is evaluated in terms of service capability, serviceability, availability, throughput and FN's energy consumption. A brief description of these performance metrics is given as follows.

3.2.1 Service Capability

The service capability of a FN is the ratio of remaining resources to the total number of resources in that FN at time slot t. Similarly, network service capability is obtained by dividing the sum of remaining resources at each FN by the total resource capacity of the network at time slot t. Detailed mathematical descriptions of a FN's and network's service capability are provided in Chapter 4.

3.2.2 Serviceability and Availability

Serviceability and availability of the network are defined as the percentage of vehicles getting services within the desired and minimum data rate (e.g., throughput and delay), respectively [39, 123].

3.2.3 Throughput

Throughput is the network's capability to provide new RBs to vehicles. However, network throughput at time slot t depends on the vehicle's achievable rate and time fraction allotted to the vehicle. We use the Shannon formula to find the vehicles' achievable rate at each time slot from each FN. A detailed description of finding network throughput at time slot t is given in Chapter 5.

3.2.4 FN's Energy Consumption

This thesis considers two different ways of FN's energy consumption. Firstly, the FN energy consumption in transmitting data in a downlink channel to a vehicle using transmission power. Secondly, the energy used by FN to process the task and transmit the data using transmission rate. The energy consumption of a FN using transmission power depends on the bandwidth of a FN and the distance of the vehicle from that FN, which is described in Chapter 6. Similarly, the energy consumption of FN depends on processing the task and amount of transmission rate required to transmit the data to a vehicle, described in Chapter 7.

3.3 Summary

This chapter discusses the preliminaries of FVNETs, establishing connections between vehicles and FNs using WAVE protocol and available RBs according to FN channel bandwidth and common terminologies. Then, it discusses the various performance metrics considered to compare the performance of the proposed algorithms with the existing algorithms.

Chapter 4

A Dynamic Resource Management Algorithm for Maximizing Service Capability in FVNETs

This chapter presents dynamic resource management (DRM), a polynomial time algorithm, for efficient resource allocation among FNs to improve their service capability and resource utilization efficiency. In this chapter, we consider the set of vehicles in overlapped coverage regions of two or more FNs and served by those FNs. The allocated RBs of vehicles in the overlapped coverage regions are migrated between pairs of FNs. This migration of RBs reduces the allocated RBs of vehicles in overlapped coverage regions to maximize the service capability. The existing algorithms [24, 25, 39, 69] migrate the RBs of vehicles between pairs of FNs without considering the load on FN. On the contrary, the proposed algorithm migrates the RBs of vehicles between pairs of FNs, such that the load on FN is minimal. The differences between the proposed and existing algorithms in terms of performance metrics, fog environment and vehicles in overlapped coverage regions are shown in Table 4.1.

The proposed algorithm is simulated by taking 10 to 50 FNs and 300 to 2100 vehicles at an arrival rate of 10 vehicles/s. We compare the simulation results with DRO [24], SA [25], DRO + SA and RO in terms of service capability, serviceability, availability, throughput, and resource utilization efficiency. The simulation outcomes show that the

Table 4.1: The differences between the proposed and existing algorithms

Work	**Performance metrics**	**Fog environment**	**Analysis**	**Vehicles in overlapped coverage region**
Vu et al. [24]	Service capability, serviceability, availability and throughput	√	There is a reduction in the service capability and serviceability as vehicles connecting to the network increase.	×
Liu et al. [25]	Throughput	×	They have not considered the service capability, serviceability and availability.	×
Dao et al. [39]	Serviceability, availability and throughput	×	They have not considered the service capability.	×
He et al. [69]	Throughput, bandwidth utilization	×	There is a reduction in the serviceability of the network.	×
Proposed algorithm (DRM)	Service capability, serviceability, availability and throughput	√	We enhance the service capability, serviceability, availability, throughput and resource utilization efficiency as vehicles connecting to the network increase.	√

proposed algorithm reduces occupied RBs among FNs by migrating RBs of the set of vehicles and achieves better service capability, serviceability, availability, throughput and resource utilization efficiency than other migration algorithms, such as DRO, SA, DRO + SA and RO. The major contributions of this chapter are listed as follows.

1. We consider allocating RBs to newly arrived vehicles by migrating RBs between pairs of FNs without affecting their services. The RBs of vehicles are migrated to minimize overall allocated RBs.

2. The optimal RBs migration problem in FVNETs is formulated into ILP by considering the variables that impact FNs' resource constraints and the network's service capability.

3. We propose a polynomial-time DRM algorithm for optimal migration of RBs between pairs of FNs to minimize occupied RBs among FNs and maximize the service capability and resource utilization efficiency of the network by migrating RBs of a set of vehicles.

4. We present extensive simulations to show that the DRM algorithm can achieve better performance than the four existing algorithms in terms of service capability, serviceability, availability, throughput, and resource utilization efficiency of the network.

The simulation outcomes show that the proposed DRM algorithm minimizes the occupied RBs among FNs by migrating RBs of vehicles in the overlap coverage regions and enhances the network's service capability, serviceability, availability, throughput and resource utilization efficiency when compared to DRO [24], SA [25], DRO + SA and RO. The results also show the impact of the increased number of vehicles connecting to the network.

The rest of this chapter is structured as follows. The problem statement is presented in Section 4.1. Section 4.2 gives the polynomial time DRM algorithm for RB migration by formulating the RB migration problem into ILP. Section 4.3 demonstrates the evaluation of the proposed algorithm with existing algorithms. The summary of this chapter is presented in Section 4.4.

4.1 Problem Statement

Consider a FVNETs in a city area A in which $\mathcal{G}$ number of FNs are deployed. Each FN i, $1 \leq i \leq \mathcal{G}$, has a communication range R and it can overlap with the communication range of neighboring FN j, $1 \leq j \leq \mathcal{G}$, $i \neq j$. A FN i provides services to the set of vehicles $\mathcal{P}_t(i)$ reaching its communication range at time slot t. Note that the Poisson distribution is used for the arrival rate (departure rate) of vehicles to (from) the network with mean value λ (or μ). Table 4.2 summarizes the different notations and their definitions used in this chapter.

Table 4.2: Important notations and their descriptions in the system model

Notation	**Description**
$\mathcal{G}$	Number of FNs
λ (or μ)	Mean arrival (or departure) rate of vehicles
L_i	Capacity of ith FN in terms of RB units
$\mathcal{Z}$	Set of overlapped coverage regions in FVNETs
$\mathbb{N}$	A set of natural numbers
$\mathbb{E}$	Expected value
$\mathcal{P}_t(i)$	A set of vehicles served by i^{th} FN at time slot t
$O_t(i)$	Currently occupied RBs of i^{th} FN at time slot t
$A_t(i)$	Available RBs of i^{th} FN at time slot t
$Sc_t(i)$	Current service capability of i^{th} FN at time slot t
Sc_t	Service capability of the network at time slot t
$\mathcal{P}_t^{in}(i)$	A set of vehicles reaching to the i^{th} FN at time slot t
$\mathcal{P}_t^{out}(i)$	A set of vehicles leaving from the i^{th} FN at time slot t
$\mathcal{P}_t(ij)$	A set of vehicles in overlapped coverage area between pairs of FNs i and j at time slot t
$p_t^*(ij)$	An optimal set of vehicles for migrating RBs between pairs of FNs i and j at time slot t
$\mathcal{P}_t^*(i)$	An optimal set of vehicles served by i^{th} FN after resource blocks migration at time slot t
b_k^i	Number of RBs allocated by i^{th} FN to k^{th} vehicle

Let b_k^i be the total number of RBs required for k^{th} vehicle with data rate r^k from the i^{th} FN, when k^{th} vehicle is present in the coverage area of i^{th} FN. It can be obtained as follows [24, 124].

$$b_k^i = \left\lceil \frac{1}{B \times \log_2(1+\psi(ik))} \times r^k \right\rceil \tag{4.1}$$

where B denotes the bandwidth that is used by one RB in a period of 1 ms, which is equivalent to 180 KHz [125], $b_k^i \in \mathbb{N}$, and $\psi(ik)$ is the signal strength between the i^{th} FN and k^{th} vehicle on the data channel [126].

Let $\mathcal{P}_t^{in}(i)$ be a set of vehicles arriving to i^{th} FN at time slot t and $\mathcal{P}_t^{out}(i)$ be a set of vehicles departing from i^{th} FN at time slot t, after successful completion of their tasks. In this situation, the mean departure rate of the vehicles from the network (μ) is obtained as follows [24].

$$\mu = \mathbb{E}\left[\sum_{i=0}^{\mathcal{G}} |\mathcal{P}_t^{out}(i)|\right] \tag{4.2}$$

At time slot t, the number of vehicles served by FNs is denoted by $\mathcal{P}_t(i)$ and can be defined as follows [24].

$$\mathcal{P}_t(i) = \mathcal{P}_{t-1}(i) \cup \mathcal{P}_t^{in}(i) \setminus \mathcal{P}_t^{out}(i) \tag{4.3}$$

In a particular scenario, the number of vehicles getting service is limited due to resource constraints. In this circumstance, the remaining RBs, $A_t(i)$ of i^{th} FN, after assigning the RBs to vehicles in $\mathcal{P}_t(i)$, can be obtained as follows [24].

$$A_t(i) = L_i - O_t(i) = L_i - \sum_{k=1}^{|\mathcal{P}_t(i)|} b_k^i \tag{4.4}$$

where $O_t(i)$ and L_i are the occupied RBs and capacity of the i^{th} FN at time instant t, respectively. The service capability of i^{th} FN at time slot t is given as follows [24].

$$Sc_t(i) = A_t(i) \times \frac{1}{L_i} \tag{4.5}$$

In the similar fashion, the network service capability at time slot t can be determined as follows [24].

$$Sc_t = \sum_{i=0}^{\mathcal{G}} A_t(i) \times \frac{1}{\sum_{i=0}^{\mathcal{G}} L_i} \tag{4.6}$$

The deluge of vehicles arriving at the network can connect to preferred FNs based on the signal strength and favorite contents. The number of vehicles connecting to these FNs increase rapidly over time. It can exhaust the capacity of FNs and become impotent to provide the services because of limited resource constraints. As a result, the new incoming vehicles connect to the FNs that own inadequate resources. In order to provide better services, the services of the vehicles are migrated between pairs of FNs without increasing the number of RBs and balancing the load among the FNs. Alternatively, FNs are required to assign a large number of RBs to satisfy these vehicles' required latency and data rate. Otherwise, it leads to a shrink in the network service capability and resource utilization efficiency.

In this chapter, we consider allocating RBs to newly arrived vehicles among the FNs without increasing the number of RBs, such that the following objectives are fulfilled.

1. Network service capability is maximized.
2. Network serviceability is maximized.
3. Network availability is maximized.
4. Network throughput is maximized.
5. Resource utilization efficiency is maximized.

4.2 Dynamic Resource Management Algorithm

The proposed algorithm, DRM, is a resource management algorithm for maximizing service capability in FVNETs. The objective of this algorithm is to maximize the service capability, serviceability, availability, throughput, and resource utilization efficiency of the networks without increasing the number of RBs of the FNs. The basic idea of the proposed algorithm is as follows. Firstly, DRM identifies the vehicles in the overlapped region of the pairs of FNs and determines the number of RBs allocated to those vehicles from the FNs. Then DRM migrates the RBs of vehicles between the pairs of FNs in order to reduce the occupied RBs on the FNs. It is noteworthy to mention that the reduction of RBs leads to an increase in service capability, serviceability, availability, throughput, and resource utilization efficiency of the networks. The detailed description is discussed as follows.

At time slot t, assume that the number of vehicles $|\mathcal{P}_t(i)|$ connecting to preferred FNs increases, for which the resources of the FNs are tends to exhaust. Next, consider the vehicles present in a coverage area of two or more FNs and served by those FNs. Let $\mathcal{P}_t(ij)$ be a set of vehicles in an overlapped coverage area of i^{th} FN and j^{th} FN at time slot t. The set of vehicles, $\mathcal{P}_t(ij)$, between pairs of FNs i and j, $1\leq i, j \leq \mathcal{G}$, $i \neq j$ can be obtained as follows.

$$\mathcal{P}_t(ij) = \mathcal{P}_t(i) \cap \mathcal{P}_t(j) \tag{4.7}$$

The $\mathcal{P}_t(ij)$ represents vehicles for which RBs can be migrated between pairs of FNs. Let $\mathcal{Z}$ be the set of overlapped coverage areas in FVNETs. The vehicles' RB migration between pairs of FNs takes place only if there exists a minimum of one vehicle in the overlapped coverage area $(i, j) \in \mathcal{Z}$ and is served by those pairs of FNs. Mathematically, $\mathcal{P}_t(ij) \neq \emptyset$.

Let $p_t^*(ij)$ be an optimal set of vehicles whose RBs can migrate from i^{th} FN to j^{th} FN and its converse $p_t^*(ji)$ be an optimal set of vehicles whose RBs can migrate from j^{th} FN to i^{th} FN. Let $\mathcal{P}_t^*(i)$ and $\mathcal{P}_t^*(j)$ be an optimal set of vehicles served by i^{th} and j^{th} FNs after optimal migration of RBs is performed. They can be derived as follows.

$$\left.\begin{aligned} \mathcal{P}_t^*(i) &= (\mathcal{P}_t(i) \setminus p_t^*(ij)) \cup p_t^*(ji) \\ \mathcal{P}_t^*(j) &= (\mathcal{P}_t(j) \setminus p_t^*(ji)) \cup p_t^*(ij) \end{aligned}\right\} \tag{4.8}$$

The optimal RB migration among FNs in resource-constrained FVNETs can be formulated as an ILP problem, which is discussed in Section 4.2.1.

4.2.1 ILP Formulation for Optimal RB Migration

We define a boolean variable x_{ij}^k to denote a vehicle $k \in \mathcal{P}_t(ij)$ chosen for migrating RBs between i^{th} and j^{th} FNs, $\forall(i,j) \in \mathcal{Z}$.

$$x_{ij}^k = \begin{cases} 1, & \text{If } k^{th} \text{ vehicle is chosen for migrating} \\ & \text{RBs between pairs of FNs } i \text{ and } j \\ 0, & \text{Otherwise} \end{cases}$$

To maximize the network resource utilization efficiency, the vehicles in the overlapped coverage area (i,j), $\forall(i,j) \in \mathcal{Z}$, are considered for optimal migration of RBs between pairs of FNs i and j, such that the overall occupied RBs by these vehicles is minimized. The optimal service migration among pairs of FNs i and j can be formulated as follows.

$$min \sum_{\forall(i,j)\in\mathcal{Z}} \left(\sum_{\forall k \in \mathcal{P}_t(ij)} x_{ij}^k r_{ij}^k \right) \tag{4.9}$$

subjected to

$$\sum_{\forall k \in \mathcal{P}_t(ij)} x_{ij}^k r_{ij}^k \leq \delta_{ij}, \forall(i,j) \in \mathcal{Z} \tag{4.10}$$

$$\sum_{(i,j)\in\mathcal{Z}} x_{ij}^k \leq |\mathcal{P}_t(ij)|, \forall k \in \mathcal{P}_t(ij), 1 \leq i,j \leq \mathcal{G}, i \neq j \tag{4.11}$$

$$\sum_{\forall(i,j)\in\mathcal{Z}} x_{ij}^k \leq 1, k \in \mathcal{P}_t(ij) \tag{4.12}$$

$$x_{ij}^k \in \{0,1\}, \forall k \in \mathcal{P}_t(ij), \forall(i,j) \in \mathcal{Z} \tag{4.13}$$

where r_{ij}^k and δ_{ij} are given by

$$r_{ij}^k = \begin{cases} b_k^i, & \text{if RBs of vehicle } k \text{ are migrated from} \\ & j^{th} \text{ FN to } i^{th} \text{ FN} \\ b_k^j, & \text{if RBs of vehicle } k \text{ are migrated from} \\ & i^{th} \text{ FN to } j^{th} \text{ FN} \end{cases}$$

and

$$\delta_{ij} = \begin{cases} L_i = A_t(i) + \sum_{\forall k \in \mathcal{P}_t(ij)} (b_k^i) \\ L_j = A_t(i) + \sum_{\forall k \in \mathcal{P}_t(ij)} (b_k^j) \end{cases} \tag{4.14}$$

In Eq. (4.9), the objective is to minimize the overall allocated RBs in the network by considering the vehicles in overlapped coverage regions in order to migrate the allocated RBs between pairs of FNs subjected to the following constraints. The constraint (given in Eq. (4.10)) guarantees that the RBs occupied by the vehicles, when optimal migration is administered, do not surpass the capacity of the destination FN. The constraint (shown in Eq. (4.11)) ensures that the at most $|\mathcal{P}_t(ij)|$ vehicles in a region $(i, j) \in \mathcal{Z}$ can be chosen for migrating services between i^{th} and j^{th} FNs. The constraints in (Eq. (4.12)) and (Eq. (4.13)) ensure a vehicle from all overlapped coverage regions is chosen only once for RB migration. By solving objective function (Eq. (4.9)), we get the optimal set $p_t^*(ij)$, and $p_t^*(ji)$ of vehicles (from Eq. (4.8)) for the optimal service migration between the pairs of FNs i and j.

Theorem 4.2.1. *The ILP optimization problem in Eq.* (4.9) *is NP-hard.*

Proof: We use a well-known NP-hard problem, called seminar assignment problem (SAP), which is a special case of general assignment problem (GAP) [127] and reduce it to our ILP optimization problem in order to prove this theorem. Consider an instance of SAP having a number of n students and m seminar halls. Each seminar hall $r \in m$ has a capacity of $B_r \in \mathbb{N}$. The assignment of student $s \in n$ to a seminar hall r has a profit p_{rs}. The aim is to assign a subset of students to seminar halls, such that the number of students in each seminar hall r should be at most B_r, and the total profit is maximized.

The instance of SAP is reduced to an instance of our ILP problem by mapping (one-to-one) the seminar halls to FNs, the students to the set of vehicles, and r^{th} seminar hall capacity B_r to i^{th} FN capacity (given as Q_i). However, the profit of assigning s^{th} student to r^{th} seminar hall is negated in the mapping of the allocated RBs of i^{th} FN to k^{th} vehicle, $k \in \mathcal{P}_t(ij)$. Here, we assume that there exists a set of vehicles to do RB migration. This reduction can be carried out in polynomial time. Therefore, the instance of SAP has an assignment if and only if the instance of ILP problem has an assignment. This establishes the NP-hardness of the ILP optimization problem.

Theorem 4.2.2. *ILP optimization problem in Eq.* (4.9) *can be optimally solved in polynomial time when $\mathcal{G}$ is fixed.*

Proof: From Theorem 4.2.1, it is clear that the ILP optimization problem is NP-hard. Now consider a graph model, $G(V, E)$, representing the FVNETs with $\mathcal{G}$ FNs (Figure 4.1a), in which set of vertices V represents the FNs and E is the set of edges in G. There exists an edge between i^{th} FN and j^{th} FN if and only if there exists one or more vehicle(s) in the overlapped coverage region and connected to those FNs as shown in Figure 4.1b. As we know, this graph model can be solved in polynomial time when V is fixed [127]. Therefore, the ILP problem can also be solved in polynomial time when $\mathcal{G}$ is fixed.

4.2.2 Algorithm Description

Algorithm 4.1 Dynamic Resource Management

```
Inputs: $\mathcal{G}$, $G(V, E)$
Outputs: An optimal set of vehicles for RBs migration, service capability, serviceability, availability, throughput and Resource utilization efficiency
for $i \leftarrow 1$ to $\mathcal{G}$ do
    $\mathcal{P}_t \leftarrow \emptyset$
    $O_t(i) \leftarrow \sum_{k=0}^{|\mathcal{P}_t(i)|} b_k^i$
    $C \leftarrow O_t(i)$
    for each neighbor $FN_j$ of $i$, $1 \le j \le \mathcal{G}$, $i \neq j$ do
        Find $\mathcal{P}_t(ij)$
        $\mathcal{P}_t \leftarrow \mathcal{P}_t \cup \mathcal{P}_t(ij)$
        $O_t(j) \leftarrow \sum_{k=0}^{|\mathcal{P}_t(j)|} b_k^j$
        $C \leftarrow C \cup O_t(j)$
    end for
    MIGRAGERBS($\mathcal{P}_t$, $i$, $C$)
end for
Find service capability, serviceability, availability, throughput and resource utilization efficiency
```

Algorithm 4.1 presents the proposed algorithm, DRM, for the optimal RB migration in FVNETs. The graph model $G(V, E)$ and $\mathcal{G}$ are given as input to Algorithm 4.1 and generates an optimal set of vehicles $\mathcal{P}_t$ for RBs migration. Upon the arrival of vehicles in the overlapped coverage area (i, j), $\forall\ (i, j) \in \mathcal{Z}$ at time slot t, it finds union of all vehicles $\mathcal{P}_t$, which is present in the overlapped coverage area of each neighbor of i^{th} FN, say, j^{th} FN, $1 \leq i, j \leq \mathcal{G}$, $i \neq j$. Also, it finds the set of the occupied capacity of each neighboring FN, C (Step 5 to Step 10). The vehicles present in the set $\mathcal{P}_t$ are served by both i^{th} and j^{th} FNs. For optimal RB migration, it invokes the MigrateRBs (Procedure 1) with $\mathcal{P}_t$ and C for FN i in Step 11 of Algorithm 4.1. For a given set of vehicles $\mathcal{P}_t$ and the set of occupied capacities C, the MigrateRBs in Procedure 1 finds the vehicle k_{min} with minimum RBs and the corresponding f^{th} FN, such that the load on f^{th} FN is minimum in Step 2. Procedure 1 invokes FindMinVehicle ,given in Procedure 2, to find the vehicle k_{min} and the corresponding f^{th} FN by taking $\mathcal{P}_t$ and C as input. In Procedure 1, the RBs of vehicle k_{min} are migrated from i^{th} FN to f^{th} FN only if the remaining RBs of f^{th} FN satisfy the desired requirements of vehicle k_{min} and the vehicle k_{min} not served by FN f in Steps 3 and 4. Then it updates the occupied capacity and remaining RBs of i^{th} and f^{th} FNs in Step 5. Otherwise, it skips the vehicle k_{min} from the process of RB migration in Step 6. The vehicle k_{min} is removed from the set $\mathcal{P}_t$ in Step 9. This process is repeated from Step 1 to Step 9 until the set $\mathcal{P}_t$ becomes empty.

Procedure 1 MigrateRBs($\mathcal{P}_t$, i, C)

```
      Input: An optimal set of vehicles $\mathcal{P}_t$
      Output: RB migration
 1: while $\mathcal{P}_t \neq \emptyset$ do
 2:     $k_{min}, f \leftarrow$ FindMinVehicle($\mathcal{P}_t$,$C$)
 3:     if $k_{min}$ not served by $f$ and $A_t(f) \geq b^{f}_{k_{min}}$ then
 4:         Migrate the RBs of vehicle $k_{min}$ from FN $i$ to FN $f$
 5:         Update the remaining RBs and occupied capacity of $f^{th}$ and $i^{th}$ FNs
 6:     else
 7:         Skip the vehicle $k_{min}$ from RB migration
 8:     end if
 9:     $\mathcal{P}_t \leftarrow \mathcal{P}_t - \{k_{min}\}$
10: end while
```

4.2.3 An Illustration

Figure 4.1 shows as an example of FVNET for optimal RB migration between FNs using the proposed algorithm, DRM, to reduce the occupied RBs in the group of FNs. This figure

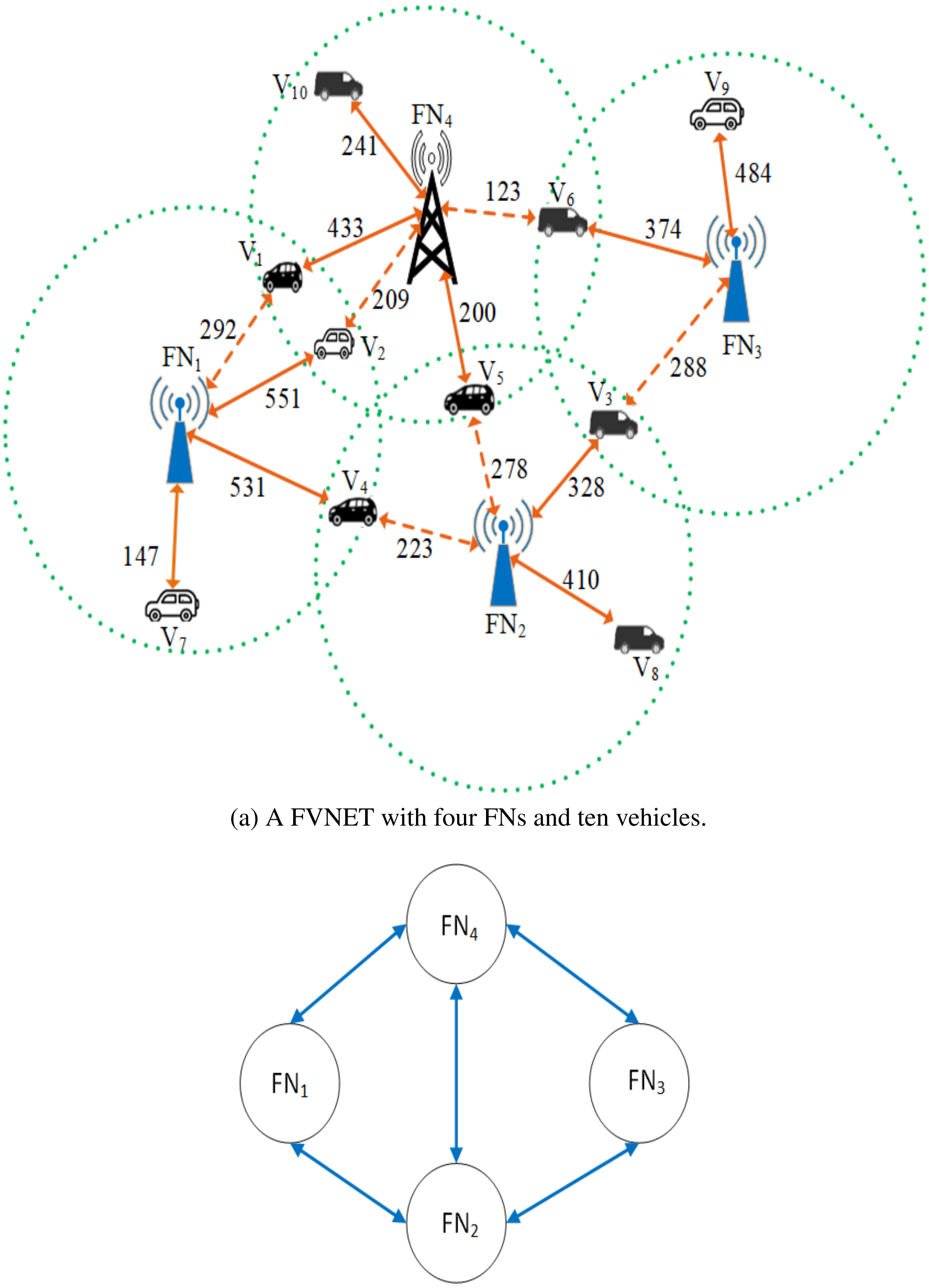
Procedure 2 FindMinVehicle($\mathcal{P}_t$, C)

```
function FINDMINVEHICLE($\mathcal{P}_t$, $C$)
    $min\_rb \leftarrow max\_value$
    $k_{min} \leftarrow -1$
    for vehicle $k \in \mathcal{P}_t$ do
        if $b_k^{f_1} + O_t(f_1) < b_k^{f_2} + O_t(f_2)$ then
            if $b_k^{f_1} + O_t(f_1) \leq min\_rb$ then
                $min\_rb \leftarrow b_k^{f_1} + O_t(f_1)$
                $f \leftarrow f_1$
                $k_{min} \leftarrow k$
            end if
        else
            if $b_k^{f_2} + O_t(f_2) \leq min\_rb$ then
                $min\_rb \leftarrow rb_k^{f_2} + O_t(f_2)$
                $f \leftarrow f_2$
                $k_{min} \leftarrow k$
            end if
        end if
    end for
    return $k_{min}, f$
end function
```

illustrates a scenario of FVNET with four FNs and ten vehicles. The vehicles that FNs currently serve are indicated by solid orange lines, whereas dashed orange lines indicate the vehicles whose services can be migrated to FNs. The number of RBs required from FNs to satisfy desired requirements of vehicles is represented by numbers beside the orange lines. Figure 4.1b is a graph representation of FVNET, corresponding to Figure 4.1a in which vertices represent FNs and an edge between vertices exists if vehicles are located in an overlapped coverage area of two FNs.

Firstly, the set of vehicles is connected to each FN given by $FN_1 = \{V_1, V_2, V_4, V_7\}$, $FN_2 = \{V_3, V_4, V_5, V_8\}$, $FN_3 = \{V_3, V_6, V_9\}$ and $FN_4 = \{V_1, V_2, V_5, V_6, V_{10}\}$. The union of vehicles in overlapped coverage region of two or more FNs is given by $\{V_1, V_2, V_3, V_4, V_5, V_6\}$. The current occupied RBs of each FN for vehicles in the overlapped coverage region are shown in Figure 4.2 (i.e., before RB migration). The set of vehicles connected to each FN is given as input to Algorithm 4.1 for optimal RB migration among FNs.

When FN_1 is chosen (i.e., iteration i = 1 of Algorithm 4.1), the set of all vehicles, which are in the overlapped coverage area of FN_1 and its neighbor FNs, is determined.



(a) A FVNET with four FNs and ten vehicles.

(b) A graph model of Figure 4.1a.

Figure 4.1: An example for RB migration using the proposed algorithm.

They are $\mathcal{P}_t = \{V_1, V_2, V_4\}$. Here, the neighboring FN is FN_j = FN_2 and $\mathcal{P}_t(ij)$ = $\mathcal{P}_t$ = $\{V_4\}$. Similarly, when FN_j = FN_4, $\mathcal{P}_t(ij)$ = $\{V_1$, $V_2\}$ and $\mathcal{P}_t$ = $\{V_1$, V_2, $V_4\}$. The migration algorithm MIGRATRBS defined in Procedure 1 is invoked with the set $\mathcal{P}_t$ and C that consists of a set of vehicles for RB migration and occupied capacities of neighboring

	FN_1	FN_2	FN_3	FN_4
V_1	292			433
V_2	551			209
V_3		328	288	
V_4	531	223		
V_5		278		200
V_6			374	123
$O_t(i)$ =	1082	328	374	633

(a)

	FN_1	FN_2	FN_3	FN_4	
V_2				209	
V_4		223			
V_1	292				
$O_t(i)$ =	292	223	-	209	After iteration i = 1
V_3			288		
$O_t(i)$ =	292	223	288	209	
V_5				200	
$O_t(i)$ =	292	223	288	409	After iteration i = 2
V_6				123	
$O_t(i)$ =	292	223	288	532	After iteration i = 3

(b)

Figure 4.2: An optimal RB migration using DRM scheme for FVNET. (a) Occupied RBs of FNs before migration. (b) Occupied RBs of FNs after migration using DRM.

FNs, respectively. The vehicle $k_{min} = V_2 \in \mathcal{P}_t$ and corresponding FN_j = FN_4 are chosen for RB migration from FN_1, since vehicle V_2 is required minimum RBs (i.e., $rb_j^{k_{min}}$ = 209) from FN_4 using FINDMINVEHICLE. After successful migration, the occupied RBs of FN_4 are updated to 209, and vehicle V_2 is removed from $\mathcal{P}_t$. Subsequently, vehicles V_4 and V_1 are chosen from $\mathcal{P}_t$ for RB migration. Note that these vehicles are selected based on the minimum number of RBs. After the completion of iteration i = 1, the occupied RBs of each FN are shown in Figure 4.2. In the next iteration (i.e., iteration i = 2 of Algorithm 4.1), the union of all vehicles in overlapped coverage region of FN_2 and its neighbor FNs is $\mathcal{P}_t = \{V_3, V_5\}$.

After the iteration i = 2, the occupied RBs of FNs FN_1, FN_2, FN_3, and FN_4 are 292, 223, 288, and 409, respectively. In the iteration i = 3, the FN_3 is chosen and $\mathcal{P}_t = \{V_6\}$ is obtained. For migrating RBs of vehicle V_6, the load on FN_3 and FN_4 is 662 (i.e., 288 + 374) and 532 (i.e., 409 + 123), respectively. Since FN_4 contains minimum load, the RBs are migrated from FN_3 to FN_4. The overall occupied RBs are 2417 before RB migration and RBs are reduced to 1335 after the migration. Therefore, the percentage of reduction is 44.76% (i.e., $\frac{(2417-1335)}{2417}$). Note that this reduction greatly improves the service capability, serviceability, availability, throughput, and resource utilization efficiency of the network. On the other hand, the percentage of reduction is 21.64% in the DRO [24], 43.67% in the SA [25] and 16.75% in the RO. This clearly shows the superior performance of DRM over the existing algorithms.

Theorem 4.2.3. *The number of vehicles in the overlapped coverage region of two or more FNs for RB migration from i^{th} FN to each neighboring FN_j (1 $\leq j \leq \mathcal{G}$, $i \neq j$) is at most $|\mathcal{P}_t|$.*

Proof: The j^{th} FN allocates the required RBs, b_k^j, when migrating service of a vehicle k from i^{th} FN *if and only if* the j^{th} FN has sufficient available RBs (i.e., $A_t(i) \geq b_k^j$). Therefore, the total number of vehicles in the overlapped coverage region for RB migration from i^{th} FN to all the neighboring FNs is at most $|\mathcal{P}_t|$. Note that the set $\mathcal{P}_t$ is the union of the set of vehicles in the overlapped coverage region of each j^{th} FN ($1 \leq j \leq \mathcal{G}$, $i \neq j$), which is neighbor to i^{th} FN (i.e., $\mathcal{P}_t = \mathcal{P}_t \cup \mathcal{P}_t(ij)$).

Theorem 4.2.4. *The proposed algorithm DRM migrates the RBs of vehicle $k_{min} \in \mathcal{P}_t$ with minimum load $b_{k_{min}}^f$ to f^{th} FN, such that the load on f^{th} FN is minimum.*

Proof: Consider the FVNET example with $\mathcal{G}$ = 4 as shown in Figure 4.1a. When FINDMINVEHICLE( ) is invoked in Procedure 1 with $\mathcal{P}_t$ = $\{V_3$, $V_5\}$ for FN_2 (i.e. in iteration i =2), suppose vehicle V_3 is selected. Then the load on FN_2 and FN_3 is 551 (i.e., 223 + 328) and 228 (i.e., 0 + 228), respectively. When the vehicle V_5 is selected, the load on FN_2 and FN_4 is 501 (i.e., 223 + 278) and 409 (i.e., 209 + 200), respectively. The proposed algorithm DRM using FINDMINVEHICLE( ) selects vehicle V_3 and the corresponding FN_3 instead of vehicle V_5 with RBs 200. Since the load on FN_3 is minimum, the RBs of vehicle V_3 are migrated from FN_2 to FN_3. Therefore, it is proved that the proposed algorithm migrates the vehicle's RBs to the corresponding FN, such that the load on FN is minimum.

4.2.4 Complexity Analysis

The proposed algorithm DRM is presented in Algorithm 4.1, which invokes MIGRATERBS( ) (Procedure 1) with the occupied capacities and a set of vehicles in the overlapped coverage area (i, j), $\forall(i, j) \in \mathcal{Z}$ of pairs of FNs as input. In Procedure 1, Step 2 (i.e. Procedure 2) takes $\mathcal{O}(|\mathcal{P}_t|)$ time in worst case as it iterates for $\mathcal{O}(|\mathcal{P}_t|)$ times. Steps 3-9 take constant time. The while loop iterates for $\mathcal{O}(|\mathcal{P}_t|)$ times. Therefore, the overall running time of Procedure 1 is $\mathcal{O}(|\mathcal{P}_t|^2)$ in the worst case.

In the Algorithm 4.1, for a given FN_i, Step 5 finds a set of vehicles in the overlapped coverage regions of each neighbor, say FN_j, of i^{th} FN, $1 \leq i, j \leq \mathcal{G}$, $i \neq j$. It takes $\mathcal{O}(\mathcal{G})$ time in the worst case. The outer for loop in Algorithm 4.1 takes $\mathcal{O}(\mathcal{G})$ time. Therefore, the worst case time complexity of Algorithm 4.1 is given as $\mathcal{O}(\mathcal{G}(\mathcal{G} + |\mathcal{P}_t|^2)) \approx \mathcal{O}(\mathcal{G}^2 + \mathcal{G}(|\mathcal{P}_t|^2))$.

4.3 Performance Evaluation

The performance of the proposed algorithm is evaluated in terms of service capability, serviceability, availability, throughput, and resource utilization efficiency. The simulated results were compared with existing service migration algorithms, such as DRO [24], SA [25] and RO, in which FNs and vehicles are selected randomly.

4.3.1 Simulation Setup

The simulations were carried out by creating a virtual environment using Python (version 3.8) on PyCharm IDE 2020.1.3. This IDE was running on an Intel(R) Xeon(R) Gold 622R CPU @ 2.90GHz 2.89 GHz processor, 64-bit operating system and 64.0 GB installed RAM. The simulation environment, including network traffic details, was set up based on the setup given in [24, 25]. We evaluate the execution of the proposed algorithm DRM in a network

Figure 4.3: A road map of [5000 × 5000] square meters.

model with [10 $\sim$ 50] FNs deployed in a region size of [5000 $\times$ 5000] square meters as shown in Figure 4.3. This road map shows an area of a city in which the deployment of FNs ranges from 10 to 50. These FNs are deployed in such a way that their coverage area overlaps with one or more FNs. We consider the mobility patterns of vehicles that arrived and/or departed to/from the network. They follow the Poisson distribution in each time slot. However, the mean arrival rate (λ) and mean departure rates (μ) are 10 vehicles/s and 5 vehicles/s, respectively. It is noteworthy to mention that the velocity of vehicles is not explicitly shown as it is modelled in the form of λ and μ. When FNs cannot allocate required RBs, vehicles are served with minimum data rates ranging from 0.5 to 2 Mbps. As the mean arrival rate exceeds the mean departure rate, the number of vehicles available at the network increases in each time slot. The simulation results show the impact of the increase in the number of vehicles on the service capability, serviceability, availability and throughput of the network. The average results were obtained by conducting the Monte-Carlo simulations up to one hundred five times (i.e., 15 times for each result) with 210 time slots using queuing model. We use IEEE 802.11p as our MAC protocol to enable wireless access in FVNETs. The Nakagami model is used to model the data/signal propagation. This model is considered as the most realistic model. The different parameters considered for the simulations with their values are given in Table 4.3.

Table 4.3: Parameters and their values for simulations

Parameter		**Value**
$\mathcal{G}$		[10 $\sim$ 50]
Vehicle density	Mean arrival rates (λ)	10 vehicles/s
	Mean departure rates (μ)	5 vehicles/s
Number of vehicles with respect to one simulation		2100
Network area		5000 m × 5000 m
Radio communication range in single-hop communications		500 m
Bandwidth of FNs		{10, 15, 20} MHz
Coverage radius of a FN		500 m
Cumulative number of service connections		2100
Required data rate		[0.5 $\sim$ 2] Mbps
Time slot duration		1 s
Confidence interval or simulation duration		210 s

4.3.2 Service Capability

Service capability is the ratio between the remaining resources and the total number of resources in a network [24]. Figure 4.4 shows the network service capability by the proposed algorithm DRM algorithm and other existing migration algorithms, such as DRO, SA, and RO. It is observed that the FNs possess sufficient RBs to satisfy requests of the vehicles during initial time slots. Hence, all the algorithms produce better service capability in the initial duration. However, as the deluge arrival of vehicles to the network increases, the network service capability reduces due to the reduction in the available RBs of FNs. The DRO algorithm uses the solution of maximum weight matching, which is the set of edges without common vertices and reduces the RBs of those vehicles in the set of edges returned by the matching solution. Therefore, there is a severe reduction in the service capability using the DRO algorithm.

The SA and RO algorithms enhance the service capability of the network as the number of vehicles arriving at the network increases. The SA algorithm improves the service capability using the graph colouring solution up to 97.80% and 24.03% when compared to DRO and RO, respectively. However, the proposed algorithm outperforms the service capability when compared to SA, RO and DRO + SA combined. The proposed algorithm

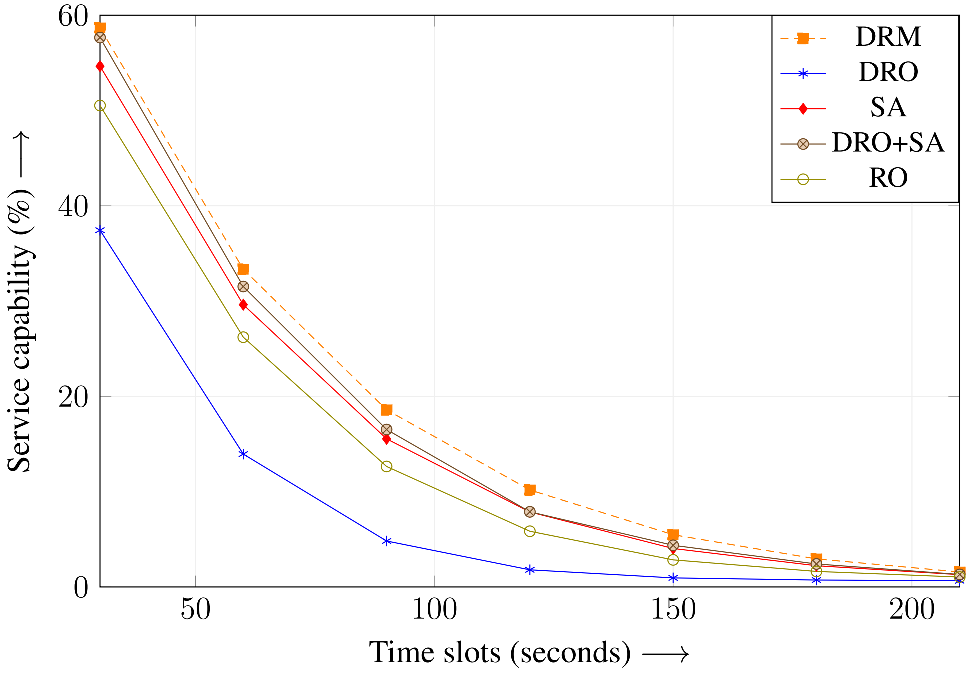


Figure 4.4: Pictorial comparison of network service capability for DRM, DRO, SA, DRO + SA and RO algorithms.

enhances the service capability by 21.78%, 51.05% and 19.65% when compared to SA, RO and DRO + SA, respectively. The rationality behind this is that it is greedy to select a vehicle with minimum RBs.

4.3.3 Serviceability

The network serviceability is the percentage of vehicles getting served with desired requirements in a network [39]. The serviceability of the network by the proposed algorithm and other migration algorithms is shown in Figure 4.5. The simulation results show that all the algorithms behave similarly during the initial time slot, as FNs contain enough RBs. However, the serviceability of the network reduces as the deluge arrival of vehicles at the network causes the FNs to be impotent in providing services to arrived vehicles. Therefore, there is a reduction in the serviceability of the network. The SA and the DRO augmented by SA (i.e., DRO + SA) behave similarly as vehicles arriving at the network increase. The SA and RO algorithm improves the serviceability of the network. The SA algorithm improves the serviceability by 52.42% and 13. 03% when compared to DRO and RO algorithms, re-

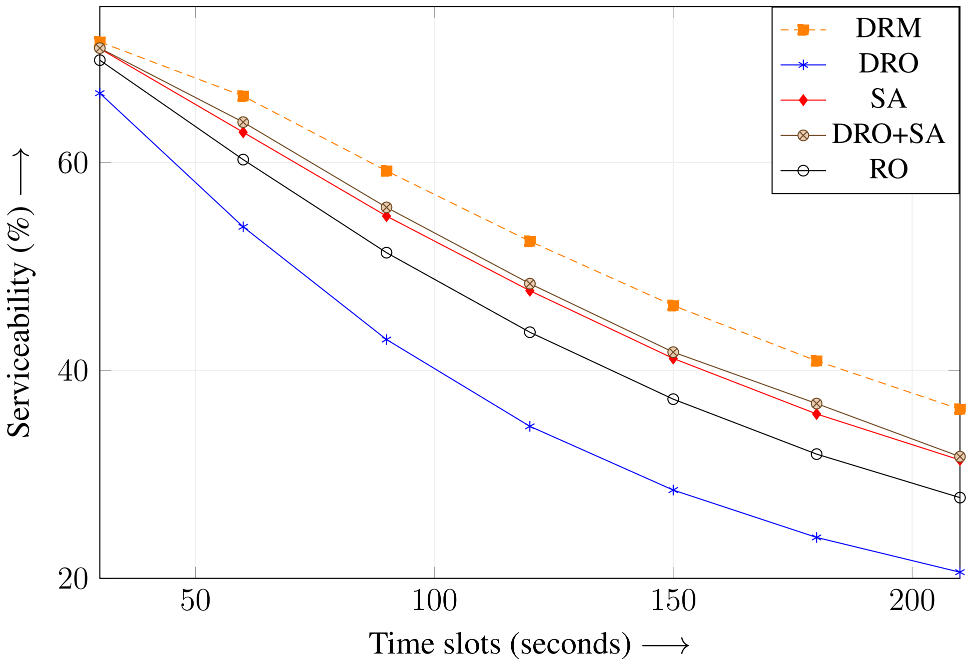


Figure 4.5: Pictorial comparison of network serviceability for DRM, DRO, SA, DRO + SA and RO algorithms.

spectively. However, the proposed algorithm is greedy in selecting vehicles with minimum RBs for migrating RBs between pairs of FNs. Furthermore, it maximizes the RBs reduction of vehicles in the overlapped coverage regions to accommodate vehicles with their desired data rates. Therefore, the proposed algorithm enhances the serviceability by 15.55%, as the vehicles connecting to the network increase when compared to the SA algorithm.

4.3.4 Availability

Availability is the percentage of vehicles getting served with minimum requirements in a network [39]. Figure 4.6 presents the availability of the network satisfying incoming vehicles with a minimum data rate of 0.5 Mbps. As incoming vehicles connecting to the network increase, the capacity of FNs tends to exhaust in servicing vehicles. As a result, the network availability reduces gradually. The DRO algorithm enhances the availability better than the other algorithm. However, as vehicles arriving at the network increase at time slot 150, the availability reduces. This reduction in availability is due to a few RB migrations among pairs of FNs for the vehicles in overlapped coverage regions. However,

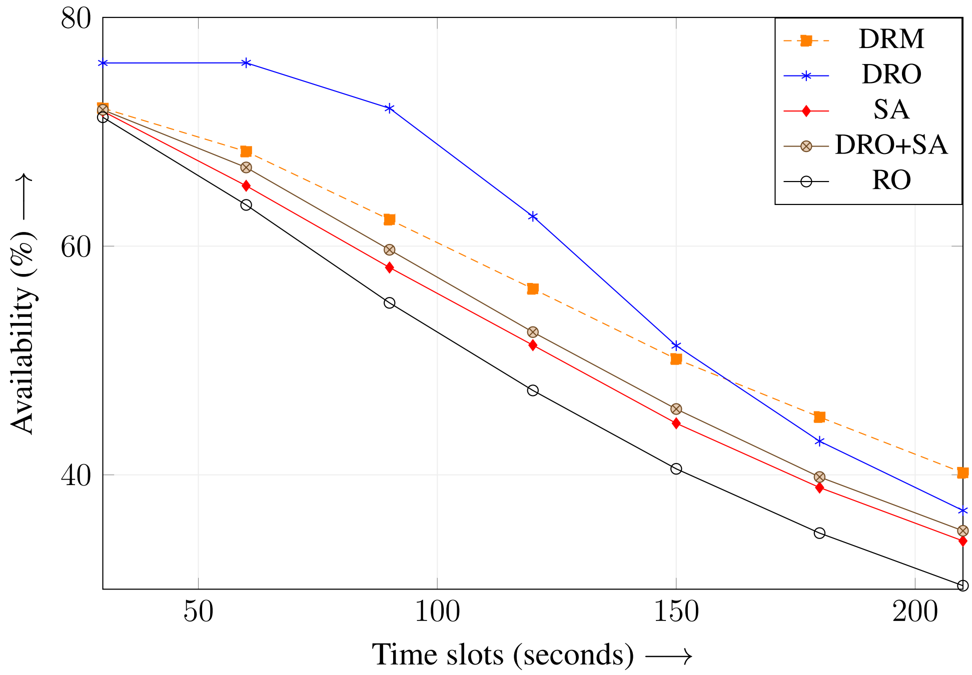


Figure 4.6: Pictorial comparison of network availability for DRM, DRO, SA, DRO + SA and RO algorithms.

the proposed algorithm reduces the allocated RBs by migrating RBs among pairs of FNs. Figure 4.7 shows the percentage of RB reduction using the proposed algorithm and existing algorithms. The proposed algorithm maximizes the RBs reduction of vehicles up to 2.0% on average. Therefore, it is able to accommodate the vehicles with minimum data rates when not allocated with required RBs. Moreover, the proposed algorithm improves the availability of the network as vehicles connecting to the network increase by 08.94%, 17.41%, and 32.57% when compared to DRO, SA and RO, respectively.

4.3.5 Throughput

Network throughput is obtained using the vehicle's achievable rates at different time slots and from different FNs. Using the Shannon formula, the achievable rate of vehicle k from FN $i \in \mathcal{G}$, at time slot t, is $\mathcal{T}_t(ik) = B\log_2(1+\psi(ik))$, where B is the available bandwidth at i^{th} FN. If $\alpha_t(ik)$ is an allocated time fraction to vehicle k, then the throughput of vehicle k from FN i is $\tau_t(ik) = \alpha_t(ik) \times \mathcal{T}_t(ik)$. The network throughput is obtained from $\sum_{\forall k \in \|\mathcal{P}_t(i)\|} \log\left(\sum_t \sum_{i \in \mathcal{G}} \tau_t(ik)\right)$ [25]. The network's throughput is directly proportional

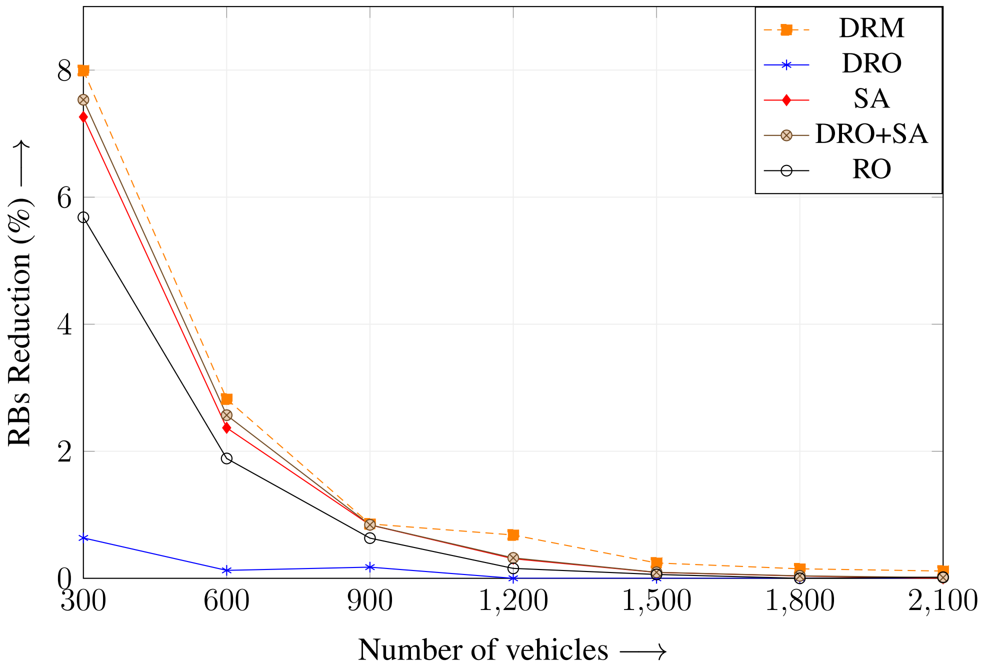


Figure 4.7: Pictorial comparison of RBs reduction for DRM, DRO, SA, DRO + SA and RO algorithms.

to the time fraction allocated to the vehicles getting services. As a result, the throughput increases as the number of vehicles getting services increases. The network's throughput for the proposed and existing algorithms is shown in Figure 4.8, in which the throughput increases gradually as the vehicles getting services from the network increase. Furthermore, the proposed algorithm can accommodate newly arrived vehicles at FNs by reducing allocated RBs of vehicles that have already arrived without affecting their services. As a result, the proposed algorithm improves the network throughput up to 57.31%, 20.74%, and 39.13%, compared to DRO, SA, and RO algorithms, respectively.

4.3.6 Resource Utilization Efficiency

Resource utilization efficiency is the percentage of occupied RBs in a network. Figure 4.9 shows the distribution of the average percentage of RBs used in the network obtained from the proposed and existing algorithms. The height of the box represents the distribution of the percentage of occupied RBs of FNs. The blue diamond symbol denotes the average percentage of RBs utilization of the entire network, and the red line inside the box denotes

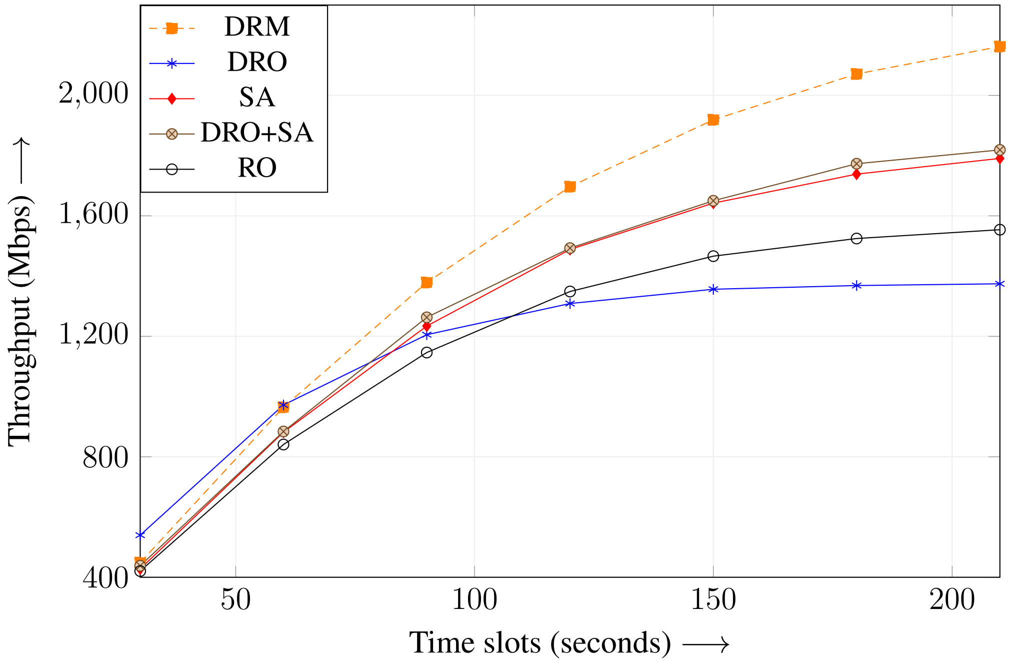


Figure 4.8: Pictorial comparison of throughput for DRM, DRO, SA, DRO + SA and RO algorithms.

the median. When the arrival of vehicles at the network increases, it leads the FNs to be overloaded with their capacity. As a result, FNs become impotent to provide services to the new arriving vehicles. Thus, the existing algorithms, DRO, SA, and RO, provide 90.05%, 81.00%, and 83.38% of average RBs utilization, respectively. The DRO augmented by SA provide 82.12% of RBs utilization. On the contrary, the proposed algorithm minimizes the allocated RBs of vehicles that have arrived early in order to allocate RBs to newly arrived vehicles. Thus, the proposed algorithm provides 78.46% of average resource utilization by reducing allocated RBs of vehicles in overlapped coverage regions. Furthermore, the allocated RBs of vehicles are reduced by migrating allocated RBs between pairs of FNs. The percentage of RBs reduction as vehicles connecting to the network increases for the proposed and existing algorithms is shown in Figure 4.7. The reduced RBs of vehicles from FNs are reused to provide services to the vehicles that arrive. The simulation results show that the proposed algorithm improves RB utilization efficiency by reducing allocated RBs by 12.87%, 03.13%, and 05.90% as compared to DRO, SA, and RO algorithms, respectively. Therefore, the proposed algorithm improves resource utilization efficiency by allocating released RBs to newly arrived vehicles.

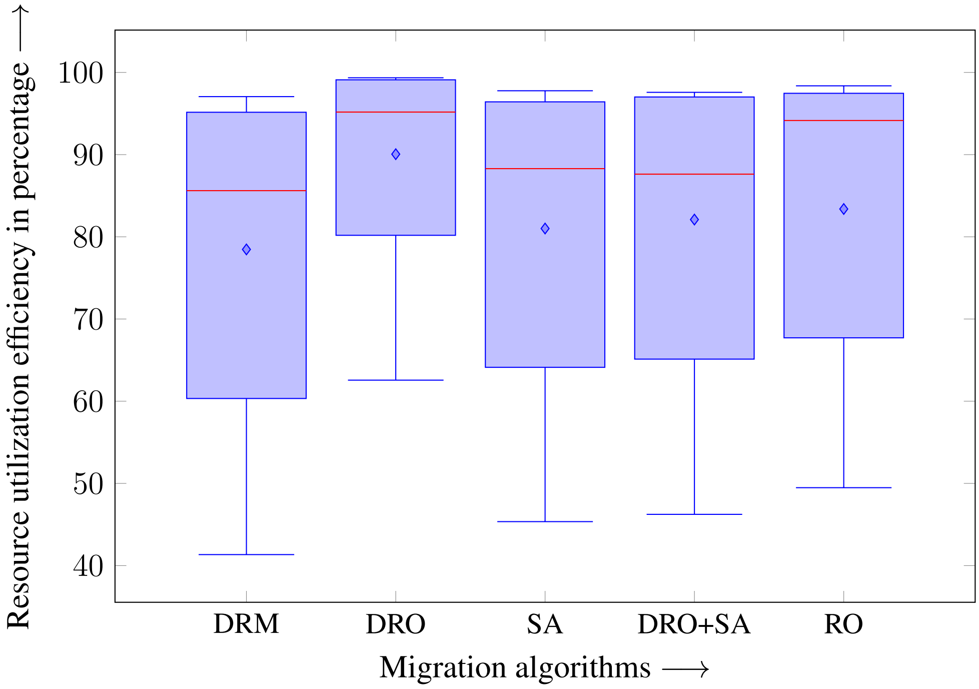


Figure 4.9: Pictorial comparison of network resource utilization efficiency for DRM, DRO, SA, DRO + SA and RO algorithms.

4.4 Summary

In this chapter, we propose a DRM algorithm to manage RBs allocation in FVNETs by considering vehicles in overlapped coverage regions of two or more FNs and migrating RBs of a set of vehicles among pairs of FNs. The objective of the proposed algorithm is to improve the network service capability, serviceability, availability, throughput, and resource utilization efficiency by minimizing allocated RBs. The proposed algorithm maximizes the service capability of the network by minimizing the occupied RBs of vehicles that have already arrived. This reduction in allocated RBs is achieved by migrating allocated RBs of a set of vehicles among FNs, and it is addressed by formulating ILP. The simulation outcomes show that the proposed algorithm reduces occupied RBs among FNs by migrating RBs of the set of vehicles and achieves better service capability, serviceability, availability, throughput and resource utilization efficiency when compared to other migration algorithms, such as DRO, SA, DRO + SA and RO.

Chapter 5

An Efficient Resource Orchestration Algorithm for Enhancing Throughput in FVNETs

This chapter presents an efficient resource orchestration (ERO) algorithm for coordinating RB allocation and offloading upstream services among FNs to maximize the network throughput. ERO algorithm partitions the FNs' coverage region into restricted and non-restricted coverage areas. The restricted coverage area is a coverage region that does not overlap with neighbouring FNs' coverage. Similarly, FN's coverage region overlaps with neighbouring FN's coverage regions, which are called non-restricted coverage areas. The maximizing throughput problem is formulated to reduce assigned RBs of vehicles in the non-restricted coverage areas. Hence, the assigned RBs of vehicles in the non-restricted coverage areas are migrated between pairs of FNs to reduce allotted RBs. Further, a minimum priority queue is constructed based on the occupied capacities of FNs to perform optimal RB migration. The existing algorithms [24,25,39,68,69] from the above literature have been designed to maximize the performance measures, such as throughput, serviceability, availability, and service capability, while satisfying the vehicles' desired requirements in the FVNETs. Table 5.1 shows the difference between the existing and proposed algorithms in terms of fog environment and vehicles in restricted and non-restricted coverage regions. However, the proposed algorithm differs from the state-of-the-art in the following forms.

1. Unlike [24, 25, 39, 68, 69], the proposed algorithm constructs the minimum priority queue using occupied capacities of FNs.

2. Unlike [24, 68, 69], the proposed algorithm for migrating RBs amidst pairs of FNs is greedy in choosing the vehicle with minimum RBs and greedy in choosing FNs having the minimum occupied capacity.

The proposed algorithm, ERO, is simulated by considering the vehicle arrival rate as 5 to 10 vehicles/s and 150 to 3000 vehicles with 10 to 50 FNs in FVNETs. This chapter considers the influence of the rise in the number of vehicles on the network's throughput, serviceability, availability and service capability for three scenarios: the mean arrival rate is greater than the mean departure rate, the mean arrival rate is equal to the mean departure rate and the mean arrival rate is less than the mean departure rate. The simulation outcomes are compared with DRO [24], ARB [39], SA [25], and RO in terms of throughput, serviceability, availability and service capability of the network. The simulation results show that the ERO performs better than existing algorithms regarding throughput, serviceability, availability, and service capability. The novel contributions of this chapter are listed in the following points.

1. The throughput is maximized by migrating allotted RBs of vehicles in non-restricted coverage regions such that the allotted RBs of these vehicles are minimized among pairs of FNs.

2. We formulated the RBs migration problem in FVNETs to an ILP by scrutinizing the variables influencing the network throughput and FNs resource constraints.

3. We propose an ERO algorithm, a polynomial time algorithm, which constructs the minimum priority queue for optimal RBs migration between pairs of FNs to augment the network's throughput.

4. The ERO algorithm synchronizes the RBs allocation for offloading upstream services such that throughput is maximized by partitioning the coverage of FNs into restricted and non-restricted coverage regions.

Table 5.1: The difference between the existing and proposed algorithms

Work	**Algorithm**	**Approach**	**Analysis and/or scope of improvement**	**Fog environment**	**Vehicles in restricted coverage regions**	**Vehicles in non restricted coverage regions**
Liu et al. (2016) [25]	SA	Graph coloring for resource management	They have not scrutinized the network serviceability, availability, and service capability.	×	√	×
Vu et al. (2020) [24]	DRO	Maximum matching problem for RBs reduction between pairs of FNs	There is a decline in throughput, serviceability, availability, service capability and resource usage efficiency when vehicles entering the network increase.	√	×	√
Dao et al. (2017) [39]	ARB	Backpressure algorithm and Hungarian method for solving imbalance among nodes	They have not examined the resource utilization efficiency and service capability.	√	√	×
Ge et al. (2018) [68]	UA-US	Utility maximization problem is used for load balancing	They have not scrutinized the network serviceability and service capability.	√	√	×
He et al. (2016) [69]	SMDP-RA	SMDP based resource allocation	There is a shrinkage in the network serviceability.	×	√	×
Proposed algorithm	ERO	Greedy in choosing a vehicle having the least RBs and builds binary heap in selecting FNs with minimum occupied capacity	Magnifies the reduction of RBs amid pairs of FNs such that the throughput, serviceability, availability, and service capability are augmented as vehicles enter the network increase spike in demand.	√	√	√

5. We present the simulation results showing that the ERO algorithm outperforms the existing algorithms regarding throughput, serviceability, availability, and service capability. The results are obtained by considering the influence of the rise in the arrival of vehicles to the network.

The remainder of this chapter is structured as follows. The system model and problem statement are illustrated in Section 5.1. The ILP formulation of the problem is presented in the same section. The description of the proposed ERO algorithm with an illustration is presented in Section 5.2. Its complexity analysis is also given in Section 5.2. Section 5.3 shows the extensive simulation outcomes of the proposed ERO algorithm and its performance with the existing algorithms. Finally, the summary of this chapter is in Section 5.4.

5.1 System Model and Problem Statement

This section presents the system model for downlink communication and problem statement for maximizing network throughput.

5.1.1 System Model

Consider a city area A in which FVNET is fixed with $\mathcal{G}$ number of FNs. Let R be the communication range of each FN i, $1 \leq i \leq \mathcal{G}$, and it overlaps with the communication range of neighbouring FN j, $1 \leq j \leq \mathcal{G}$, $i \neq j$. Let $\mathcal{P}_t(i)$ be a set of vehicles within the communication range of the FN i for getting services at time slot t. The mean arrival rate and mean departure rate of vehicles to the network follow a Poisson distribution with the mean values of λ and μ, respectively. The values for the Poisson distribution are computed as the expected number of arrived vehicles ($\mathbb{E}$) by taking the timeslot duration (δs) on the mean arrival rate λ and are computed as $\lambda.\delta s$ as per the previous research [24]. The values are extracted randomly using mean and variance within the time interval (T). The interarrival time of vehicles follows an exponential distribution. Further, a free flow discrete time traffic model is considered for uninterrupted and homogenous vehicular traffic over a fixed length of overlapping coverage regions [38, 50]. It is worth mentioning that the velocity of

Table 5.2: Description of notations

Notation	**Description**
$\mathcal{Z}$	A set of non-restricted coverage areas in FVNETs
λ, μ	Mean arrival and departure rate of vehicles, respectively
$\mathcal{P}_t^r(i)$	A set of vehicles in restricted region of i^{th} FN at time slot t
$\mathcal{P}_t^{nr}(ij)$	A set of vehicles in non-restricted region of i^{th} FN and j^{th} FN at time slot t
$\mathcal{T}_t(ik)$	Achievable rate of k^{th} vehicle from i^{th} FN at time slot t
A	A city area in which FVNET is fixed
B	Bandwidth of a FN
D_{ki}	Euclidean distance between i^{th} FN and k^{th} vehicle
$\rho_t(ik)$	Transmission power of i^{th} FN when connecting with k^{th} vehicle at time slot t
N_k, I_k	Gaussian noise and co-channel inference of k^{th} vehicle, respectively
$G(V, E)$	A graph of FVNET with a set of vertices V and a set of edges E
$\alpha_t(ik)$	An allocated time fraction of k^{th} vehicle from i^{th} FN at time slot t
$\tau_t(ik)$	k^{th} vehicle throughput from i^{th} FN at time slot t
$\psi(ik)$	SINR of k^{th} vehicle from i^{th} FN
β	Pathloss exponent
γ	Scaling co-efficient
T	Serviceable time of the network
t_q	Time slot t_q, $1 \leq q \leq T$
δ_s	Time slot duration

vehicles is not explicitly considered. However, vehicles are generated by varying data rates of vehicles and distance from FN.

Table 5.2 summarises the various notations and their definitions used in our work. In each time slot, the FNs provide services to the vehicles in their coverage region based on the SINR of those vehicles. However, the signal strength depends on the distance between vehicle k and FN i, which is obtained by considering the vehicle k located at (x_k, y_k) and FN i at the origin in 2D space using Euclidean distance, as $D_{ki} = \sqrt{(x_k^2 + y_k^2)}$ [77].

The FN uses the transmit power control in downlink communication to achieve a constant bit rate in each time slot, irrespective of the vehicle's position in the coverage region of the FN [85]. The approximate transmission power of FNs can be obtained using the distance-dependent exponential radio model [128, 129]. In this model, as the distance from FN increases, the transmission power also increases [130]. Therefore, assuming a constant bitrate (B bits per each slot), the transmission power of FN i when communicating with vehicle k at time slot t is indicated by $\rho_t(ik)$ and is obtained by

$$\rho_t(ik) = D_{ki}^{\beta} \times \frac{B}{\gamma} \tag{5.1}$$

where γ is a scaling co-efficient and β is a path loss exponent [130]. Thus, the SINR of vehicle k from FN i is obtained as follows.

$$\psi(ik) = \frac{\rho_t(ik)}{N_k + I_k} \tag{5.2}$$

where N_k is the Gaussian noise power of vehicle k and I_k is the co-channel inference power of vehicle k [95].

Suppose the vehicle k has requested services with data rate r^k in the coverage area of FN i. Then total RBs, b_k^i, allotted from FN i to the vehicle k, can be obtained from Eq. (4.1).

5.1.2 Problem Statement

Consider the vehicles reaching to and leaving from FN i is represented by the sets $\mathcal{P}_t^{in}(i)$ and $\mathcal{P}_t^{out}(i)$, respectively. Then, the mean departure rate of vehicles, μ, in the network is obtained using expectation $\mathbb{E}$ as discussed in Eq. (4.2).

As stated Chapter 1, the total duration of a FN, T, is partitioned into equal time slots, t_q, $1 \leq q \leq T$, such that each time slot has a duration of δs as shown in Figure 1.2. The FNs provide services to the vehicles reaching them at each time slot t_q. Let the set of vehicles getting services from FN i at time slot t be $\mathcal{P}_t(i)$ and it can be defined in Eq. (4.3).

At time slot t, $|\mathcal{P}_t(i)|$ gives the number of vehicles getting services from FN i. Due to the finite resource restraints of FNs, the network provides services to a limited number of vehicles. Thus, the FNs become impuissant in providing services to the vehicles when the vehicles arriving at the network increase at a particular instant of time t. In this situation, the available RBs of i^{th} FN, service capability of i^{th} FN and service capability of the network are obtained using Eq. (4.4), Eq. (4.5) and Eq. (4.6), respectively.

Similarly, the percentage of vehicles getting services within the desired and minimum data rate (e.g., throughput and delay) defines the serviceability and availability of the network, respectively [39, 123].

We use the Shannon formula to find the vehicles' achievable rate at each time slot from each FN. The achievable rate of a vehicle $k \in \mathcal{P}_t(i)$ from FN i, at time slot t, is given as [25]

$$\mathcal{T}_t(ik) = B \times \log_2(1 + \psi(ik)) \tag{5.3}$$

where B is the bandwidth available at FN i. If $\alpha_t(ik)$ is a time fraction allotted to vehicle k, then, the k^{th} vehicle throughput from FN i is given in [25] as

$$\tau_t(ik) = \alpha_t(ik) \times \mathcal{T}_t(ik) \tag{5.4}$$

Then, the resource allocation problem $\mathbb{P}$ is represented as

$$\mathbb{P} : \max_{\vec{\tau}, \vec{t}} \sum_{k=1}^{|\mathcal{P}_t(i)|} \log \left(\sum_{t}^{T} \sum_{i \in \mathcal{G}} \tau_t(ik) \right), 1 \leq i \leq \mathcal{G} \tag{5.5}$$

s.t

$$\sum_{k=1}^{|\mathcal{P}_t(i)|} \frac{\tau_t(ik)}{\mathcal{T}_t(ik)} \leq t_q, 1 \leq i \leq \mathcal{G}, 1 \leq q \leq T \tag{5.6}$$

$$\tau_t(ik) \geq 0 \tag{5.7}$$

$$t_q \geq 0 \tag{5.8}$$

where notations $\vec{\tau}$ and $\vec{t}$ are the collections of throughput variables $\tau_t(ik)$ and time slot variables t_q, respectively. Constraint shown in Eq. (5.6) corresponds to the available time fraction for the i^{th} FN, $1 \leq i \leq \mathcal{G}$, in time slot t_q, $1 \leq q \leq T$. Moreover, minimizing the standard deviation of FNs using occupied capacities (i.e. $\min \sigma(O_t)$) to measure load balance among FNs and minimizing the FNs' energy consumption in transmitting services to vehicles can be considered other objective functions.

In practical systems, only one FN provides services to a vehicle for downlink communication. The vehicle in the restricted region of a FN gets services from a single FN, and the vehicle in the non-restricted region of a FN gets services from either one of the FNs of that non-restricted region. However, when the number of vehicles reaching the network rises over time, the FNs become impuissant in providing services due to the finite resource restraints of FNs. As a result, the FNs allocate inadequate resources to the arrived vehicles. This results in the degradation of vehicles' throughput and affects the QoS. Hence, the allocated RBs of vehicles in the non-restricted regions of FNs are migrated between pairs of FNs to maximize throughput and provide better QoS. Additionally, various performance metrics, including serviceability, availability, and service capability, are examined with the growing number of vehicles connecting to the network.

5.2 Efficient Resource Orchestration Algorithm

In this work, the proposed algorithm partitioned the coverage region of a FN as a non-restricted coverage region and a restricted coverage region. The non-restricted coverage region of a FN is a coverage region which is overlapped with the neighbouring FN coverage regions. The restricted coverage region of a FN is a non-overlapped coverage region of that FN. The restricted and non-restricted coverage regions of FNs in FVNETs are shown in Figure 5.1a. The ERO algorithm maximizes the objective given in Eq. (5.5) by minimizing the allotted RBs of vehicles in the non-restricted coverage areas of FNs. This minimization of RBs is carried out by migrating allocated RBs amid pairs of FNs. Therefore, the ERO algorithm constructs a minimum priority queue using the binary tree data structure for the occupied capacities of FNs. The migration of RBs of vehicles in the non-restricted regions is carried out by fetching a FN with minimum occupied capacity from the priority queue. The proposed algorithm is presented in Algorithm 5.1 and described in Section 5.2.2.

The set of non-restricted coverage regions in FVNETs is denoted by $\mathcal{Z}$. The ERO algorithm identifies the vehicles in the non-restricted and restricted coverage parts of FNs. The set of vehicles in the non-restricted part of FN i and FN j at time slot t is denoted by $\mathcal{P}_t^{nr}(ij)$, $1 \leq i, j \leq \mathcal{G}$, $i \neq j$ and the set of vehicles in the restricted coverage part of FN i at time slot t is denoted by $\mathcal{P}_t^{r}(i)$. The sets $\mathcal{P}_t^{nr}(ij)$ and $\mathcal{P}_t^{r}(i)$ are defined as follows.

$$\mathcal{P}_t^{nr}(ij) = \mathcal{P}_t(i) \cap \mathcal{P}_t(j) \tag{5.9}$$

$$\mathcal{P}_t^{r}(i) = \mathcal{P}_t(i) \setminus \mathcal{P}_t^{nr}(ij) \tag{5.10}$$

In this chapter, we assume that there exists at least a single vehicle in the non-restricted coverage region $(i, j) \in \mathcal{Z}$ of i^{th} and j^{th} FNs to migrate the allotted RBs of vehicles in the set $\mathcal{P}_t^{nr}(ij)$ (i.e., $\mathcal{P}_t^{nr}(ij) \neq \emptyset$). Subsequently, consider the vehicles in the set $\mathcal{P}_t^{nr}(ij)$ whose RBs migrate from FN i to FN j, and its converse defined as $p_t^*(ij) \in \mathcal{P}_t^{nr}(ij)$ and $p_t^*(ji) \in \mathcal{P}_t^{nr}(ij)$, respectively. They can be accomplished using Eq. (4.8).

The RBs migration problem for the vehicles in the non-restricted coverage regions in the FVNETs is mapped as an ILP problem, which is explained in Section 5.2.1.

5.2.1 ILP Formulation of Problem

The aggregated throughput is maximized by considering the vehicles in the non-restricted regions, $\mathcal{P}_t^{nr}(ij)$, $\forall(i, j) \in \mathcal{Z}$ for reducing allocating RBs among pairs of FNs. This RBs reduction is carried out by migrating RBs of vehicles in the set $\mathcal{P}_t^{nr}(ij)$, such that allocated RBs of these vehicles are minimized. Therefore, the RBs migration amidst pairs of FNs i and j is mapped as an ILP problem, which is described in Section 4.2.1.

The ILP problem in Eq. (4.9) is a SAP, a well-known NP-Hard problem. SAP is a particular case of GAP, which maps the l number of students to p number of seminar halls to maximize the profit [127]. The SAP instance is transformed into our ILP problem instance by relating the students to a set of vehicles in the non-restricted coverage areas, seminar halls to FNs, and the capacity of the s^{th} seminar hall, denoted as B_s, to the capacity of the i^{th} FN, labelled as L_i. However, it is essential to note that the profit derived from assigning student r to seminar hall s is reversed when establishing connections between vehicles $k \in \mathcal{P}_t^{nr}(ij)$ and FN $i \in \mathcal{G}$. This transformation is carried out in polynomial time. On the other hand, the ILP problem can be solved when the number of FNs is fixed (i.e., the number of seminar halls is fixed) [127]. Therefore, we propose an ERO algorithm to maximize the throughput by fixing the number of FNs to a finite number.

5.2.2 Algorithm Description

The proposed algorithm is given in Algorithm 5.1 for relocating RBs amidst pairs of FNs in FVNETs. An example of FVNET is shown in Figure 5.1a. Now consider a graph, $G(V, E)$, with a set of vertices and edges, representing the FVNETs with $\mathcal{G}$ number of FNs. The set V constitutes the FNs, and the set E comprises the edges in G. An arc between FN i and FN j in G exists if and only if there exists at least a vehicle in a non-restricted area $(i, j) \in \mathcal{Z}$ as shown in Figure 5.1b. The ERO algorithm takes the graph, $G(V, E)$, and the number of FNs, $\mathcal{G}$, as input. It performs the RBs migration by constructing a minimum priority queue using the binary heap in Procedure 3. The ERO algorithm finds, $\mathcal{D}_t$, the set of vehicles in all non-restricted coverage areas of FNs, $\forall (i, j) \in \mathcal{Z}$, $1 \leq i, j \leq \mathcal{G}$, $i \neq j$, for RBs migration in a time instant t. Further, it also finds the occupied capacity, L, of all FNs at a particular time instant t (Line 5 to Line 14 in Algorithm 5.1). For migrating RBs of vehicles in $\mathcal{D}_t$, it invokes the `Migrate( )` (Procedure 3) with $\mathcal{D}_t$ and L as parameters in Line 15 of Algorithm 5.1.

The `Migrate( )` (Procedure 3) constructs the minimum priority queue, Q_p, for the given set of vehicles in $\mathcal{D}_t$ using occupied capacities of FNs in L (Line 2). It finds the FN g with minimum occupied RBs and the vehicle $k_{min} \in \mathcal{D}_t$ with minimum RBs with respect to FN g. The FN g is obtained using the `GetMinKey( )`, which is an operation on the priority queue, Q_p, in Line 3, and the vehicle k_{min} is obtained by invoking `SelectVehicle( )` (Procedure 4). If `SelectVehicle( )` returns the vehicle k_{min} having the least RBs with respect to FN g in Line 9, then the RBs of vehicle k_{min} are migrated to the FN g in the lines from Lines 13 to 15. Otherwise, the FN g is removed from L in Line 11. From Lines 12 to 18, the Procedure 3 relocates the RBs of vehicle k_{min} from FN i to FN g only if k_{min} is not served by FN g and desired RBs of that vehicle satisfied by the FN g. Then it updates the occupied RBs and available RBs of FN i and FN g in Line 15. Otherwise, it skips the k_{min} vehicle from RBs migration in Line 17. After the successful migration of RBs, the vehicle k_{min} is discarded from the set $\mathcal{D}_t$ in Line 18. This practice is repeated until the set $\mathcal{D}_t$ becomes empty.

Algorithm 5.1 ERO Algorithm

```
Inputs: $G(V, E)$, $\mathcal{G}$
Outputs: A set of vehicles for RBs migration, throughput, service capability, serviceability, and availability
$\mathcal{D}_t \leftarrow \emptyset$            ▷ The set of vehicles in the non-restricted coverage areas of FNs
$L \leftarrow \emptyset$            ▷ The set of occupied capacities of FNs
$i \leftarrow 1$            ▷ Initialization of $i^{th}$ FN
while $i \leq \mathcal{G}$ do
    $\mathcal{D}_i \leftarrow \emptyset$
    for each neighbor FN $j$ of $i$, $1 \leq j \leq \mathcal{G}, i \neq j$ do
        $\mathcal{P}_t^{nr}(ij) \leftarrow \mathcal{P}_t(i) \cap \mathcal{P}_t(j)$
        $\mathcal{P}_t^{r}(i) \leftarrow \mathcal{P}_t(i) \setminus \mathcal{P}_t^{nr}(ij)$
        $\mathcal{P}_t^{r}(j) \leftarrow \mathcal{P}_t(j) \setminus \mathcal{P}_t^{nr}(ij)$
        $\mathcal{D}_i \leftarrow \mathcal{D}_i \cup \mathcal{P}_t^{nr}(ij)$
    end for
    $\mathcal{D}_t \leftarrow \mathcal{D}_t \cup \mathcal{D}_i$
    $O_t(i) \leftarrow \sum_{k=1}^{|\mathcal{P}_t(i)|} b_k^i$
    $L \leftarrow L \cup O_t(i)$
    $i \leftarrow i + 1$
end while
Migrate($\mathcal{D}_t$, L)
Find throughput, serviceability, availability and service capability
```

5.2.3 An Illustration

Consider an FVNET with four FNs and eight vehicles for RBs migration, as shown in Figure 5.1a and its corresponding graph model, shown in Figure 5.1b, is given as input to the ERO algorithm. In FVNET, shown in Figure 5.1a, the vehicles which FNs serve are illustrated by thick red lines, and the vehicles whose RBs can be shifted to FNs are depicted by dashed red lines. A number beside the red lines denotes the number of RBs required by the vehicle from the corresponding FN. In the graph representation of FVNET, given in Figure 5.1b, the vertices denote the FNs, and the edge between vertices represents the existence of vehicles in the non-restricted coverage region of FNs. The FVNET depicted in Figure 5.1a is a sample snapshot of a traffic scenario for easy understanding of the proposed algorithm.

The proposed algorithm finds the vehicles in FNs' non-restricted and restricted coverage regions using Algorithm 5.1. The algorithm ERO always strives to reduce the allotted

Procedure 3 Migrate($\mathcal{D}_t$, L)

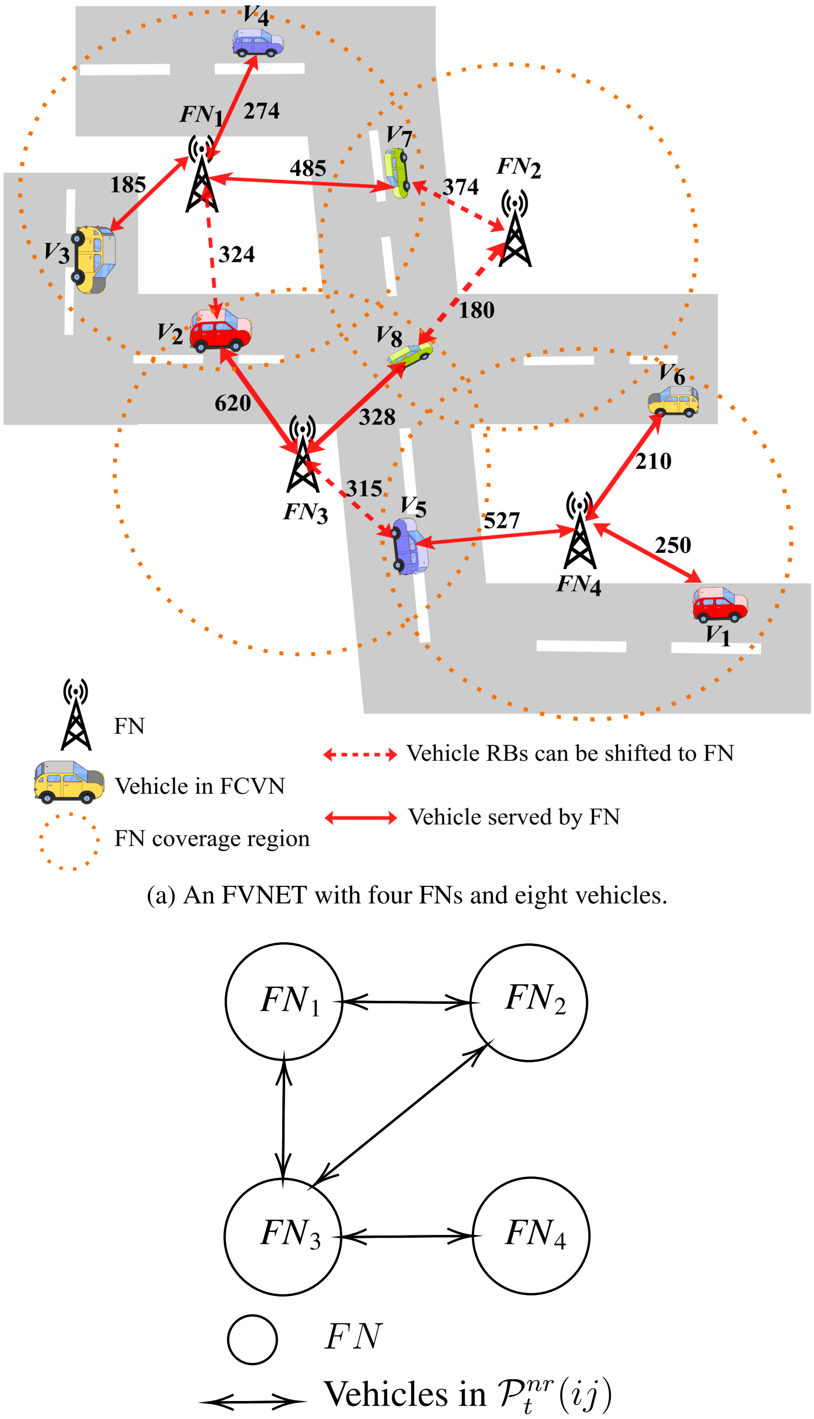
Inputs: A set of vehicles in non-restricted coverage regions, $\mathcal{D}_t$ and allocated capacity of FNs, L

Outputs: RBs Migration

```
while $\mathcal{D}_t \neq \emptyset$ do
    $Q_p \leftarrow$ MinPriorityQueue$(L)$
    $g \leftarrow Q_p$.GetMinKey()
    $list \leftarrow \mathcal{D}_t(g)$
    $migrateList \leftarrow \emptyset$
    for $\forall\, k \in list$ do
        if $k \in \mathcal{D}_t$ then
            $migrateList \leftarrow migrateList \cup k$
        end if
    end for
    $k_{min} \leftarrow$ SelectVehicle$(migrateList, g)$
    if $k_{min} = -1$ then
        $L$.Remove$(g)$
    else
        if $A_t(g) \geq b^g_{k_{min}}$ and $k_{min}$ is not served by $g$ then
            Migrate the RBs of $k_{min}$ from FN $i$ to FN $g$
            Update the $O_t(i)$, $O_t(g)$, $A_t(i)$ and $A_t(g)$ of $i^{th}$ and $g^{th}$ FNs
        else
            Skip the vehicle $k_{min}$ from RBs migration
        end if
        $\mathcal{D}_t \leftarrow \mathcal{D}_t - \{k_{min}\}$
    end if
end while
```

Procedure 4 SelectVehicle($migrateList$, g)

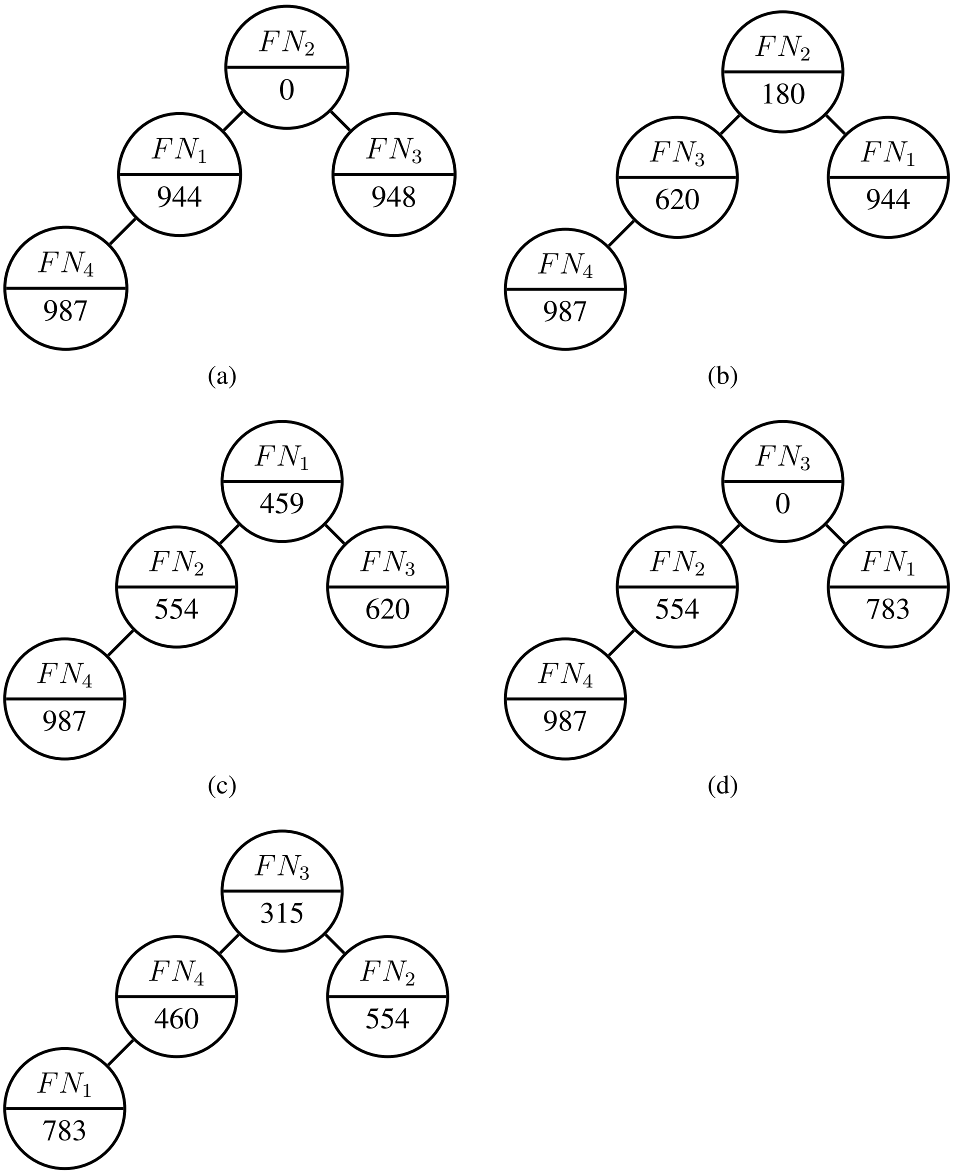
```
$min\_rbs \leftarrow max\_number$
$\vartheta_{min} \leftarrow -1$
for $\forall\vartheta \in migrateList$ do
    if $b^{\vartheta}_g \leq min\_rbs$ then
        $min\_rbs \leftarrow b^{\vartheta}_g$
        $\vartheta_{min} \leftarrow \vartheta$
    end if
end for
return $\vartheta_{min}$
```



(a) An FVNET with four FNs and eight vehicles.

(b) A graph model of Figure 5.1a.

Figure 5.1: An FVNET for RBs migration using the proposed algorithm.



(a)

(b)

(c)

(d)

(e)

Figure 5.2: Binary heap construction using allocated capacities of FNs while migrating RBs of vehicles using the proposed algorithm.

RBs of vehicles in non-restricted coverage areas by relocating RBs among pairs of FNs. For migrating RBs, the algorithm selects the vehicle $k_{min} \in \mathcal{P}_t^{nr}$ with minimum RBs, such that the occupied RBs of FN g is minimum. To ensure the minimum occupied RBs of FN g, the ERO constructs the minimum priority queue using the occupied capacities of FNs. From the Algorithm 5.1, we obtain the union of vehicles in all non-restricted coverage regions of FNs, i.e., $\mathcal{D}_t$ = $\{V_2$, V_5, V_7, $V_8\}$. The obtained occupied capacities of FNs for the vehicles in the restricted and non-restricted regions is L = $\{O_1$ = 944, O_2 = 0, O_3 = 948, O_4 = 987$\}$. Then, Figure 5.2a shows that the binary heap is built using these occupied capacities. The RBs migration is performed using Procedure 3 by giving the set of vehicles in non-restricted regions and occupied capacities, $\mathcal{D}_t$ and L, respectively, as inputs. However, by constructing a minimum priority queue, the `Migrate( )` (Procedure 3) always finds the FN g with minimum occupied RBs for migration. The minimum priority queue is built using the binary heap data structure in the first iteration, as shown in Figure 5.2a. The Procedure 3 also finds the vehicle k_{min} having minimum RBs required from the FN g. Thus, it finds g = FN_2 from Figure 5.2a. Then the set of vehicles that are in the vicinity of FN g = FN_2 is $list$ = $\{V_7$, $V_8\}$ and the set of vehicles whose RBs can be migrated to FN g = FN_2 is $migrateList$ = $\{V_7$, $V_8\}$. The vehicle k_{min} = V_8 is obtained from the vehicles in the set $migrateList$ using the `SelectVehicle( )` (Procedure 4). Then the RBs of vehicle k_{min} = V_8 are shifted to FN g from FN i (i.e., FN_2 from FN_3) only if the FN g = FN_2 satisfies the desired RBs of vehicle k_{min}. After successful migration, the vehicle k_{min} = V_8 is removed from $\mathcal{D}_t$. Then the updated occupied capacity of FNs and vehicles in non-restricted regions are L = $\{O_1$ = 944, O_2 = 180, O_3 = 620, O_4 = 987$\}$ and $\mathcal{D}_t$ = $\{V_2$, V_5, $V_7\}$, respectively. In the same way, the proposed algorithm constructs the binary heap in each iteration until the RBs migration of vehicles in the non-restricted coverage areas is completed in FVNET (i.e., the set $\mathcal{D}_t$ becomes empty).

Figure 5.3 shows the overall occupied capacity of FNs before migration is 2879, and it is reduced to 2112 after migration using the proposed algorithm. Therefore, the percentage of RBs reduced using ERO is 26.64%. This minimization greatly improves the number of vehicles getting services from the network, which alternatively improves the throughput of the network. Note that this minimization also enhances the network's throughput, service-

	FN_1	FN_2	FN_3	FN_4
V_1				250
V_2	324		620	
V_3	185			
V_4	274			
V_5			315	527
V_6				210
V_7	485	374		
V_8		180	328	
$O_t(i)$ =	944	0	948	987

(a)

	FN_1	FN_2	FN_3	FN_4	
V_1				250	
V_3	185				
V_4	274				
V_6				210	
$O_t(i)$ =	459	-	-	460	Occupied capacity for vehicles in $\mathcal{P}_t^r(i)$
V_8		180			
V_7		374			
V_2	324				
V_5			315		
$O_t(i)$ =	783	554	315	460	Overall occupied capacities after migration

(b)

Figure 5.3: An optimal RB migration using ERO algorithm for FVNET. (a) Occupied RBs of FNs before migration. (b) Occupied RBs of FNs after migration using ERO.

ability, availability, and service capability. On the other hand, the existing algorithms like RO, ARB [39], SA [25], and DRO [24] reduce the allocated RBs up to 21.5%, 26%, 26%, and 10.28%, respectively. This exhibits the remarkable performance of the ERO algorithm over the existing algorithms.

5.2.4 Complexity Analysis

Procedure 4, invoked with the set $migrateList$ as input, iterates $\mathcal{O}(|migrateList|)$ times. It is worth noting that the set $migrateList$ is finite, so this algorithm operates in constant time to find ϑ_{min}. In Procedure 3, a binary heap is constructed with $|L|$ nodes. If the size of set L is $|L| = \mathcal{G}$, in the worst case, the construction of the binary heap in Line 2 takes $\mathcal{O}(\mathcal{G} \log \mathcal{G})$. In Procedure 3, operations such as retrieving the FN with the minimum occupied capacity in Line 3 and removing the FN from the set L in Line 13 take $\mathcal{O}(1)$. Furthermore, the binary heap is constructed in each iteration of the while loop, which runs $\mathcal{O}(|\mathcal{D}_t|)$ times in Procedure 3. Therefore, it runs for $\mathcal{O}(|\mathcal{D}_t|\mathcal{G} \log \mathcal{G})$ times in the worst case.

In Algorithm 5.1, the set of vehicles in the restricted and non-restricted coverage regions, from Line 6 to Line 10, is obtained for a given i^{th} FN and its neighbouring j^{th} FN, with $1 \leq i, j \leq \mathcal{G}$ and $n \neq m$. This operation is completed in $\mathcal{O}(\mathcal{G})$ time. However, the outer while loop also iterates $\mathcal{O}(\mathcal{G})$ times and invokes Procedure 3 in Line 15. Therefore, the worst-case running time complexity of the proposed algorithm is $\mathcal{O}(\mathcal{G}^2 + |\mathcal{D}_t|\mathcal{G} \log \mathcal{G})$.

5.3 Performance Evaluation

The effectiveness of the proposed algorithm ERO is assessed in terms of throughput, serviceability, availability, and service capability. In addition, the simulation results are analogized with current algorithms, such as RO, ARB [39], SA [25], and DRO [24]. As stated earlier, the vehicles and FNs are chosen randomly in the RO algorithm.

5.3.1 Simulation Setup

A virtual environment is created for simulation runs using Python 3.8 on PyCharm IDE 2020.1.3. A computing device with 64.0 GB installed RAM, a 64-bit operating system, and Intel(R) Xeon(R) Gold 622R CPU @ 2.90 GHz 2.89 GHz processor is used to run this IDE. Note that 2.90 GHz and 2.89 GHz are the clock speeds of the processor base frequency and the frequency computed by Windows, respectively. The simulation environment, including traffic parameters, is configured by following the setup provided in [24, 39]. The proposed algorithm's performance is assessed within a network where FNs are deployed across a city measuring an area of $[5000 \times 5000]^2$ meters, as depicted in Figure 5.4. The Voronoi tessellation, indicated by the dotted lines, illustrates a city area accommodating 10-50 FNs. The FNs are deployed in a way that the coverage areas overlap with one or more neighbouring FNs. Vehicle arrivals and departures from the network adhere to a Poisson

Table 5.3: Simulation parameters and their values

Parameter	**Value**
Number of FNs $(\mathcal{Z})$	[10 $\sim$ 50]
Radius of FN coverage (R)	500 m
Bandwidth of FNs (B)	[10, 15, 20] MHz
Vehicles arrival rate (λ)	[5 $\sim$ 10] vehicles/s
Vehicles departure rate (μ)	[5 $\sim$ 10] vehicles/s
Data rate required (r^k)	[0.5 $\sim$ 2] Mbps
Network area (A)	5000 m $\times$ 5000 m
Number of vehicles	[150 $\sim$ 3000] vehicles
Gaussian Noise (N_k)	-104 dBm
Co-channel Inference (I_k)	-75 dBm
Pathloss exponent (β)	3
Scaling co-efficient (γ)	1
Time slot duration (δs)	1 s

distribution, with mean arrival and departure rates varying between 5 vehicles/s and 10 vehicles/s. The number of vehicles with the required data rates and distance from FNs is generated randomly within a range from 150 to 3000 in the environment. A free flow discrete time traffic model is considered for uninterrupted and homogenous vehicular traffic over a fixed length of non-restricted coverage regions [38,50]. The FNs serve the vehicles meeting data rates [0.5 $\sim$ 2] Mbps. A dynamic hierarchical topology is used to organize HPNs, RSUs and vehicles in FVNETs. However, vehicles are connected to a FN using the star topology. Further, the Monte-Carlo simulations are conducted using a queuing model with 300 time slots, considering three traffic scenarios. The average results are obtained by running simulations equal to one hundred and fifty times. The Nakagami model, the most realistic model, is used to illustrate signal propagation. IEEE 802.11p is used in FVNETs as a MAC protocol to facilitate wireless connectivity. The simulation outcomes display the consequence of the rise in the number of connections to the network on the throughput, serviceability, availability, and service capability. The various parameters used for the simulations are given in Table 5.3.

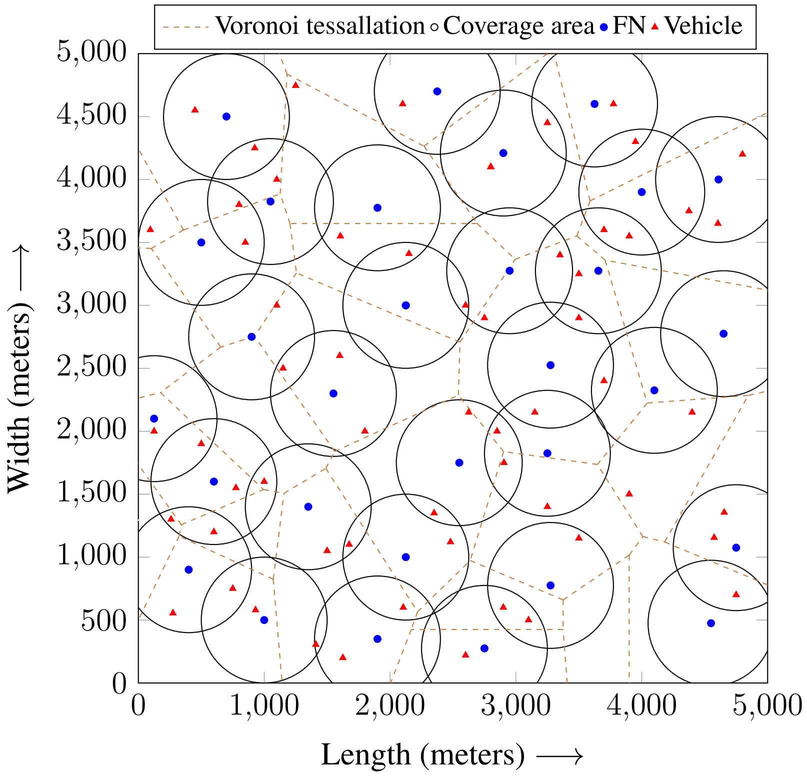


Figure 5.4: Voronoi tessellation (dotted lines) and the distribution of thirty FNs and sixty vehicles in an area of $[5000 \times 5000]^2$ meters in the simulation where the coverage regions are formed corresponding to the distance threshold (circles centered at blue dots).

5.3.2 Results and Discussions

The different resource management algorithms proposed in literature [24, 25, 39] such as DRO, ARB, SA, and RO, respectively, are considered for performance comparison. The DRO algorithm performs the RBs migration to reduce the RBs of vehicles in the non-restricted area. To maximize the RBs reduction, DRO uses the maximum weight matching solution, i.e., it reduces the RBs of vehicles in the non-restricted areas obtained from the set of edges. The set of edges is drawn from the maximum weight matching problem. The SA algorithm relocates the RBs of vehicles from the selected FN to the neighbouring FN. However, the FN is selected using the graph colouring solution of the network. The ARB algorithm uses the Hungarian method for associating vehicles and FNs [131]. Then, using the back pressure approach, it migrates the RBs of vehicles in the non-restricted regions from a higher load FN to a lower load neighbouring FN. The RO algorithm selects the random FN for migrating RBs of vehicles selected randomly in the non-restricted regions.

However, the ERO algorithm always selects a FN with minimum occupied capacity using the minimum priority queue. Then, it selects the vehicle that needs minimum RBs from the selected FN to perform RBs migration.

The performance of ERO is analyzed with the DRO, ARB, SA, and the basic algorithm RO. However, the network throughput is obtained using the achievable rate of vehicles getting services from different FNs in different time slots [25]. The other performance metrics like serviceability, availability, and service capability are defined in Section 5.1.2. The effectiveness of ERO is measured in terms of throughput, serviceability, service capability, and availability in each time slot when the number of vehicles connecting to the network rises.

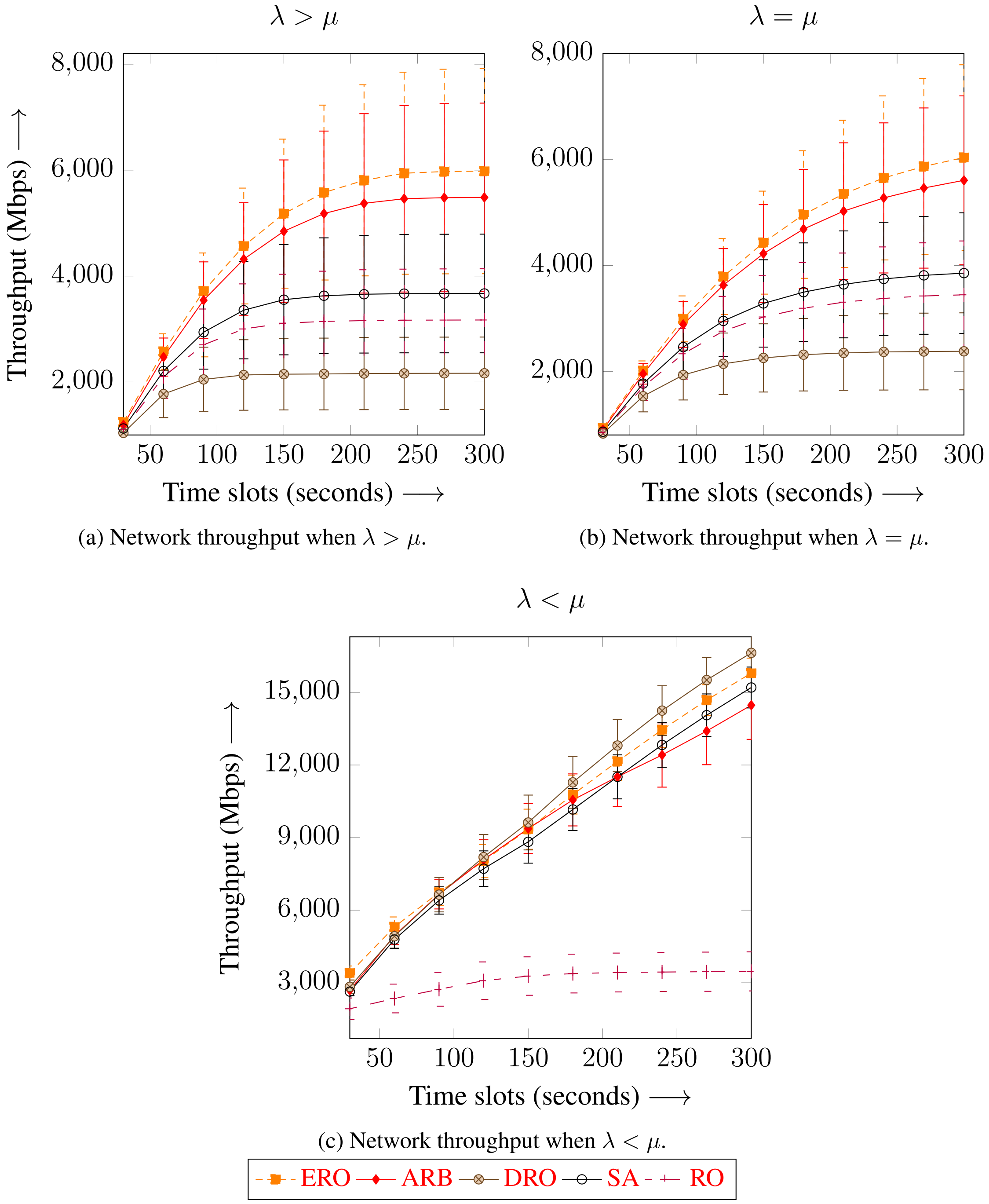
5.3.3 Influence of the Rise in the Number of Vehicles

In this chapter, we consider the influence of the rise in the number of vehicles on the network's throughput, serviceability, availability and service capability for three scenarios: (a) the mean arrival rate is greater than the mean departure rate ($\lambda > \mu$), (b) the mean arrival rate is equal to the mean departure rate ($\lambda = \mu$), and (c) the mean arrival rate is less than the mean departure rate ($\lambda < \mu$). Subsequently, the average results of three scenarios are obtained for the network's throughput, serviceability, availability, and service capability.

5.3.3.1 Throughput

The mean departure (arrival) rate indicates the number of vehicles leaving (entering) the network. The achievable rate of the vehicle is obtained from the Shannon formula as shown in Eq. (5.3). The throughput of the network is proportional to the time assigned to the vehicles receiving utilities. Therefore, as the number of vehicles obtaining services increases, the throughput also increases. The influence of the rise in the number of vehicles coming at the network on throughput when $\lambda > \mu$, $\lambda = \mu$ and $\lambda < \mu$ is shown in Figure 5.5a, Figure 5.5b and Figure 5.5c, respectively.

In the scenario $\lambda > \mu$, the network experiences a continuous increase in the number of vehicles connecting in every time slot, as the mean arrival rate is greater than the mean de-



(a) Network throughput when $\lambda > \mu$.

(b) Network throughput when $\lambda = \mu$.

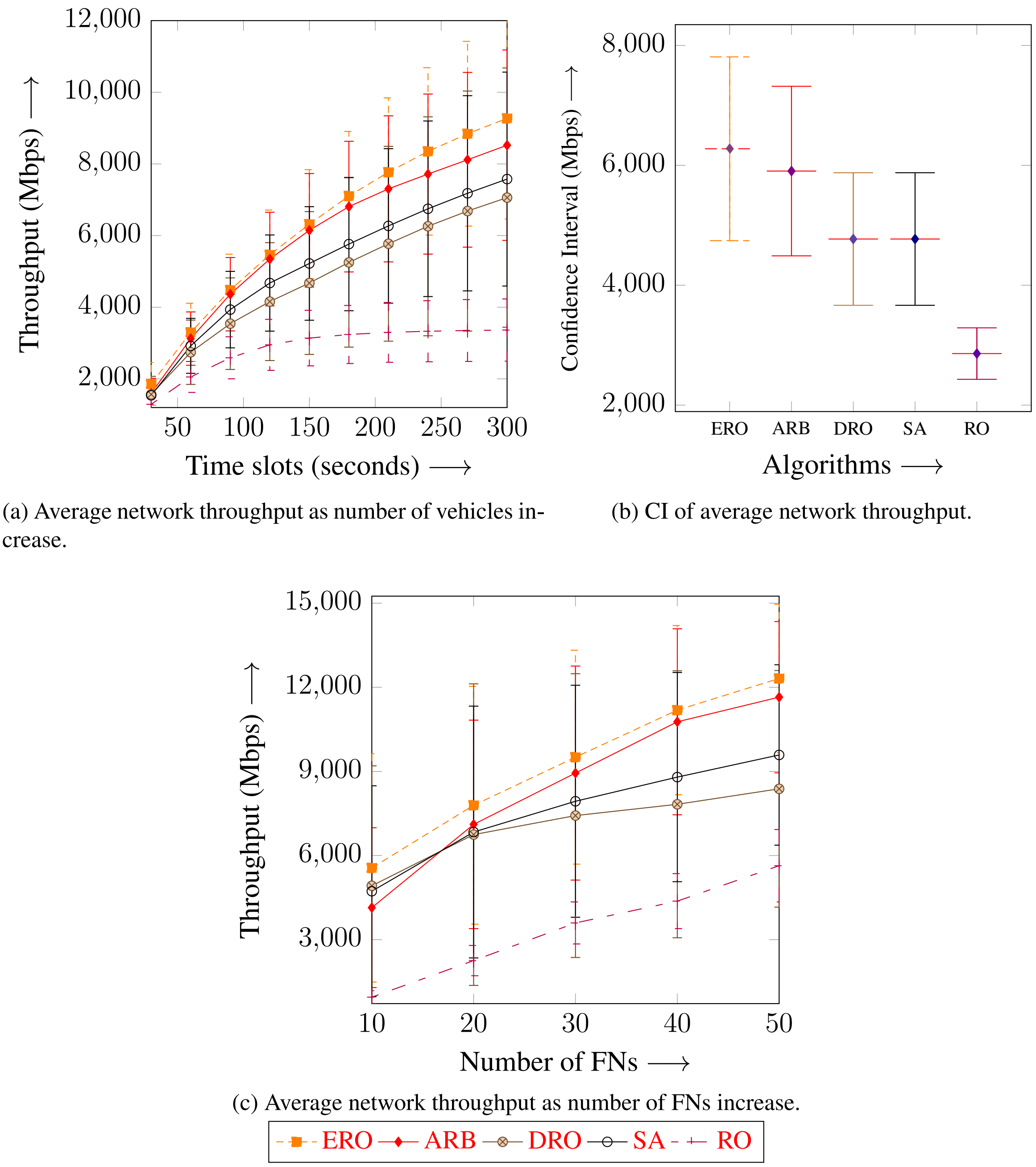
(c) Network throughput when $\lambda < \mu$.

Figure 5.5: The network throughput with confidence intervals in each time slot using the ERO, RO, ARB, SA, and DRO algorithms.

parture rate. However, it is noticed that the FNs have abundant resources during initial time slots to furnish utilities to the vehicles. Therefore, the rise in the coming of vehicles in the network makes the FNs helpless to provide desired services, which impacts the throughput of the network. The network's throughput increases gradually during the initial time slots for all algorithms. Subsequently, the throughput stabilizes as the network becomes congested due to growing traffic. The reduction of RBs using the DRO scheme is declining due to the maximum weight-matching solution. Hence, there is a stable throughput using DRO. Similarly, the RO and SA algorithms demonstrate consistent network throughput as the number of vehicles entering the network rises. This consistency is due to less RBs reduction for the vehicles in the non-restricted areas using RO, SA and DRO algorithms. At the same time, the ARB algorithm enhances the throughput by coordinating RBs migration to reduce allocated RBs for the vehicles in the non-restricted areas. However, the ERO algorithm experiences a gradual increase in the throughput. More specifically, the throughput reaches 6000 Mbps at time slot 240. Subsequently, the throughput stabilizes as resource blocks become fully occupied by previously arrived vehicles. As a result, some vehicles are not admitted to the network. In summary, the proposed algorithm improves the throughput by 61.76%, 6.86% and 43.56% on average compared to RO, ARB and SA, respectively.

When $\lambda = \mu$, where the mean arrival rate equals the mean departure rate, it is evident that the FNs retain adequate resources to deliver services to vehicles in every time slot. The RO, SA and DRO algorithms perform the minimum number of RB migrations to reduce the allocation of RBs as the network experiences an influx of vehicles. As a result, there is a stable throughput using RO, SA and DRO algorithms due to decreased available RBs in the network. Nonetheless, the proposed ERO algorithm enhances network throughput by migrating RBs of vehicles in non-restricted regions between pairs of FNs. As a result, newly arrived vehicles get services in the network. Here, the throughput reaches 6000 Mbps at time slot 300 and it stabilizes in the long perspective. On average, ERO surpasses RO, ARB, SA, and DRO by 47.06%, 5.39%, 35.71%, and 95.36%, respectively.

When $\lambda < \mu$, we initialize the network with a set of [(μ - λ) $\times$ T] vehicles. The rationality behind this initialization is that there are enough vehicles for departure to realize



(a) Average network throughput as number of vehicles increase.

(b) CI of average network throughput.

(c) Average network throughput as number of FNs increase.

Figure 5.6: Average network throughput and CI of ERO, RO, ARB, SA, and DRO algorithms.

the $\lambda < \mu$ scenario. Otherwise, $\lambda < \mu$ scenario is the same as $\lambda = \mu$ scenario as the maximum number of vehicles that can leave equals λ. We observe different trends in the $\lambda > \mu$ and $\lambda < \mu$ scenarios in Figure 5.5. Particularly, in the $\lambda < \mu$ scenario, throughput increases linearly up to the time slot 300 and is not stabilized until that time. This is due to a decrease in the number of vehicles and an increase in the available RBs in the network. As a result, there is no exhaustion of available RBs in the networks. The RO algorithm provides stable throughput in all time slots as the network is already fully occupied with available RBs from the initial time slot. The DRO provide better throughput as vehicle congestion decreases in the network. However, the ERO algorithm enhances the throughput by reducing the RBs of vehicles in non-restricted regions. Therefore, it provides throughput better than DRO in initial time slots, and it exhibits throughput similar to DRO on average as traffic decreases in the network. Notably, the ERO algorithm outperforms the ARB and SA algorithms by an average of 6.85% and 8.00%, respectively.

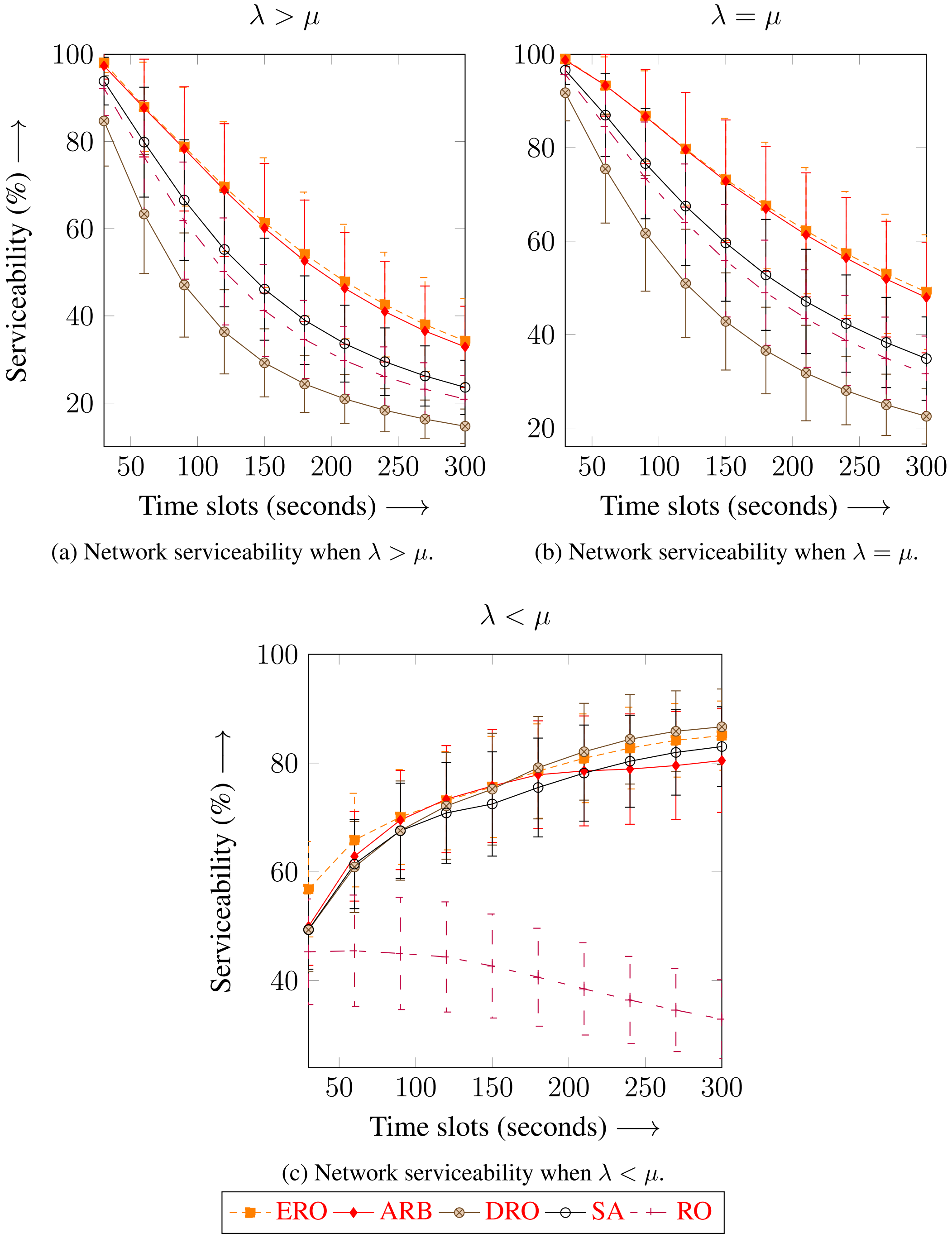
We calculate the average network throughput for the three scenarios at every time slot as depicted in Figure 5.6a. It is evident that with an increasing number of vehicles entering the network, throughput gradually rises using both proposed and existing algorithms. Notably, the proposed algorithm outperforms the existing algorithms due to its greedy approach in selecting vehicles with minimal RBs requirements and FNs with the least occupied capacity. Consequently, it achieves a significant throughput improvement, surpassing RO, ARB, SA, and DRO algorithms by 97.6%, 6.65%, 20.19%, and 29.90%, respectively. We compute the 95% confidence intervals (CIs) for every time slot in all scenarios with 150 simulation runs and show CIs of average throughput for every time slot in Figure 5.5 and Figure 5.6a, respectively. The CI is determined by the formula $CI = \bar{x} \pm Z\,(\frac{\sigma}{\sqrt{n}})$, where $\bar{x}$ is the sample mean, σ is the standard deviation, n is the sample size and Z is the value from standard normal distribution corresponding to desired confidence level (e.g., 1.96 for a 95% CI). The top and bottom lines denote the upper and lower limits, and the middle line denotes the mean. Figure 5.6b shows the CIs by averaging the network throughput values for each algorithm over all the points from Figure 5.6a, which in turn are average values of corresponding points over the three scenarios (Figure 5.5). Further, Figure 5.6c shows the average throughput of three scenarios with CIs for the number of FNs in the network using proposed and existing algorithms. The ERO algorithm enhances the throughput as number of FNs increases on average by 11.94%, 21.39%, and 29.27% when compared to ARB, SA, and DRO algorithms, respectively.

5.3.3.2 Serviceability

The network's serviceability and CIs for every time slot of the proposed and existing algorithms are assessed by considering three distinct scenarios (i.e., $\lambda > \mu$, $\lambda = \mu$, and $\lambda < \mu$), as depicted in Figure 5.7a, Figure 5.7b, and Figure 5.7c, respectively. When $\lambda > \mu$, as the number of vehicles entering the network increases over time, the serviceability of the network diminishes using both proposed and existing algorithms. This decline is attributed to the escalating traffic within the network, leading to a depletion of resources and a consequent decline in the network's ability to serve arriving vehicles effectively. However, in contrast, the ERO algorithm enhances network serviceability as the influx of vehicles into the network grows. On average, it improves serviceability by 44.49%, 2.35%, 30.78%, and 97.2% compared to the RO, ARB, SA, and DRO algorithms, respectively.

In the scenario $\lambda = \mu$, both the proposed and existing algorithms exhibit enhanced serviceability during the initial time slots. Notably, the ARB algorithm provides better serviceability in the early time slots. However, the network experienced increased vehicles over time due to reduced available RBs. Therefore, the serviceability of the network declines in ARB. Simultaneously, the proposed ERO algorithm demonstrates serviceability levels similar to ARB during the initial time slots. Subsequently, it improves serviceability as the vehicles arrive in the network by migrating RBs of vehicles in non-restricted regions between pairs of FNs. Consequently, the ERO algorithm maximizes serviceability on average by 32.43%, 1%, 23.88%, and 71.57% compared to RO, ARB, SA, and DRO algorithms, respectively.

In the scenario $\lambda < \mu$, there is an increase in the available RBs and a decrease in the number of vehicles in the network in each time slot. This increase in the available RBs enhances the serviceability in every time slot. The serviceability using RO declines because the network is initialized with vehicles from the initial time slot, leaving the network with no available RBs. The DRO algorithms provide superior serviceability as the congestion of



(a) Network serviceability when $\lambda > \mu$.

(b) Network serviceability when $\lambda = \mu$.

(c) Network serviceability when $\lambda < \mu$.

Figure 5.7: The network serviceability with CIs in each time slot using the ERO, RO, ARB, SA, and DRO algorithms.

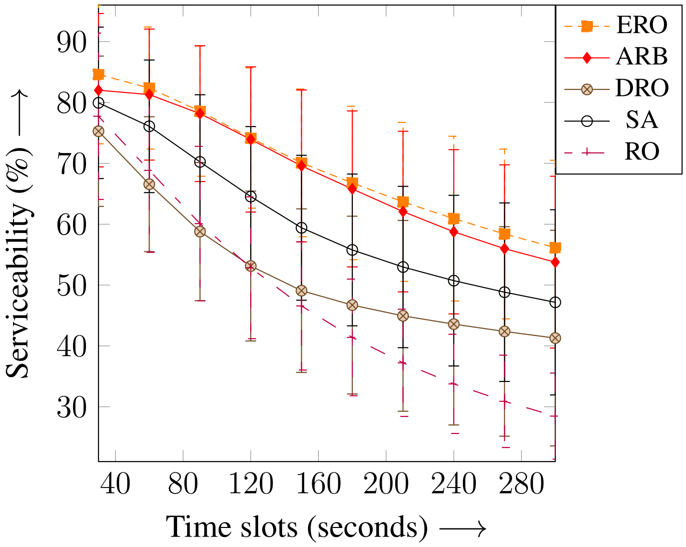
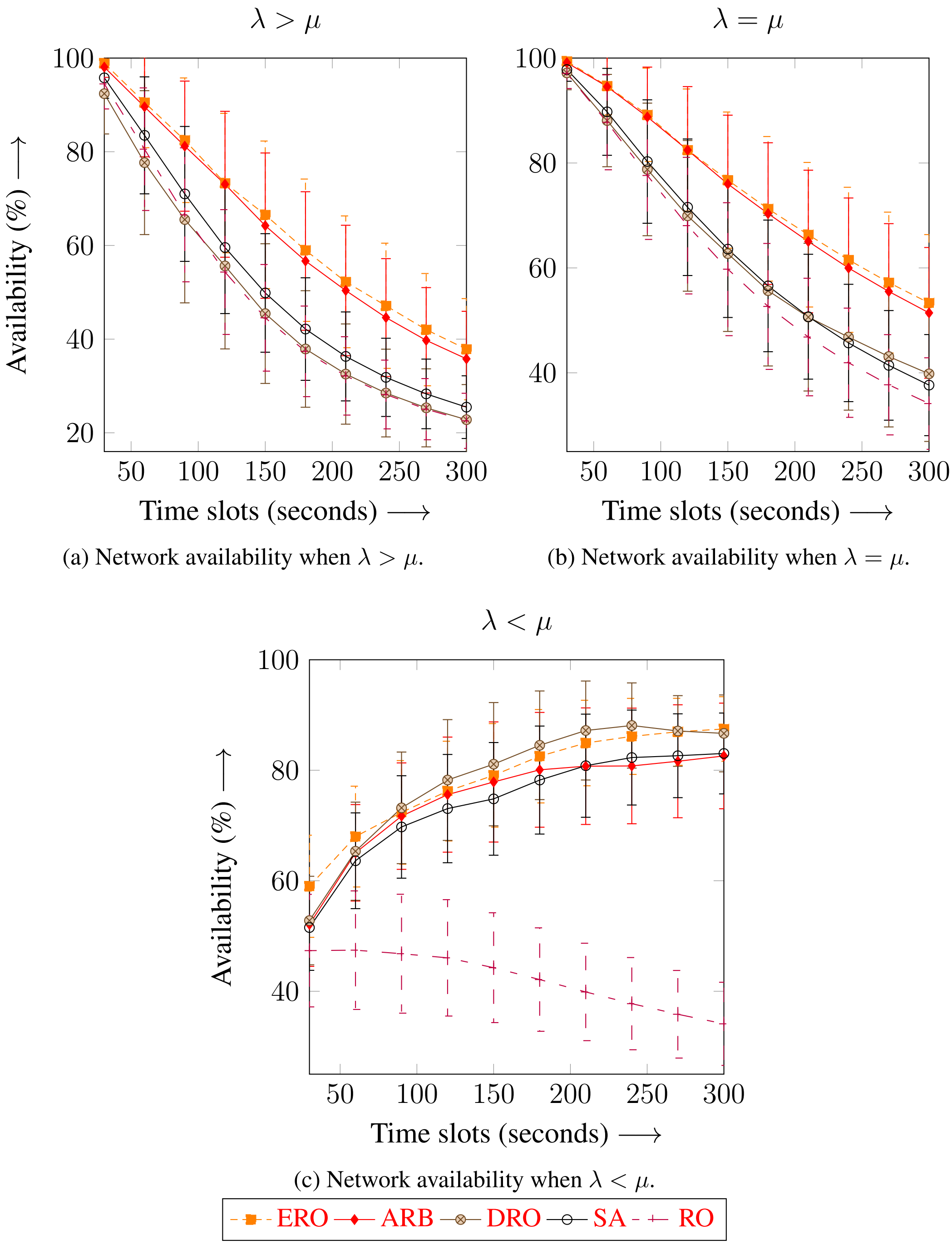


Figure 5.8: Average network serviceability of $\lambda > \mu$, $\lambda = \mu$ and $\lambda < \mu$ scenarios.

vehicles in the network decreases. Nevertheless, the ERO algorithm ultimately surpasses the DRO by 2.09% on average. The ERO algorithm prioritizes the FNs with minimum occupied capacity and vehicles requiring the least RBs for relocation between pairs of FNs. Consequently, the ERO algorithm outperforms the RO, ARB, and SA algorithms by an average of 90.03%, 3.89% and 4.91%, respectively.

Figure 5.8 shows the average serviceability of the network obtained by averaging the serviceability of the three scenarios. It also shows the CIs of average serviceability for every time slot. The serviceability of the network reduces gradually in each time slot. This reduction is due to a decrease in the available RBs in the network, causing the FNs to exhaust and become inoperable in delivering utilities to arrived vehicles. It is noted that the ARB provide better serviceability in the initial time slots. Simultaneously, the proposed ERO algorithm exhibits serviceability similar to ARB and enhances serviceability as the number of vehicles arriving in the network grows. As a result, the ERO algorithm surpasses the RO, ARB, SA, and DRO by an average of 54.95%, 1.73%, 15.79%, and 35.08%, respectively.



(a) Network availability when $\lambda > \mu$.

(b) Network availability when $\lambda = \mu$.

(c) Network availability when $\lambda < \mu$.

Figure 5.9: The network availability with CIs in each time slot using the ERO, RO, ARB, SA, and DRO algorithms.

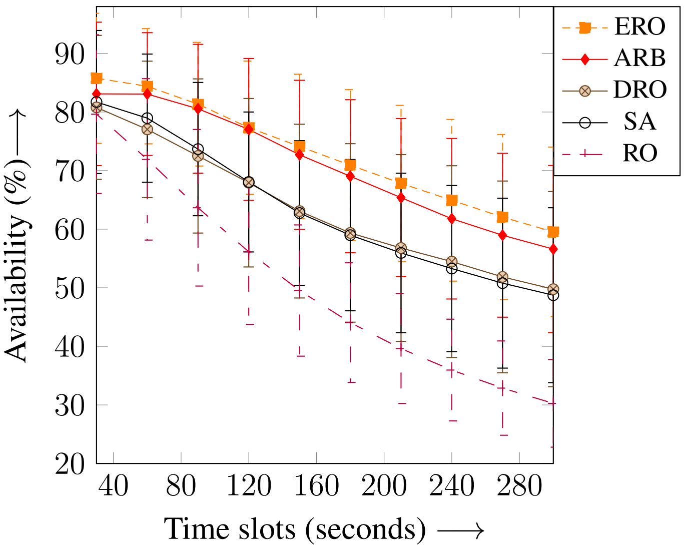


Figure 5.10: Average network availability of $\lambda > \mu$, $\lambda = \mu$ and $\lambda < \mu$.

5.3.3.3 Availability

FNs deliver the services to the vehicles meeting a minimum data rate of 0.512 Mbps as FNs allocate RBs to the vehicles that guarantee the minimum rate. Figure 5.9 shows the availability of the network with CIs at each time slot for the three scenarios. In the scenario where $\lambda > \mu$, both the proposed and existing algorithms demonstrate similar availability in the initial time slot, which proceeds to decline as the number of vehicles arriving at the network increases, as shown in Figure 5.9a. It is observed that the use of the DRO algorithm results in a significant decrease in network availability due to a reduction in the available RBs in the network. In contrast, the proposed algorithm effectively manages RB reduction by relocating RBs among pairs of FNs for vehicles in non-restricted regions. As a result, it maximizes network availability, surpassing the RO, ARB, SA, and DRO algorithms by an average of 44.82%, 3.22%, 31.3%, and 44.09%, respectively. Figure 5.9b shows that the proposed ERO and ARB algorithms experience similar availability in initial time slots when $\lambda = \mu$. Subsequently, as vehicles arrive in the network, the ERO algorithm provides better availability than existing algorithms due to the availability of RBs. As a result, the ERO algorithm excels in delivering higher availability compared to the RO, ARB, SA, and DRO algorithms by an average of 30.6%, 1.44%, 22.52%, and 22.04%, respectively.

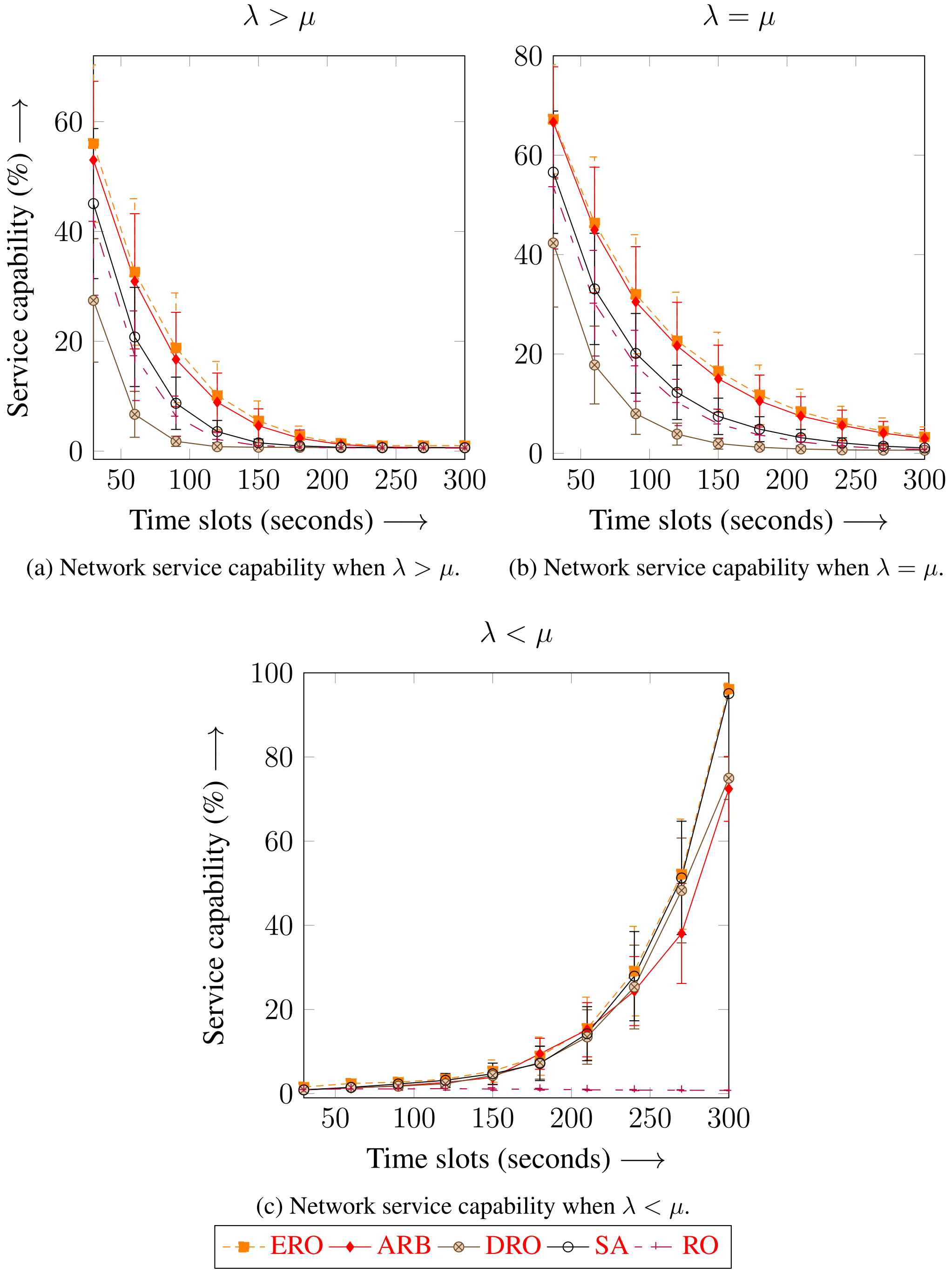
When $\lambda < \mu$, the network is assessed by randomly distributing vehicles at the network's initial time slot. Subsequently, the availability is obtained as shown in Figure 5.9c. The availability increases gradually with the decreasing number of vehicles in the network. The DRO algorithm provides better availability. However, there is an increase in availability using the ERO algorithm due to reduced congested traffic in the network. Moreover, the ERO algorithm is greedy in selecting the vehicles requiring minimum RBs and FNs with minimum occupied capacity. As a result, it enhances availability by 90.28%, 4.84%, and 6.1%, on average, compared to RO, ARB and SA, respectively, with the decrease in the number of vehicles in the network.

The average network availability across the three scenarios with CIs is illustrated in Figure 5.10. The average availability of the network decreases with the decrease in the available RBs of FNs. During the initial time slots, the ARB algorithm attains superior availability. Initially, the ERO algorithm reaches the availability levels of the ARB. However, with decreasing available RBs, the ERO algorithm eventually enhances availability by 3.01% compared to the ARB. Furthermore, the ERO algorithm consistently exhibits better average availability, outperforming the RO, SA, and DRO algorithms by margins of 53.85%, 16.17%, and 15.67%, respectively.

5.3.3.4 Service Capability

Figure 5.11 illustrates the service capability of the network with CIs in each scenario as the number of vehicles reaching the network grows. Service capability reflects the percentage of available resources for allocating RBs to arriving vehicles in the network, according to Eq. (4.6). When $\lambda > \mu$, it is observed that service capability is significantly reduced when employing the RO, SA, and DRO algorithms as the number of vehicles connecting the network increases, as shown in Figure 5.11a. This decrease is due to an increase in the number of vehicles, resulting in a reduction in RBs for vehicles in non-restricted regions. In comparison, the ARB provide better service capability. However, the ERO algorithm optimizes service capability by efficiently reducing allocated RBs for vehicles in non-restricted regions through RB migration between pairs of FNs. As a result, it significantly enhances service capability, outperforming the ARB algorithm by averages of 9.59%.

In the scenario where $\lambda = \mu$, the service capability obtained is shown in Figure 5.11b.



(a) Network service capability when $\lambda > \mu$.

(b) Network service capability when $\lambda = \mu$.

(c) Network service capability when $\lambda < \mu$.

Figure 5.11: The network service capability with CIs in each time slot using the ERO, RO, ARB, SA, and DRO algorithms.

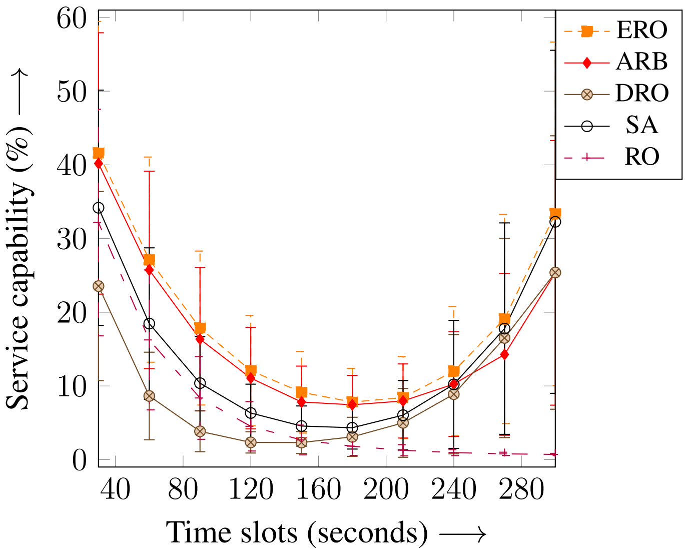


Figure 5.12: Average network service capability of $\lambda > \mu$, $\lambda = \mu$ and $\lambda < \mu$.

The ERO algorithm excels in optimizing RBs reduction due to the ample availability of RBs for migration between pairs of FNs. As a result, the ERO algorithm enhances the network's service capability as congestion-free traffic in the network. It improves service capability by an average of 7.8% compared to the ARB algorithms. Similarly, the service capability is obtained in the $\lambda < \mu$ scenario as shown in Figure 5.11c. It is observed that the service capability is close to zero in the initial time slots due to the lack of available RBs in the network. Subsequently, service capability increases as more vehicles departing the network release the allocated RBs. The RO algorithm provides lower service capability due to fewer RBs migration between pairs of FNs, resulting in vehicle congestion in the network. However, the ERO algorithm maximizes the service capability as vehicle congestion decreases. It enhances the service capability by 35.72%, 22.43% and 36.64% on average compared to ARB, SA and DRO, respectively. We compute the average service capability for the three scenarios with CIs at each time slot shown in Figure 5.12. It is observed that the service capability is more in the initial time slots as there are ample available RBs in the network. Then, it decreases due to an increase in the number of vehicles. Subsequently, the number of vehicles leaving the network increases, and there is an increase in the available RBs in the network. As a result, there is an increase in the service capability of the network. The ERO algorithm exhibits enhanced average service capability compared to existing al-

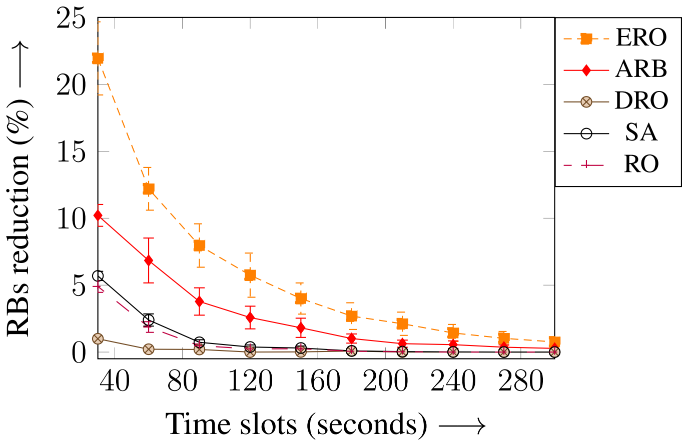


Figure 5.13: Average RBs reduction of $\lambda > \mu$, $\lambda = \mu$ and $\lambda < \mu$.

gorithms. It improves the service capability by 13.75% and 48% on average compared to ARB and SA, respectively.

Figure 5.13 illustrates the average percentage of RBs reduction for the three scenarios in the network with CIs using proposed and existing algorithms. On average, the RO, SA, and DRO algorithms achieve a reduction of less than 1% in allocated RBs, primarily aimed at minimizing the occupied capacity of FNs. In contrast, the RO, ARB, DRO and ERO algorithms achieve average RB reductions of 1%, 2.8%, 0.2% and 5.98%, respectively. The proposed ERO algorithm optimizes the occupied capacity of FNs by coordinating the RBs migration between pairs of FNs. This optimization, in turn, leads to improvements in the network's throughput, serviceability, availability, and service capability. We calculate the standard deviation of FNs based on the average occupied capacity of FNs across three scenarios with CIs, as depicted in Figure 5.14. Standard deviation serves as a metric for evaluating the load balance among FNs in the network. According to simulation results, it is observed that the DRO algorithm initially yields a higher standard deviation, which subsequently decreases to its lowest point. This decline is attributed to a drop in the remaining RBs in the network. On the other hand, the proposed algorithm exhibits a lower deviation in occupied capacity among FNs compared to the existing algorithms in the initial time slot. Furthermore, it consistently maintains a lower standard deviation among FNs as the number of vehicles in the network increases.

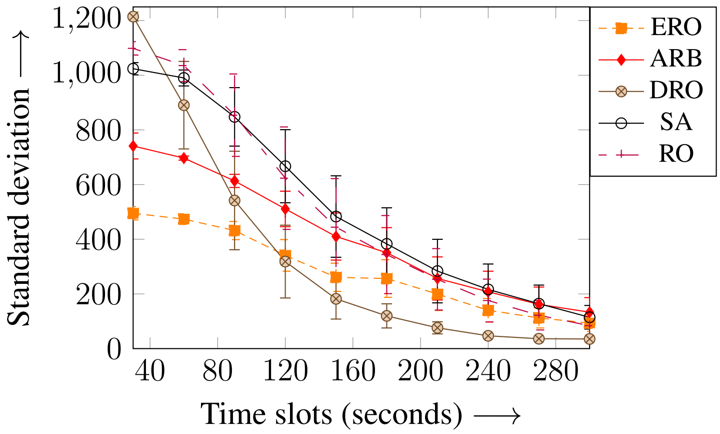


Figure 5.14: Standard deviation of occupied capacities for ERO, RO, ARB, SA, and DRO algorithms.

5.4 Summary

In this chapter, we present an ERO algorithm to synchronize RBs allotment between pairs of FNs to maximize the network's throughput. First, the coverage region of FNs is partitioned into restricted and non-restricted coverage regions. Then, the throughput maximization problem is formulated as reducing allocated RBs of vehicles in the non-restricted coverage areas. This reduction is carried out by migrating RBs of the vehicles among pairs of FNs. Hence, the minimum priority queue is constructed using occupied RBs of FNs to perform optimal migration. As a result, the ERO algorithm always selects a FN with minimum occupied capacity and a vehicle with the minimum required RBs using the minimum priority queue. Therefore, it can maximize the RBs reduction among pairs of FNs, which maximizes the network's throughput. Moreover, the ERO algorithm is assessed in terms of throughput, serviceability, availability, and service capability in each time slot as the number of arriving vehicles within the network grows. The simulation outcomes show that the ERO algorithm surpasses existing algorithms regarding the network's throughput, serviceability, availability, and service capability.

Chapter 6

An Energy-Efficient Resource Allocation Algorithm for Managing On-Demand Services in FVNETs

This chapter presents a resource allocation algorithm, energy-efficient resource allocation (EERA), for offloading upstream services by coordinating RBs allocation among FNs, such that the energy usage of FNs in the downlink is diminished, and the network resource utilization efficiency is maximized. The EERA algorithm builds the B+ tree using occupied RBs of FNs to reduce the allocated RBs of vehicles in the overlap coverage areas of FNs. In addition, the allocated RBs of vehicles are minimized by relocating RBs between pairs of FNs. Further, this reduction of allocated RBs minimizes FNs' energy usage in furnishing vehicles' services. We simulate the proposed algorithm considering the vehicle arrival and departure rates as 10 and 5 vehicles/s, respectively, with a range of 300 to 2100 vehicles and 10 to 50 FNs in FVNETs. The simulation results are compared with MCF [40], DRO, and SA in terms of the percentage of RBs occupied, the energy consumption of FNs and the resource utilization efficiency. The outcome of simulations shows that the suggested EERA algorithm surpasses when analogized with other existing algorithms. The major contributions of this chapter are listed below.

1. We consider the coordinating RBs of vehicles in the overlap coverage areas of FNs in

FVNETs while meeting the vehicle's desired requirements by relocating RBs among pairs of FNs.

2. The optimal migration of RBs in FVNETs is framed as ILP by assessing the resource parameters of FNs.

3. We propose an EERA algorithm using a B+ tree for optimal RBs migration among FNs for offloading upstream services, such that the energy utilization of FNs in the downlink is reduced and resource usage efficiency is maximized.

4. We demonstrate the simulations that depict the growing number of vehicles arriving on the energy utilization of FNs and the resource usage efficiency of the network. Also, the influence of increasing the number of FNs on the energy utilization of FNs shows the applicability of the proposed algorithm over the existing algorithms.

The rest of this chapter is catalogued as follows. The system model and problem statement are illustrated in Section 6.1. The description of the proposed EERA algorithm and its complexity analysis are explained in Section 6.2. An illustration for EERA is also presented in the same section. Section 6.3 shows the extensive simulation results of the proposed EERA algorithm and its execution with the existing algorithms. Ultimately, Section 6.4 summarises the work.

6.1 System Model and Problem Statement

This section presents the system model and problem statement. The system model of FVNETs contains the FN's energy consumption and communication model for FNs and vehicles in FVNETs.

6.1.1 System Model

Consider a FVNET deployed in a city area A with $\mathcal{G}$ number of FNs. Let the coverage range of FN i, $1 \leq i \leq \mathcal{G}$ be R, and it can overlap with the coverage range of neighbouring FN j, $1 \leq i \leq \mathcal{G}$, $i \neq j$. Vehicles' occurrence time and release time follow a Poisson distribution with a mean occurrence rate λ and leave rate μ. The interarrival time of vehicles follows an exponential distribution. Figure 1.2 shows the downlink medium time of FN, which is split into small time intervals, each with a time fraction of $\delta\tau$. Within each timeslot, the FNs can serve vehicles within their coverage region. FNs assign RBs to the vehicles available in their coverage region to provide services based on the SINR signal strength of those vehicles [77]. However, the strength of the signal depends on the distance between i^{th} FN and k^{th} vehicle, which is calculated in a 2D space by considering FN located at the origin and vehicles at (x_k, y_k) using Euclidean distance $D_{ki} = \sqrt{(x_k^2 + y_k^2)}$. In this system model, we consider the vehicles in overlap coverage parts of FNs and the power consumption of FNs in the downlink communication when RBs are assigned by the FNs. As discussed in Chapter 1, HPNs and RSUs are regarded as FNs since these devices provide communication, storage, computation and wide-area coverage with other FNs and the cloud [24]. The dynamic hierarchical topology is used to organize HPNs, RSUs and vehicles in FVNETs as shown in Figure 6.1.

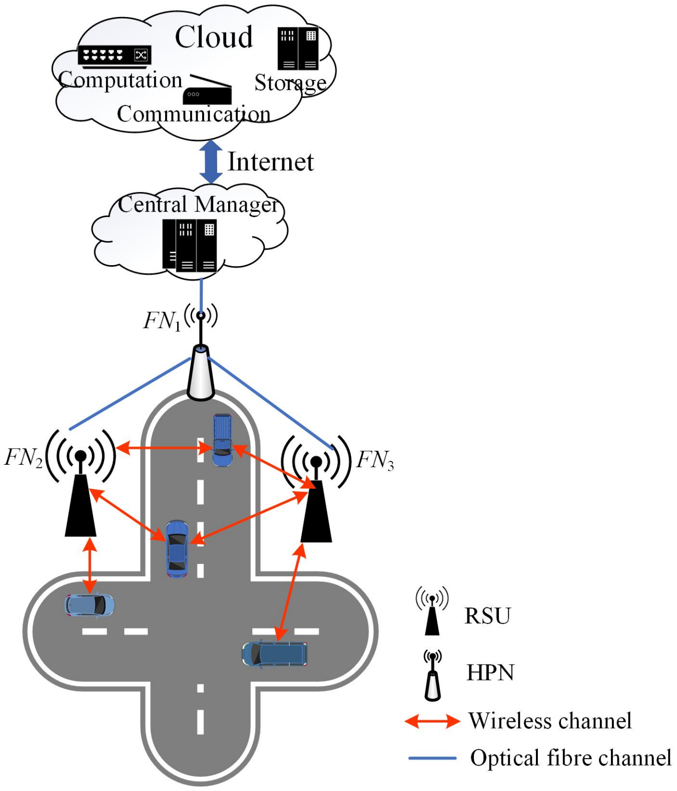


Figure 6.1: Hierarchical organization of HPNs, RSUs and vehicles in FVNETs

6.1.1.1 Communication Model

The FNs use the transmit power control in downlink transmission to achieve the sustained bit rate in each timeslot [85]. The estimated downlink transmission energy cost of FNs can be obtained by assuming the distance-depended exponential radio path loss model [128, 129]. In this model, FN power consumption increases as the distance from FN increases [130]. Hence, the power consumption of FN i, when communicating with vehicle k, assuming a constant bitrate (B bits per each slot) at timeslot t is given by

$$\rho_t(ik) = (D_{ki})^{\beta} \times \frac{B}{\alpha} \tag{6.1}$$

where $\rho_t(ik)$ is the transmission power of i^{th} FN, D_{ki} is the distance between i^{th} FN and k^{th} vehicle at timeslot t, β is a path loss exponent and α is a scaling co-efficient [130]. Then the SINR of k^{th} vehicle from i^{th} FN with Gaussian noise power N_k and co-channel inference power I_k can be obtained using Eq. (5.2) [95]. Consider the total number of RBs required by k^{th} vehicle from the i^{th} FN with data rate r^k be b_k^i. The required RBs by k^{th} vehicle with bandwidth B from i^{th} FN is obtained using Eq. (4.1).

In a FVNET, a vehicle can communicate with one or more FN(s). We assume that the vehicles available in coverage areas overlapped by two or more FNs coverage. In a typical scenario, each vehicle is close to one FN and far from another FN. Eq. (6.1) gives the power consumption by the FN to serve a vehicle. It is regarded that the power consumption by the FN to a nearby vehicle is minimum when analogised to a vehicle at a far distance within the coverage area of that FN [85].

6.1.2 Problem Statement

Let us consider a set of vehicles coming at i^{th} FN be $\mathcal{P}_t^{in}(i)$ and a set of vehicles exiting at i^{th} FN be $\mathcal{P}_t^{out}(i)$, at timeslot t. In this environment, the mean leave rate of vehicles leaving the network, μ, is calculated from Eq. (4.2).

As discussed in Chapter 4.1, $\mathcal{P}_t(i)$, $A_t(i)$ and $\mathcal{P}_t(ij)$ denote the set of vehicles getting services from i^{th} FN, the remaining RBs of i^{th} FN and the set of vehicles in the overlap coverage region of i^{th} FN and its neighbouring j^{th} FN at timeslot t, respectively. These $\mathcal{P}_t(i)$, $A_t(i)$ and $\mathcal{P}_t(ij)$ can be accomplished from Eqs. (4.3), (4.4) and (4.7), respectively.

Let $\mathcal{Z}$ be the set of overlap coverage parts in FVNETs. A vehicle $k \in \mathcal{P}_t(ij)$ can be served by either FN i or FN j in the overlap region $(i, j) \in \mathcal{Z}$. Therefore, we define a binary variable to indicate the vehicle k in the overlap region $(i, j) \in \mathcal{Z}$ is served by FN i.

$$y_i^k = \begin{cases} 1, & \text{If vehicle } k \text{ served by FN } i \\ 0, & \text{Otherwise} \end{cases}$$

$$\sum_{\forall (i,j) \in \mathcal{Z}} (y_i^k + y_j^k) \leq 1, k \in \mathcal{P}_t(ij) \tag{6.2}$$

The Eq. (6.2) ensures that the vehicle k is served by either FN i or FN j from all overlap coverage regions of FVNETs. As vehicles joining the network increases, the number of vehicles gaining services also enriches. This growth in on-demand services directs to the immense energy utilization of FNs in FVNETs. Therefore, the problem is to mitigate the FNs' total energy usage in FVNETs, which is depicted as follows

$$\mathbb{P} : \min \sum_{n=1}^{\mathcal{G}} \rho_t(ik), \forall k \in \mathcal{P}_t(ij) \tag{6.3}$$

subject to

$$\sum_{n=1}^{\mathcal{G}} y_i^k \leq 1, \forall k \in \mathcal{P}_t(ij) \tag{6.4}$$

$$b_k^i \leq A_t(n), 1 \leq i \leq \mathcal{G}, \forall k \in \mathcal{P}_t(ij) \tag{6.5}$$

The objective in Eq. (6.3) indicates the reduction of the total energy cost of FNs in FVNETs considering the resource limitations of the network. The Eq. (6.4) guarantees the vehicle k in the overlap coverage area $(i, j) \in \mathcal{Z}$ is furnished by a single FN. Constraint in Eq. (6.5) ensures the required RBs of vehicle k in the region $(i, j) \in \mathcal{Z}$ should be less than remaining RBs of FN i to acquire utilities from FN i. The increase in vehicles coming to the network drives the vehicles to couple with FNs based on their signal strength. However, as joining vehicles to the network grows, the FNs become futile in providing on-demand services to the vehicles due to finite resource limitations. Consequently, there is a severe reduction in resource utilization efficiency. Alternatively, it also leads to an upsurge in the FNs' energy cost in the network. Therefore, to address this problem and enhance the network's services, the allocated RBs of vehicles in the overlap coverage parts are reduced amid pairs of FNs. In this chapter, we regard the reduction of vehicles' allocated RBs in the overlap areas of FNs by migrating those vehicles' RBs amid pairs of FNs to realize the following goals. (1) Minimizing energy cost of FNs. (2) Maximizing resource utilization efficiency.

6.2 Energy-Efficient Resource Allocation Algorithm

The presented EERA algorithm recedes occupied RBs by harmonizing RBs allocation among FNs. The RBs reduction is accomplished by migrating RBs of vehicles in the overlapping coverage of pairs of FNs. The objective of the EERA algorithm is to degrade the energy consumption cost of FNs and maximize the resource utilization efficiency in the network. The idea of the proposed algorithm is presented as follows: first, the algorithm constructs a B+ tree using occupied RBs of FNs. Further, it chooses the FN f having minimum occupied RBs. Then, it strives to find the vehicle v_{min} with minimum RBs to that FN f. Subsequently, it migrates the RBs of vehicle v_{min} to FN f such that the allocated RBs of FN f is minimized. However, the migration of RBs to the FN f is successful only if the available RBs of the FN f meet the vehicle's v_{min} desired necessities. The proposed algorithm is depicted in Algorithm 6.1 and clarified in Section 6.2.2.

At a particular instant of time t, consider a situation in which vehicles reaching the network increases, and the available resources of the FNs tend to drain. The EERA algorithm identifies the set of vehicles in the overlapping coverage part $(i, j) \in \mathcal{Z}$ of FVNETs. The set of vehicles in the overlapping range of FNs (i.e., $\mathcal{P}_t(ij)$) is obtained using Eq. (4.7). $|\mathcal{P}_t(ij)|$ gives the total vehicles in the overlapping range of i^{th} and j^{th} FNs. Then, the allocated RBs of vehicles in the set $\mathcal{P}_t(ij)$ can be migrated between FNs by assuming that there exists at least one vehicle in the overlapping coverage $(i, j) \in \mathcal{Z}$ of i^{th} and j^{th} FNs, i.e., $\mathcal{P}_t(ij) \neq \emptyset$. Consider a set of vehicles whose RBs can relocate from i^{th} FN to j^{th} FN (denoted as $p_t^*(ij)$) and its converse, i.e., a set of vehicles whose RBs can relocate from j^{th} FN to i^{th} FN (denoted as $l_t^*(mn)$). At timeslot t, the set of vehicles, furnished by i^{th} and j^{th} FNs after RBs are migrated, is denoted as $\mathcal{P}_t^*(i)$ and $\mathcal{P}_t^*(j)$, respectively. The sets, $\mathcal{P}_t^*(i)$ and $\mathcal{P}_t^*(j)$, can be accomplished from Eq. (4.8). The RBs deduction of vehicles in the overlapping coverage of FNs is framed as an ILP problem, which is examined in Section 6.2.1.

6.2.1 Framing ILP for RBs Reduction

The assigned RBs of the vehicles in the overlapping coverage are reduced to solve the objective $\mathbb{P}$ and maximize the resource utilization efficiency. This deduction of RBs is carried out by migrating RBs of the vehicles in $\mathcal{P}_t(ij)$, $\forall(i,j) \in \mathcal{Z}$ between pairs of FNs, such that the allocated RBs to these vehicles from FNs is minimum. The RBs reduction of vehicles in $\mathcal{P}_t(ij)$, $\forall(i,j) \in \mathcal{Z}$, can be mapped as ILP, which is described in Section 4.2.1.

Note that the ILP problem in Eq. 4.9, selecting vehicles from overlapped coverage regions for RBs migration between pairs of FNs, is a SAP problem which is a prominent NP-Hard problem that aims to maximize the profit [127]. The model of SAP is reduced to the model of our ILP problem by drawing students to vehicles in overlapping coverage parts, seminar halls to FNs, and r^{th} seminar hall capacity, B_r, to i^{th} FN capacity (C_n). However, the profit of mapping s^{th} student to r^{th} seminar hall is negated in the mapping of k^{th} vehicle RBs to i^{th} FN. This reduction is achieved in polynomial time. On the contrary, The ILP problem can be cracked in polynomial time when the number of seminar halls is fixed (i.e., the number of FNs ($\mathcal{G}$) is fixed in our problem) [127]. In this context, we propose an EERA algorithm to diminish the energy consumption cost of FNs and enhance the resource utilization efficiency, which is discussed in Section 6.2.2.

6.2.2 Algorithm Description

The EERA algorithm provides a solution for FVNETs to lower the energy usage of FNs and enhance resource utilization efficiency by employing a well-known data structure called the B+ tree. The proposed algorithm, EERA, is demonstrated in Algorithm 6.1.

Algorithm 6.1 Energy-Efficient Resource Allocation

```
Input: $G(V, E)$, $\mathcal{G}$
Output: Energy consumption, resource utilization efficiency
 1: $\mathcal{L}_t \leftarrow \emptyset$
 2: $Q \leftarrow \emptyset$
 3: for $i \leftarrow 1$ to $\mathcal{G}$ do
 4:     $\mathcal{L}_i \leftarrow \emptyset$
 5:     for each neighbour FN $j$ of $i$, $1 \leq j \leq \mathcal{G}$, $i \neq j$ do
 6:         Obtain $\mathcal{P}_t(ij)$
 7:         $\mathcal{L}_i \leftarrow \mathcal{L}_i \cup \mathcal{P}_t(ij)$
 8:     end for
 9:     $\mathcal{L}_t \leftarrow \mathcal{L}_t \cup \mathcal{L}_i$
10:     $O_t(i) \leftarrow \sum_{k=1}^{|\mathcal{P}_t(i)|} b_k^i$
11:     $Q \leftarrow Q \cup O_t(i)$
12: end for
13: energy $\leftarrow$ Compute_Energy($\mathcal{L}_t$, Q)
14: Find resource utilization efficiency
```

Algorithm 6.1 takes a graph $G(V, E)$ of FVNETs and $\mathcal{G}$ as inputs and produces energy consumed by FNs and resource utilization efficiency in FVNETs. It also invokes the procedure `Compute_Energy()` (i.e., Procedure 5) to find the energy utilization of FNs by passing the vehicles set in the overlapping range areas and occupied capacities of FNs as parameters. At time instant t, when vehicles arrive in the overlapping coverage (i, j), $\forall (i, j) \in \mathcal{Z}$ (Lines 3-11), the algorithm finds the $\mathcal{L}_t$, which is a vehicles set in the overlapping range of i^{th} FN by considering each neighbouring FN (say, j^{th} FN), $i \neq j$. Also, it computes the allocated capability $O_t(i)$ of each FN at timeslot t in Line 10. The set Q, which is a union of the allocated capability of each FN, is obtained in Line 11. In Line 13, it invokes Procedure 5 (i.e., `Compute_Energy()`) by passing $\mathcal{L}_t$ and Q as arguments.

The Procedure 5 performs the RBs relocation by finding the FN f and vehicle v_{min}. After finding the FN f and vehicles v_{min}, it strives to relocate the RBs of vehicle v_{min} to FN f. Relocation of RBs of vehicle v_{min} to FN f possible only if the vehicles v_{min} not served by FN f and the required RBs of vehicle v_{min} is satisfied by FN f from Line 6 to Line 15. Otherwise, the vehicle v_{min} is skipped from the RBs relocation in Line 12. After successfully migrating RBs, the allocated capability and available RBs of FN f are updated in Line 9. Subsequently, the energy consumed by FN f in downlink communication is

Procedure 5 Compute_Energy($\mathcal{L}_t$, Q)

Input: A set of vehicles, $\mathcal{L}_t$, in overlapping parts of FNs and set of allocated capacity, Q, of FNs

Output: RBs Migration, FNs energy consumption

```
 1: while $\mathcal{L}_t \neq \emptyset$ do
 2:     energy_used ← 0
 3:     $f$,$v_{min}$ ← FindFNandVehicle($\mathcal{L}_t$, $Q$)
 4:     if $v_{min} = -1$ then
 5:         $Q$.delete($f$)
 6:     else
 7:         if $v_{min}$ is not served by $f$ and $A_t(f) \geq b^f_{v_{min}}$ then
 8:             Migrate the RBs of vehicle $v_{min}$ to FN $f$ from FN $i$
 9:             Update the allocated capacity and remaining RBs of $f^{th}$ and $i^{th}$ FNs
10:             energy_used ← energy_used + $\rho_t(fv_{min})$
11:         else
12:             Skip the vehicle $v_{min}$ from RBs migration
13:         end if
14:         $\mathcal{L}_t \leftarrow \mathcal{L}_t - \{v_{min}\}$
15:     end if
16: end while
17: return energy_used
```

computed in Line 10 and the vehicle v_{min} is withdrawn from the set $\mathcal{L}_t$ in Line 14. The FN f having minimum occupied RBs and vehicle v_{min} with minimum RBs for FN f are obtained by invoking the Procedure 6 in Line 3. If Procedure 5 finds no vehicle to be migrated, then the obtained FN f is deleted from the set of the occupied capability of FN, Q, in Line 5. These Lines from 2 to 15 are iterated until the set $\mathcal{L}_t$ becomes empty. The total energy FNs consumed is returned in Line 17. `FindFNandVehicle()` (i.e. Procedure 6) is invoked from Procedure 5 in Line 3, constructs the B+ Tree data structure with an order of seven keys using the set of the occupied capacity of FNs, Q in Line 1 of Procedure 6. The FN f with minimum occupied capacity is obtained using the operation `GetMinFN()` on B+ Tree in Line 2. Consequently, it finds the set, $rb\_migrate$, with the vehicles in the overlapping coverage part of FN f from Line 3 to Line 9. Subsequently, from Line 10 to Line 17, Procedure 6 finds the vehicle v_{min} having minimum desired RBs for the FN f from the set $rb\_migrate$. Then, it returns the FN f and vehicle v_{min} in Line 18.

Procedure 6 FindFNandVehicle($\mathcal{L}_t$, Q)

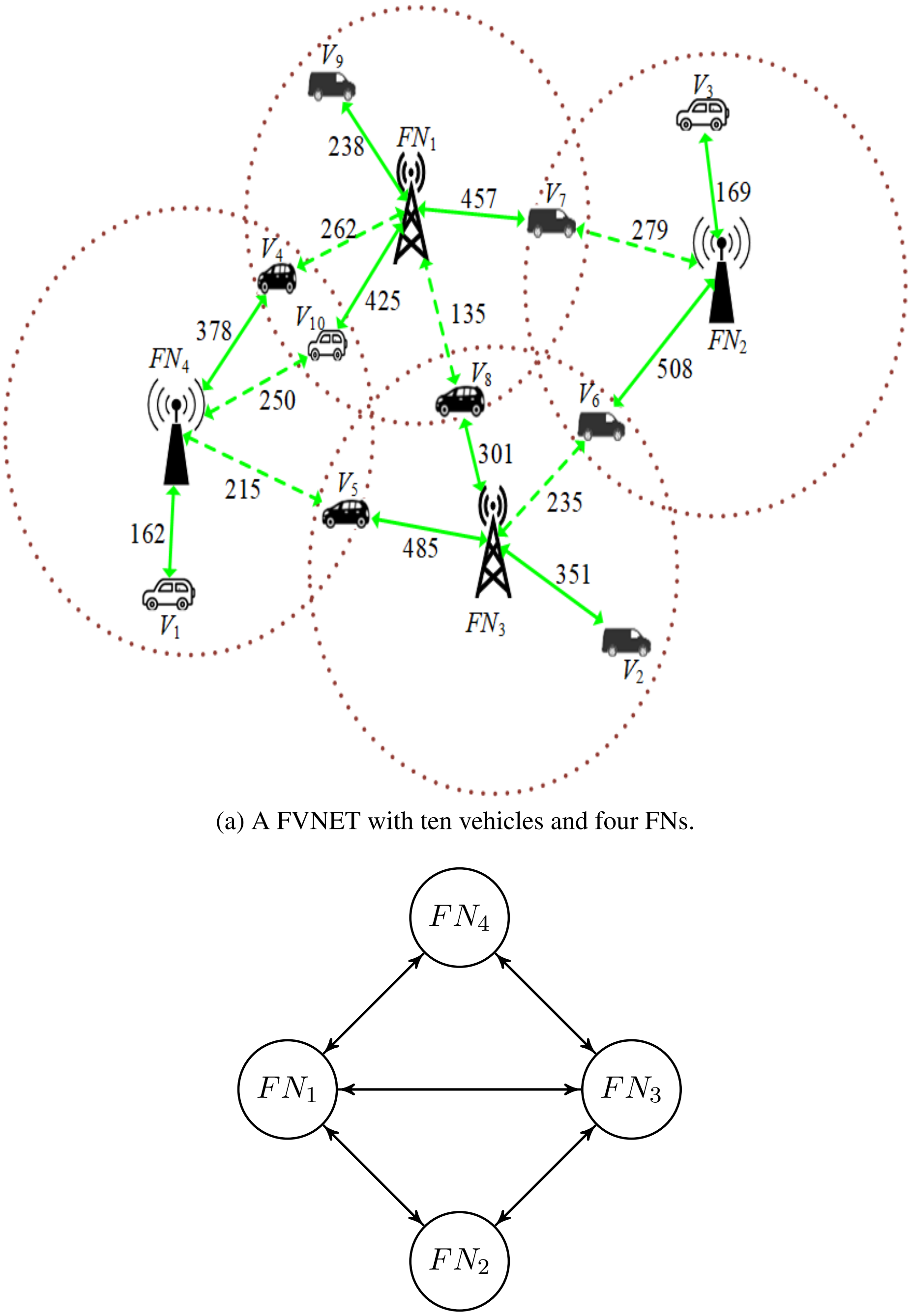
```
$tree \leftarrow$ B+_Tree($Q$)
$f \leftarrow tree$.GetMinFN()
$V\_list \leftarrow \mathcal{P}_t(f)$
$rb\_migrate \leftarrow \emptyset$
for vehicle $v \in V\_list$ do
    if $v \in \mathcal{L}_t$ then
        $rb\_migrate \leftarrow rb\_migrate \cup v$
    end if
end for
$min\_rb \leftarrow max\_number$
$v_{min} \leftarrow -1$
for vehilce $v \in rb\_migrate$ do
    if $b_v^f \leq min\_rb$ then
        $min\_rb \leftarrow b_k^f$
        $v_{min} \leftarrow v$
    end if
end for
return $f$,$v_{min}$
```

6.2.3 An Illustration

Let us consider a FVNET with four FNs and ten vehicles, as shown in Figure 6.2a. The solid green lines between FNs and vehicles indicate that the vehicles are connected to FNs, and the dashed green lines denote that the RBs of those vehicles can be migrated to the corresponding FNs. A number beside the green lines shows the total RBs that must be allocated from the FN to the corresponding vehicle. The graph representation of FVNETs is shown in Figure 6.2b. The vertices designate the FNs, and there exists an edge amid FNs only if vehicles present in the overlap coverage areas of corresponding FNs. The graph model, $G(V, E)$, of FVNETs and the number of FNs, $\mathcal{G}$, are given as input to the EERA algorithm (i.e., Algorithm 6.1). The Algorithm 6.1 finds the vehicles set in overlap coverage regions and occupied capacity of FNs as $\mathcal{L}_t = \{V_4, V_5, V_6, V_7, V_8, V_{10}\}$ and $Q = \{Q_1 = 1120, Q_2 = 677, Q_3 = 1137, Q_4 = 540\}$, respectively. Figure 6.3 shows the occupied capacity of FNs before and after RBs migration in Figure 6.3a and Figure 6.3b, respectively. Subsequently, Algorithm 6.1 invokes the `Compute_Energy()` (i.e., Procedure 5) with the sets $\mathcal{L}_t$ and Q as parameters to compute the energy consumption of FNs. Then the FN f and vehicle v_{min} are obtained by calling `FindFNandVehicle()` (i.e., Procedure 6) in

Procedure 5). In each iteration of Procedure 5, Procedure 6 constructs a B+ tree to find the FN f with minimum occupied capacity. In the first iteration, Procedure 6 returns FN f = FN_4 and v_{min} = V_5 (i.e., 215) for RBs migration. Then, the Procedure 5 migrates the RBs of vehicles v_{min} = V_5 from FN_3 to f = FN_4. Consequently, it removes the vehicle v_{min} = V_5 from the set $\mathcal{L}_t$ and updates the sets $\mathcal{L}_t$ and Q as $\{V_4, V_6, V_7, V_8, V_{10}\}$ and $\{Q_1$ = 1120, Q_2 = 677, Q_3 = 652, Q_4 = 755$\}$, respectively. In the second iteration, Procedure 6 returns FN f = FN_3 by constructing B+ tree and v_{min} = V_6 (i.e., 235). Then Procedure 5 updates the sets $\mathcal{L}_t$ and Q after relocating RBs of vehicle v_{min} = V_6 from FN FN_2 to FN f = FN_3, and it also remove v_{min} = V_6 from the set $\mathcal{L}_t$. The updated sets after the iteration are $\mathcal{L}_t$ = $\{V_4, V_7, V_8, V_{10}\}$ and Q = $\{Q_1$ = 1120, Q_2 = 169, Q_3 = 887, Q_4 = 755$\}$. In the third iteration, Procedure 6 returns FN f = FN_2 using B+ tree and vehicle v_{min} = V_7 (i.e., 279) for RBs relocation between FNs FN_1 and f = FN_2. The allocated RBs of vehicle v_{min} = V_7 are reduced by relocating from FN FN_1 to FN f = FN_2 in Procedure 5. Consequently, after relocation, the updated sets $\mathcal{L}_t$ and Q are $\{V_4, V_8, V_{10}\}$ and $\{Q_1$ = 663, Q_2 = 448, Q_3 = 887, Q_4 = 755$\}$, respectively.

In the fourth iteration, Procedure 6 returns the FN $f = FN_2$ but fails to return vehicle v_{min} for RBs migration. Procedure 6 fails to return vehicle v_{min} when no vehicles are present for RBs migration in the overlap regions of given FN $f = FN_2$. As a result, Procedure 5 deletes FN f = FN_2 from the set Q. In the fifth iteration, Procedure 6 returns the FN $f = FN_1$ from the set Q = $\{Q_1$ = 663, Q_3 = 887, Q_4 = 755$\}$ and it also return vehicle v_{min} = V_8 (i.e., 135) for RBs migration. Then, Procedure 5 migrates RBs for the vehicle v_{min} = V_8 from FN FN_3 to FN f = FN_1 which results in the sets $\mathcal{L}_t$ and Q are $\{V_4, V_{10}\}$ and $\{Q_1$ = 798, Q_3 = 586, Q_4 = 755$\}$, respectively. In the sixth iteration, Procedure 6 returns the FN f = FN_3 but fails to return vehicle v_{min} for RBs migration. Consequently, Procedure 5 deletes FN f = FN_3 from the set Q resulting to Q = $\{Q_1$ = 798, Q_4 = 755$\}$. In the seventh iteration, the B+ tree is constructed using the values of Q and the values for $f = FN_4$ and $v_{min} = V_{10}$ (i.e., 250) are obtained from Procedure 6. Then, the RBs of the vehicle $v_{min} = V_{10}$ is relocated from FN FN_1 to FN f = FN_4. After migration the vehicle $v_{min} = V_{10}$ is removed from the set $\mathcal{L}_t$ and the values of Q are updated as $\{Q_1$ = 373, Q_4 = 1005$\}$. In the eighth iteration, Procedure 5 fetches the values of f = FN_1 and v_{min} = V_4 (i.e.,



(a) A FVNET with ten vehicles and four FNs.

(b) A graph represntation of Figure 6.2a.

Figure 6.2: An FVNET for RBs migration using the proposed algorithm.

262) from Procedure 6. Then it migrates the RBs of the vehicle $v_{min} = V_4$ from FN FN_4 to FN $f = FN_1$ and $v_{min} = V_4$ is deleted from $\mathcal{L}_t$ after successful migration. Subsequently, the updated values of $\mathcal{L}_t$ and Q are $\{\emptyset\}$ and $\{Q_1 = 635, Q_4 = 627\}$, respectively. The allocated capacity of FNs was 3474 before the migration of RBs. After successfully migrating the

	FN_1	FN_2	FN_3	FN_4
V_1				162
V_2			351	
V_3		169		
V_4	262			378
V_5			485	215
V_6		508	235	
V_7	457	279		
V_8	135		301	
V_9	238			
V_{10}	425			250
Occupied RBs	1120	677	1137	540

(a)

	FN_1	FN_2	FN_3	FN_4	
V_1				162	
V_2			351		
V_3		169			
V_9	238			215	
V_5			485	215	
FNs occupied RBs	1120	677	652	755	After 1st iteration
V_6		508	235		
FNs occupied RBs	1120	169	887	755	After 2nd iteration
V_7	457	279			
FNs occupied RBs	663	448	887	755	After 3rd iteration
V_8	135		301		
FNs occupied RBs	798	448	586	755	After 5th iteration
V_{10}	425			250	
FNs occupied RBs	373	448	586	1005	After 7th iteration
V_4	262			378	
FNs occupied RBs	635	448	586	627	After 8th iteration

(b)

Figure 6.3: An optimal RB migration using EERA algorithm for FVNET. (a) Occupied RBs of FNs before migration. (b) Occupied RBs of FNs after migration using EERA.

RBs of all the vehicles in $\mathcal{L}_t$, the total allocated capacity of FNs is reduced to 2296, as shown in Figure 6.3b. Alternatively, the proposed algorithm reduces the allocated RBs up to 33.91%. This RBs reduction minimizes the energy consumption in the downlink services of FNs and enhances resource utilization efficiency. However, existing algorithms, such as RO, DRO, SA, and MCF, reduce the allocated RBs up to 26.71%, 16.23%, 17.87%, and 25.53%, respectively.

6.2.4 Complexity Analysis

Procedure 6 construct the B+ tree with $|Q|$ number of keys. The size of set Q is $|Q| = \mathcal{G}$, hence the construction of B+ tree consume $\mathcal{O}(\log \mathcal{G})$ time in the worst case. Finding the set of vehicles, $rb\_migrate$, from Line 3 to Line 9 and finding the vehicle v_{min} from Line 10 to Line 17 in Procedure 6 takes $\mathcal{O}(|rb\_migrate| + |V\_list|)$ times which is finite amount of time. Therefore, Procedure 6 completed $\mathcal{O}(\log \mathcal{G})$ time in the worst case complexity. Whereas in Procedure 5, relocating RBs of vehicle v_{min} from Line 6 to Line 15 takes a constant amount of time. Also, it invokes the Procedure 6 in Line 3. However, Lines from Line 1 to Line 16 are repeated $\mathcal{O}(|\mathcal{L}_t|)$ times. Hence, the general time complexity of Procedure 5 is $\mathcal{O}(|\mathcal{L}_t| \log \mathcal{G})$. The Lines from 3 to Line 12 in Algorithm 6.1 iterates for $\mathcal{O}(\mathcal{G}^2)$ times in the worst case. Consequently, the comprehensive time complexity of the EERA algorithm takes $\mathcal{O}(\mathcal{G}^2 + |\mathcal{L}_t| \log \mathcal{G})$ time complexity in the worst case.

6.3 Performance Evaluation

The proposed EERA algorithm performance is evaluated by considering the energy usage of FNs and the resource utilization efficiency of the network. The simulation setup is explained in Section 6.3.1. The simulation results are analogized with existing algorithms, such as RO, MCF [40], DRO [24] and SA [25]. We demonstrated the influence of the growth in vehicles arriving at the network on the energy consumption of FNs and resource utilization efficiency of the network. Further, the influence of the increase in the number of FNs on the energy consumption of FNs is also demonstrated.

Table 6.1: Parameters and their values for simulations

Parameter	**Value**
Network area (A)	5000 m × 5000 m
Number of FNs $(\mathcal{G})$	[10 $\sim$ 50]
Radius of FN coverage (R)	500 m
Bandwidth of FNs (W)	[10, 15, 20] MHz
Vehicles arrival rate (λ)	10 vehicles/s
Vehicles departure rate (μ)	5 vehicles/s
Required data rate	[0.5 $\sim$ 2] Mbps
Pathloss exponent (β)	3
Scaling co-efficient (α)	1
Gaussian Noise (N_k)	-104 dBm
Co-channel Inference (I_k)	-75 dBm
Time slot duration	1 s

6.3.1 Simulation Setup

The proposed algorithm considers a FVNET consisting of energy-limited FNs ranging from 10 to 50. The network is deployed in a city area of [5000 × 5000] m^2. The vehicles joining and leaving the network follow the Poisson process with a mean coming rate λ = 10 vehicles/s and mean leave rate μ = 5 vehicles/s. The vehicles served with the minimum 0.5 Mbps of data rate when FNs could not assign the demanded RBs. The energy cost of FNs is readily available in [128, 129]. The energy cost is obtained from the distance-dependent path-loss model using the path-loss exponent $\beta = 3$. The FN estimates the energy usage cost in the downlink transmission, assuming the constant bit rate. The media access control protocol, IEEE 802.11p, provides wireless connectivity in FVNETs. The Monte-Carlo simulations are conducted with 210 timeslots. The simulations are conducted by creating an environment in PyCharm IDE 2023.1.2 using Python 3.11, Intel(R) Xeon(R) Gold 622R CPU @ 2.90 GHz processor, and 64.0 GB RAM. This IDE runs on the 64-bit Windows operating system. The average results are obtained over ninety simulations. The additional parameters assumed for the simulations are specified in Table 6.1.

6.3.2 Results and Discussions

The literature in [24, 25] proposes various resource management algorithms, including DRO, and SA, respectively. The literature in [40] proposes an energy-efficient scheduler, namely MCF. The RO algorithm selects the vehicles randomly for RBs migration from the FN picked randomly. The DRO algorithm determines the maximum weight-matching solution to migrate RBs of vehicles amid pairs of FNs. In contrast, the SA finds the graph colouring solution to relocate the RBs of vehicles among pairs of FNs. The MCF is an energy-efficient scheduler that minimizes the FN's total energy consumption while providing the vehicles' services in overlapped coverage regions in each time slot. We analyze the efficacy of EERA with the performance of RO, MCF, DRO, and SA algorithms. The effectiveness of the EERA is illustrated in terms of the energy usage of FNs and resource usage efficiency when vehicles coming to the network rise. The energy usage per time slot is defined as the average energy usage of FN to provide services to the vehicle with B bits of data. Similarly, resource utilization efficiency is the rate of allocated RBs in the network. Moreover, the performance is also measured in terms of the average energy usage of FNs when the number of FNs increases in the network.

6.3.3 Energy Consumption of FNs

The mean coming (leaving) rate implies the number of vehicles entering (leaving) the network. In our case, the mean coming rate is greater than the mean departure rate. As a result, the number of vehicles connecting to the network grows in all time slots. The assigned RBs of FNs increase as the vehicles connected to FNs grow. Therefore, the allocated RBs of vehicles in overlapping coverage are migrated to minimise the overall occupied RBs of FNs. Figure 6.4 shows the influence of the growth in the vehicles joining the network on occupied RBs of the network after lessening allotted RBs of vehicles in the overlap coverage parts for the EERA and existing algorithms. To reduce the allocated RBs, the DRO employs the maximum weight-matching solution, which returns the set of edges with no shared vertices. Thus, only the vehicles in the matching edges are used to reduce allocated RBs. Therefore, the allocated RBs after reduction using DRO is high and increases as ve-

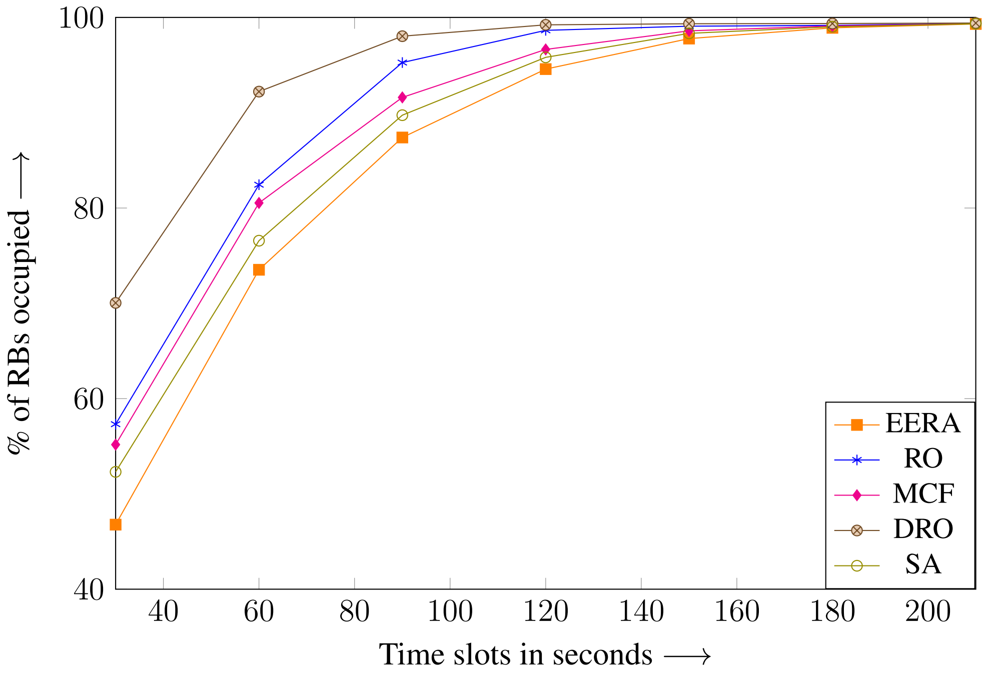


Figure 6.4: Total allocated RBs for EERA, RO, MCF, DRO, and SA algorithms after RBs reduction.

hicles increase in the network. The algorithms (i.e., RO, MCF, and SA) perform better in reducing the RBs of vehicles in overlapping parts of FNs as the increase of vehicles joining the network. The SA and MCF algorithms reduce the RBs of vehicles in overlapping coverage parts using the graph colouring solution and by selecting vehicles with minimum RBs at each time slot, respectively. The SA strives to reduce the RBs of vehicles from overlap regions such that the burden on FNs is low. Therefore, the SA algorithm enhances the lowering of RBs as vehicles increase in the network by 3.63%, 1.91%, and 7.94% on average when analogised to RO, MCF and DRO, respectively. However, the EERA algorithm, which builds the B+ tree using occupied capacities of FNs, reduces the RBs and enhances the occupied RBs significantly as vehicles increase in the network by 2.73% on average compared to the SA algorithm.

An increase in the vehicles joining the network increases the bandwidth consumption, resulting in competing among vehicles for finite bandwidth [132]. As a result, some vehicles with good signals strive to utilize FNs heavily, depriving services to other vehicles. However, this poses a significant challenge in terms of tremendous energy usage by FNs to

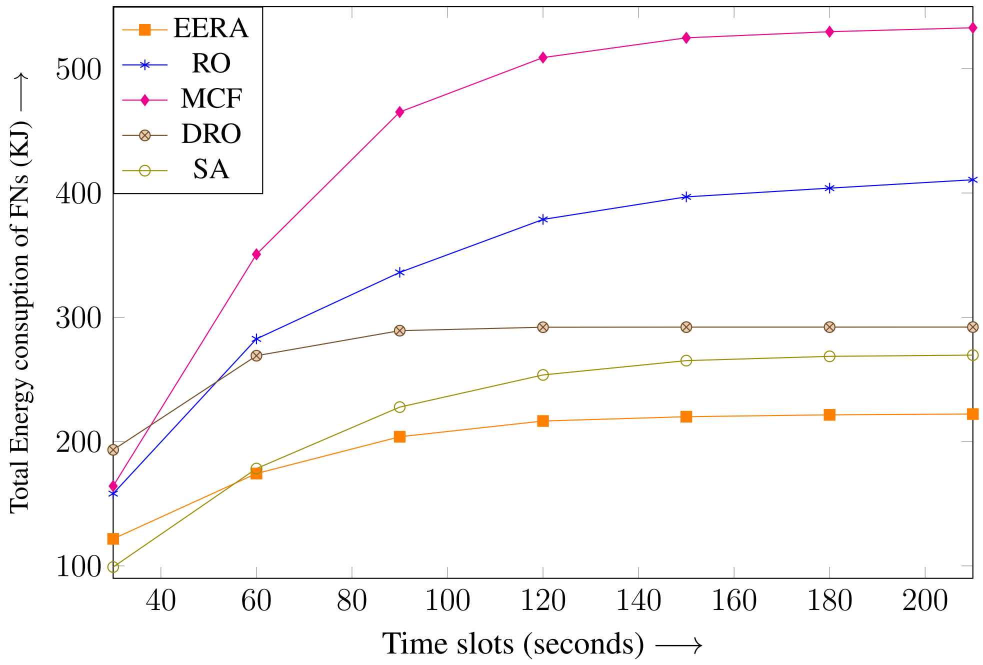


Figure 6.5: Energy consumption of FNs per time slot for EERA, RO, MCF, DRO, and SA algorithms.

provide continuous utilities to the vehicles when limited energy sources power FNs. Once the RBs reduction is successful for vehicles in the overlapping coverage parts, FNs are ready to deliver the data. The proposed algorithm selects vehicles nearer to the FNs (i.e., vehicles with the least RBs) by relocating the RBs of vehicles in the overlapping coverage parts among pairs of FNs. As a result, there is a reduction in the energy consumption of the FNs. Figure 6.5 shows the energy consumption by FNs as the vehicles joining the network grow. When analogized with other algorithms, the proposed algorithm remarkably lessens the FNs' energy usage in downlink communication. Here, the SA performs better because it uses the graph colouring solution to reduce the RBs of vehicles in the overlapping regions in each time slot, and it improves the reduction in the energy consumption by the FNs up to 34.37%, 48.07% and 22.31% on average when analogized to RO, MCF and DRO algorithms, respectively. However, the proposed EERA algorithm reduces the energy consumption of FNs. It outperforms the SA as it is greedy in selecting FN with minimum occupied capacity and greedy in deciding the vehicle with the least RBs. Therefore, it reduces the energy consumption of FNs by 6.62% on average when analogized to the SA algorithm.

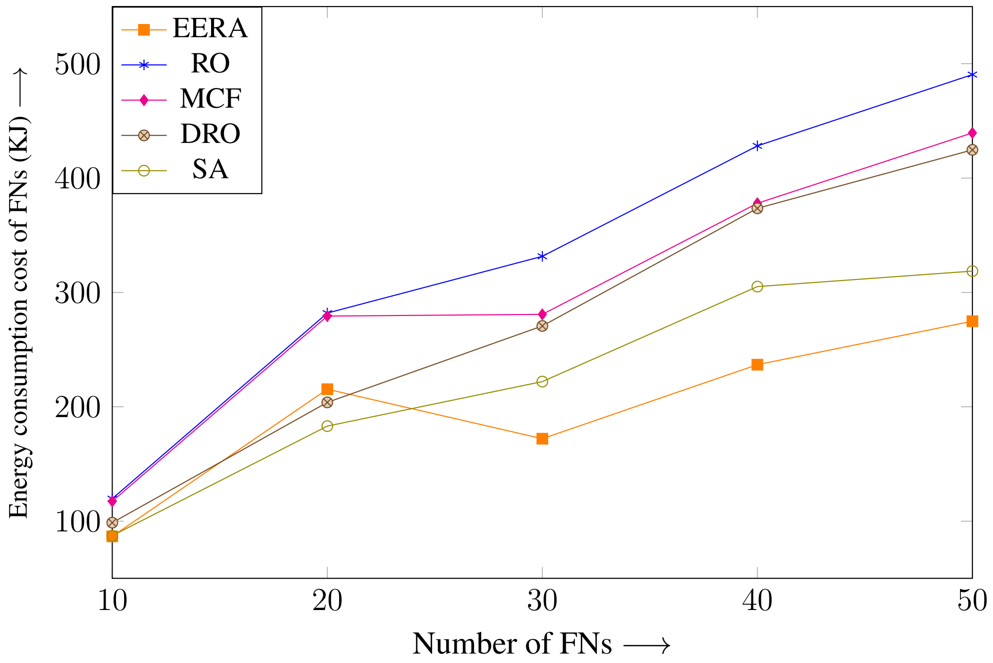


Figure 6.6: Energy consumption cost of FNs as FNs in FVNETs increase for EERA, RO, MCF, DRO, and SA algorithms.

The energy consumption of FNs as FNs in the FVNETs increase is pictured in Figure 6.6. As deploying the number of FNs in the network rises, more FNs are available to provide services as vehicles joining the network grow. MCF, DRO, and SA algorithms reduce FNs' energy consumption as the FNs increase. The SA algorithm reduces FNs' energy consumption by FNs by 31.78%, 25.55% and 16.59% on average when analogized to RO, MCF and DRO algorithms, respectively. Nevertheless, there is a notable reduction in the FN energy utilization using the EERA algorithm. The EERA algorithm diminishes the energy consumption of FNs by 37.58%, 32.51%, 22.95%, and 8.33% on average analogized to RO, MCF, DRO, and SA algorithms, respectively.

6.3.4 Resource Utilization Efficiency

The percentage of assigned RBs in the network indicates resource usage efficiency. The distribution of the percentage of average RBs occupied for the EERA algorithm and existing algorithms is portrayed in Figure 6.7. The black diamond mark represents the average percentage of RBs used. The blue line inside the box signifies the median, and the

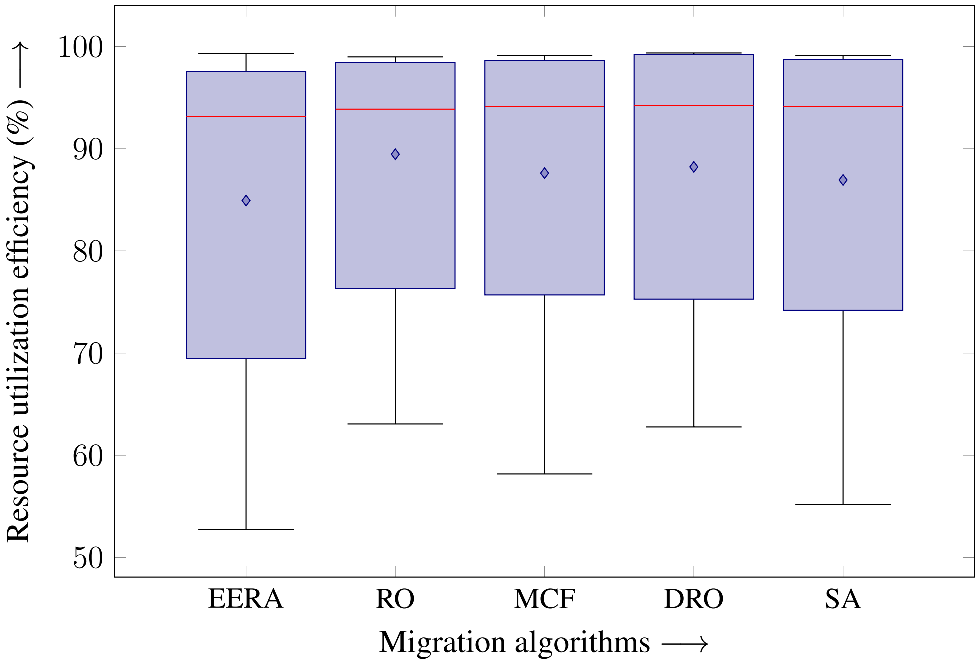


Figure 6.7: Resource utilization efficiency for EERA, RO, MCF, DRO, and SA algorithms.

boxplot's height denotes the distribution of assigned RBs of FNs in FVNETs. The FNs become ineffective when vehicles joining the network increase due to inadequate available RBs. Consequently, the algorithms RO, MCF, DRO and SA deliver an average of 89.46%, 87.61%, 88.22%, and 86.94% of RBs utilization in the network, respectively. In contrast, the proposed algorithm is greedy in deciding FNs with the least occupied capacity and reduces the allocated RBs by selecting vehicles with minimum RBs in the overlap regions. Therefore, the EERA algorithm exhibits an average of 84.92% RBs utilization. The reduced allocated RBs from FNs are reused for the vehicles in overlap regions. Therefore, the simulation results depict that the EERA algorithm enhances the efficiency of RBs utilization by 4.21%, 3.15%, 3.87%, and 2.37% when analogue to RO, MCF, DRO and SA algorithms, respectively.

6.4 Summary

In this chapter, we proposed an EERA algorithm to coordinate RBs allocation among FNs to minimize the energy usage of FNs and augment the resource usage efficiency of the network. The allocated RBs of vehicles are lessened by relocating RBs among pairs of FNs so that the occupied capacity of FNs is minimized. We have considered the vehicles in the overlapping coverage parts of FNs to migrate the allocated RBs of vehicles. However, as vehicles joining the network grow, FNs become impuissant to furnish services to the vehicles due to limited resources and pose a challenge in terms of massive energy consumption by FNs for continuously serving vehicles. Therefore, the EERA algorithm reduces assigned RBs of vehicles in overlapping regions by relocating RBs among pairs of FNs. Consequently, the FNs' energy consumption is minimized with the growing number of vehicles joining the network. Further, we have analyzed FNs' energy consumption as the number of FNs in the network grows. The proposed algorithm has been evaluated with existing algorithms, namely RO, MCF, DRO, and SA. The simulation outcomes illustrate that the proposed algorithm outruns the energy consumption of FNs by 37.58%, 32.51%, 22.95% and 8.33% on average compared to RO, MCF, DRO and SA algorithms, respectively.

Chapter 7

Energy-Efficient and Delay-Aware Task Scheduling in Energy-Limited FVNETs using Q-Learning

This chapter presents an RL-based EEDA task scheduling algorithm in intersecting regions of energy-limited FVNETs. This algorithm mitigates the energy consumption of FNs while discharging and improving the network's throughput by satisfying the delay constraints of the tasks. EEDA is a greedy-based RL algorithm that provides a sub-optimal solution to task scheduling in the intersecting regions of FVNETs. It uses the Q-learning approach to train FNs for different vehicle arrival rates of traffic scenarios. It chooses the FN, which consumes minimum energy to schedule tasks in each time slot while meeting the deadline of tasks. Further, the selection of FNs for scheduling tasks depends on the sojourn time of the vehicle in the intersecting region, the task deadline, the response time from the FN and the average energy usage of the FN. The literature [10, 108, 133] considers various objectives like latency, waiting time, FN energy usage, reliability, etc. Moreover, these works need to adequately discuss the challenges in the finite resource FVNETs when the coverage region of FNs intersects with neighbouring FNs. In contrast, the proposed algorithm considers the task deadline, task data size, energy usage of FNs, and vehicle's sojourn time in the intersecting region for scheduling tasks in a suitable FN of an intersecting region. Subsequently, the FNs are trained for various traffic conditions such that the average energy usage of FNs is reduced, and the network's throughput is improved.

The proposed algorithm is simulated for the mean arrival rate of vehicles ranging from 2 to 6 per second in each intersecting region to train the FNs for free-flow traffic of 100 seconds duration. Subsequently, simulated results compared with benchmark algorithms, such as PSG [41], EDF [42], FCFS [43], and RO [44] in terms of average energy usage, network throughput, total transmitted data, completed service request and total service time of FN. The simulation results show that the RL-based EEDA algorithm performs better than benchmark algorithms in minimizing the average energy consumption of FN and enhancing the network throughput. The main contributions of this chapter can be summarized as follows.

1. This chapter investigates the scheduling of time-critical tasks among FNs in the intersecting regions to reduce the average energy usage of FN while considering the task's deadline, FN's energy usage and vehicle's sojourn time in the intersecting region.

2. The scheduling of tasks among FNs is transformed into an ILP by evaluating the task's deadline, response time from the FN and the residing time of the vehicle in the intersecting region.

3. Since the ILP is NP-hard, we design a greedy-based task scheduling in the intersecting regions of FVNETs using the Q-learning-based RL technique to mitigate the energy consumption of FN and satisfy the delay constraints of vehicles.

4. Performance evaluation has been carried out by considering the impact of the increase in the number of vehicles in the intersecting regions for various vehicle arrival rates in the network concerning the energy usage of FN and throughput of the network.

The remainder of this chapter is structured as follows. The system model and the problem formulation are imparted in Section 7.1. Next, Section 7.2 describes the energy-efficient task scheduling approach using the RL technique. Then, the simulation setup and performance evaluation are explained in Section 7.3. Finally, Section 7.4 summarizes the chapter.

7.1 System Model and Problem Formulation

The architecture of a FVNET is shown in Figure 1.1 in Chapter 1. Each smart vehicle is fitted with OBUs for storage and processing resources in ITS. In matters of famine of resources, HPNs and RSUs are employed along the road to supply vehicular services. These HPNs and RSUs are roadside infrastructures regarded as FNs [24]. The central component is connected to the cloud over the Internet and manages the HPNs. HPNs have more resource capabilities than RSUs, and they coordinate the RSUs nearby. Further, the 5G technology's key feature, C-V2X, which supports V2I and I2V communications, is used for connection between vehicles and FNs [28]. Figure 1.2 shows the network's total serviceable time, T, is partitioned into equal timeslots $\delta\tau$. In each timeslot, the FNs are available for vehicular services to one or more vehicles [25].

This chapter assumes that the FNs are powered by rechargeable batteries, which require periodic recharging due to the lack of a permanent power source. Hence, human intervention or solar energy is required to revive FNs when their batteries are depleted to ensure uninterrupted services. This reviving FNs to deliver services and ensure energy preservation till the successive recharge cycle is challenging when the FNs cover highway segments of remote areas (i.e., forests or hill terrain or military bases or airports) where consistent power sources are unavailable [38]. In this circumstance, consider that the FVNETs cover remote areas' highway segments such that the FNs' coverage region intersects with neighbouring FNs' coverage. Further, the dealy-sensitive tasks generated by each arrived vehicle in the intersecting region are offloaded into FN (i.e., RSU or HPN) for computation. However, HPN acts as a central node, which decides the scheduling of tasks among FNs of intersecting regions, including itself. It is worth mentioning that the RSUs are exclusively operated for executing the allotted tasks. The increase in arrived vehicles in the intersecting region rapidly generates compute-intensive and delay-sensitive tasks in the network, causing increases in the processing of tasks at FNs [37]. It may lead to high energy consumption by FNs and influence the end-user experience without proper management. In contrast, efficient energy utilization in limited energy FVNETs enhances the network's throughput and lifetime. Therefore, we propose a RL based EEDA task scheduling in intersecting regions of energy-limited FVNETs to mitigate energy consumption of FNs while discharging and improving the network's throughput by satisfying delay constraints of the delay-sensitive tasks.

EEDA is a greedy-based RL algorithm which provides a sub-optimal solution to task scheduling in the intersecting regions of FVNETs. It uses the Q-learning approach to train FNs for different vehicle arrival rates of traffic scenarios. It chooses the FN, which consumes minimum energy to schedule tasks in each time slot while meeting the deadline of tasks. Further, the selection of FNs for scheduling tasks depends on the sojourn time of the vehicle in the intersecting region, the task deadline, the response time from the FN and the average energy usage of the FN. The system model, communication model, execution model, energy consumption model and formulation of energy consumption of FNs into an ILP problem are described in the following sections.

7.1.1 System Model

Consider $\mathcal{G}$ as the number of energy-limited FNs deployed near the highway forming the FVNET such that their coverage area intersects with neighbouring FNs as shown in Figure 7.1. Assume each FN i, $1 \leq i \leq \mathcal{G}$, has the coverage range R and the $\mathcal{Z}$ number of intersecting regions in FVNETs. Then the neighbouring FN i and FN j form an intersecting region denoted by $(i, j) \in \mathcal{Z}$. Further, a free flow discrete time traffic model is considered using uninterrupted and homogeneous vehicular traffic over a fixed-length highway segment of L, $L \geq R$. The total serviceable time, T, of the network is partitioned into fixed-duration time slots, $\delta\tau$. FNs can provide services to one or more vehicles in every time slot. However, the Poisson process is used for vehicles' mean arrival at the network, and the interarrival time of vehicles follows the exponential distribution for free-flow traffic conditions. Let $V_t(i)$ and $T_t(i)$ be the set of vehicles and set of vehicle's respective tasks in the vicinity of FN i, respectively, at time slot t. Then the set of vehicles and tasks in the

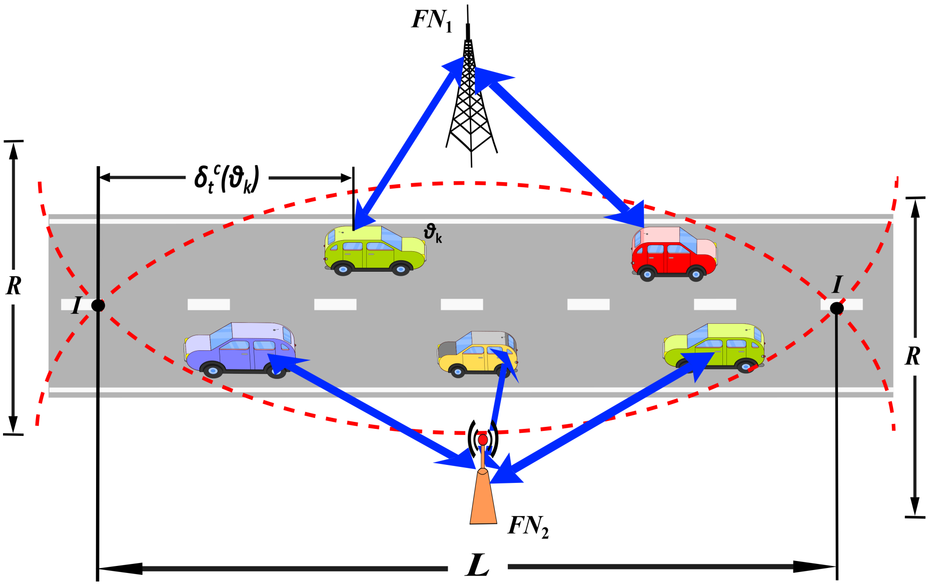


Figure 7.1: Energy limited FNs and vehicles in FVNETs.

intersecting region of FNs i and j at time slot t is obtained as follows.

$$V_t(ij) = V_t(i) \cap V_t(j) \tag{7.1}$$

$$T_t(ij) = T_t(i) \cap T_t(j) \tag{7.2}$$

In every time slot, the HPN accumulates the set of independent tasks $T_t(ij)$ received from corresponding the set of vehicles $V_t(ij)$ in the intersecting region $(i, j) \in \mathcal{Z}$. Each task k is denoted by a tree-tuple (c_k, h_k, d_k). Here, c_k denotes the required CPU cycles in a million instructions per second (MIPS) of task k, h_k describes the data size (in bytes), and d_k gives the task's deadline k. Table 7.1 summarises the notations and their description. In each time slot, vehicles in the intersecting region send the service request to FN. Subsequently, FN forwards the set of tasks to HPN to schedule the tasks for processing. HPN schedules the tasks based on deadline constraints of tasks, the sojourn time of the vehicle in the intersecting region and the energy consumption of FNs. Consequently, FNs process the tasks to provide services in the downlink to the vehicles.

Table 7.1: List of notations and their description

Notation	Description
$V_t(ij)$	Set of vehicles in the intersecting region (i, j) at time slot $t \in \mathcal{Z}$
c_k	CPU cycles (in MIPS) required by task k
F_i	CPU cycles of FN i
h_k	Data size (in bytes) of task k
$T_t(ij)$	Set of tasks in the intersection part $(i, j) \in \mathcal{Z}$ at time slot t
d_k	Deadline of task k
$\mathcal{S}_{ij}^k$	Sojourn time of the vehicle ϑ_k in the intersecting region $(i, j) \in \mathcal{Z}$
L	Length of the intersecting coverage parts of FVNETs
v_k	Velocity of vehicle $\vartheta_k \in V_t(ij)$
$\delta_{t_2}^c(\vartheta_k)$	Covered distance of vehicle ϑ_k in an intersecting region at time slot t_2
χ_{ki}	Transmission rate of FN i for the task k
T_{ki}^{tr}	Transmission time of task k from FN i
T_{ki}^e	Execution time of task k from FN i
T_{ki}^r	Response time of FN i for the task k
E_{ki}^e	FN i energy consumption in executing task k
E_{ki}^{tr}	FN i energy consumption in transmitting data h_k
$E_{ki}(t)$	Energy consumed by FN i in serving vehicle ϑ_k at time slot t
$E_{ij}(t)$	Energy consumed by FNs i and j in serving all the tasks in an intersecting region $(i, j) \in \mathcal{Z}$ at time slot t
$E(t)$	Energy consumed in FVNETs at time slot t

7.1.2 Communication Model

We presumed that the vehicles in the FNs' intersecting region maintain a constant velocity throughout their sojourn time. Every vehicle at time slot t enters the intersecting region at location I and sends the beacon message to FN to establish a connection. A beacon message from vehicle ϑ_k describes the vehicle's arrival time and the velocity at location I of an intersecting region. Subsequently, we assess the adequate communication time of each vehicle in the intersecting region as follows.

Let ϑ_k be a vehicle with the velocity v_k entering the intersecting region from the point of notice I at time slot t_1. The vehicle ϑ_k will send the service request to the FN at time slot t_2 after covering some distance, $\delta^c_{t_2}(\vartheta_k)$, in the intersecting region $(i, j) \in \mathcal{Z}$. As a result, the distance covered by the vehicle ϑ_k is obtained in the duration between t_1 and t_2 as follows.

$$\delta^c_{t_2}(\vartheta_k) = v_k \times \Delta t \tag{7.3}$$

where Δt = $|t_2$ - $t_1|$, is the difference between when the vehicle arrived at location I and when the vehicle sent a service request to FN.

The adequate communication time of the vehicle ϑ_k depends on the vehicle's sojourn time in the intersecting region. The sojourn time, $\mathcal{S}^k_{ij}$, of vehicle ϑ_k is detailed as the amount of time the vehicle ϑ_k resides in the intersecting region (i, j) of FNs i and j. Thus, the sojourn time is expressed as follows.

$$\mathcal{S}^k_{ij} = min\left(\frac{L}{v_k}, \frac{L - \delta^c_{t_2}(\vartheta_k)}{v_k}\right) \tag{7.4}$$

The FN attain a steady bit rate using the transmit power control in downlink transmission in every time slot, regardless of the position of the vehicle in the intersecting region [85]. Further, this system assumes the distant-dependent path loss communication model [32, 40] to fetch the transmit power of a FN. In this model, the transmit power of a FN increases as the vehicle distance from FN increases [130]. Therefore, considering a steady bit rate (B bits per each slot), the transmission power of FN i when contacting with vehicle ϑ_k at time slot t is denoted by $\rho_t(ik)$ and expressed using Eq. (5.1). The distance between FN and the vehicle is obtained by considering the FN i is located at the origin (0, 0). The vehicle ϑ_k is at (x_k, y_k) in the 2D coordinate space using Euclidean distance as D_{ki} = $\sqrt{(x_k^2 + y_k^2)}$ [32]. Then, the SINR between FN i and vehicle ϑ_k at time slot t on the data channel, which is denoted by $\psi_t(ik)$, is derived from Eq. (5.2). Subsequently, the transmission rate of a FN i is obtained for the task k using the Shannon formula, and it is represented as follows [95].

$$\chi_{ki} = B \times \log_2(1 + \psi_t(ik)) \tag{7.5}$$

where B is the available bandwidth at each FN.

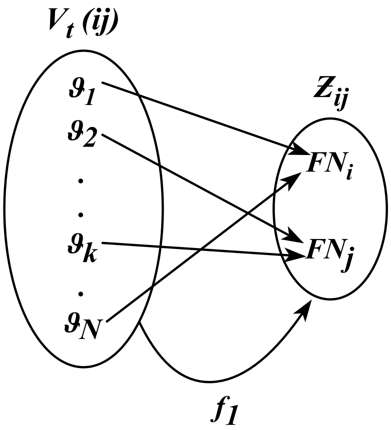


Figure 7.2: Service request from vehicles to FNs is surjective.

Theorem 7.1.1. *The service request from the set of vehicles, $V_t(ij)$, to the FNs, $\mathcal{G}_{ij}$, at time slot t in the intersecting regions, $\forall(i,j) \in \mathcal{Z}$, is surjective.*

Proof: Let $V_t(ij)$ be the set of vehicles at time slot t and $\mathcal{G}_{ij}$ be the set of FNs in an intersecting region $(i,j) \in \mathcal{Z}$. Then the function $f_1 : V_t(ij) \rightarrow \mathcal{G}_{ij}$ describes the service request to FNs from the vehicles in the intersecting region $(i,j) \in \mathcal{Z}$ as shown in Figure 7.2. The function f_1 is injective only if no two vehicles send a request to the same FN. However, in function f_1, two different vehicles, ϑ_k and ϑ_l, can send the request to the same FN. In contrast, every FN in $\mathcal{G}_{ij}$ has a service request from vehicles in $V_t(ij)$ (i.e., for every image in $\mathcal{G}_{ij}$ there is a preimage in $V_t(ij)$). Therefore, the service request from vehicles to FNs is surjective, not injective.

7.1.3 Execution Model

Let λ be the mean arrival rate of vehicles in the network and the set of vehicles at time slot t, $V_t(ij)$, in the intersecting region having N number of vehicles is expressed as $V_t(ij)$ = $\{\vartheta_1, \vartheta_2, \ldots, \vartheta_N\}$. Further, each vehicle ϑ_k, $1 \leq k \leq N$, offloads a delay-sensitive task to a FN in the intersecting region to exploit the network's services. However, it is worth noting that the time to transmit the task k from vehicle ϑ_k to FN i is not considered. Therefore, the execution time of task k in FN i is given as [134].

$$T_{ki}^{e} = \frac{c_k}{F_i} \tag{7.6}$$

where F_i denotes the CPU frequency of FN i. The transmission time to transmit the data h_k in the downlink communication from FN i to vehicle ϑ_k is given as

$$T_{ki}^{tr} = \frac{h_k}{\chi_{ki}} \tag{7.7}$$

Consequently, the response time of FN i for offloading task k to the corresponding vehicle ϑ_k is given as

$$T_{ki}^{r} = T_{ki}^{e} + T_{ki}^{tr} \tag{7.8}$$

7.1.4 Energy Consumption Model

Two kinds of energy consumption of a FN exist in this system model. Firstly, energy consumed by FN in processing the scheduled task k. Secondly, the energy consumed by FN in transmitting data h_k of task k to vehicle ϑ_k in downlink communication. The energy consumption of FN i for processing the task k of a vehicle in the intersection region depends on various factors such as clock frequency and task type. Along with existing studies, [32, 134, 135], we consider the energy consumption of FN i is given as

$$E_{ki}^{e} = \mathcal{E} \times (F_k)^2 \times c_k \tag{7.9}$$

where $\mathcal{E}$ is the energy coefficient depending on the chip architecture. Besides, after execution of task k, the energy consumed by FN i for transmitting the data h_k to vehicle ϑ_k in the intersecting region is given as

$$E_{ki}^{tr} = T_{ki}^{tr} \times \rho_t(ik) \tag{7.10}$$

Therefore, the energy consumed by FN i for providing services to the vehicle ϑ_k at time slot t is defined as

$$E_{ki}(t) = E^{e}_{ki} + E^{tr}_{ki} \tag{7.11}$$

7.1.5 Problem Formulation

We define two boolean variables, y_i^k, and y_j^k to indicate the task k in an intersecting region $(i, j) \in \mathcal{Z}$ that is selected for processing in one of the FNs i and j.

$$y_i^k = \begin{cases} 1, & \text{If the task } k \text{ is processed by FN } i \\ 0, & \text{Otherwise} \end{cases}$$

and

$$y_j^k = \begin{cases} 1, & \text{If the task } k \text{ is processed by FN } j \\ 0, & \text{Otherwise} \end{cases}$$

$$(y_i^k + y_j^k) \leq 1, \forall k \in T_t(ij), (i, j) \in \mathcal{Z} \tag{7.12}$$

The Eq. (7.12) ensures that the tasks in an intersecting region are processed by either FN i or FN j of an intersecting region $(i, j) \in \mathcal{Z}$ in FVNETs. Therefore, each task in an intersecting region is scheduled for processing in a suitable FN. The total energy consumed by FNs i and j in an intersecting region $(i, j) \in \mathcal{Z}$ for processing N number of tasks at time slot t is obtained as follows.

$$E_{ij}(t) = \sum_{k=1}^{N} 1_{\{y_i^k\}} E_{ki}(t) + \sum_{k=1}^{N} 1_{\{y_j^k\}} E_{kj}(t) \tag{7.13}$$

where $1_{\{X\}}$ is an index function that results the value 1 when X is assigned with value 1, otherwise 0. Besides, the total energy consumed in the FVNETs at time slot t is obtained as follows

$$E(t) = \sum_{\forall (i,j) \in \mathcal{Z}} E_{ij}(t) \tag{7.14}$$

The vehicles in the intersecting region of FNs can get services from either of the FNs of the intersecting region. On the other hand, a FN furnishes services to a vehicle in the downlink transmission in the practical systems. Moreover, service requests to the network increase as vehicles enter the network rises over time. This increase in the service connections yields an increase in the energy consumption of FNs for processing and downlink offloading tasks. Accordingly, the problem is stated as reducing the energy consumption of FNs formulated as an ILP problem, expressed as follows.

$$\mathbb{P} : min \sum_{\forall t \in T} E(t) \tag{7.15}$$

such that

$$T^r_{ki} y^k_i + T^r_{ki} y^k_j \leq d_k, \forall k \in T_t(ij), (i,j) \in \mathcal{Z} \tag{7.16}$$

$$(y^k_i + y^k_j)\mathcal{S}^k_{ij} \geq d_k, \forall \vartheta_k \in V_t(ij), (i,j) \in \mathcal{Z} \tag{7.17}$$

$$T^r_{ki} y^k_i + T^r_{ki} y^k_j \leq (y^k_i + y^k_j)\mathcal{S}^k_{ij}, \forall k \in T_t(ij), (i,j) \in \mathcal{Z} \tag{7.18}$$

$$\sum_{k=1}^{N} (y^k_i + y^k_j) \leq N, (i,j) \in \mathcal{Z} \tag{7.19}$$

$$\sum_{\forall (i,j) \in \mathcal{Z}} (y^k_i + y^k_j) \leq 1, k \in T_t(ij) \tag{7.20}$$

$$y^k_i \in \{0,1\}, y^k_j \in \{0,1\} \tag{7.21}$$

The objective in Eq. (7.15) denotes the minimization of total energy consumed by FNs in all time slots subjected to the various constraints in the network. Constraint in Eq. (7.16) guarantees the response time from a FN should be less than or equal to the task k deadline. Constraint in Eq. (7.17) assures the sojourn time of vehicle ϑ_k is greater than the deadline of corresponding task k in the region $(i,j) \in \mathcal{Z}$. Constraint in Eq. (7.18) ensures response time from a FN for the task k should be less than the residing time of the respective vehicle in an intersecting part. Constraint in Eq. (7.19) guarantees that the total number of tasks scheduled among FNs i and j in an intersecting region should be less or equal to the number of available tasks N in that region $(i,j) \in \mathcal{Z}$. However, the constraint in Eq. (7.20) ensures the task k is scheduled in either FN i or FN j in an intersecting region

$(i, j) \in \mathcal{Z}$ of FVNETs. Finally, Eq. (7.21) is the integer constraint.

Scheduling a task in an optimal FN generally requires an exponential complexity solution for the objective $\mathbb{P}$ in Eq. (7.15). When N number of available tasks are offloaded to HPNs at time slot t, scheduling decisions among FNs in every intersecting region is a 0/1 multiple knapsack problem, a prominent NP-hard optimization problem [95]. As a result, we propose a RL-based sub-optimal task scheduling algorithm for the objective $\mathbb{P}$.

7.2 Reinforcement Learning based Energy-Efficient Task Scheduling

The intent for RL in the system is to train the FNs for various traffic conditions (i.e., $\lambda = 2 \sim 6$) in FVNETs, such that the long-term reward is maximized in each time slot t. Here, the FN reward is a function of total transmitted bits and the average energy consumption of FN. In this approach, the RL agent learns FN's energy consumption for different arrival rates of vehicles in the network. The RL agent can achieve the highest rewards by scheduling the vehicle's task in a suitable FN, consuming minimum energy in an intersecting region for each time slot.

7.2.1 RL agent

As the RL agent is clueless about the network and state transition, it follows the ϵ-greedy approach to explore myriad actions in the network for obtaining optimal scheduling policy in each state. Moreover, this system assumes time slots and vehicles as the states and corresponding state actions, respectively. The RL agent operation is illustrated in Figure 7.3. In the ϵ-greedy method, there are two phases, first the exploration phase and second the exploitation phase. In an initial state, as the agent is unaware of state transition and network condition, it follows the exploration phase, in which the agent probes all feasible moves greedily till it manipulates the most promising actions to increase the long-term return. The RL agent uses the random numbers in the range of (0, 1) denoted by W as shown in Figure 7.3 in the exploration phase. Subsequently, in the exploitation phase, it

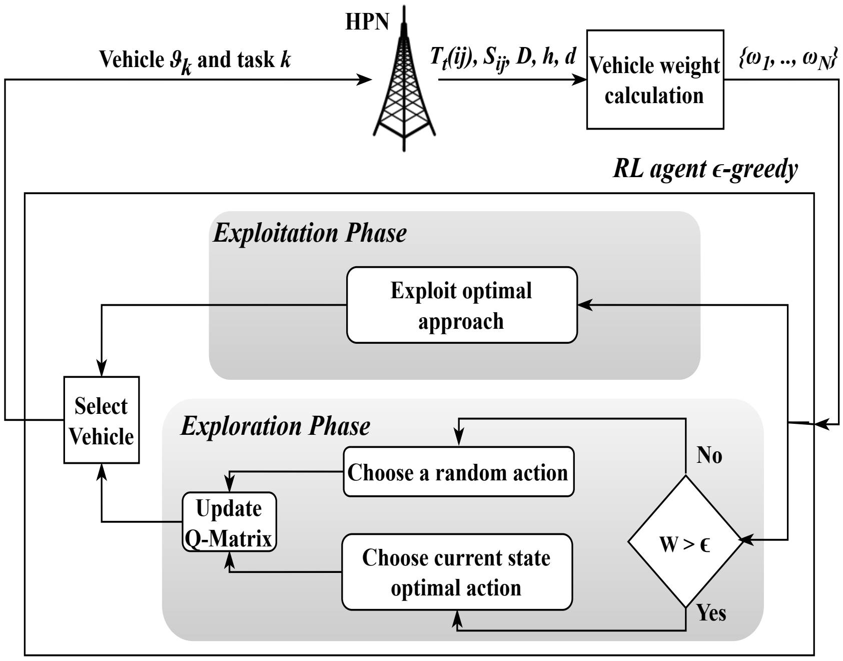

HPN
Vehicle ϑ_k and task k
$T_t(ij)$, S_{ij}, D, h, d
Vehicle weight calculation
$\{\omega_1, .., \omega_N\}$
RL agent ϵ-greedy
Exploitation Phase
Exploit optimal approach
Select Vehicle
Exploration Phase
Choose a random action
Update Q-Matrix
Choose current state optimal action
No
W > ϵ
Yes

Figure 7.3: RL agent operation.

chooses the vehicle which enhances the long-term reward.

However, the agent directly can not determine vehicles as actions in this dynamic circumstance. Further, an action refers to vehicles identified by their respective weights rather than their specific identities. Therefore, the agent is instructed to treat the weights of vehicles as actions to take. The weights of vehicles are calculated using the parameters such as the vehicle's distance from FN, sojourn time in the intersecting region, respective task data size and task deadline. Then, the agent's current policy for the action is determined by selecting the vehicle with the highest weight for each FN in the intersecting region. As a result, the off-policy performance of the RL system converges towards the optimal selection of the vehicle ϑ_k and the vehicle's task k for the FNs i and j in the intersecting region to maximize the reward.

Theorem 7.2.1. *The service from the set of FNs, $\mathcal{G}_{ij}$, to the vehicles in the set $V_t(ij)$, at time slot t in an intersecting region (i, j), $\forall(i, j) \in \mathcal{Z}$, is injective.*

Proof: Consider the function $f_2 : \mathcal{G}_{ij} \rightarrow V_t(ij)$, which describes the FNs in the set $\mathcal{G}_{ij}$ serving the vehicles belonging to the set $V_t(ij)$ at time slot t in an intersecting region $(i, j) \in \mathcal{Z}$, as shown in Figure 7.4. The FNs, FN i and FN j, in an intersecting region

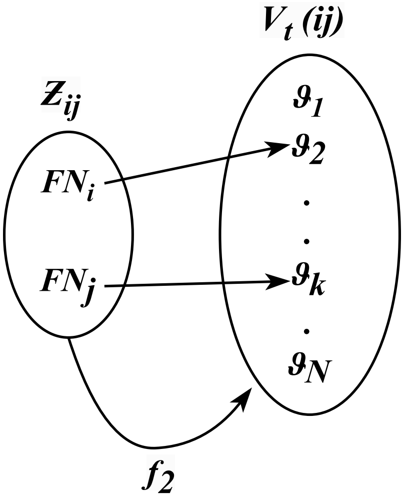


Figure 7.4: Services from FNs to vehicles is injective.

$(i, j) \in \mathcal{Z}$ serve the vehicles ϑ_k and ϑ_l, $k \neq l$, belong to the set $V_t(ij)$, respectively. In other words, every FN in an intersecting region will serve the vehicles belonging to the set $V_t(ij)$ at time slot t. However, no two vehicles ϑ_k and ϑ_l, $k \neq l$, belong to the set $V_t(ij)$ get services from the same FN in an intersecting region at time slot t. Therefore, the service from FNs to vehicles in an intersecting region is injective but not surjective.

7.2.2 Vehicle Weight Calculation

FNs in an intersection region maintain the list of vehicles in their coverage area and forward the beacon messages disseminated by vehicles to HPN in each time slot. Further, FN forwards the set of vehicle tasks in the intersection region to HPN. Then, HPN compiles the following information at the beginning of every time slot t.

- HPN computes the vehicles set in each intersecting region at time slot t. Let there are N number of vehicles in the intersecting region $(i, j) \in \mathcal{Z}$ denoted by $V_t(ij)$ = $\{\vartheta_1,..., \vartheta_N\}$ and their respective tasks are denoted by the set $T_t(ij)$ = $\{1, 2,..., k,..., N\}$.
- S_{ij} = $\{S^1_{ij}, S^2_{ij},..., S^N_{ij}\}$ is a set of sojourn time of vehicles.
- D = $\{D_{1i}, D_{2i},..., D_{Ni}\}$ is a set of distances of vehicles from FN i or FN j in an

intersecting region (i, j).

- h = $\{h_1$, h_2,. . ., $h_N\}$ is a set of data sizes of vehicles tasks.
- d = $\{d_1$, d_2,. . ., $d_N\}$ is a set of deadlines of vehicles tasks.
- ω = $\{\omega_1$, ω_2,. . ., $\omega_N\}$ is a set of weights of vehicles in the intersecting region (i, j).

HPN calculates the vehicles' weight using the sets S_{ij}, D, h, d. It is worth mentioning that the FN energy consumption increases when communicating with vehicles as the vehicle's distance from FN increases [136]. Hence, FNs strive to serve vehicles which are close to FN to mitigate energy consumption. However, this kind of operational policy deteriorates the QoS of vehicles when leaving the intersecting region without completing the service request. Therefore, the RL agent considers this event as undesired to bypass the penalty. The vehicle's weight, denoted by ω_k, is obtained to train the FN to serve the vehicle with minimum energy consumption and bypass undesired events. The vehicle weight is influenced by its proximity to the FN, task data size, task deadline and sojourn time in the intersecting region. Vehicle weight increases as vehicle distance from FN decreases to make FN transmit the data at a high rate instead of transmitting data to distant vehicles using the exact quantity of energy, and vehicle weight increases as data size increases. Further, vehicle weight increases as task deadlines decrease to prioritise the tasks with lower deadlines, and vehicle weight increases as vehicle sojourn time decreases to prioritise the vehicles leaving the intersecting region without receiving complete service (to bypass penalty). Hence, vehicle weight is directly proportional to task data size and indirectly proportional to vehicle distance, vehicle sojourn time, and task deadline. The mathematical vehicle weight is defined as follows.

$$\omega_k = \frac{h_k}{(D_{ki} \times S_{ij}^k \times d_k)} \tag{7.22}$$

Finally, the RL agent uses the set of weights, ω, to schedule the tasks among suitable FNs in an intersecting region such that the energy consumption of FNs is reduced.

7.2.3 Task Scheduling in an Intersection Region using RL

The RL agent is trained to schedule tasks in optimal FNs of an intersecting region while collaborating with its environment. Here, the RL system considers the Markov Decision model with horizon T containing the data set $(Ss, A, r_t, \Theta, \eta)$ that is described as follows.

Procedure 7 RL agent's optimal policy (Max-weight)

```
Input: $V_{ij}(t)$, $\omega$
Output: $A(s_t)$ = $\{a^t_1$,. . . ,$a^t_N\}$ and $g$ number of vehicles from $A(s_t)$ for FNs in $(i, j) \in \mathcal{Z}$ (i.e.,$\binom{N}{g}$)
Let $V_{ij}(t)$ be the set N number of vehicles in the intersecting region of FNs $i$ and $j$
$A(s_t) \leftarrow \{a^t_p$ = $\vartheta_p$ | $\forall$ $\vartheta_p \in V_{ij}(t)$, 1≤ p ≤ $N\}$
Find action $a^t_k$ = max($A(s_t)$) such that $\omega_k \geq \ldots \geq \omega_1$
return $a^t_k$
```

1. *System space*: Ss is the system state space in which $s_t \in Ss$ is a state at time slot t, $1 \leq t \leq T$.

2. *Action space*: The available actions at state s_t is $A(s_t)$ = $\{a^t_1$,. . . , $a^t_N\}$. The action a^t_k represents the vehicle ϑ_k with corresponding weight ω_k. The number of available actions equals N number of vehicles available at state s_t in an intersection region $(i, j) \in \mathcal{Z}$ and vehicles are selected for each FN of an intersecting region. If g is the number of FNs available in an intersecting region, then the combination of vehicles selected from the action space, g, is given by $\binom{N}{g}$. As g and N remain constant, the number of actions remains consistent in each state. Moreover, the available actions $A(s_t)$ in each state are uniquely identified. Suppose the vehicles in the available actions at state s_t may differ from those at state s_{t+1}. However, the agent chooses the vehicle (i.e., action) found on the current policy (i.e., maximum-weight policy), which is given in Procedure 7. The agent selects action a^t_k at state s_t if the vehicle ϑ_k with the highest weight ω_k is chosen.

3. *Reward*: The agent's goal is to maximize the reward it receives from the environment, and r_{s_t} is the reward at state s_t received from the environment for taking action a^t_k. It is defined as the ratio of bits transmitted to the selected vehicles and the average

energy consumed by the FN at time slot t. However, when a vehicle leaves the intersecting region without receiving complete service, the reward is penalized by the remaining data that must be transmitted to complete the vehicle's request. Hence, the agent always tries to transmit a large amount of data from a FN, consuming minimum energy at time slot t to maximize the reward while avoiding the vehicles leaving intersecting regions without complete service. On the other hand, FNs consume less energy to transmit the data to closer vehicles. Hence, the RL agent tries to select a vehicle close to the FN. Therefore reward for action a_k^t at state s_t is given as,

$$r_{s_t} = r(s_t, a_k^t) = \frac{h_k}{f(E_{ki}(t))}, \forall t \in T \tag{7.23}$$

where $f(.)$ is a function of average energy consumption of FN i at time slot t given as $f(E_{ki}(t)) = \frac{\sum_{\bar{t}=1}^{t} E_{ki}(\bar{t})}{t}$, $\forall t \in T$. In this system, each FN can provide services to one vehicle at each time slot.

4. Θ is the transition probability for the state-action pair $(s_t, a_k^t) \in Ss$. The probability of possible pair of next state s_{t+1} and reward r_{s_t} is denoted by $\Theta(s_{t+1}, r_{s_t} | s_t, a_k^t)$. The transition probability from any pair $(s_t, a_k^t) \in Ss \times A$, gives the probability of transition to next state s_{t+1} from current state s_t provided that the action a_k^t is taken at state s_t. In this RL system, the transition probability equals one when the agent moves to the next state s_{t+1} from the current state s_t.

5. η is the learning rate set between 0 and 1.

The RL agent uses the policy π to obtain action a_k^t at state $s_t \in Ss$ denoted by $a_k^t = \pi(s_t)$, $\forall a_k^t \in A(s_t)$ to maximize the rewards. The agent can adjust its policy towards optimal policy π^* when following the policy π while observing the next state s_{t+1} and reward $r(s_t, a_k^t)$. The objective is to find the optimal policy π^* that maximizes the overall discounted rewards over horizon T. Then, the total discounted rewards, Υ is expressed as

$$\Upsilon = r_1 + \gamma r_2 + \gamma^2 r_3 + \cdots + \gamma^{T-1} r_T = \sum_{s_t=1}^{T} \gamma^{s_t - 1} r_{s_t} \tag{7.24}$$

where γ is the discounting factor that takes a value in the $(0, 1)$ range. When following the policy π, the total discounted reward becomes

$$\Upsilon_\pi = \sum_{s_t=1}^{T} \gamma^{s_t-1} r(s_t, \pi(s_t)) \tag{7.25}$$

Consider the set Π represents all possible policies, then the optimal approach is given by

$$\pi^* = \arg\max_{\pi\in\Pi} \Upsilon_\pi \tag{7.26}$$

The value function $V_\pi : Ss \to \mathbb{R}$ quantify the envisioned value for following the policy π using transition probability from state s_t to state s_{t+1}. Then the value function can be expressed as

$$V_\pi(s_t) = r(s_t, \pi(s_t)) + \gamma \sum_{s_{t+1}} \Theta(s_{t+1}, r_{s_t}|s_t, a_k^t), \forall s_t \in Ss \tag{7.27}$$

Consequently, the optimal policy π^* gives the optimal value function $V_{\pi^*} : Ss \to \mathbb{R}$, expressed as follows

$$V_{\pi^*}(s_t) = \max_{\pi\in\Pi} V_\pi(s_t) \tag{7.28}$$

The optimal policy in Eq. (7.28) earns the maximum reward using Eq. (7.25) [38]. Eq. (7.28) can be solved by using transition probability function Θ. However, state transition mapping in the Markov model is unavailable. Hence, the RL system uses Q-Learning to train optimal policy for FNs by observing the outcome of various actions without transition function. The RL agent implements the iterative Q-Learning to realize the optimal policy π^*.

In Q-learning, the agent memorises optimal approach π^* by initiating the Q-values $Q(s_t, a_k^t)$. Subsequently, at each state $s_t \in Ss$ it maps best action $a_k^t \in A(s_t)$. In other words, the agent first observes the current state s_t in the learning phase and takes the corresponding best action a_k^t. As a result, it earns the reward r_{s_t} and next state s_{t+1}. Then it updates the Q-values $Q(s_t, a_k^t)$ for the observed outcomes r_{s_t} and s_{t+1}. Similarly, the process above is replicated in the next iteration till the agent learns the optimal policy π^*,

Algorithm 7.1 Q-learning based Reinforcement Learning

```
Initialize $Q(s_t, a_k^t) = 0, \forall \in Ss$
Initialize $\epsilon$, $\eta_{s_t}$, $\gamma$
for each episode do
    for each $(i, j) \in \mathcal{Z}$ do
        for $t \leftarrow 1$ to $T, \forall s_t \in Ss$ do
            for each FN in $(i, j) \in \mathcal{Z}$ do
                take action $a_k^t$, observe $r_{s_t}$ and next state $s_{t+1}$
                if RANDOM(0, 1) $< \epsilon$ then
                    Select a random $a_k^t$ from state $s_{t+1}$
                else
                    Select a $a_k^t$ from state $s_{t+1}$ using $\epsilon$-greedy approach
                end if
                Update $Q(s_t, a_k^t)$ using Eq. (7.31)
            end for
            $s_t \leftarrow s_{t+1}$
        end for
    end for
end for
```

especially when the agent earns the maximum rewards by taking action in the given state using optimal policy π^*. Then the optimal Q-values for all state action pair $(s_t, a_k^t) \in Ss \times A(s_t)$ is $Q^*(s_t, a_k^t)$ and is expressed as follows.

$$Q^*(s_t, a_k^t) = r(s_t, a_k^t) + \gamma \sum_{s_{t+1} \in Ss} \Theta(s_{t+1}, r_{s_t} | s_t, a_k^t) V_{\pi^*}(s_t) \tag{7.29}$$

However, the optimal value function is $V_{\pi^*}(s_t) = \max_{a_k^t} Q^*(s_t, a_k^t)$, then the optimal policy for the state s_t is given as

$$\pi^*(s_t) = \arg\max_{a_k^t} Q^*(s_t, a_k^t) \tag{7.30}$$

Hence, the Q-values can be obtained using the stochastic increment Q-learning, using the below equation.

$$\begin{aligned} Q(s_t, a_k^t) =& Q(s_t, a_k^t) + \\ & \eta_{s_t} \left\{ r(s_t, a_k^t) + \gamma \max_{a_k^{t+1}} Q(s_{t+1}, a_k^{t+1}) - Q(s_t, a_k^t) \right\} \end{aligned} \tag{7.31}$$

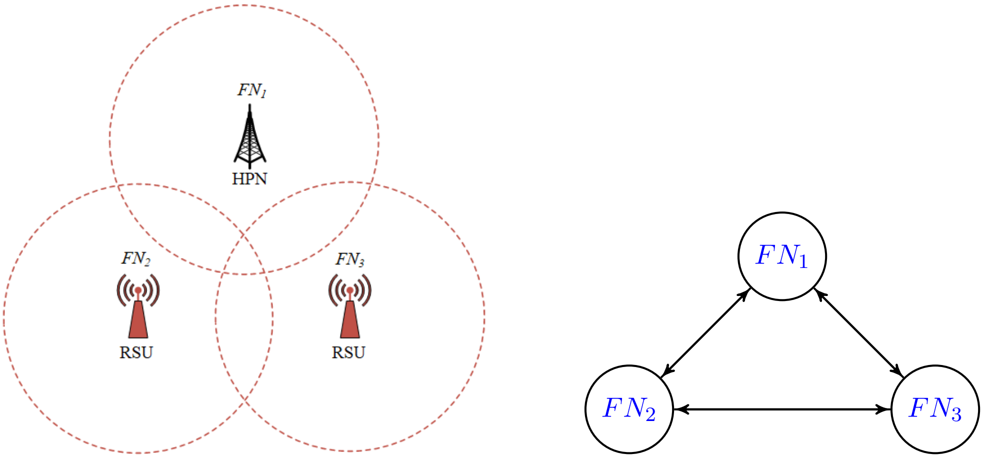
where η_{s_t} is the learning rate, which takes a value between 0 and 1. Moreover, if $\eta_{s_t} = 0$, learning will not occur, i.e., Q-values are not updated. However, setting η_{s_t} with higher values makes learning fast. Therefore, η_{s_t} satisfies the condition $\sum_{s_t} \eta_{s_t} = \infty$ and $\sum_{s_t} \eta_{s_t}^2 < \infty$. The former condition guarantees the convergence of the algorithm pre-maturely. The latter condition ensures fading the noise in the algorithm [38]. In most cases η_{s_t} is set to $1/s_t$. Once the algorithm convergence, the optimal policy and the optimal Q-values are obtained. It is worth noting that γ plays an essential role in converging Q-values. The value of $\gamma = 0$ makes the agent consider only the current state reward. If the discount factor is near 1, the agent tries for maximum future reward. The off-policy Q-learning-based reinforcement learning for FVNETs is given in Algorithm 7.1.

7.3 Performation Evaluation

In this section, the performance of the EEDA algorithm is investigated. The simulation model is described in Section 7.3.1, and results are presented by comparing with PSG [41], EDF [42], FCFS [43] and RO [44] in Section 7.3.3.

7.3.1 Simulation Setup

A virtual environment is created for evaluating EEDA's performance on the integrated development kit PyCharm 2020.1.3 using Python 3.8. Windows 11 64-bit operating system, 64 GB RAM and CPU @ 2.90 GHz Intel(R) Xeon(R) Gold 622R processor is used to run the environment. The FNs in the environment are arranged in an area of $[3000 \times 3000]^2$ meter, as shown in Figure 7.5a. The figure shows one HPN and two RSUs forming three intersecting regions in FVNET. The graph model $G(V, E)$ of corresponding FVNET is shown in Figure 7.5b in which vertices represent the FNs and an arc between vertices denotes the intersecting region of respective FNs. The length of each intersecting region in FVNETs is approximately 900 meters. The Monte-Carlo simulations are conducted for various arrival rates of vehicles from 2 vehicle/s to 6 vehicle/s in each intersecting region. The mean arrival rate of vehicles follows Poisson distribution. Moreover, the most realistic propagation model, IEEE 802.11p, is used for wireless connectivity in FVNETs. Subsequently,



(a) (b)

Figure 7.5: Topology of FVNET and graph model with one HPN and three RSUs considered for simulations.

the average results are obtained by carrying simulations in each intersecting region up to 100 timeslots. The simulation outcome shows the impact of the increase in the number of vehicles in the intersecting regions for various vehicle arrival rates in the network on FN's energy usage and throughput. Other performance metrics, such as the percentage of transmitted data in the network, the number of completed service requests, and service time, are also considered to show the performance of the proposed algorithm. The parameters used for the simulation setup are shown in Table 7.2.

7.3.2 Benchmark Algorithms

The RL-based EEDA scheduling technique is evaluated with the performance of the following baselines: PSG [41], EDF [42], FCFS [43] and RO [44]. The PSG algorithm is the scheduling algorithm that schedules tasks with lower deadlines. The PSG is semi-greedy and achieves optimal energy consumption by scheduling tasks in a FN consuming minimum energy among available FNs. The task scheduling in the EDF algorithm is based on the lower deadlines in the FN selected randomly among the FNs. The tasks having the least arrival time are scheduled first in the arbitrally selected FN in the FCFS scheduling. Similarly, RO scheduling schedules the tasks selected randomly in a FN selected randomly. On the other hand, in the EEDA algorithm, the scheduling tasks in suitable FNs are decided

Table 7.2: Parameters and their values for simulations

Parameter	Value
Covered distance of vehicle ϑ_k, $\delta_t^c(\vartheta_k)$	[100 $\sim$ 799] m
Mean arrival rate of vehicles, λ	[2 $\sim$ 6] vehicles/s
Task k data size, h_k	[100 $\sim$ 150] MB
Vehicle speed, v_k	[15 $\sim$ 25] m/s
Task k CPU requirements, c_k	[1000 $\sim$ 2500] Mega cycles
FN CPU frequency, F_i	3.0 GHz
Task k deadline, d_k	[1 $\sim$ 10] s
Link bandwidth, B	20 MHz
Length of Intersecting region, L	900 m
Radius of FN coverage, R	500 m
Channel Inference, I_{ki}	-70 dBm
Gaussian Noise, N_{ki}	-80 dBm
Pathloss exponent, β	3
Scaling coefficient, α	1
Learning rate, η_{s_t}	1/s_t
Discounting factor, γ	0.9
Epsilon, ϵ	0.2
Time slot length	1 s
Simulation duration	100 s

by examining the response time from the FN, the task's deadlines, the energy usage of FNs and the vehicle's sojourn time in the intersecting region. Hence, the RL agent is trained for various arrival rates of vehicles in the network to obtain better results.

7.3.3 Results and Discussion

The performance of EEDA is analyzed with PSG, EDF, FCFS and RO algorithms in terms of average energy consumption, network throughput, percentage of transmitted data, percentage of tasks meeting their deadlines and total service time for different vehicle arrival rates from 2 to 6 vehicles/s.

7.3.3.1 Average Energy Consumption

The energy usage per time slot is the average energy usage of FN to deliver utilities to the vehicle with the required bits of data. We assume that each arrived vehicle in each time slot offloads a task into a FN for services, and a FN provides services to a vehicle in each time slot. Further, the impact of the increase in the number of vehicles in FVNETs on the average energy consumption of FN is assessed for different arrival rates (λ) of vehicles. Figure 7.6 shows the average energy usage of FN in processing tasks for each vehicle arrival rate (λ) from 2 to 6 vehicles/s for the benchmark and proposed algorithm. Subsequently, FN's energy usage for the different arrival rates of vehicles in the network is shown in Figure 7.7. From Figure 7.6, it is observed that the energy consumption of FN increases linearly as the number of tasks increases. The PSG is greedy in deciding FNs consuming minimum energy to schedule tasks in the intersecting region while meeting delay constraints of tasks. However, PSG suffers from the increased response time from FN. As it is unaware of the residing time of the vehicle in the intersecting region, the FNs process the tasks even after the vehicle leaves the intersecting region. Hence, using the PSG algorithm, FN's energy usage increases for λ values between 2 and 6.

As mentioned earlier, the intersecting regions in FVNETs formed with two FNs having finite resources. Hence, the number of vehicles getting services in the intersecting regions is approximately the same as the vehicle arrival rate increases. In the case of the EDF algorithm, the lower deadline tasks are prioritised. Hence, for each vehicle arrival rate (λ), energy usage increases as vehicles arriving in the network increase using EDF. However, the FN's energy consumption is approximately the same for vehicle arrival rates ranging from 2 to 6 using EDF. The FCFS schedules the first arrived task, and the RO schedules the task arbitrarily among FNs selected arbitrarily. FCFS and RO strive to schedule tasks in randomly chosen FNs in the intersecting region without considering the energy usage of FNs and the sojourn time of vehicles. Consequently, the energy usage using FCFS and RO increases when vehicles arriving in the network increase for the arrival rates ranging from 2 to 6. Nevertheless, energy usage in the network is reduced when the arrival rate increases from 3 to 5 using FCFS and RO, as shown in Figure 7.7.

On the other hand, the RL agent in the EEDA algorithm is trained for different arrival rates in the FVNETs. The RL agent schedules tasks among two FNs in each intersecting region by selecting a vehicle with maximum weight among many arrived vehicles in all timeslots. The number of vehicles in the intersecting regions increases as the vehicle arrival rate increases. However, the EEDA algorithm schedules tasks in a FN consuming minimum energy in each timeslot to satisfy deadline constraints of tasks while considering the sojourn time of the vehicle in the intersecting region. Moreover, the EEDA algorithm optimizes the energy usage of FNs as vehicles arriving in the network increase compared to other algorithms for arrival rates from 2 to 6, as shown in Figure 7.6. Thus, the average energy usage of FN for different vehicle arrival rates using the EEDA algorithm is minimized to 53.06%, 52.59%, 56.24% and 56.36% on average compared to PSG, EDF, FCFS and RO algorithms, respectively.

7.3.3.2 Network Throughput

The network's throughput is obtained using the transmission rate from FNs to vehicles at timeslot t. The allocated time fraction at time slot t from FN i to the vehicle ϑ_k is $\sigma_t(ki)$, then the throughput is obtained using the Shannon formula in Eq. (7.5). The throughput from FN i to the vehicle ϑ_k is given as $\Gamma_t(ki) = \sigma_t(ki) \times \chi_{ki}$. Subsequently, the network throughput is given as $\sum_{\forall(i,j)\in\mathcal{Z}} \log(\sum_t \sum_{i\in(i,j)} \Gamma_t(ki)) \; \forall\vartheta_k \in V_t(ij)$ [25]. Figure 7.8 shows the network throughput achieved when there is an increase in vehicles reaching the network for each arrival rate from 2 to 6. The network's throughput depends on the time fraction allocated to the vehicles getting services. Therefore, the network's throughput is proportional to the number of vehicles getting services. The figure shows that the throughput increases as the number of vehicles in the network increases, and the EEDA algorithm maximizes the network throughput for each value of λ.

The throughput decreases gradually when the vehicle arrival rate increases from 2 to 6, as shown in Figure 7.9. As mentioned earlier, in this chapter, the intersecting regions in FVNETs consist of two FNs yielding limited resources. Therefore, the network throughput is reduced for values of λ from 2 to 6. The benchmark algorithms provide services to vehicles about to leave the intersecting region. Hence, the throughput is significantly reduced

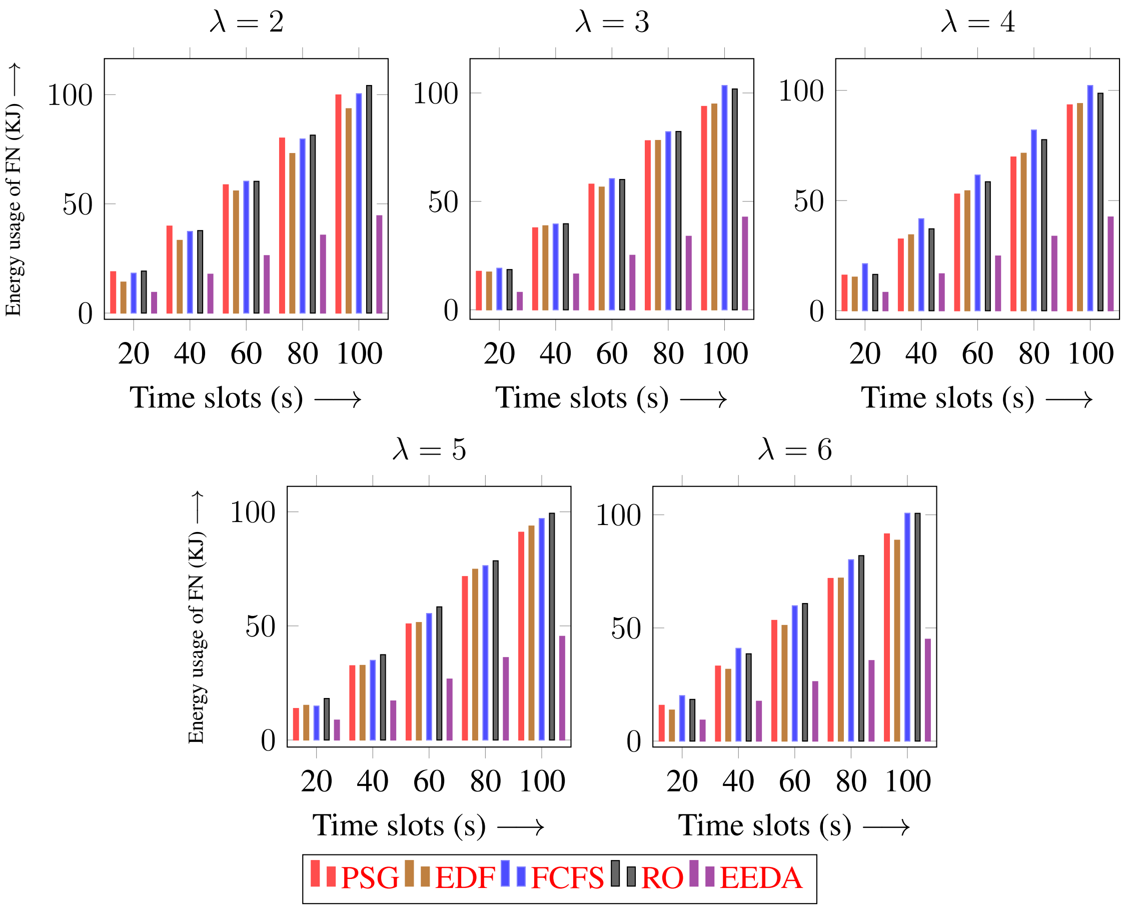


Figure 7.6: Average energy consumption of a FN when there is an increase in the number of tasks for various arrival rates in FVNETs using PSG, EDF, FCFS, RO and EEDA algorithms.

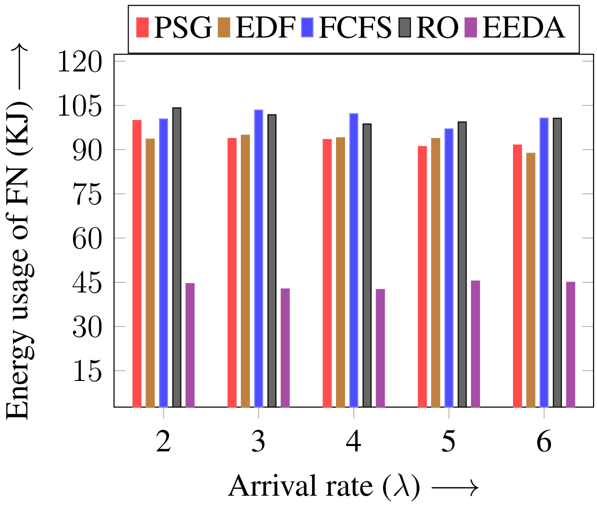


Figure 7.7: Average energy consumption of a FN in FVNETs for various arrival rates of vehicles using PSG, EDF, FCFS, RO and EEDA algorithms.

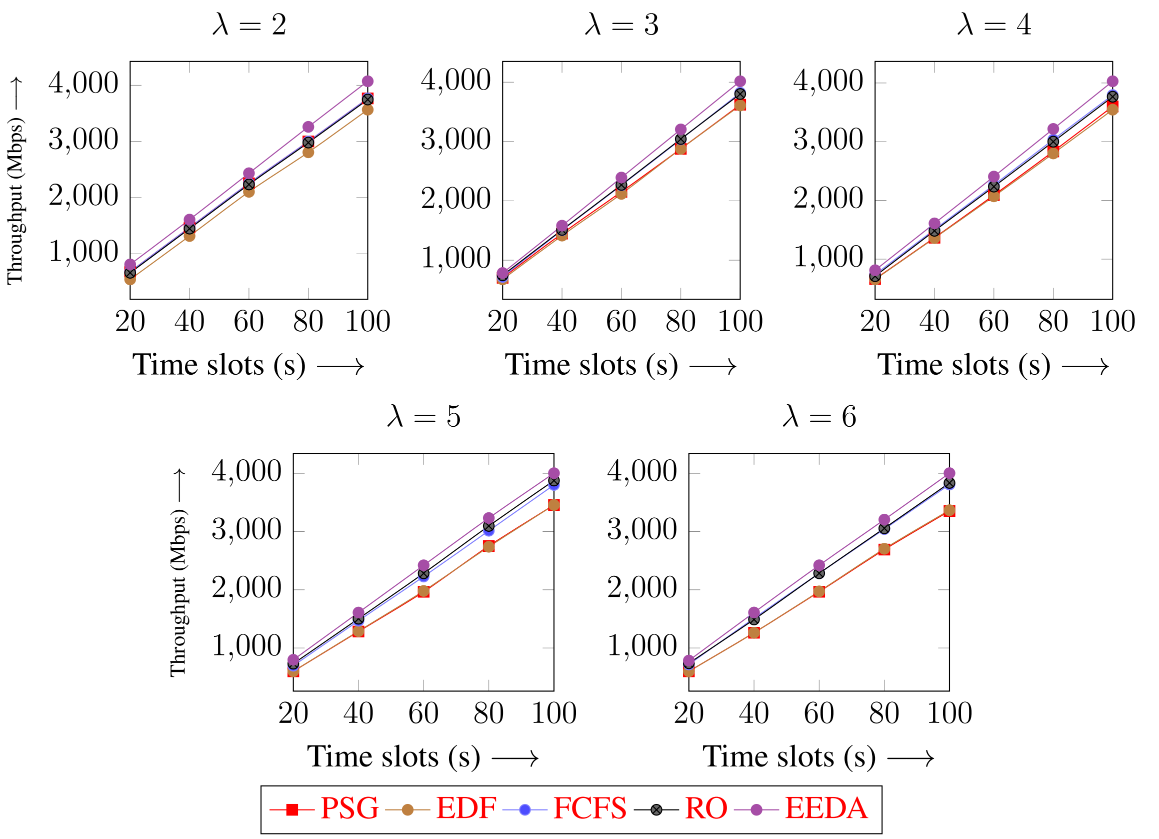


Figure 7.8: Network throughput when there is an increase in the number of vehicles for the arrival rates from 2 to 6 using PSG, EDF, FCFS, RO and EEDA algorithms.

using PSG, EDF and FCFS algorithms. Using the RO algorithm, due to its randomness, the throughput increases and then decreases arbitrarily. However, the EEDA algorithm strives to avoid serving the vehicles leaving the intersecting region to bypass the penalty. Therefore, the RL-based EEDA algorithm enhances the network throughput by 13.21%, 14.77%, 5.96% and 5.70% on average when compared to PSG, EDF, FCFS and RO algorithms, respectively, for various values of λ.

7.3.3.3 Percentage of Transmitted Data

The percentage of transmitted data is defined as the number of bits transmitted to the vehicle over the total amount of bits requested during its sojourn time in the intersecting region of the network. Figure 7.10 shows the percentage of data transmitted to the vehicles in the intersecting regions of the network for different values of λ. The benchmark algorithms

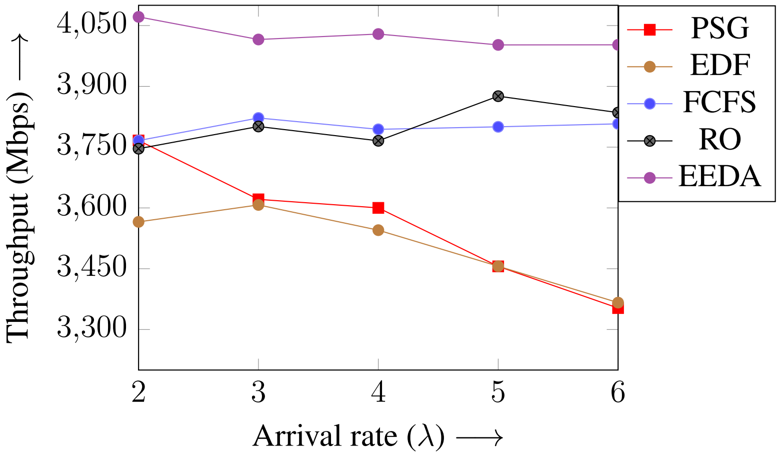


Figure 7.9: Network throughput for various arrival rates of vehicles using PSG, EDF, FCFS, RO and EEDA algorithms.

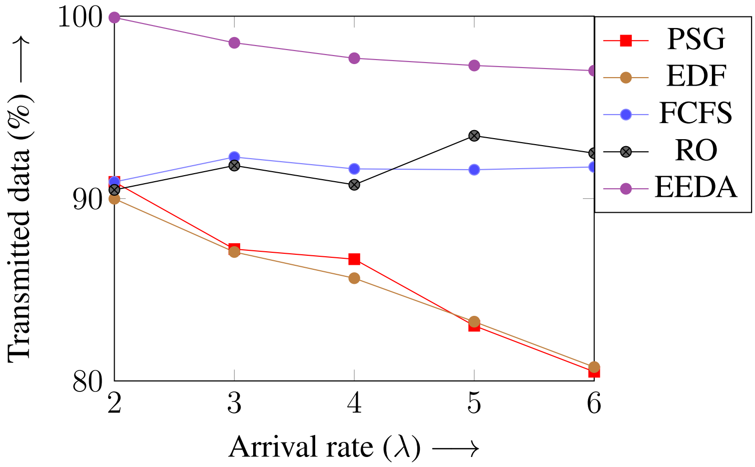


Figure 7.10: Transmitted data (%) in the FVNETs for various arrival rates of vehicles using PSG, EDF, FCFS, RO and EEDA algorithms.

fail to transmit total request data to the vehicles as the vehicle arrival rate in the network increases due to selecting the vehicles leaving the intersecting regions. However, the RL agent in the EEDA algorithm is trained to avoid selecting vehicles leaving the intersecting regions for maximizing long-term reward. Consequently, the EEDA algorithm completes the transmission data to the vehicle during its sojourn time as arrival rates into the network increase. Hence, the EEDA enhances percentage data transmission by an average of 14.66%, 15.06%, 7.06%, and 6.88% compared to PSG, EDF, FCFS and RO algorithms, respectively.

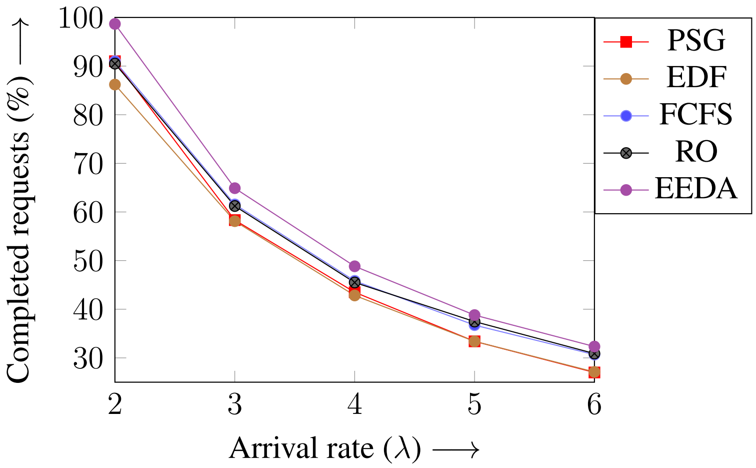


Figure 7.11: Completed requests in FVNETs for various arrival rates of vehicles using PSG, EDF, FCFS, RO and EEDA algorithms.

7.3.3.4 Percentage of Completed Requests

Figure 7.11 indicates the percentage of vehicles leaving the intersecting regions with completed service requests for the vehicle arrival rate from 2 to 6. Similarly, Figure 7.12 shows the percentage of vehicles leaving the intersecting regions with incomplete service from FN for different arrival rates. Figure 7.11 shows that as the network's vehicle arrival rate increases, the percentage of vehicles receiving completed requests reduces. Alternatively, Figure 7.12 shows that the percentage of vehicles leaving with incomplete requests increases as the arrival rate increases. However, the RL agent prioritizes the serving vehicles leaving the intersecting region in case of satisfying the task's deadline constraints. Otherwise, it bypasses the penalty by not scheduling the tasks of vehicles leaving the intersecting regions in the EEDA algorithm. As a result, it enhances the percentage of completed services on an average of 13.57%, 15.12%, 6.28%, and 6.11% compared to PSG, EDF, FCFS and RO algorithms, respectively. Similarly, the EEDA algorithm diminishes the percentage of vehicles leaving with incomplete services by 25.15%, 26.46%, 21.00%, and 21.15% on average compared to PSG, EDF, FCFS and RO algorithms, respectively.

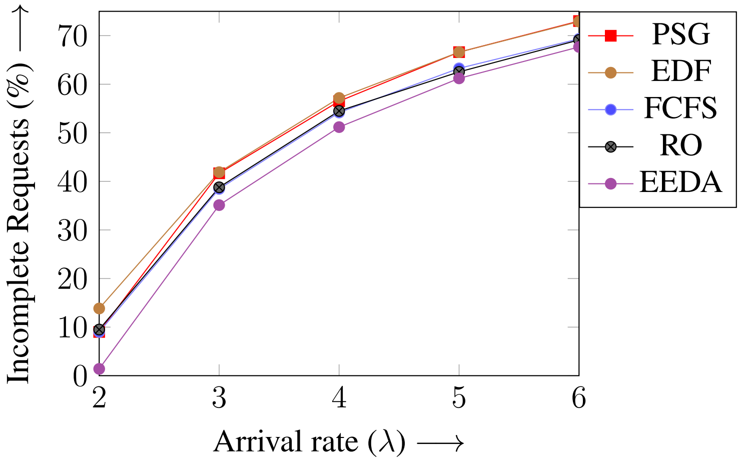


Figure 7.12: Incomplete requests in FVNETs for various arrival rates of vehicles using PSG, EDF, FCFS, RO and EEDA algorithms.

7.3.3.5 Total Service Time

The service time of a FN is the total time needed to process the tasks when scheduled in a FN. FNs' service time depends on the processing task type (i.e., data size and CPU cycles). The CPU requirement of a task is represented in megacycles per second. The total service time in FVNETs is shown in Figure 7.13 for different values of λ. The figure shows that the service time in FVNETs reduces as the vehicle's arrival rates increase using the PSG, EDF and FCFS algorithms. There is consistent service time using the RO algorithm for the vehicle arrival rate from 2 to 4. However, the service time using the EEDA algorithm for the vehicle arrival rate from 2 to 6 is consistent. Overall, the EEDA algorithm performs better in service time than the competing benchmarks. In other words, the EEDA algorithm, on average, performs better in reducing service time by 2.36%, 1.00%, 8.31%, and 8.48% compared to PSG, EDF, FCFS and RO algorithms, respectively.

7.4 Summary

This chapter considers the intersecting regions of FNs in FVNETs covering remote areas and schedules the delay-sensitive tasks offloaded by the vehicles in the intersecting regions among FNs. The RL-based EEDA algorithm is proposed to schedule the tasks in intersecting regions to minimize the energy usage of FNs in serving the vehicles while discharging.

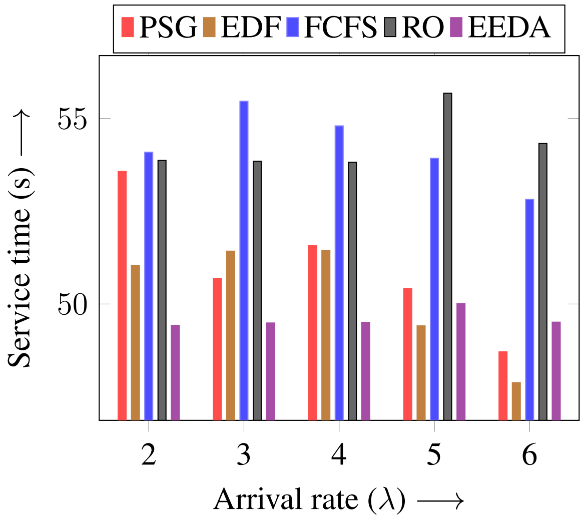


Figure 7.13: Service time in FVNETs for various arrival rates of vehicles using PSG, EDF, FCFS, RO and EEDA algorithms.

Hence, the RL agent is trained for free-flow traffic, considering vehicle arrival rates ranging from 2 to 6 in the FVNETs. Further, to decide on suitable FN for processing tasks, the RL agent considers the task deadline, FN response time, FN energy usage, and vehicle's sojourn time in the intersecting region. Henceforth, a greedy-based sub-optimal solution, EEDA, using RL is obtained to mitigate energy usage of FNs as vehicles arriving in the network increase for different arrival rates. Moreover, the EEDA algorithm is assessed in FN's energy usage, network throughput, percentage of transmitted data, percentage of services completed and total service time. The simulation results show that the RL-based EEDA algorithm performs better than benchmark algorithms in minimizing the average energy consumption of FN and enhancing the network throughput. The proposed EEDA algorithm reduces the energy consumption of FNs in serving vehicles up to 53.06%, 52.59%, 56.24% and 56.36% on average, compared to PSG, EDF, FCFS and RO algorithms, respectively.

Chapter 8

Conclusion and Future Scope

This thesis investigates the resource management and task scheduling algorithms in FVNETs to enhance the user QoS and minimize the energy consumption of FNs while meeting deadline constraints of tasks, respectively. Specifically, it mainly dealt with vehicles in the intersecting regions and FVNETs in which FNs are deployed such that the FN's coverage region intersects with neighbouring FNs. First, it introduces the FVNETs, followed by the background of FC in VANETs, along with various applications of FVNETs and challenges in FVNETs. The main contributions of this thesis are discussed from Chapter 4 to Chapter 7, where it proposes various resource management and task scheduling algorithms. In this thesis, contributions have been made by considering the significant challenges, such as service capability, serviceability, availability, throughput, energy consumption of FNs, resource utilization efficiency, response time of FNs and vehicles sojourn time in the intersecting regions to satisfy the delay constraints of tasks.

8.1 Conclusion

In Chapter 4, the DRM algorithm manages the RBs allocation in FVNETs by considering vehicles in overlapped coverage regions of FNs. The objective of the DRM algorithm is to minimize the occupied RBs of vehicles that have already arrived. This reduction in allocated RBs is achieved by migrating allocated RBs of a set of vehicles among FNs, and it is addressed by formulating ILP. The DRM algorithm maximizes the network's service capability by minimizing allocated RBs. The simulation outcome shows that the DRM algorithm enhances the service capability by migrating RBs of the set of vehicles compared to other migration algorithms, such as DRO, SA, DRO+SA and RO.

In Chapter 5, an ERO algorithm is proposed to synchronize RB allocation among pairs of FNs to maximize the network's throughput. The ERO algorithm partitions the FNs' coverage region into restricted and non-restricted coverage regions. Then, it reduces the assigned RBs of vehicles in the non-restricted coverage regions by migrating the RBs of those vehicles among pairs of FNs. The ERO algorithm uses the minimum priority queue to select the FNs with minimum occupied capacity and a vehicle with the minimum required RBs. It constructs the minimum priority queue using the occupied capacity of FNs to perform optimal migration. As a result, the RBs reduction among pairs of FNs maximizes the network's throughput. Further, the performance of the ERO algorithm is assessed in each time slot as the number of vehicles grows within the network. The simulation outcomes show that the ERO algorithm enhances the network's throughput compared to existing algorithms, such as RO, ARB, SA and DRO.

In Chapter 6, the EERA algorithm reduces the energy consumption of FNs with the increasing number of vehicles joining the network while maximizing the resource utilization efficiency. Here, the EERA algorithm considers the vehicles in the overlapping coverage parts of FNs to reduce the allocated RBs of vehicles. The EERA has been evaluated with RO, MCF, DRO and SA in terms of the percentage of RBs occupied, the energy consumption of FNs and the resource utilization efficiency. Further, FN's energy consumption as the number of FNs in the network increases is also analyzed using EERA. The simulation outcomes show that the EERA algorithm reduces the energy consumption of FNs by 37.58%, 32.51%, 22.95% and 8.33% on average compared to RO, MCF, DRO and SA algorithms, respectively.

Chapter 7 presents the RL-based task scheduling algorithm in FVNETs to minimize the energy consumption of FNs during their battery-depleting time. The coverage region of each FN intersects with the neighbouring FN(s) to provide services in remote areas where consistent power sources are unavailable. Vehicles in such regions offload delay-sensitive tasks into FNs to get services. However, when the number of vehicles arriving into the network grows over peak hours, the energy dissipation of FNs for processing tasks increases. Consequently, energy-limited FNs become ineffective in delivering services without efficient task scheduling. Therefore, the RL-based EEDA task scheduling among FNs is proposed to reduce the energy dissipation of FNs. The RL agent is trained for different vehicle arrival rates to schedule tasks in a suitable FN of the intersecting regions. The performance of EEDA is evaluated by considering FN energy dissipation, FN response time and vehicle living time in the intersecting region to meet delay constraints of tasks. The simulation outcomes show that the EEDA algorithms lower the FN's energy dissipation by 53.06%, 52.59%, 56.24% and 56.36% on average compared to benchmark algorithms PSG, EDF, FCFS and RO, respectively.

8.2 Future Scope

Although the proposed resource management and task scheduling algorithms show promising performance improvements compared to the existing algorithms available in the recent literature, other techniques, namely evolutionary computing, deep learning and game theory, need to be investigated to improve the proposed algorithms further. Some of the potential extensions of our research work presented in this thesis are listed as follows.

As a future extension of the resource management algorithms, to make them more robust and realistic, these algorithms can be extended by considering the dynamic conditions of the network, such as the mobility of the vehicles. The load of FNs can be determined periodically to identify the overloaded and underloaded FNs and provide further services to the incoming vehicles. Further, a proactive algorithm can be developed to avoid the load imbalance among the FNs. The multi-level feedback queue type of algorithm can experiment with delay and non-real-time tasks in the FVNETs as the extension of current works.

Task scheduling in intersecting regions of FVNETs can be extended by offloading delay-sensitive tasks into mobile FNs and vehicles in the parking lots to minimize the energy consumption of FNs. Further, different approaches, such as fuzzy logic and metaheuristic-based, can be investigated to schedule the tasks among mobile FNs for efficient energy preservation.

Recently, osmotic computing has emerged as a related paradigm of edge, fog and cloud computing. It provides a promising solution to delay-sensitive tasks. Therefore, it can be integrated into VANETs to provide efficient ITS and smart city solutions as a further extension. In this paradigm, energy preservation is a primary challenge that needs more attention from the research community in the future.

The data transmission between FNs (i.e., transmitter) and vehicles (i.e., receiver) is affected by the intersymbol interference (ISI) problems in the communication channel. This inference is due to the interference of allotted time intervals with neighbouring pulses. Moreover, RB migration has demerits, such as noise disruption in communication. It leads to the channel fading also. Hence, we will consider addressing the ISI problems and noise disruption in FVNETs in our future work.

Bibliography

[1] Muhammad Saleh Bute, Pingzhi Fan, Gang Liu, Fakhar Abbas, and Zhiguo Ding. A collaborative task offloading scheme in vehicular edge computing. In *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, pages 1–5. IEEE, 2021.

[2] Rob van der Meulen and JR Gartner Says By. a quarter billion connected vehicles will enable new in-vehicle services and automated driving capabilities. 2015, 2020.

[3] Sanjaya K Panda, Indrajeet Gupta, and Prasanta K Jana. Task scheduling algorithms for multi-cloud systems: allocation-aware approach. *Information Systems Frontiers*, 21:241–259, 2019.

[4] Ibrar Yaqoob, Ibrahim Abaker Targio Hashem, Yasir Mehmood, Abdullah Gani, Salimah Mokhtar, and Sghaier Guizani. Enabling communication technologies for smart cities. *IEEE Communications Magazine*, 55(1):112–120, 2017.

[5] Qian Luo, Yurui Cao, Jiajia Liu, and Abderrahim Benslimane. Localization and navigation in autonomous driving: Threats and countermeasures. *IEEE Wireless Communications*, 26(4):38–45, 2019.

[6] Rida Zojaj Naeem, Saman Bashir, Muhammad Faisal Amjad, Haider Abbas, and Hammad Afzal. Fog computing in internet of things: Practical applications and future directions. *Peer-to-Peer Networking and Applications*, 12(5):1236–1262, 2019.

[7] Huifeng Wu, Danfeng Sun, Lan Peng, Yuan Yao, Jia Wu, Quan Z Sheng, and Yi Yan. Dynamic edge access system in iot environment. *IEEE Internet of Things Journal*, 7(4):2509–2520, 2019.

[8] MLM Peixoto, AHO Maia, E Mota, E Rangel, DG Costa, D Turgut, and LA Villas. A traffic data clustering framework based on fog computing for vanets. *Vehicular Communications*, 31:100370, 2021.

[9] Anirudh Paranjothi and Mohammed Atiquzzaman. A statistical approach for enhancing security in vanets with efficient rogue node detection using fog computing. *Digital Communications and Networks*, 8(5):814–824, 2022.

[10] Peng Qin, Yang Fu, Guoming Tang, Xiongwen Zhao, and Suiyan Geng. Learning based energy efficient task offloading for vehicular collaborative edge computing. *IEEE Transactions on Vehicular Technology*, 71(8):8398–8413, 2022.

[11] Mithun Mukherjee, Suman Kumar, Qi Zhang, Rakesh Matam, Constandinos X Mavromoustakis, Yunrong Lv, and George Mastorakis. Task data offloading and resource allocation in fog computing with multi-task delay guarantee. *IEEE Access*, 7:152911–152918, 2019.

[12] Roberto Beraldi, Claudia Canali, Riccardo Lancellotti, and Gabriele Proietti Mattia. Distributed load balancing for heterogeneous fog computing infrastructures in smart cities. *Pervasive and Mobile Computing*, 67:101221, 2020.

[13] KE Srinivasa Desikan, Vijeth J Kotagi, and C Siva Ram Murthy. Topology control in fog computing enabled iot networks for smart cities. *Computer networks*, 176:107270, 2020.

[14] Kamran Sattar Awaisi, Assad Abbas, Hasan Ali Khattak, Abbas Khalid, Hafiz Tayyab Rauf, and Seifedine Kadry. A dynamic load balancing mechanism for fog computing environment. *International Journal of Web and Grid Services*, 18(3):337–360, 2022.

[15] Mir Salim Ul Islam, Ashok Kumar, and Yu-Chen Hu. Context-aware scheduling in fog computing: A survey, taxonomy, challenges and future directions. *Journal of Network and Computer Applications*, 180:103008, 2021.

[16] Cheng Huang, Rongxing Lu, and Kim-Kwang Raymond Choo. Vehicular fog computing: architecture, use case, and security and forensic challenges. *IEEE Communications Magazine*, 55(11):105–111, 2017.

[17] Amir Vahid Dastjerdi, Harshit Gupta, Rodrigo N Calheiros, Soumya K Ghosh, and Rajkumar Buyya. Fog computing: Principles, architectures, and applications. In *Internet of things*, pages 61–75. Elsevier, 2016.

[18] Charles C Byers. Architectural imperatives for fog computing: Use cases, requirements, and architectural techniques for fog-enabled iot networks. *IEEE Communications Magazine*, 55(8):14–20, 2017.

[19] Khaled S El Gayyar, Ahmed I Saleh, and Labib M Labib. A new fog-based routing strategy (FBRS) for vehicular ad-hoc networks. *Peer-to-Peer Networking and Applications*, 15(1):386–407, 2022.

[20] Zainab H Ali, Mahmoud M Badawy, and Hesham A Ali. A novel geographically distributed architecture based on fog technology for improving vehicular ad hoc network (VANET) performance. *Peer-to-Peer Networking and Applications*, 13(5):1539–1566, 2020.

[21] Julian Bellendorf and Zoltán Ádám Mann. Classification of optimization problems in fog computing. *Future Generation Computer Systems*, 107:158–176, 2020.

[22] Tom Goethals, Filip De Turck, and Bruno Volckaert. Self-organizing fog support services for responsive edge computing. *Journal of Network and Systems Management*, 29(2):1–33, 2021.

[23] Kim-Kwang Raymond Choo, Rongxing Lu, Liqun Chen, and Xun Yi. A foggy research future: Advances and future opportunities in fog computing research, 2018.

[24] Duc-Nghia Vu, Nhu-Ngoc Dao, Woongsoo Na, and Sungrae Cho. Dynamic resource orchestration for service capability maximization in fog-enabled connected vehicle networks. *IEEE Transactions on Cloud Computing*, 2020.

[25] Chunshan Liu, Min Li, Stephen V Hanly, and Philip Whiting. Joint downlink user association and interference management in two-tier hetnets with dynamic resource partitioning. *IEEE Transactions on Vehicular Technology*, 66(2):1365–1378, 2016.

[26] Francesco Pantisano, Mehdi Bennis, Walid Saad, and Mérouane Debbah. Match to cache: Joint user association and backhaul allocation in cache-aware small cell networks. In *2015 IEEE International Conference on Communications (ICC)*, pages 3082–3087. IEEE, 2015.

[27] Qiaoyang Ye, Ozgun Yilmaz Bursalioglu, Haralabos C Papadopoulos, Constantine Caramanis, and Jeffrey G Andrews. User association and interference management in massive MIMO hetnets. *IEEE Transactions on Communications*, 64(5):2049–2065, 2016.

[28] Fakhar Abbas, Gang Liu, Pingzhi Fan, Zahid Khan, and Muhammad Saleh Bute. A vehicle density based two-stage resource management scheme for 5g-v2x networks. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pages 1–5. IEEE, 2020.

[29] Arindam Ghosh, Vishnu Vardhan Paranthaman, Glenford Mapp, Orhan Gemikonakli, and Jonathan Loo. Enabling seamless v2i communications: toward developing cooperative automotive applications in VANET systems. *IEEE Communications Magazine*, 53(12):80–86, 2015.

[30] Yansong Li, Qian Luo, Jiajia Liu, Hongzhi Guo, and Nei Kato. Tsp security in intelligent and connected vehicles: Challenges and solutions. *IEEE Wireless Communications*, 26(3):125–131, 2019.

[31] Claudia Campolo, Antonella Molinaro, and Antoine O Berthet. Full-duplex communications to improve platooning control in multi-channel VANETs. In *2017 IEEE international conference on communications workshops (ICC workshops)*, pages 936–941. IEEE, 2017.

[32] Satish Vemireddy and Rashmi Ranjan Rout. Clustering based energy efficient multi-relay scheduling in green vehicular infrastructure. *Vehicular Communications*, 25:100251, 2020.

[33] Shanzhi Chen, Jinling Hu, Yan Shi, and Li Zhao. LTE-V: A TD-LTE-based V2X solution for future vehicular network. *IEEE Internet of Things journal*, 3(6):997–1005, 2016.

[34] 3GPP. *Technical Specification Group Radio Access Network: A Study on LTE-based V2X Services*. document 3GPP TR 36.885 V2.0.0 (Release 14), 2016.

[35] Sobia Jangsher, Haojie Zhou, Victor OK Li, and Ka-Cheong Leung. Joint allocation of resource blocks, power, and energy-harvesting relays in cellular networks. *IEEE Journal on Selected Areas in Communications*, 33(3):482–495, 2015.

[36] Nhu-Ngoc Dao, Minho Park, Joongheon Kim, Jeongyeup Paek, and Sungrae Cho. Resource-aware relay selection for inter-cell interference avoidance in 5G heterogeneous network for internet of things systems. *Future Generation Computer Systems*, 93:877–887, 2019.

[37] Haibin Zhang, Qian Zhang, Jiajia Liu, and Hongzhi Guo. Fault detection and repairing for intelligent connected vehicles based on dynamic bayesian network model. *IEEE Internet of Things Journal*, 5(4):2431–2440, 2018.

[38] Ribal F Atallah, Chadi M Assi, and Jia Yuan Yu. A reinforcement learning technique for optimizing downlink scheduling in an energy-limited vehicular network. *IEEE Transactions on Vehicular Technology*, 66(6):4592–4601, 2016.

[39] Nhu-Ngoc Dao, Junwook Lee, Duc-Nghia Vu, Jeongyeup Paek, Joongheon Kim, Sungrae Cho, Ki-Sook Chung, and Changsup Keum. Adaptive resource balancing for serviceability maximization in fog radio access networks. *IEEE Access*, 5:14548–14559, 2017.

[40] Abdulla A Hammad, Terence D Todd, George Karakostas, and Dongmei Zhao. Downlink traffic scheduling in green vehicular roadside infrastructure. *IEEE Transactions on Vehicular Technology*, 62(3):1289–1302, 2012.

[41] Sadoon Azizi, Mohammad Shojafar, Jemal Abawajy, and Rajkumar Buyya. Deadline-aware and energy-efficient iot task scheduling in fog computing systems: A semi-greedy approach. *Journal of network and computer applications*, 201:103333, 2022.

[42] John A Stankovic, Marco Spuri, Krithi Ramamritham, and Giorgio Buttazzo. *Deadline scheduling for real-time systems: EDF and related algorithms*, volume 460. Springer Science & Business Media, 1998.

[43] Wei Zhao and John A Stankovic. Performance analysis of fcfs and improved fcfs scheduling algorithms for dynamic real-time computer systems. In *1989 Real-Time Systems Symposium*, pages 156–157. IEEE Computer Society, 1989.

[44] Sanjaya K Panda and Prasanta K Jana. An energy-efficient task scheduling algorithm for heterogeneous cloud computing systems. *Cluster Computing*, 22(2):509–527, 2019.

[45] Puneet Kansal, Manoj Kumar, and Om Prakash Verma. Classification of resource management approaches in fog/edge paradigm and future research prospects: a systematic review. *The Journal of Supercomputing*, 78(11):13145–13204, 2022.

[46] Jingyun Feng, Zhi Liu, Celimuge Wu, and Yusheng Ji. Ave: Autonomous vehicular edge computing framework with aco-based scheduling. *IEEE Transactions on Vehicular Technology*, 66(12):10660–10675, 2017.

[47] Xueshi Hou, Yong Li, Min Chen, Di Wu, Depeng Jin, and Sheng Chen. Vehicular fog computing: A viewpoint of vehicles as the infrastructures. *IEEE Transactions on Vehicular Technology*, 65(6):3860–3873, 2016.

[48] Hakim Badis and Abderrezak Rachedi. Modeling tools to evaluate the performance of wireless multi-hop networks. In *Modeling and Simulation of Computer Networks and Systems*, pages 653–682. Elsevier, 2015.

[49] Muhammad Rizwan Ghori, Kamal Z Zamli, Nik Quosthoni, Muhammad Hisyam, and Mohamed Montaser. Vehicular ad-hoc network (VANET). In *2018 IEEE international conference on innovative research and development (ICIRD)*, pages 1–6. IEEE, 2018.

[50] RP Roess, ES Prassas, and WR McShane. Traffic engineering,(4thedn), 2011.

[51] Yu Wang, Jun Zheng, and Nathalie Mitton. Delivery delay analysis for roadside unit deployment in vehicular ad hoc networks with intermittent connectivity. *IEEE Transactions on Vehicular Technology*, 65(10):8591–8602, 2015.

[52] Baihong Dong, Jian Deng, Weigang Wu, and Tianyu Meng. Topology analysis system for vehicular ad hoc network. In *2016 17th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, pages 299–302. IEEE, 2016.

[53] Wasan Kadhim Saad, Ibraheem Shayea, Abdulraqeb Alhammadi, Muntasir Mohammad Sheikh, and Ayman A El-Saleh. Handover and load balancing self-optimization models in 5G mobile networks. *Engineering Science and Technology, an International Journal*, 42:101418, 2023.

[54] Jayashree Patil, Nandini Sidnal, et al. Comparative study of intelligent computing technologies in VANET for delay sensitive applications. *Global Transitions Proceedings*, 2(1):42–46, 2021.

[55] Asim Ihsan, Wen Chen, Shunqing Zhang, and Shugong Xu. Energy-efficient NOMA multicasting system for beyond 5G cellular V2X communications with imperfect CSI. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):10721–10735, 2021.

[56] Pavle Belanovic, Danilo Valerio, Alexander Paier, Thomas Zemen, Fabio Ricciato, and Christoph F Mecklenbrauker. On wireless links for vehicle-to-infrastructure communications. *IEEE Transactions on vehicular technology*, 59(1):269–282, 2009.

[57] Syed S Husain, Andreas Kunz, Athul Prasad, Emmanouil Pateromichelakis, and Konstantinos Samdanis. Ultra-high reliable 5G V2X communications. *IEEE Communications Standards Magazine*, 3(2):46–52, 2019.

[58] Mehrdad Shirkhani, Zeinab Tirkan, and Abbas Taherpour. Performance analysis and optimization of two-way cooperative communications in inter-vehicular networks. In *2012 International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1–6. IEEE, 2012.

[59] Tuan-Duc Nguyen, Olivier Berder, and Olivier Sentieys. Energy-efficient cooperative techniques for infrastructure-to-vehicle communications. *IEEE Transactions on Intelligent Transportation Systems*, 12(3):659–668, 2011.

[60] Yaser P Fallah, Ching-Ling Huang, Raja Sengupta, and Hariharan Krishnan. Analysis of information dissemination in vehicular ad-hoc networks with application to cooperative vehicle safety systems. *IEEE Transactions on Vehicular Technology*, 60(1):233–247, 2010.

[61] Junhua Wang, Kai Liu, Ke Xiao, Chao Chen, Weiwei Wu, Victor CS Lee, and Sang Hyuk Son. Dynamic clustering and cooperative scheduling for vehicle-to-vehicle communication in bidirectional road scenarios. *IEEE Transactions on Intelligent Transportation Systems*, 19(6):1913–1924, 2017.

[62] Venkatesh Ramaiyan, Eitan Altman, and Anurag Kumar. Delay optimal scheduling in a two-hop vehicular relay network. *Mobile Networks and Applications*, 15:97–111, 2010.

[63] Yuanjie Wang, Yinsheng Liu, Jiayi Zhang, Haina Ye, and Zhenhui Tan. Cooperative store–carry–forward scheme for intermittently connected vehicular networks. *IEEE Transactions on Vehicular Technology*, 66(1):777–784, 2016.

[64] Jieqiong Chen, Ammar Zafar, Guoqiang Mao, and Changle Li. On the achievable throughput of cooperative vehicular networks. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, 2016.

[65] Byungjin Ko, Kai Liu, Sang Hyuk Son, and Kyung-Joon Park. RSU-assisted adaptive scheduling for vehicle-to-vehicle data sharing in bidirectional road scenarios. *IEEE Transactions on Intelligent Transportation Systems*, 22(2):977–989, 2020.

[66] Syed Hassan Ahmed, Muhammad Azfar Yaqub, Safdar Hussain Bouk, and Dongkyun Kim. Smartcop: Enabling smart traffic violations ticketing in vehicular named data networks. *Mobile Information Systems*, 2016, 2016.

[67] Duc-Nghia Vu, Nhu-Ngoc Dao, and Sungrae Cho. Downlink sum-rate optimization leveraging hungarian method in fog radio access networks. In *2018 International Conference on Information Networking (ICOIN)*, pages 56–60. IEEE, 2018.

[68] Xin Ge, Xiuhua Li, Hu Jin, Julian Cheng, and Victor CM Leung. Joint user association and user scheduling for load balancing in heterogeneous networks. *IEEE Transactions on Wireless Communications*, 17(5):3211–3225, 2018.

[69] Hongli He, Hangguan Shan, Aiping Huang, and Long Sun. Resource allocation for video streaming in heterogeneous cognitive vehicular networks. *IEEE Transactions on Vehicular Technology*, 65(10):7917–7930, 2016.

[70] Ermioni Qafzezi, Kevin Bylykbashi, Makoto Ikeda, Keita Matsuo, and Leonard Barolli. Coordination and management of cloud, fog and edge resources in SDN-VANETs using fuzzy logic: a comparison study for two fuzzy-based systems. *Internet of Things*, 11:100169, 2020.

[71] Entesar Hosseini, Mohsen Nickray, and Shamsollah Ghanbari. Optimized task scheduling for cost-latency trade-off in mobile fog computing using fuzzy analytical hierarchy process. *Computer Networks*, 206:108752, 2022.

[72] Thaha Mohammed, Behrouz Jedari, and Mario Di Francesco. Efficient and fair multi-resource allocation in dynamic fog radio access network slicing. *IEEE Internet of Things Journal*, 9(24):24600–24614, 2022.

[73] Muhammad Adnan, Jawaid Iqbal, Abdul Waheed, Noor Ul Amin, Mahdi Zareei, Shidrokh Goudarzi, and Asif Umer. On the design of efficient hierarchic architecture for software defined vehicular networks. *Sensors*, 21(4):1400, 2021.

[74] Tianqing Zhou, Yongming Huang, Lixing Fan, and Luxi Yang. Load-aware user association with quality of service support in heterogeneous cellular networks. *IET Communications*, 9(4):494–500, 2015.

[75] Qiang Zheng, Kan Zheng, Haijun Zhang, and Victor CM Leung. Delay-optimal virtualized radio resource scheduling in software-defined vehicular networks via stochastic learning. *IEEE Transactions on Vehicular Technology*, 65(10):7857–7867, 2016.

[76] Md Mehedi Hasan, Sungoh Kwon, and Jee-Hyeon Na. Adaptive mobility load balancing algorithm for LTE small-cell networks. *IEEE transactions on wireless communications*, 17(4):2205–2217, 2018.

[77] Dantong Liu, Lifeng Wang, Yue Chen, Maged Elkashlan, Kai-Kit Wong, Robert Schober, and Lajos Hanzo. User association in 5G networks: A survey and an outlook. *IEEE Communications Surveys & Tutorials*, 18(2):1018–1044, 2016.

[78] Rose Qingyang Hu and Yi Qian. An energy efficient and spectrum efficient wireless heterogeneous network framework for 5G systems. *IEEE Communications Magazine*, 52(5):94–101, 2014.

[79] Dantong Liu, Yue Chen, Kok Keong Chai, Tiankui Zhang, and Maged Elkashlan. Opportunistic user association for multi-service hetnets using nash bargaining solution. *IEEE Communications Letters*, 18(3):463–466, 2014.

[80] Abdulla A Hammad, Ghada H Badawy, Terence D Todd, Amir A Sayegh, and Dongmei Zhao. Traffic scheduling for energy sustainable vehicular infrastructure. In *2010*

IEEE global telecommunications conference GLOBECOM 2010, pages 1–6. IEEE, 2010.

[81] Lei Zhang and Yu Wang. An offline roadside unit ON-OFF scheduling algorithm for energy efficiency of ad hoc networks. *IEEE Access*, 6:59742–59751, 2018.

[82] Wassim Sellil Atoui, Wessam Ajib, and Mounir Boukadoum. Offline and online scheduling algorithms for energy harvesting RSUs in VANETs. *IEEE Transactions on Vehicular Technology*, 67(7):6370–6382, 2018.

[83] Om-Kolsoom Shahryari, Hossein Pedram, Vahid Khajehvand, and Mehdi Dehghan TakhtFooladi. Energy-efficient and delay-guaranteed computation offloading for fog-based iot networks. *Computer Networks*, 182:107511, 2020.

[84] Youngsu Jang, Jinyeop Na, Seongah Jeong, and Joonhyuk Kang. Energy-efficient task offloading for vehicular edge computing: Joint optimization of offloading and bit allocation. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pages 1–5. IEEE, 2020.

[85] Amir Khezrian, Terence D Todd, George Karakostas, and Morteza Azimifar. Energy-efficient scheduling in green vehicular infrastructure with multiple roadside units. *IEEE Transactions on Vehicular Technology*, 64(5):1942–1957, 2014.

[86] Mohammadreza Nazeri, Mohammadreza Soltanaghaei, and Reihaneh Khorsand. A predictive energy-aware scheduling strategy for scientific workflows in fog computing. *Expert Systems with Applications*, page 123192, 2024.

[87] Pejman Hosseinioun, Maryam Kheirabadi, Seyed Reza Kamel Tabbakh, and Reza Ghaemi. A new energy-aware tasks scheduling approach in fog computing using hybrid meta-heuristic algorithm. *Journal of Parallel and Distributed Computing*, 143:88–96, 2020.

[88] Rodrigo Roman, Javier Lopez, and Masahiro Mambo. Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges. *Future Generation Computer Systems*, 78:680–698, 2018.

[89] Mehdi Sookhak, F Richard Yu, Ying He, Hamid Talebian, Nader Sohrabi Safa, Nan Zhao, Muhammad Khurram Khan, and Neeraj Kumar. Fog vehicular computing: Augmentation of fog computing using vehicular cloud computing. *IEEE Vehicular Technology Magazine*, 12(3):55–64, 2017.

[90] Arnav Thakur and Reza Malekian. Fog computing for detecting vehicular congestion, an internet of vehicles based approach: A review. *IEEE Intelligent Transportation Systems Magazine*, 11(2):8–16, 2019.

[91] Yuxuan Jiang and Danny HK Tsang. Delay-aware task offloading in shared fog networks. *IEEE Internet of Things Journal*, 5(6):4945–4956, 2018.

[92] Indranil Sarkar, Mainak Adhikari, Neeraj Kumar, and Sanjay Kumar. Dynamic task placement for deadline-aware iot applications in federated fog networks. *IEEE Internet of Things Journal*, 9(2):1469–1478, 2021.

[93] Mithun Mukherjee, Mian Guo, Jaime Lloret, Razi Iqbal, and Qi Zhang. Deadline-aware fair scheduling for offloaded tasks in fog computing with inter-fog dependency. *IEEE Communications Letters*, 24(2):307–311, 2019.

[94] Mainak Adhikari, Mithun Mukherjee, and Satish Narayana Srirama. DPTO: A deadline and priority-aware task offloading in fog computing framework leveraging multilevel feedback queueing. *IEEE Internet of Things Journal*, 7(7):5773–5782, 2019.

[95] Chaogang Tang, Xianglin Wei, Chunsheng Zhu, Yi Wang, and Weijia Jia. Mobile vehicles as fog nodes for latency optimization in smart cities. *IEEE Transactions on Vehicular Technology*, 69(9):9364–9375, 2020.

[96] Chao Zhu, Giancarlo Pastor, Yu Xiao, Yong Li, and Antti Ylae-Jaeaeski. Fog following me: Latency and quality balanced task allocation in vehicular fog computing. In *2018 15th Annual IEEE international conference on sensing, communication, and networking (SECON)*, pages 1–9. IEEE, 2018.

[97] Mainak Adhikari, Satish Narayana Srirama, and Tarachand Amgoth. Application offloading strategy for hierarchical fog environment through swarm optimization. *IEEE Internet of Things Journal*, 7(5):4317–4328, 2019.

[98] Seung-seob Lee and SuKyoung Lee. Resource allocation for vehicular fog computing using reinforcement learning combined with heuristic information. *IEEE Internet of Things Journal*, 7(10):10450–10464, 2020.

[99] Dong Uk Kim, Seong Bae Park, Choong Seon Hong, and Eui Nam Huh. Resource allocation and user association using reinforcement learning via curriculum in a wireless network with high user mobility. In *2023 International Conference on Information Networking (ICOIN)*, pages 382–386. IEEE, 2023.

[100] Taha Alfakih, Mohammad Mehedi Hassan, Abdu Gumaei, Claudio Savaglio, and Giancarlo Fortino. Task offloading and resource allocation for mobile edge computing by deep reinforcement learning based on sarsa. *IEEE Access*, 8:54074–54084, 2020.

[101] Abir Mchergui and Tarek Moulahi. A novel deep reinforcement learning based relay selection for broadcasting in vehicular ad hoc networks. *IEEE Access*, 10:112–121, 2021.

[102] Shangdong Yang and Yang Gao. An optimal algorithm for the stochastic bandits while knowing the near-optimal mean reward. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):2285–2291, 2020.

[103] Richard S Sutton and Andrew G Barto. Reinforcement learning: An introduction. *Robotica*, 17(2):229–235, 1999.

[104] Yifei Wei, F Richard Yu, Mei Song, and Zhu Han. Joint optimization of caching, computing, and radio resources for fog-enabled IoT using natural actor–critic deep reinforcement learning. *IEEE Internet of Things Journal*, 6(2):2061–2073, 2018.

[105] Chi Harold Liu, Zipeng Dai, Yinuo Zhao, Jon Crowcroft, Dapeng Wu, and Kin K Leung. Distributed and energy-efficient mobile crowdsensing with charging stations by deep reinforcement learning. *IEEE Transactions on Mobile Computing*, 20(1):130–146, 2019.

[106] Xingjian Li, Jun Fang, Wen Cheng, Huiping Duan, Zhi Chen, and Hongbin Li. Intelligent power control for spectrum sharing in cognitive radios: A deep reinforcement learning approach. *IEEE access*, 6:25463–25473, 2018.

[107] Ying He, F Richard Yu, Nan Zhao, Hongxi Yin, and Azzedine Boukerche. Deep reinforcement learning (drl)-based resource management in software-defined and virtualized vehicular ad hoc networks. In *Proceedings of the 6th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications*, pages 47–54, 2017.

[108] Satish Vemireddy and Rashmi Ranjan Rout. Fuzzy reinforcement learning for energy efficient task offloading in vehicular fog computing. *Computer Networks*, 199:108463, 2021.

[109] Lina Altoaimy and Imad Mahgoub. Fuzzy logic based localization for vehicular ad hoc networks. In *2014 IEEE symposium on computational intelligence in vehicles and transportation systems (CIVTS)*, pages 121–128. IEEE, 2014.

[110] Elnaz Limouchi, Imad Mahgoub, and Ahmad Alwakeel. Fuzzy logic-based broadcast in vehicular ad hoc networks. In *2016 IEEE 84th vehicular technology conference (VTC-fall)*, pages 1–5. IEEE, 2016.

[111] Ejaz Ahmed and Hamid Gharavi. Cooperative vehicular networking: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 19(3):996–1014, 2018.

[112] Jaehoon Jeong, Hohyeon Jeong, Eunseok Lee, Tae Oh, and David HC Du. SAINT: Self-adaptive interactive navigation tool for cloud-based vehicular traffic optimization. *IEEE Transactions on Vehicular Technology*, 65(6):4053–4067, 2015.

[113] Yiwen Shen, Jinho Lee, Hohyeon Jeong, Jaehoon Jeong, Eunseok Lee, and David HC Du. SAINT+: Self-adaptive interactive navigation tool+ for emergency service delivery optimization. *IEEE Transactions on Intelligent Transportation Systems*, 19(4):1038–1053, 2017.

[114] Sebastian Van De Hoef, Karl Henrik Johansson, and Dimos V Dimarogonas. Fuel-efficient en route formation of truck platoons. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):102–112, 2017.

[115] Emmanouil Koukoumidis, Li-Shiuan Peh, and Margaret Rose Martonosi. Signalguru: leveraging mobile phones for collaborative traffic signal schedule advisory. In *Proceedings of the 9th international conference on Mobile systems, applications, and services*, pages 127–140, 2011.

[116] Yiwen Shen, Jaehoon Jeong, Tae Oh, and Sang Hyuk Son. Casd: a framework of context-awareness safety driving in vehicular networks. In *2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pages 252–257. IEEE, 2016.

[117] Chaojie Wang, Siyuan Gong, Anye Zhou, Tao Li, and Srinivas Peeta. Cooperative adaptive cruise control for connected autonomous vehicles by factoring communication-related constraints. *Transportation Research Part C: Emerging Technologies*, 113:124–145, 2020.

[118] Alessandro Bazzi, Barbara M Masini, Alberto Zanella, and Ilaria Thibault. On the performance of IEEE 802.11p and LTE-V2V for the cooperative awareness of connected vehicles. *IEEE Transactions on Vehicular Technology*, 66(11):10419–10432, 2017.

[119] Stephan Eichler. Performance evaluation of the IEEE 802.11p WAVE communication standard. In *2007 IEEE 66th Vehicular Technology Conference*, pages 2199–2203. IEEE, 2007.

[120] Moray Rumney et al. *Physical layer: LTE and the evolution to 4G wireless: Design and measurement challenges*. John Wiley & Sons, 2013.

[121] Qiong Yang, Lin Wang, Weiwei Xia, Yi Wu, and Lianfeng Shen. Development of on-board unit in vehicular ad-hoc network for highways. In *2014 international conference on connected vehicles and expo (ICCVE)*, pages 457–462. IEEE, 2014.

[122] Claudia Campolo, Antonella Molinaro, Alexey Vinel, and Yan Zhang. Modeling and enhancing infotainment service access in vehicular networks with dual-radio devices. *Vehicular communications*, 6:7–16, 2016.

[123] Jason W Rupe. Network nodal independence, hierarchical path search, and model reuse for network availability computation. *IEEE Transactions on Reliability*, 65(4):1842–1851, 2016.

[124] Z Bai et al. Evolved universal terrestrial radio access (e-utra); physical layer procedures. *3GPP, Sophia Antipolis, Technical Specification*, 36, 2013.

[125] L ETSI. Evolved universal terrestrial radio access (e-utra); physical channels and modulation. *ETSI TS*, 136(211):V9, 2011.

[126] Jin Li, Ju Bin Song, and Zhu Han. Network connectivity optimization for device-to-device wireless system with femtocells. *IEEE Transactions on Vehicular Technology*, 62(7):3098–3109, 2013.

[127] Sven O Krumke and Clemens Thielen. The generalized assignment problem with minimum quantities. *European Journal of Operational Research*, 228(1):46–55, 2013.

[128] Shie-Yuan Wang. The effects of wireless transmission range on path lifetime in vehicle-formed mobile ad hoc networks on highways. In *IEEE International Conference on Communications, 2005. ICC 2005. 2005*, volume 5, pages 3177–3181. IEEE, 2005.

[129] Christoph Sommer, David Eckhoff, Reinhard German, and Falko Dressler. A computationally inexpensive empirical model of IEEE 802.11 p radio shadowing in urban environments. In *2011 Eighth international conference on wireless on-demand network systems and services*, pages 84–90. IEEE, 2011.

[130] Zhigang Rong and Theodore S Rappaport. *Wireless communications: Principles and practice, solutions manual.* Prentice Hall, 1996.

[131] Dieter Jungnickel. Weighted matchings. *Graphs, Networks and Algorithms*, pages 441–479, 2013.

[132] Saeedeh Parsaeefard, Vikas Jumba, Atoosa Dalili Shoaei, Mahsa Derakhshani, and Tho Le-Ngoc. User association in cloud RANs with massive MIMO. *IEEE Transactions on Cloud Computing*, 9(2):821–833, 2018.

[133] Hongjia Wu, Jiao Zhang, Zhiping Cai, Fang Liu, Yangyang Li, and Anfeng Liu. Toward energy-aware caching for intelligent connected vehicles. *IEEE Internet of Things Journal*, 7(9):8157–8166, 2020.

[134] Sudip Misra and Niloy Saha. Detour: Dynamic task offloading in software-defined fog for iot applications. *IEEE Journal on Selected Areas in Communications*, 37(5):1159–1166, 2019.

[135] Tuyen X Tran and Dario Pompili. Joint task offloading and resource allocation for multi-server mobile-edge computing networks. *IEEE Transactions on Vehicular Technology*, 68(1):856–868, 2018.

[136] Fan Yang, Chong Zhao, Xu Ding, and Jianghong Han. An analytical model for energy harvest road side units deployment with dynamic service radius in vehicular ad-hoc networks. *IEEE Access*, 8:122589–122598, 2020.

List of Publications

List of International Journals:

1. **Thanedar Md Asif** and Sanjaya Kumar Panda, "A dynamic resource management algorithm for maximizing service capability in fog-empowered vehicular ad-hoc networks", *Peer-to-Peer Networking and Applications*, Springer, Vol. 16, pp. 932–946, 2023. (**Indexed in SCI/SCIE and Scopus, Impact Factor (IF): 4.2, Published**)
   DOI: https://doi.org/10.1007/s12083-023-01451-7

2. **Thanedar Md Asif** and Sanjaya Kumar Panda, "Energy and priority-aware scheduling algorithm for handling delay-sensitive tasks in fog-enabled vehicular networks", *The Journal of Supercomputing*, Springer. (**Indexed in SCI/SCIE and Scopus, IF: 3.3, Published**)
   DOI: https://doi.org/10.1007/s11227-024-06004-0

3. **Thanedar Md Asif** and Sanjaya Kumar Panda, "An energy-efficient resource allocation algorithm for managing on-demand services in fog-enabled vehicular adhoc networks", *International Journal of Web and Grid Services*, Inderscience, 2024. (**Indexed in SCI/SCIE and Scopus, IF: 3.8, Published**)
   DOI: https://doi.org/10.1504/IJWGS.2024.10061105

4. **Thanedar Md Asif** and Sanjaya Kumar Panda, "An efficient resource orchestration algorithm for enhancing throughput in fog computing-enabled vehicular networks", *Pervasive and Mobile Computing*, Elsevier. (**Indexed in SCI/SCIE and Scopus, IF: 4.3, Under Revision**)

5. **Thanedar Md Asif** and Sanjaya Kumar Panda, "Energy-efficient and delay-aware task scheduling in energy-limited fog-empowered vehicular networks using Q-learning", *IEEE Transactions on Vehicular Technology*. (**Indexed in SCI/SCIE and Scopus, IF: 6.8, Under Review**)

List of International Conferences:

1. **Thanedar Md Asif** and Sanjaya Kumar Panda, "Delay-aware task offloading for mobile fog nodes in smart cities: A fuzzy and q-learning approach", in Proceedings of the 15^{th} Student Research Symposium, 30^{th} IEEE International Conference on High Performance Computing, Data, and Analytics (HiPC), 18-21 December 2023, Goa, India. (**Core Rank: B**)