Resource Management Optimization in Next-Generation Wireless Networks: From Network Slicing to Vehicular Communications

Thesis presented and defended by

SOLOMON ORDUEN YESE

PhD thesis in

SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

Defended on December 18th, 2025

Composition du jury

GUNES KARABULUT KURT	Professeure, Polytechnique Montréal	President
NADJIB AIT SAADI	Professeur, UVSQ Paris-Saclay	Rapporteur
PERIKLIS CHATZIMISIOS	Professeur, International Hellenic University, Greece	Rapporteur
VERONIQUE VEQUE	Professeure, Université Paris-Saclay / CentraleSupélec	Examiner
SAHAR HOTEIT	Associate Professor, Université Paris-Saclay / CentraleSupélec	Examiner
ABDUL KARIM GIZZINI	Associate Professor, Université Paris-Est Créteil	Examiner
ARSENIA CHORTI	Professeure, ETIS, CY Cergy Paris, ENSEA, CNRS	Directrice
SARA BERRI	Associate Professor, ETIS, CY Cergy Paris, ENSEA, CNRS	Supervisor

Titre: Optimisation de la Gestion des Ressources dans les Réseaux sans Fil de Nouvelle Génération : du Découpage du Réseau aux Communications Véhiculaires

Mots clés: contrôle d'admission, découpage du réseau, 5G/B5G, utilisation des ressources, priorité dynamique, tarification dynamique, O-RAN, association utilisateur, consommation d'énergie, délai, économie d'énergie, réseau, optimisation, faible latence, fiabilité, V2X.

Résumé: L'évolution vers les réseaux 5G et au-delà a introduit des défis sans précédent en matière de gestion des ressources, d'optimisation des réseaux et d'approvisionnement en termes de qualité de service (QoS) dans des architectures réseau hétérogènes. Cette thèse aborde les problèmes d'optimisation ouverts dans les réseaux sans fil de nouvelle génération en se concentrant sur trois domaines clés : *i)* le contrôle d'admission au découpage du réseau *ii)* la gestion des ressources du réseau d'accès radio ouvert (O-RAN) et *iii)* la sélection intelligente de la technologie d'accès radio (RAT) pour les communications véhiculaires.

Les nouvelles solutions de contrôle d'admission que nous proposons pour le découpage du réseau intègrent des mécanismes de surréservation et des processus de rachat de ressources dynamiques, maximisant les revenus des fournisseurs d'infrastructure (InP) tout en respectant les accords de niveau de service (SLA). Nos solutions utilisent des mécanismes d'adaptation dynamique des priorités, de mise en commun progressive des ressources et de partage des capacités pour gérer efficacement les demandes de tranches de réseau dans des scénarios hors ligne et en ligne où les informations sur les demandes futures sont limitées et où les demandes peuvent être annulées après expiration du délai d'attente. L'évaluation des performances montre que les solutions de contrôle d'admission proposées génèrent des profits plus élevés, une meilleure utilisation des ressources, ainsi que des taux d'acceptation des tranches légèrement plus élevés comparé aux solutions de l'état de l'art.

Pour les architectures O-RAN, nous développons des algorithmes d'association d'utilisateurs écoénergétiques qui optimisent conjointement l'association des équipements utilisateurs (UE) avec les unités radio (RU), les unités distribuées (DU) et les unités centrales (CU) tout en garantissant les délais. Nos approches minimisent la consommation d'énergie grâce à une association dynamique des unités et à des mécanismes de contrôle de puissance qui permettent aux composants inactifs du réseau de passer en mode veille, tout en respectant les exigences strictes en matière de qualité de service.

Dans le domaine des communications véhiculaires, nous présentons un environnement multi-objectif d'apprentissage par renforcement profond pour la sélection des RAT dans les scénarios véhicule-à-infrastructure/réseau (V2I/V2N). Notre solution permet une prise de décision en temps réel dans des conditions d'incertitude en optimisant simultanément la latence, la fiabilité et la fréquence des transferts dans des environnements véhiculaires en évolution rapide.

Les solutions proposées apportent collectivement une progression à l'état de l'optimisation des réseaux dans les systèmes sans fil de nouvelle génération tout en répondant aux défis pratiques du déploiement dans les infrastructures réseau modernes.



Title: Resource Management Optimization in Next-Generation Wireless Networks: From Network Slicing to Vehicular Communications

Keywords: admission control, network slicing, 5G/B5G, resource utilization, dynamic priority, dynamic pricing, O-RAN, UE/user association, energy consumption, delay, energy saving, network, optimisation, low latency, reliability, V2X

Abstract: The evolution toward 5G and beyond networks has introduced unprecedented challenges in resource management, network optimization, and quality of service (QoS) provisioning across heterogeneous network architectures. This thesis addresses some open optimization problems in next-generation wireless networks, focusing on three key domains: *i)* network slicing admission control *ii)* open radio access network (O-RAN) resource management and *iii)* intelligent radio access technology (RAT) selection for vehicular communications.

The novel admission control schemes we propose for network slicing incorporate overbooking mechanisms and dynamic buyback processes, maximizing infrastructure provider (InP) revenue while upholding service level agreements (SLAs). Our solutions employ dynamic priority adaptation, step-wise pooling, and capacity sharing mechanisms to efficiently manage slice requests in both offline and online scenarios where future request knowledge is limited and requests can renege after waiting time expiration. The evaluation of the proposed slice admission control solutions against state-of-the-art (SoA) solutions show that they yield higher profit and better resource utilization as well as marginally higher slice acceptance rates.

As far as our contribution in the domain of O-RAN is concerned, we develop energy-efficient user association algorithms that jointly optimize user equipment (UE) association with radio units (RUs), distributed units (DUs) and central units (CUs) while ensuring delay guarantees. Our approaches minimize energy consumption through dynamic unit association and power control mechanisms that enable idle network components to enter into sleep mode, while maintaining stringent QoS requirements. We also evaluate our proposed solutions against SoA solutions and our solutions yield significant reductions in energy consumption at very minimal delay cost but without violating the delay requirements of the UEs.

In the vehicular communication domain, we present a multi-objective deep reinforcement learning framework for RAT selection in vehicle to-infrastructure/network (V2I/V2N) scenarios. Our solution enables real-time decision making under uncertainty, simultaneously optimizing latency, reliability, and handover frequency in rapidly changing vehicular environments. Comprehensive performance evaluations of our proposed solution via simulations demonstrate significant improvements over SoA solutions, yielding significant reduction in the number of handovers with improved reliability and delay.

The proposed solutions collectively advance the state of network optimization in next-generation wireless systems while addressing practical deployment challenges in modern network infrastructures.

This thesis is dedicated to my late Parents (Mr. Yese and Mrs Bridget Tsegh) who could not live to see the fruition of the wonderful educational foundation they laid.

Acknowledgement

My heartfelt gratitude goes to the Almighty God, the creator of heaven and earth for His enduring mercies towards me over these years. May His name forever be praised. I am particularly indebted to my thesis supervisor; Professor **Sara Berri**, and thesis director Professor **Ersi Arsenia Chorti** for their unwavering support, patience, guidance and contribution towards the success of this work. Through their help, I was able to overcome the numerous obstacles I faced throughout my PhD journey. I feel deeply honored to have been supervised by them. My profound gratitude also goes to the members of the jury: Proffesors **Gunes Karabulut Kurt**, **Veronique Veque**, **Nadjib Ait Saadi**, **Periklis Chatzimisios**, **Sahar Hoteit** and **Abdul Karim Gizzini** for agreeing to and taking time out of their busy schedules to be a part of this memorable occasion of my PhD journey. Thank you all!

I wish to express my profound gratitude generally to the nation and people of Nigeria and in particular to the **Petroleum Technology Development Fund (PTDF)** for the fully funded sponsorship that enabled me to purse this PhD. I also wish to sincerely thank the management and members of the **ETIS laboratory** for the support received and for making the journey quite memorable. Not forgetting Professors **Eleftherios Mamatas** (who did not live to the need of my PhD) and **Aikaterini Tzompanaki** who consistently took time out of their busy schedules for the three consecutive years to be a part of my CSI committe.

My special appreciation also goes my beloved wife Geraldine Yese and kids (Aondoyima and Yesukator Yese) whose love and support made my PhD journey worthwhile. Also, very worthy of note are the families of Dr. Jimin Kwaghsende and Mr. Joakin Elogie, my siblings (Mrs Mwuese kwaghsende, Mrs Jacinta Elogie, Eng. Peter Yese, Miss Karen Yese and Mrs Cynthia Yese) for their supports, not forgetting my spiritual family (RCCG Jesus House for All Nations). Pulling through these trying times without your supports would have been near impossible. May God richly bless you all.

Contents

Acronyms xv

1 Introduction 1

1.1 Context . . . 1

1.2 Research Objectives . . . 5

1.3 Thesis Organization and Dissemination . . . 6

1.4 List of Publications . . . 6

1.4.1 Journals . . . 6

1.4.2 Conferences . . . 7

2 Background Concepts 8

2.1 Evolution of Wireless Networks . . . 8

2.2 Network Virtualization and Software-Defined Networking . . . 11

2.3 Network Slicing: Concepts and Challenges . . . 12

2.4 Vehicular Communications and V2X Technologies . . . 14

2.5 Admission Control in Network Systems . . . 16

2.6 Radio Access Technology Selection . . . 17

2.7 Open Radio Access Network (O-RAN) Architecture . . . 18

2.8 Open Research Questions and Motivation . . . 20

3 Slice Admission Control Schemes in Next Generation Networks 22

3.1 Introduction . . . 23
3.2 Review of Related Works . . . 26
3.3 Slice Admission Control Scheme with Overbooking and Dynamic Buyback Process . . . 34
3.3.1 System Model (P1) . . . 35
3.3.2 Problem Formulation (P1) . . . 36
3.3.3 Proposed Algorithm for P1 . . . 37
3.3.3.1 Time Complexity Analysis for Algorithm 1 . . . 38
3.3.4 Performance Evaluation of the Solution for P1 . . . 40
3.3.5 Simulation Settings for P1 . . . 40
3.3.6 Proposed Priority Schemes . . . 40
3.3.6.1 Revenue-Cost Ratio Based Scheme (RCRS) . . . 41
3.3.6.2 Revenue Based Scheme (RS) . . . 41
3.3.6.3 Priority-Duration Ratio Based Scheme (PDRS) . . . 41
3.3.6.4 Class Based Scheme (CS) . . . 42
3.3.7 Comparison of Proposed Solution for P1 to SoA . . . 42
3.4 Efficient Online Slice Admission Control Scheme in Next Generation Networks . . . 46
3.4.1 System Model (P2) . . . 46
3.4.2 Problem Formulation (P2) . . . 47
3.4.3 Proposed Algorithm for P2 . . . 48
3.4.3.1 Time Complexity Analysis for Algorithm 2 . . . 49
3.4.4 Performance Evaluation of the Proposed Solution for P2 . . . 51
3.4.5 Simulation Settings for P2 . . . 51
3.4.6 Comparison of the Proposed Solution for P2 to SoA . . . 52
3.5 Chapter Summary . . . 56

4 Delay-Guaranteed and Energy Efficient User Association Functions placement in O-RAN 58

4.1 Introduction . . . 59

4.2 Review of Related Works . . . 60
4.3 Energy Efficient and Delay-Guaranteed User Association for O-RAN . . . 67
4.3.1 System Model (P3) . . . 67
4.3.2 Access Network Delay Model . . . 68
4.3.3 RU Power Consumption Model . . . 68
4.3.4 Problem Formulation (P3) . . . 69
4.3.5 Proposed Algorithm for P3 . . . 70
4.3.5.1 Time Complexity Analysis for Algorithm 3 . . . 71
4.3.6 Performance Evaluation of the Proposed Solution for P3 . . . 72
4.3.6.1 Simulation Settings for P3 . . . 72
4.3.6.2 Comparison of Proposed Solution for P3 to SoA . . . 73
4.4 End to End Delay-Guaranteed and Energy Efficient User Association with Functions Placement for O-RAN . . . 77
4.4.1 System model (P4) . . . 77
4.4.2 End to End Delay Model . . . 78
4.4.3 Energy Consumption Model (P4) . . . 81
4.4.3.1 CU Power Consumption Model . . . 81
4.4.3.2 DU Power Consumption Model . . . 82
4.4.4 Problem Formulation (P4) . . . 82
4.4.5 Proposed Algorithm for P4 . . . 84
4.4.5.1 Time Complexity Analysis for Algorithm 4 . . . 86
4.4.6 Performance Evaluation of the Solution Proposed for P4 . . . 86
4.4.6.1 Simulation Settings for P4 . . . 86
4.4.6.2 Comparison of the Proposed Solution for P4 to the Solver Solution . . . 88
4.4.6.3 Comparison of the Proposed Solution for P4 to SoA . . . 91
4.4.6.4 Impact of the Number of RUs . . . 93
4.5 Chapter 4 Summary . . . 97

5 Multi-Objective Deep Reinforcement Learning Based RAT Selection So-

lution for Vehicle-to-Infrastructure Communication **99**

5.1 Introduction . . . 99

5.2 Related Works . . . 101

5.3 RAT Selection in Vehicular Communication . . . 104

5.3.1 System Model (P5) . . . 104

5.3.2 Problem Formulation (P5) . . . 106

5.4 Proposed Algorithm for P5 . . . 107

5.5 Performance Evaluation of the Proposed Solution for P5 . . . 111

5.5.1 Simulation Settings for P5 . . . 111

5.5.2 Comparison of Proposed Solution of P5 to SoA . . . 113

5.5.3 Impact of Reward Function Weighting . . . 114

5.6 Chapter Summary . . . 117

6 Conclusions and Perspectives **118**

6.1 Conclusions . . . 118

6.2 Perspectives . . . 119

6.2.1 Towards Multi-level Network Slicing . . . 120

6.2.2 Holistic O-RAN Optimization . . . 120

6.2.3 Intelligent Vehicular Networking . . . 120

6.2.4 Further Evaluation . . . 120

6.3 Chapter Summary . . . 121

Bibliography **122**

List of Figures

2.1 Mobile networks evolution: from 1 to 6G. 11

2.2 Network slicing architecture. 14

2.3 O-RAN Architecture. 19

3.1 A typical sliced network framework. 24

3.2 System Model (P1). 36

3.3 Comparison in terms of revenue between the proposed algorithm for P1 and the SoA algorithm, for different priority schemes. 43

3.4 Comparison in terms of buyback cost between the proposed algorithm for P1 and the SoA algorithm. 44

3.5 Comparison in terms of net revenue between the proposed algorithm for P1 and the SoA algorithm. 44

3.6 Comparison in terms of acceptance rate between the proposed algorithm for P1 and the SoA algorithm. 45

3.7 System Model (P2). 47

3.8 Comparison of the proposed solution for P2 against SoA algorithms in terms of profit. 53

3.9 Comparison of the proposed solution for P2 against SoA algorithms in terms of resource utilization rate. 54

3.10 Comparison of the proposed solution for P2 against SoA algorithms in terms of profit, for different prices. 56

4.1 System Model (P3). 67

4.2 Total energy consumption of proposed algorithm for P3 and the SoA algorithms. . . . 75
4.3 Comparison in terms of total access network delay for the proposed algorithm for P3 and SoA algorithms. . . . 76
4.4 Comparison in terms of delay per UE between the proposed algorithm for P3 and the SoA algorithms. . . . 76
4.5 System Model (P4). . . . 78
4.6 Comparison of the proposed algorithm for P4 against the exact solution in terms of total energy consumption. . . . 89
4.7 Comparison of the proposed algorithm for P4 against the exact solution in terms of total UEs delay. . . . 90
4.8 Comparison of the proposed algorithm for P4 against the exact solution in terms of per UE delay. . . . 90
4.9 Comparison of the proposed algorithm for P4 against the exact solution in terms of execution time. . . . 91
4.10 Total and component-wise energy consumption of the proposed algorithm for P4 and the SoA algorithms. . . . 93
4.11 Comparison in terms of total end to end delay for the proposed algorithm for P4 and SoA algorithms. . . . 94
4.12 Comparison in terms of per UE end to end delay for the proposed algorithm for P4 and SoA algorithms. . . . 95
4.14 Proposed algorithm for P4 vs SoA algorithms in terms of total UEs delay for different network area and number of RUs setting. . . . 95
4.13 Proposed algorithm for P4 vs SoA algorithms in terms of total energy consumption for different network area and number of RUs setting. . . . 96
4.15 Proposed algorithm for P4 vs SoA algorithms in terms of average delay per UE for different network area and number of RUs. . . . 97

5.1 System Model (P5). . . . 106
5.2 Proposed algorithm for P5 vs SoA algorithm in terms of total vehicle delay. 114
5.3 Proposed algorithm for P5 vs SoA algorithm in terms of achieved reliability. 114
5.4 Proposed algorithm for P5 vs SoA algorithm in terms of number of handovers. 115
5.5 Total vehicle delay. . . . 116

5.6 Achieved reliability. 116
5.7 Comparison in terms of number of handovers. 117

List of Tables

3.1 A Recap of some literature. 34

3.2 Simulation Parameters for P1. 41

3.3 Simulation Parameters for P2. 51

3.4 Acceptance rate for $N \in \{100, \ldots, 800\}$. 53

3.5 The acceptance rate of requests by slice type. 54

4.1 Simulation Parameters for P3. 73

4.2 Simulation Parameters for P4. 88

5.1 Simulation Parameters for P5. 112

5.2 DRL specific Parameters for P5. 113

List of Algorithms

1 Proposed admission control algorithm for P1. 39

2 Proposed algorithm for P2. 50

3 Proposed UE association algorithm for P3. 72

4 Proposed UE association and functions placement algorithm for P4. 87

5 Deep Reinforcement Learning based RAT Selection Algorithm for P5. 110

Acronyms

1G First Generation

2G Second Generation

3G Third Generation

3GPP 3rd Generation Partnership Project

4G Fourth Generation

5G Fifth Generation

6G Sixth Generation

AN Access Network

B5G Beyond Fifth Generation

C-V2X Cellular Vehicle-to-Everything

CAC Connection Admission Control

CAM Cooperative Awareness Message

CAPEX Capital Expenditure

CDMA Code Division Multiple Access

CPU Central Processing Unit

CU Central Unit

DENM Decentralized Environmental Notification Message

DiffServ Differentiated Services

DQL Deep Q-learning

DQN Deep Q-Network

DRL Deep Reinforcement Learning

DSRC Dedicated Short-Range Communications

DU Distributed Unit

E2E End-to-End

ECGSMIOT Extended Coverage GSM IoT

eMBB Enhanced Mobile Broadband

ETSI European Telecommunications Standards Institute

FH Fronthaul

GSM Global System for Mobile Communications

HetNets Heterogeneous Networks

HSPA High-Speed Packet Access

InP Infrastructure Provider

IntServ Integrated Services

IoT Internet of Things

IP Internet Protocol

ITS Intelligent Transportation Systems

LSTM Long Short-Term Memory

LTE Long Term Evolution

MAC Medium Access Control

MCS Modulation and Coding Scheme

MEC Multi-Access Edge Computing

MH Midhaul

MIMO Multiple-Input Multiple-Output

mMTC Massive Machine-Type Communications

NB-IOT Narrowband Internet of Things

near-RT RIC near-Real-Time RAN Intelligent Controller

NFV Network Function Virtualization

NFVI Network Functions Virtualization Infrastructure

NFVO Network Function Virtualization Orchestrator

Non-RT RIC non-Real-Time RAN Intelligent Controller

NR New Radio

O-RAN Open Radio Access Network

OPEX Operational Expenditure

PC5 Direct communication interface (device-to-device)

PDCP Packet Data Convergence Protocol

Q-learning Quality-learning

QoE Quality of Experience

QoS Quality of Service

RAN Radio Access Network

RAT Radio Access Technology

RB Resource Block

RIS Reconfigurable Intelligent Surfaces

RL Reinforcement learning

RLC Radio Link Control

RRC Radio Resource Control

RSSI Received Signal Strength Indicator

RSVP Resource Reservation Protocol

RU Radio Unit

SDN Software-Defined Networking

SINR Signal-to-Interference-plus-Noise Ratio

SLA Service Level Agreement

SoA State-of-the-Art

THz Terahertz

UE User Equipment

UMTS Universal Mobile Telecommunications System

uRLLC Ultra-Reliable Low-Latency Communications

Uu Cellular network interface (UE to infrastructure)

V2I Vehicle-to-Infrastructure

V2N Vehicle-to-Network

V2P Vehicle-to-Pedestrian

V2V Vehicle-to-Vehicle

V2X Vehicle-to-Everything

VIM Virtualized Infrastructure Manager

VNF Virtual Network Function

VNFM Virtual Network Function Manager

XR Extended Reality

1 Introduction

Contents

1.1 Context 1

1.2 Research Objectives 5

1.3 Thesis Organization and Dissemination 6

1.4 List of Publications 6

1.4.1 Journals 6

1.4.2 Conferences 7

1.1 Context

The telecommunications industry is experiencing a profound transformation as it transitions from traditional monolithic network architectures to highly flexible, programmable and service-oriented systems. This evolution, driven by the deployment of fifth-generation (5G) networks and the anticipation of sixth-generation (6G) networks, represents not just a merely incremental improvement in wireless communication capabilities, but rather a fundamental paradigm shift that redefines how network resources are conceptualized, allocated and managed [1], [2].

The emergence of next-generation wireless networks has been catalyzed by several converging technological and societal trends. The exponential growth in mobile data traffic, projected to increase by a factor of five between 2022 and 2028 according to industry forecasts, has created unprecedented demands for network capacity and performance [3]. Simultaneously, the diversification of application requirements has introduced a complex landscape of service classes, each with distinct and often conflicting quality

of service (QoS) requirements. Ultra-reliable low-latency communications (uRLLC) verticals, such as autonomous vehicle control systems and industrial automation, require end-to-end latencies of less than 1-10 millisecond with reliability levels exceeding 99.999% [4]. In contrast, enhanced mobile broadband (eMBB) services prioritize high throughput for applications such as 4K/8K video streaming and augmented reality experiences, while massive machine-type communications (mMTC) focus on connecting billions of Internet of things (IoT) devices with energy efficiency and wide-area coverage as primary concerns [5].

The traditional approach to network design, characterized by over-provisioning of resources and static allocation strategies, has proven inadequate to address this diversity of requirements while maintaining economic viability. The capital expenditure (CAPEX) and operational expenditure (OPEX) associated with deploying and maintaining separate physical networks for each service class or deploying sufficient network infrastructure to meet these ever-growing demands would be prohibitively expensive, necessitating the development of innovative resource sharing and virtualization technologies [6].

Network virtualization has emerged as a fundamental enabler of this transformation, allowing multiple virtual networks to coexist on a shared physical infrastructure while maintaining isolation and providing customized service characteristics [7].

The virtualization is achieved through software-defined networking (SDN) principles which have been extended beyond traditional data center environments to encompass the entire network stack, from radio access networks to core network functions [8]. This software-centric approach enables programmable network control, dynamic resource allocation and rapid service deployment, fundamentally altering the economics and operational models of telecommunications networks.

Central to realizing these economic and operational benefits is network slicing which represents perhaps the most significant architectural innovation in this context. Network slicing enables infrastructure providers (InP) to partition their physical network resources into multiple logical networks, each optimized for specific service requirements and customer segments [9]. Each network slice operates as an independent virtual network with dedicated resources, customized network functions and tailored service level agreements (SLAs). This approach allows a single physical infrastructure to simultaneously support diverse use cases ranging from mission-critical industrial applications to consumer entertainment services. However, the realization of network slicing benefits introduces complex technical challenges in resource management, particularly in admission control and resource allocation [10]. The dynamic nature of slice requests, coupled with uncertain resource demands and the desire to maximize infrastructure utilization, necessitates sophisticated algorithms that can make real-time decisions under incomplete information.

Traditional admission control mechanisms, designed for relatively predictable traffic patterns and homogeneous service requirements, are inadequate for the multi-tenant, multi-service environment of sliced networks [11]. These challenges are further complicated by the evolving architecture of the radio access network (RAN) domain, which has undergone its own transformation through the open RAN (O-RAN) initiative.

Traditional RAN deployments have been characterized by vendor-specific, inflexible base station implementations that integrate radio, baseband processing and network control functions in proprietary hardware and software platforms [12]. This approach, while providing performance optimization through tight integration, has resulted in vendor lock-in, limited innovation and high deployment and operational costs. The O-RAN architecture addresses these limitations by disaggregating the traditional base station into three distinct functional units: the central unit (CU), distributed unit (DU) and radio unit (RU) [13]. This disaggregation is accompanied by the standardization of open interfaces between components, enabling multi-vendor deployments and promoting innovation through specialized optimization of individual components.

The CU hosts higher-layer protocol functions including the radio resource control (RRC) and packet data convergence protocol (PDCP) layers and can be implemented as cloud-native network functions running on general-purpose hardware. The DU manages lower-layer functions such as the radio link control (RLC) and medium access control (MAC) protocols, providing real-time processing capabilities while maintaining interface flexibility. The RU handles radio frequency functions and analog-to-digital conversion, representing the physical interface to the wireless medium [14]. This architectural transformation introduces new degrees of freedom in network design and optimization, but also creates complex interdependencies that must be carefully managed. The placement of CU and DU functions affects both the fronthaul and backhaul network requirements, influencing latency, bandwidth consumption and overall system performance [15]. User equipment (UE) association decisions must consider not only radio conditions but also the computational and transport resources available at different network layers. The energy efficiency implications of O-RAN deployments are particularly significant, as the disaggregated architecture enables fine-grained power management, allowing individual components to be dynamically activated or deactivated based on traffic demand [16]. However, the optimization of energy consumption must be balanced against QoS requirements, particularly delay constraints that are sensitive to the geographic distribution of processing functions. The complex resource management challenges posed by both network slicing and O-RAN architectures are particularly evident in specialized application domains like vehicular networks that demand ultra-reliable, low-latency communications.

The automotive industry's transition towards connected and autonomous vehicles has created another domain of complex resource management challenges. Vehicle-to-

everything (V2X) communications encompass a diverse range of communication scenarios, including vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P) and vehicle-to-network (V2N) interactions [17]. Each of these communication types has distinct requirements in terms of latency, reliability, range and data throughput. Modern vehicular environments are characterized by the coexistence of multiple radio access technologies (RATs), each with different coverage characteristics, performance profiles and operational parameters [18]. Dedicated short-range communications (DSRC) technologies provide low-latency, short-range connectivity optimized for safety-critical applications. Cellular vehicle-to-everything (C-V2X) technologies leverage the ubiquitous coverage of cellular networks while providing both direct communication capabilities and network-assisted coordination. Fifth-generation cellular systems introduce enhanced capabilities for vehicular communications through the support of uRLLC services and network slicing [19].

The challenge of RAT selection in vehicular environments is compounded by the high mobility of vehicles, rapidly changing channel conditions and the need to maintain service continuity across heterogeneous network deployments. Traditional RAT selection approaches, which rely on simple signal strength measurements or predefined policies, are inadequate for the dynamic and multi-objective optimization required in vehicular scenarios [20]. The consequences of suboptimal RAT selection in vehicular communications extend beyond mere performance degradation. Safety-critical applications, such as collision avoidance systems and emergency vehicle notifications, have strict latency and reliability requirements that must be maintained even under challenging network conditions. The failure to maintain these requirements can have life-threatening consequences, elevating the importance of intelligent RAT selection algorithms [21]. Furthermore, the energy consumption implications of RAT selection are particularly relevant for electric vehicles, where communication system power consumption directly impacts driving range. The optimization of communication energy efficiency must be balanced against performance requirements, creating multi-objective optimization challenges that require sophisticated algorithmic approaches [22].

The evolution of network slicing, O-RAN architectures and vehicular communications represents three parallel but interconnected developments in next-generation wireless networks. Each domain presents unique technical challenges, but they share common underlying themes that motivate the research presented in this thesis. All three domains involve complex multi-objective optimization problems where traditional single-metric optimization approaches are inadequate. Network slicing requires balancing infrastructure provider revenue against service quality and resource utilization. O-RAN optimization must simultaneously consider energy efficiency, delay performance and load balancing across disaggregated components. Vehicular RAT selection involves trade-offs between

latency, reliability, handover frequency and energy consumption.

The dynamic nature of these environments necessitates real-time or near-real-time decision-making capabilities. Network slice requests arrive dynamically with uncertain resource requirements and duration. O-RAN environments experience fluctuating traffic loads and varying channel conditions. Vehicular networks are characterized by rapid topology changes and time-varying service demands. Traditional static optimization approaches are inadequate for these dynamic scenarios. The absence of future state information creates additional complexity for decision-making algorithms. In network slicing for instance, network slice future resource demands may not be fully known at admission time. Similarly, O-RAN optimization must account for uncertain future traffic patterns and channel conditions and vehicular RAT selection must be performed with limited knowledge of network congestion and mobility patterns.

The scalability requirements of these systems demand computationally efficient solutions. Network slicing systems must handle thousands of concurrent slices. O-RAN deployments involve large numbers of UEs and network components. Vehicular networks must support massive scale connectivity with millions of vehicles. These common challenges suggest that intelligent, adaptive algorithms based on machine learning and advanced optimization techniques are essential for realizing the full potential of next-generation wireless networks. The research presented in this thesis addresses these challenges through the development of novel algorithmic approaches that leverage multi-objective optimization and artificial intelligence to provide efficient and effective solutions.

1.2 Research Objectives

The overarching objective of this thesis is to develop intelligent resource management and optimization algorithms for next-generation wireless networks that can effectively address complex, multi-objective and dynamic optimization challenges inherent in network slicing, O-RAN deployments and vehicular communications.

Specifically, this thesis is aimed at achieving the following research objectives:

1. **Network slicing optimization:** develop admission control algorithms that can make real-time decisions under dynamic resource demands while maximizing infrastructure provider revenue and maintaining service level agreements.
2. **O-RAN resource management:** design energy-efficient optimization approaches that dynamically allocate computational resources across disaggregated network functions while satisfying latency and performance constraints.
3. **Vehicular RAT selection:** create intelligent algorithms for dynamic radio ac-

cess technology selection that optimize multi-objective criteria including latency, reliability and handover frequency in high-mobility scenarios.

4. **Methodological contributions:** advance the state-of-the-art in multi-objective optimization for wireless networks through novel applications of optimization and reinforcement learning techniques.

1.3 Thesis Organization and Dissemination

This thesis is organized into six chapters that provide a comprehensive treatment of intelligent resource management in next-generation wireless networks.

Chapter 1 provides the background, motivation and research objectives for the thesis, establishing the context and significance of the research contributions.

Chapter 2 provides a detailed background on network evolution, network virtualization, network slicing, vehicular and V2X communications, admission control mechanisms and RAT selection techniques.

Chapter 3 introduces dynamic slice admission control algorithms for network slicing environments, focusing on revenue optimization through intelligent overbooking strategies and online admission control with incomplete information.

Chapter 4 presents optimization frameworks for user association and function placement in O-RAN networks, emphasizing energy efficiency while maintaining QoS requirements and end-to-end delay guarantees.

Chapter 5 introduces a deep reinforcement learning approach for multi-objective RAT selection in vehicular communication environments.

Chapter 6 presents the conclusions and perspectives. It summarizes the thesis contributions, discusses its implications for future network deployments and identifies opportunities for future research.

1.4 List of Publications

1.4.1 Journals

- S. O. Yese, S. Berri and A. Chorti, "Efficient Online Slice Admission Control Scheme in Next Generation Networks," in IEEE Networking Letters, vol. 6, no. 3, pp. 163-167, Sept. 2024.

- S. O. Yese, S. Berri and A. Chorti, " End to End Delay-Guaranteed and Energy Efficient User Association Algorithm with Functions placement for O-RAN," 2025 IEEE Transaction on Mobile Computing, 2025 (**under review**).

1.4.2 Conferences

- S. O. Yese, S. Berri and A. Chorti, "Novel Slice Admission Control Scheme with Overbooking and Dynamic Buyback Process," 2023 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Dresden, Germany, 2023.

- S. O. Yese, S. Berri and A. Chorti, "Energy Efficient and Delay-Guaranteed User Association Algorithm for O-Ran," ICC 2025 - IEEE International Conference on Communications, Montreal, QC, Canada, 2025.

- S. O. Yese, S. Berri and A. Chorti, " Multi-Objective Deep Reinforcement Learning Based RAT Selection solution for Vehicle-to-Infrastructure Communication," 8th International Conference on Machine Learning for Networking (MLN'2025) (**accepted**).

2 Background Concepts

Contents

2.1 Evolution of Wireless Networks 8
2.2 Network Virtualization and Software-Defined Networking . . 11
2.3 Network Slicing: Concepts and Challenges 12
2.4 Vehicular Communications and V2X Technologies 14
2.5 Admission Control in Network Systems 16
2.6 Radio Access Technology Selection 17
2.7 Open Radio Access Network (O-RAN) Architecture 18
2.8 Open Research Questions and Motivation 20

This chapter provides a comprehensive background on the fundamental concepts, technologies and research areas that form the foundation of this thesis. The chapter is structured to present the evolution of wireless networks, the emergence of network virtualization and slicing, the development of O-RAN architectures and the advancement of vehicular communications.

2.1 Evolution of Wireless Networks

The evolution of wireless communication systems has been marked by continuous innovation driven by increasing user demands, technological advancements and the need for improved resource efficiency [23]. The progression from first-generation analog systems to the current fifth-generation digital networks represents a journey of remarkable technological transformation that has fundamentally changed how society communicates and accesses information [24].

The first generation (1G) of cellular networks, deployed in the 1980s, introduced mobile voice communications using analog transmission techniques. These systems provided basic mobility but were limited by poor voice quality, limited capacity and lack of security features. The transition to second-generation (2G) systems in the early 1990s marked the shift to digital transmission, introducing the global system for mobile communications (GSM) and code division multiple access (CDMA) technologies [25], [23]. 2G systems enabled text messaging and basic data services while significantly improving voice quality and network capacity.

Third-generation (3G) networks, standardized in the late 1990s and deployed in the early 2000s, introduced packet-switched data services and enabled mobile internet access [26]. Universal mobile telecommunications system (UMTS) and CDMA2000 technologies provided data rates of several megabits per second, enabling applications such as video calling and mobile web browsing. The evolution continued with 3.5G and 3.75G enhancements that increased data rates through technologies such as high-speed packet access (HSPA) [27].

Fourth-generation (4G) long term evolution (LTE) networks, standardized by the 3rd generation partnership project (3GPP) and widely deployed in the 2010s, represented a fundamental architectural shift towards all-IP networks [28]. LTE systems moved away from circuit-switched voice in favor of packet-switched services, enabling data rates exceeding 100 Mbps and introducing advanced antenna techniques such as multiple-input multiple-output (MIMO) [29] arrays. LTE-advanced and LTE-advanced pro further enhanced performance through carrier aggregation, coordinated multipoint transmission and massive MIMO technologies.

The fifth generation (5G) of wireless networks represent the most significant transformation in cellular network architecture to date [30]. 5G systems are designed to support three primary service categories: enhanced mobile broadband (eMBB), ultra-reliable low-latency communications (uRLLC) and massive machine-type communications (mMTC), with each of these service categories addressing distinct application requirements and consequently introducing unique technical challenges [31].

The eMBB services focus on providing extreme data rates, with peak data rates exceeding 1 Gbps for mobile users and 10 Gbps for fixed wireless access. These services enable applications such as ultra-high-definition video streaming, augmented reality and virtual reality experiences. The technical enablers for eMBB include millimeter-wave spectrum utilization, massive MIMO with hundreds of antenna elements and advanced beamforming techniques [32], [33]. Similarly, uRLLC services target applications requiring extremely low latency and high reliability, with end-to-end latency targets of 1 millisecond and reliability levels exceeding 99.999%. These stringent requirements are driven by ap-

plications such as autonomous vehicle control, industrial automation and remote surgery. The technical solutions for uRLLC include edge computing, network function virtualization and advanced resource allocation techniques [34], [35]. On their part, mMTC services address the connectivity requirements of Internet of things (IoT) applications, targeting the connection of billions of devices with diverse requirements ranging from simple sensors to complex industrial equipment. The key characteristics of mMTC include energy efficiency, wide-area coverage and support for massive numbers of devices. Technologies such as narrowband IoT (NB-IOT) and extended coverage GSM IoT (ECGSMIOT) have been developed to address these requirements [36], [37].

The sixth generation (6G) of wireless networks is currently under research and development, with commercial deployment anticipated in the 2030s [38], [39]. While 5G networks continue their global rollout, the research community and industry stakeholders have begun defining the vision and technical requirements for 6G systems, which are expected to provide transformative capabilities beyond the current 5G paradigm [40]. 6G networks are envisioned to support data rates up to 1 Tbps, approximately 100 times faster than 5G peak rates, with latency reduced to microseconds for critical applications [41]. These enhanced performance metrics will enable emerging applications such as holographic communications, digital twins, extended reality (XR) experiences and brain-computer interfaces [42], [43]. The integration of artificial intelligence and machine learning into the network architecture itself is considered a fundamental characteristic of 6G, enabling intelligent resource management, autonomous network optimization and cognitive radio capabilities [44], [40].

Key enabling technologies for 6G include terahertz (THz) communications, reconfigurable intelligent surfaces (RIS) and integrated sensing and communications (ISAC). THz communications utilize frequencies between 0.1 and 10 THz, offering unprecedented bandwidth but pose significant challenges in terms of propagation characteristics and hardware implementation [45], [46]. RIS enables dynamic control of wireless propagation environments through programmable surfaces that can reflect, refract, or absorb electromagnetic waves [47], [48]. ISAC enables combining sensing and communication systems to ensure resource efficiency and reduced hardware cost. Additionally, 6G networks are expected to integrate terrestrial and non-terrestrial networks, incorporating satellite communications, high-altitude platform stations and unmanned aerial vehicles to provide seamless global coverage [49], [50].

The architectural evolution towards 5G and beyond has introduced several key innovations such as network function virtualization (NFV) and software-defined networking (SDN) that fundamentally change how networks are designed and operated [51]. NFV enables network functions to be implemented as software running on general-purpose hardware, replacing dedicated network appliances. SDN principles on the other hand

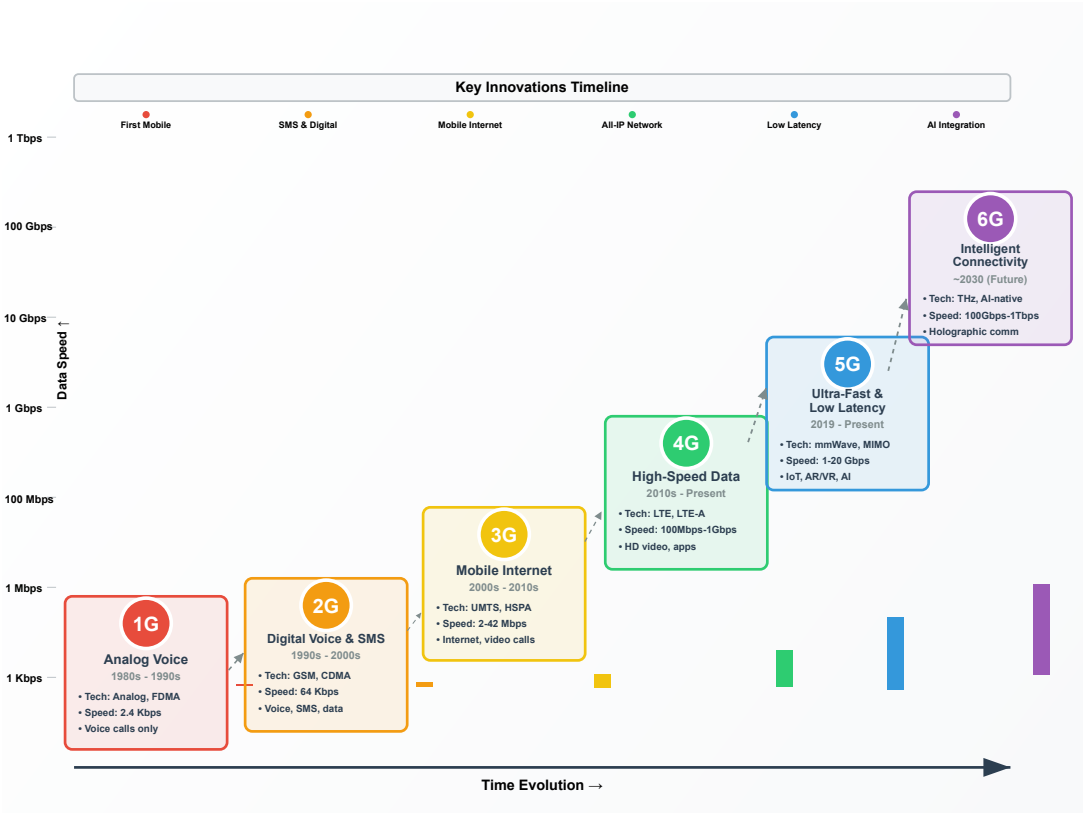


Figure 2.1: Mobile networks evolution: from 1 to 6G.

provide programmable network control, enabling dynamic resource allocation and service deployment [52]. The evolution of mobile networks is captured in Figure 2.1.

2.2 Network Virtualization and Software-Defined Networking

Network virtualization represents a fundamental shift in network architecture that enables multiple virtual networks to coexist on shared physical infrastructure while maintaining isolation and providing customized service characteristics [52]. This concept, which originated in data center environments, has been extended to encompass the entire network stack from radio access networks to core network functions [7].

The theoretical foundations of network virtualization can be traced to the concept of virtual machines in computing systems, where multiple operating systems can run simultaneously on a single physical machine. This concept has been extended to the networking context, with virtualization enabling the creation of multiple logical networks that appear as independent physical networks to their users while sharing common underlying infrastructure [53]. This approach provides several key benefits including improved resource utilization, reduced capital and operational expenditures and enhanced service flexibility

[54].

SDN provides the architectural framework that enables practical network virtualization implementations. SDN principles are based on the separation of the network control plane from the data plane, with centralized controllers managing network behavior through standardized interfaces such as OpenFlow [55], [56]. This separation enables programmable network control, where network behavior can be modified through software without requiring changes to underlying hardware infrastructure. The SDN architecture consists of three primary layers: the infrastructure layer containing network switches and routers, the control layer hosting the SDN controller and the application layer containing network applications and services [8].

NFV complements SDN by enabling network functions to be implemented as software running on standard servers, switches and storage devices. NFV decouples network functions from proprietary hardware appliances, enabling functions such as firewalls, load balancers and deep packet inspection to be instantiated as virtual network functions (VNFs). The combination of SDN and NFV provides a powerful platform for network service innovation and operational efficiency improvements [52].

The NFV architectural framework, standardized by the european telecommunications standards institute (ETSI), defines several key components including the NFV infrastructure (NFVI), VNF manager (VNFM), NFV orchestrator (NFVO) and virtual infrastructure manager (VIM) [57]. The NFVI provides the computational, storage and networking resources required to host VNFs while the VNFM is responsible for VNF lifecycle management including instantiation, scaling and termination. The NFVO coordinates the provisioning of network services across multiple VNFs and manages the overall service life cycle [58], [59].

The integration of these technologies (SDN and NFV) enables advanced network management capabilities such as dynamic service chaining, automatic scaling and intelligent resource allocation that are crucial for optimizing the VNF placement across the infrastructure to minimize costs while meeting performance requirements [59], [60].

2.3 Network Slicing: Concepts and Challenges

Network slicing is said to represent one of the most significant innovations introduced by 5G systems, enabling the creation of multiple logical networks over shared physical infrastructure, with each optimized for specific service requirements and use cases [61]. A network slice encompasses all network domains from radio access networks to core network functions, providing an end-to-end virtual network with customized functionality and performance characteristics [9].

The conceptual foundation of network slicing stems from the desire to serve the diverse QoS requirements of the different applications and services that cannot be efficiently served by a single, monolithic network design. Traditional network architectures are designed to meet the requirements of the most demanding applications, resulting in over-provisioning for less demanding services. Network slicing addresses this inefficiency by allowing each slice to be optimized for its specific requirements while sharing common infrastructure resources [10].

A network slice is defined as a logical network that provides specific network capabilities and characteristics required to serve a particular use case or business model [62], [9]. Each slice operates as an independent network with dedicated resources, customized network functions and specific performance guarantees defined through service level agreements (SLAs). The slice may span multiple network domains including radio access networks, transport networks and core networks, providing end-to-end service delivery [1].

The technical implementation of network slicing leverages SDN and NFV technologies to provide the necessary flexibility and programmability. Network slice templates define the structure and composition of network slices, specifying the required network functions, their interconnections and performance requirements. Network slice instances are created by instantiating and configuring the specified network functions according to the template requirements. The lifecycle management of network slices includes operations such as slice creation, modification, scaling and termination [2].

Network slice management involves several key challenges that must be addressed to realize the full benefits of slicing technology. Resource allocation algorithms must determine how to optimally distribute physical network resources among competing slices while meeting individual SLA requirements. Admission control mechanisms must decide whether new slice requests can be accommodated without violating existing slice agreements. Slice isolation can be maintained to ensure that performance degradation in one slice does not affect other slices sharing the same infrastructure [63].

The economic aspects of network slicing introduce additional complexity in the form of pricing models, revenue optimization and business relationship management. InPs must develop pricing strategies that balance revenue maximization with competitive positioning while ensuring sufficient resource utilization. SLAs must be carefully structured to provide appropriate incentives for both InPs and tenants while accounting for the uncertainty inherent in shared resource environments [64].

Performance isolation between network slices represents a critical technical challenge that affects both system design and operational procedures. This isolation can be in terms of radio, computational or network resources. Radio resource isolation ensures that radio

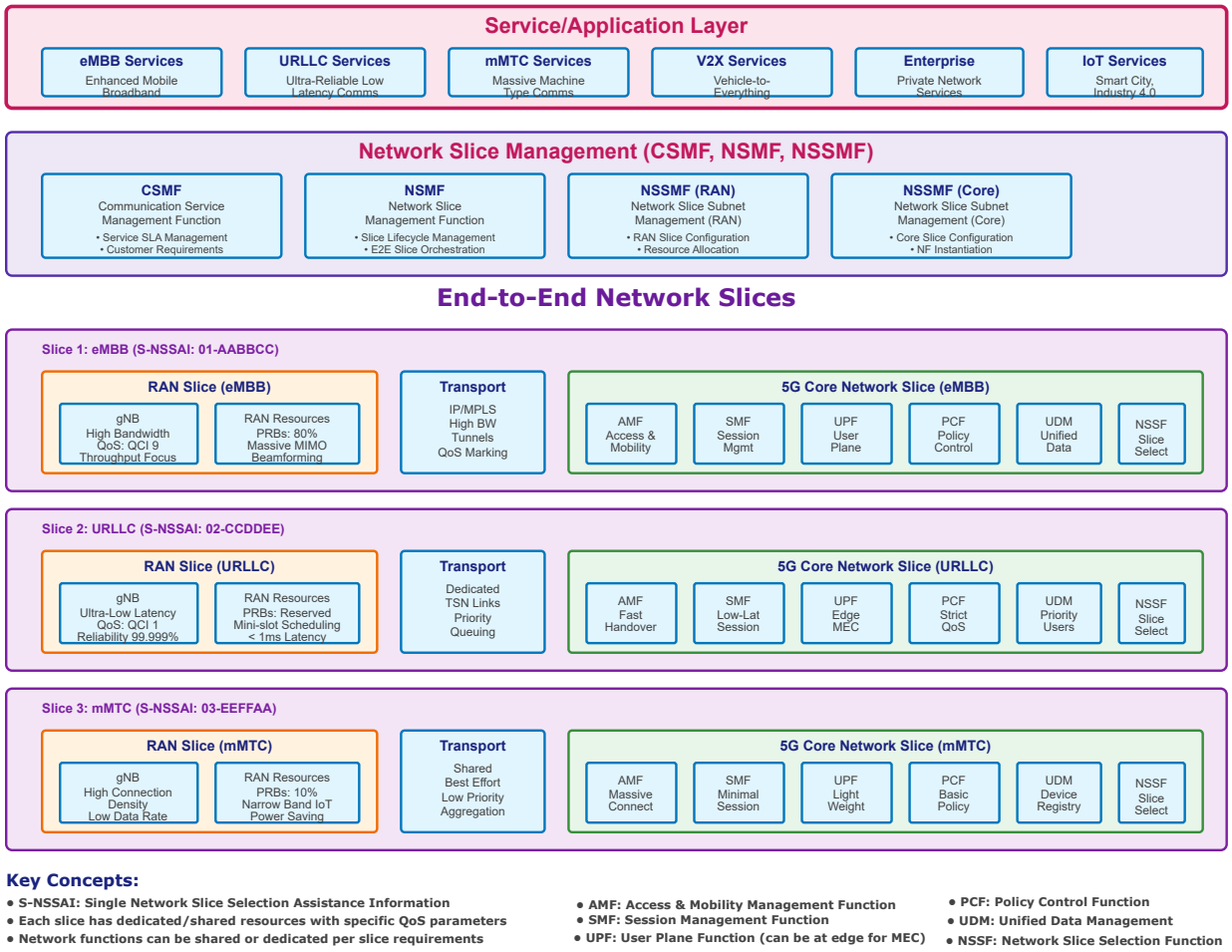


Figure 2.2: Network slicing architecture.

spectrum resources allocated to one slice are not accessed by other slices. Computational resource isolation prevents VNFs belonging to different slices from interfering with each other's processing. Network resource isolation ensures that network traffic from different slices is appropriately separated and prioritized [65].

A network slicing architecture is presented in 2.2 to further shed light on the concept of network slicing.

2.4 Vehicular Communications and V2X Technologies

Vehicle-to-everything (V2X) communications represent a rapidly evolving domain that encompasses various communication scenarios between vehicles and their surrounding environment. V2X communications encompasses multiple dimensions of vehicular interactions such as vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P) and vehicle-to-network (V2N), with each serving distinct applications

with specific performance requirements [66], [17].

The development of V2X communications has been driven by several converging trends including the advancement of autonomous driving technologies, the need for improved road safety and the desire for enhanced traffic efficiency [67]. Intelligent transportation systems (ITS) leverage V2X communications to enable cooperative driving, traffic management and safety applications that can significantly reduce accidents and improve traffic flow [68].

V2V communications enable cooperative information sharing among vehicles through direct communication between vehicles to share information about position, speed, direction and intentions. Safety applications such as collision warning, emergency braking notification and intersection collision avoidance rely on V2V communications to provide timely alerts to drivers and automated vehicle systems. The stringent latency and reliability requirements of safety applications necessitate specialized communication protocols and resource allocation mechanisms [21], [69]. V2I communications on the other hand facilitate interaction between vehicles and roadside infrastructure including traffic lights, road signs and traffic management centers [69]. These communications enable applications such as traffic light optimization, variable speed limit enforcement and real-time traffic information dissemination. V2I communications can leverage both dedicated roadside units (RSUs) and existing cellular infrastructure, providing flexibility in deployment strategies [70]. V2P communications address the safety needs of vulnerable road users including pedestrians, cyclists and motorcyclists. These communications enable vehicles to detect and respond to the presence of pedestrians and cyclists, particularly in scenarios where visibility is limited [71]. V2N communications enable vehicles to access cloud-based services and applications through cellular networks [69]. These communications support applications such as real-time traffic optimization, infotainment services and over-the-air software updates. The integration of V2N communications with 5G networks enables advanced applications leveraging network slicing, edge computing and network function virtualization [19].

Two primary technology approaches have emerged for V2X communications: dedicated short-range communications (DSRC) and cellular V2X (C-V2X). DSRC, on one hand is based on the IEEE 802.11p standard, operates in the 5.9 GHz ITSs band and provides low-latency communication optimized for safety-critical applications. DSRC systems can operate without cellular network infrastructure, providing autonomous operation capabilities that are particularly valuable in areas with limited cellular coverage [68]. C-V2X technology on the other hand leverages cellular communication systems to provide V2X services, with two complementary communication modes: direct communication (PC5 interface) and network-based communication (Uu interface). Direct communication enables V2V and V2I interactions without requiring cellular network coverage, while

network-based communication provides access to cloud services and wide-area connectivity. The evolution of C-V2X from LTE-based implementations to 5G new radio (NR) systems introduces enhanced capabilities for advanced V2X applications [72].

The coexistence of multiple V2X technologies in vehicular environments creates complex RAT selection challenges. Vehicles must dynamically select the most appropriate communication technology based on application requirements, network conditions and quality of service constraints. The heterogeneous nature of vehicular networks requires sophisticated algorithms that can adapt to rapidly changing conditions while maintaining service continuity [18].

2.5 Admission Control in Network Systems

Admission control is a fundamental mechanism in network systems for regulating access to limited network resources and maintaining service quality for admitted connections or services. The primary objective of admission control is to determine whether new service requests can be accommodated without violating the performance guarantees of previously admitted services [73].

Traditional admission control algorithms were first developed for circuit-switched networks where resources could be precisely reserved for each admitted connection. These algorithms typically employed relatively simple decision criteria based on available bandwidth and established worst-case resource requirements. The predictable nature of circuit-switched connections enabled effective admission control with relatively low computational complexity [74].

The transition to packet-switched networks introduced significant complexity in admission control design due to the statistical multiplexing of network resources and the bursty nature of packet traffic. Packet-switched admission control algorithms must account for statistical variations in traffic demands while providing probabilistic performance guarantees [75].

In Internet protocol (IP) based networks, admission control mechanisms have evolved to support quality of service (QoS) frameworks such as integrated services (IntServ) and differentiated services (DiffServ). IntServ provides per-flow resource reservations through protocols such as resource reservation protocol (RSVP), enabling precise admission control at the cost of scalability limitations. DiffServ provides class-based service differentiation with aggregate traffic management, enabling more scalable but less precise admission control [76].

Wireless network admission control introduces additional challenges due to the time-

varying nature of wireless channels, mobility of users and interference effects. Channel variations require admission control algorithms to account for link quality fluctuations and their impact on resource requirements. User mobility necessitates dynamic resource reallocation and handover management to maintain service continuity. Interference effects create dependencies between admission decisions for different users, requiring sophisticated interference management techniques [77].

The emergence of network slicing in 5G systems has fundamentally transformed admission control requirements and mechanisms. Network slice admission control must now consider the multi-tenant nature of sliced networks, where multiple service providers share common infrastructure resources. Each network slice may have distinct performance requirements, traffic characteristics and business relationships that must be considered in admission decisions. The dynamic nature of slice requests, with varying duration and resource requirements, necessitates adaptive admission control algorithms that can respond to changing network conditions [11], [78].

Economic considerations also play an increasingly important role in admission control design, particularly in multi-tenant network environments. Revenue optimization objectives must be balanced against service quality requirements and fair resource allocation among competing tenants [79].

2.6 Radio Access Technology Selection

Radio access technology (RAT) selection represents a critical network management function in heterogeneous wireless environments where multiple communication technologies coexist. The proliferation of diverse RATs, each with distinct characteristics in terms of coverage, capacity, latency and energy consumption, has created complex decision-making scenarios that require sophisticated algorithms to optimize network performance and user experience [20].

The evolution towards heterogeneous networks (HetNets) has been driven by the need to address diverse application requirements and improve overall network capacity through the deployment of multiple complementary technologies. Macro cells provide wide-area coverage, while small cells enhance capacity in high-traffic areas. WiFi networks offer high-capacity, low-cost connectivity in indoor and hotspot environments. Satellite systems provide coverage in remote areas where terrestrial infrastructure is not economically viable [80].

Traditional RAT selection algorithms rely on simple metrics such as received signal strength indicator (RSSI) or signal-to-interference-plus-noise ratio (SINR) to guide selection decisions. These approaches, while computationally efficient, fail to consider the

diverse requirements of modern applications and the complex interactions between different network technologies. Advanced RAT selection algorithms must consider multiple criteria including application requirements, network load conditions, user mobility patterns and energy consumption constraints [81], [82], [83].

The application of RAT selection in vehicular communication environments introduces unique challenges due to high user mobility, rapidly changing channel conditions and stringent application requirements. Vehicular RAT selection must consider factors such as vehicle speed, trajectory prediction, handoff costs and the critical nature of safety applications. The coexistence of DSRC and C-V2X technologies in vehicular environments requires algorithms that can dynamically select between direct communication and network-assisted communication modes based on application requirements and network conditions [84], [85], [86].

2.7 Open Radio Access Network (O-RAN) Architecture

The open radio access network (O-RAN) architecture represents a paradigm shift in RAN design that aims to create vendor-neutral, interoperable and software-defined radio access networks. The O-RAN initiative addresses the limitations of traditional RAN architectures, which are characterized by proprietary, monolithic base station implementations that create vendor lock-in and limit innovation [12].

The traditional RAN architecture integrates radio, baseband processing and control functions in vendor-specific hardware platforms with proprietary interfaces between components. This approach provides optimization opportunities through tight integration but results in several significant limitations such as vendor lock-in, limited innovation due to closed systems, high deployment costs due to specialized hardware requirements and operational complexity due to vendor-specific management systems [13].

The O-RAN architecture addresses these limitations through two key principles: disaggregation and standardization. Disaggregation involves separating the traditional base station into distinct functional units with standardized interfaces between components. Standardization ensures that interfaces between components from different vendors are interoperable, enabling multi-vendor deployments and promoting competition and innovation [87].

The O-RAN functional disaggregation defines three primary units: the central unit (CU), distributed unit (DU) and radio unit (RU). The CU hosts higher-layer protocol functions including radio resource control (RRC), service data adaptation protocol (SDAP)

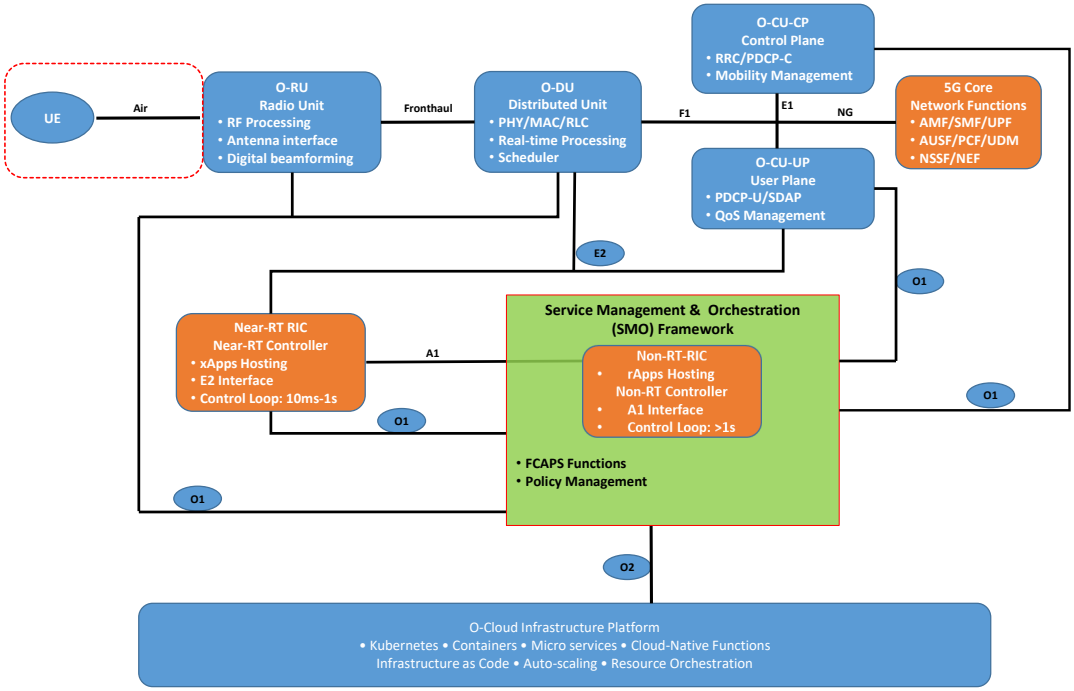


Figure 2.3: O-RAN Architecture.

and packet data convergence protocol (PDCP). The CU can be further divided into control plane (CU-CP) and user plane (CU-UP) functions to enable flexible deployment options. The DU manages lower-layer functions including radio link control (RLC), medium access control (MAC) and higher physical (PHY) layer functions. The RU handles lower PHY layer functions, radio frequency processing and the interface to antennas [14].

The interfaces between O-RAN functional units are standardized to enable interoperability between equipment from different vendors. The F1 interface connects the CU and DU, supporting both control plane and user plane protocols. The fronthaul interface connects the DU and RU, utilizing protocols such as enhanced common public radio interface (eCPRI) and radio over Ethernet (RoE). The E2 interface connects RAN elements to the near real-time RIC for intelligent RAN optimization. The A1 interface connects the Non-Real-Time RIC to the near real-time RIC for policy-based RAN management [15]. Figure 2.3 provides a high level view of the O-RAN architecture showing the various components and respective interfaces.

The O-RAN architecture introduces two key intelligence components: the non-real-time RAN intelligent controller (Non-RT RIC) and the near real-time RAN intelligent controller (near-RT RIC). The Non-RT RIC operates on timescales greater than 1 second and provides policy-based guidance for RAN optimization while the Near-RT RIC operates on timescales from 10 milliseconds to 1 second and provides real-time optimization of RAN functions through xApps (applications running on the Near-RT RIC) [88].

The deployment flexibility provided by O-RAN disaggregation enables various archi-

tectural options to optimize performance, cost and operational efficiency. CU functions can be centralized in cloud data centers to enable resource pooling and coordination benefits. DU functions can be deployed at edge locations to minimize fronthaul bandwidth requirements and reduce latency. RU functions must be deployed at cell sites to provide the radio interface. The optimal deployment strategy depends on factors such as fronthaul capacity, latency requirements and computational resource availability [15].

Energy efficiency represents a critical consideration in O-RAN deployments due to the disaggregated architecture's impact on power consumption patterns. The separation of functions across multiple locations can increase overall system power consumption due to additional networking and processing overhead. However, the flexibility provided by disaggregation enables sophisticated power management strategies, including dynamic function activation, load-adaptive processing and coordinated sleep modes across multiple network elements [16].

The user association problem in O-RAN environments becomes significantly more complex due to the disaggregated architecture and the need to consider computational and transport resources in addition to radio conditions. Traditional user association algorithms based primarily on radio signal quality are inadequate for O-RAN environments where the placement of CU and DU functions affects end-to-end performance. Joint optimization of user association and function placement represents an active area of research with significant implications for O-RAN deployment strategies [89], [15].

2.8 Open Research Questions and Motivation

The comprehensive review of existing literature reveals several open research questions that motivate the contributions presented in this thesis. These research questions span across the domains of network slicing admission control, O-RAN optimization and vehicular RAT selection, while also encompassing cross-cutting themes related to multi-objective optimization and machine learning applications in network management.

In the domain of network slicing admission control, existing research has primarily focused on static scenarios with complete information about slice requirements and network conditions. However, practical network slicing deployments involve dynamic environments where slice requests arrive online with dynamically changing and uncertain resource requirements and duration. The economic aspects of network slicing, particularly revenue optimization for infrastructure providers, have received limited attention in existing admission control algorithms. Furthermore, the potential benefits of overbooking strategies, which are well-established in other industries such as airlines and hotels, have not been thoroughly investigated in the network slicing context and where they exist are

not properly tailored to the realities of the telecommunication industry [90].

O-RAN research has primarily focused on individual aspects of the disaggregated architecture, such as fronthaul optimization or function placement, without considering the complex interdependencies between user association, function placement and energy consumption. The joint optimization of these factors represents a significant open research question with important implications for practical O-RAN deployments. Additionally, existing O-RAN research has not adequately addressed the end-to-end optimization challenges that arise when multiple O-RAN components must be jointly optimized while maintaining quality of service guarantees [89].

Vehicular RAT selection research has been limited of focus on simple, single-objective optimization approaches that do not adequately capture the complex trade-offs inherent in vehicular communication environments. The application of advanced machine learning techniques, particularly deep reinforcement learning, to vehicular RAT selection problems especially as it pertains to V2I represents an under-explored area with significant potential for performance improvements as most of the existing solutions focus on V2V communication scenario. Furthermore, existing research has not adequately addressed the unique characteristics of V2I communication scenarios, which differ significantly from V2N and traditional heterogeneous network communication environments [18].

These open research areas provide the motivation for the novel contributions presented in the subsequent chapters of this thesis, which address the corresponding research questions through the development of innovative algorithms and optimization frameworks that advance the state-of-the-art in network resource management optimization.

3 Slice Admission Control Schemes in Next Generation Networks

Contents

3.1 Introduction 23

3.2 Review of Related Works 26

3.3 Slice Admission Control Scheme with Overbooking and Dynamic Buyback Process 34

3.3.1 System Model (P1) 35

3.3.2 Problem Formulation (P1) 36

3.3.3 Proposed Algorithm for P1 37

3.3.4 Performance Evaluation of the Solution for P1 40

3.3.5 Simulation Settings for P1 40

3.3.6 Proposed Priority Schemes 40

3.3.7 Comparison of Proposed Solution for P1 to SoA 42

3.4 Efficient Online Slice Admission Control Scheme in Next Generation Networks 46

3.4.1 System Model (P2) 46

3.4.2 Problem Formulation (P2) 47

3.4.3 Proposed Algorithm for P2 48

3.4.4 Performance Evaluation of the Proposed Solution for P2 51

3.4.5 Simulation Settings for P2 51

3.4.6 Comparison of the Proposed Solution for P2 to SoA 52

3.5 Chapter Summary 56

3.1 Introduction

The fifth generation (5G) and beyond of mobile networks has attracted a lot of research interest recently. 5G promises to provide a hundred times (10 gigabit/seconds) faster data delivery than the current 4G networks [91]. Therefore, it is expected to simultaneously serve a large number of devices with multiple heterogeneous services having diverse QoS requirements such as throughput, delay, reliability, etc. [92]. This is based on the forecast by the international telecommunication union (ITU) that mobile data traffic is projected to grow at an annual rate of around 55% between 2020–2030 and reach 607 exabytes (EB) in 2025 and 5,016EB in 2030, with 2021 alone witnessing 23 times increase in data volume compared to the entire global Internet traffic in 2005 [93]. The 5G capacity will influence the aggressive evolution of IoT verticals, virtual reality and bring to life the internet of everything (IoE). To this end, 5G wireless systems are designed to support three generic services with heterogeneous requirements: extreme mobile broadband (eMBB), massive machine type Communication (mMTC) and ultra-reliable and low-latency communications (uRLLC) [94].

To provide efficient service delivery for these diverse use cases while reducing cost, the 5G architecture provides support for heterogeneous service delivery through infrastructural sharing or network slicing. Network slicing allows several logical networks called slices to share the resources of a single physical network. That is, a tenant, requests specific services from the communication service provider (CSP), i.e., the network provider and pays for the service according to agreed-upon metrics like, the number of served users, the service coverage area, or the duration of the slice instance [95]. Each slice can be customized to service specific applications or be operated by a communications provider other than the owner of the physical network infrastructure [96].

Due to the limited nature of network infrastructure resources and the quality guarantees required by network slices, mechanisms such as admission control are needed before new slice requests are accepted. This is to ensure that the new slices are only accepted if their service guarantees can be met by the network without adversely affecting the quality guarantees for existing slices [97], [98], [99]. The sharing of resources, support for low/ultra-low latency applications and the huge amount of data flowing through the network, coupled with the difficulties in forecasting the overall behavior of a system involving many different players, make traditional tools for network management such as static resource reservation, deployed in 4G systems, insufficient [97].

It is therefore necessary that novel algorithms for network slicing solutions be designed to manage the infrastructure resources, sharing them between the different slices/tenants while guaranteeing that the requirements of each slice will be met [100]. We refer to this

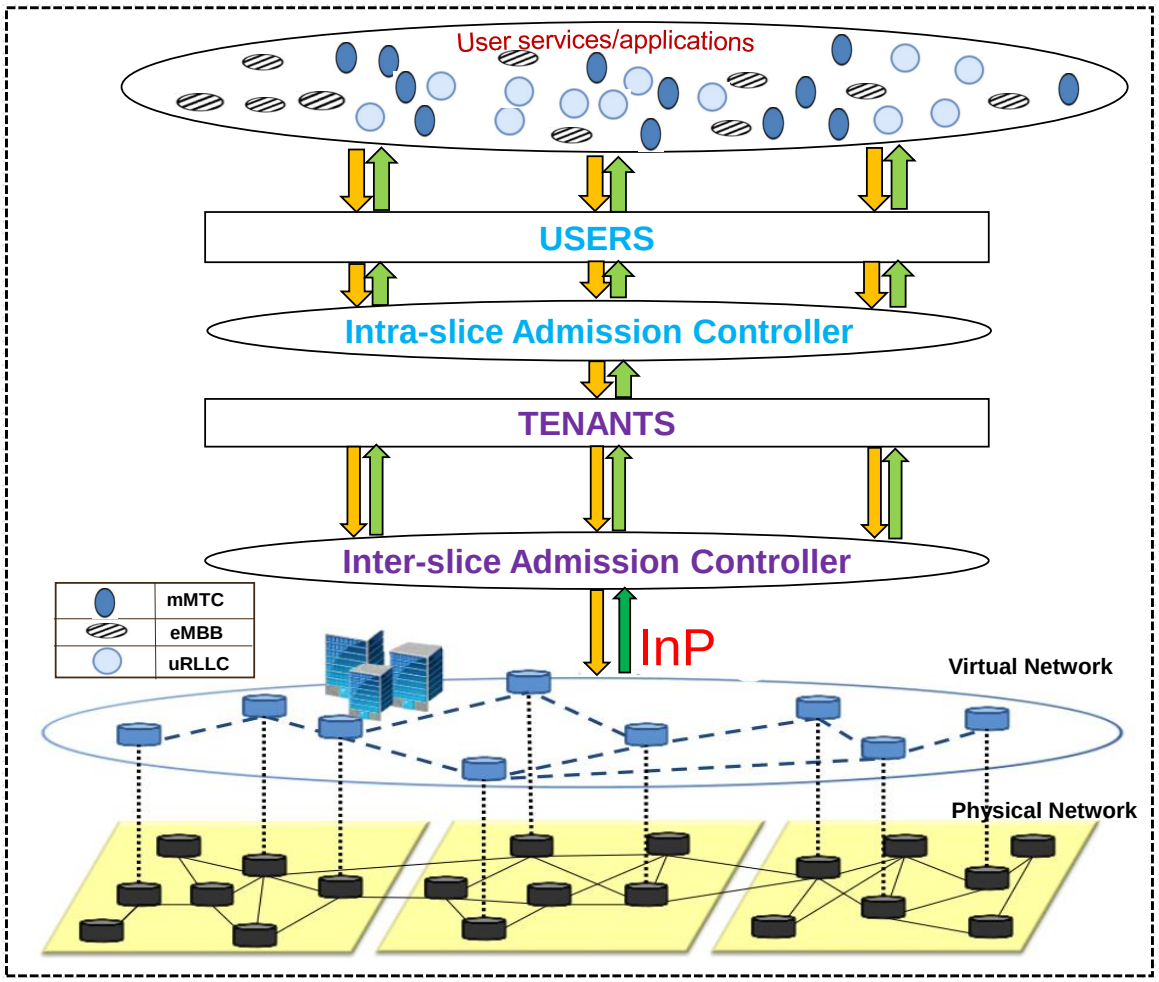


Figure 3.1: A typical sliced network framework.

procedure as inter-slice admission (admission control between slices/tenants). Note that for this thesis, we focus on inter-slice admission control without delving into intra-slice admission control (admission control between users in a slice. That is, admission control employed by a slice tenant to decide admission of the users it is serving). The framework for admission control at the various levels in a sliced network is as shown in Figure 3.1.

There are five main objectives of slice admission control: **revenue optimization**, **QoS control**, **congestion control**, **admission fairness** and **optimized resource utilization** [97]. While much research has been carried out on the subject of admission control, much of it is for the sole objective of either revenue maximization [101], [95], [102], [103], [104], [105], QoS control (SLA assurance) [106], [107], [108], [109] or improved resource utilization [98] at the expense of other strategic objectives. More recently, some researchers have attempted to simultaneously achieve two or more of these objectives by employing the concept of overbooking. Overbooking is a concept that allows the InP to accept requests beyond its nominal available resources by buying back unused resources from existing tenants when there are insufficient resources to accept new requests. While some solutions use the overbooked resources to further increase revenue by admitting only

high revenue yielding slice requests [105], [110], others use the overbooked resources to improve fairness by admitting only low priority slices thus optimizing revenue and ensuring admission fairness respectively [111], [112]. However, these existing solutions buyback all idle resources in anticipation for more requests without recourse to the resource requirements of the waiting requests thus, increasing the provider's buyback cost and decreasing the net revenue when there are no requests to admit with all the bought-back resources. Similarly, the existing solutions define priorities that mostly depend on slice type that provide short term revenue gains but lower long term revenue.

In this chapter, we present two inter-slice admission control algorithms that seek to ensure efficient management of scarce network resources and guarantee service level agreements (SLAs) for network tenants while also increasing revenue for infrastructure providers. In more detail, in Section 3.3 we consider a static scenario where a single decision instance is considered while the later considers a more realistic and dynamic scenario presented in more detail, in Section 3.4 with multiple time slots and decision instance considered over multiple time slots. The InP first admits slice requests with its available resources. However, when there are more requests but the InP does not have sufficient resources to meet the demand of these requests, the step-wise pooling mechanism is employed to determine whether part or all idle resources from already admitted tenants should be bought back in order to satisfy the new slice requests. The InP pays a certain price (the buyback price) to the tenants for the resources it is buying back. The InP then uses the resources bought back to satisfy some or all the new requests else, requests that there are no sufficient resources to accept are rejected.

The key contributions of this chapter are as follows:

- **Two admission control algorithms**: we formulate two problems of admission control as multi-objective optimization problems that maximize both InP's profit and resource utilization while meeting a slice's SLA.
- **Two greedy algorithms:** we propose two greedy algorithms to obtain an approximate solution of the formulated problems. These algorithms employ novel dynamic queue adaption priority, step-wise pooling, willingness to sell and dynamic buyback price mechanisms listed below.
- **Dynamic queue adaptation priority:** the proposed mechanism sorts the requests in queue in order of their priority. It enables the InP to ensure service differentiation as well as to make intelligent decisions that will lead to high long term revenue.
- **Step-wise and full pooling mechanisms:** these mechanisms dynamically determine at each point whether an InP buys back some or all of the available unused

resources from existing tenants. The step-wise pooling mechanism enables the InP to only buyback as much resources as is needed per time. Thus reducing the InP's buyback cost, consequently increasing the InP's net revenue.

- **A willingness to sell probability:** this creates a framework for determining the probability that an admitted request will be willing or not to sell some of its unused resources to the InP.

- **A dynamic buyback price mechanism:** the dynamic buyback price ensures a win-win situation for the InP and the already accepted tenants as the tenants are able to recover part of their cost when they overestimate their demand. Similarly, the InP is also able to satisfy more clients and build a reputation without having to lose revenue [98], [110]. This is in total contrast to what is available in literature, where it is assumed that the InP buys back resources at the same or higher rate from the initial price sold to the tenants and sells same at a higher price to the new tenants [104], [113], [105], [114].

3.2 Review of Related Works

In [115], a dynamic resource pooling and trading mechanism in network virtualization using Stackelberg game is proposed to achieve more efficient utilization of network resources. The proposed mechanism, allows a user to request a long-term resource amount at the beginning that can be lower than the peak demand. When the user cannot exhaust its resources due to low traffic, the unused resources are sold back to the carrier/operator at a reduced cost. If after the trade, the same users have increased traffic and requires extra resources, the carrier sells the resources to such a user at an increased rate. The scheme reduces the tenant's cost and increases the operators' revenue. However, the initial over-allocation of resources reduces resource utilization and operator revenue as the operator may have to reject other requests until later on when excess resources will be resold to the carrier for onward. Also, tenants operate at a loss when they resale the excess resources to the carriers as they sell at lower rates and buy at much higher rates if and when they need the resources again.

A call admission control (CAC) scheme that enables mobile network operators to maximize the revenue is proposed by [101]. The CAC scheme logically divides the network into different network slices with each belonging to a tenant. It also characterizes each network slice by its traffic type (elastic or inelastic) and its number of users. The scheme ensures that all tenant's users are provided with service guarantees as contained in the SLA as long as the tenant does not introduce more users than allowed by the slice size. A bidding system is also proposed by [101] to ensure that network slices are dynami-

cally allocated to tenants. The infrastructure provider, accepts or rejects these requests depending on the current network load. When the network is close to congestion, the scheme trades off the admission of requests that provide lower revenue for future requests that may provide higher revenue. The scheme maximizes revenue for infrastructure operators but increases the dropping rate and is unfair to requests with low revenue prospects. The scheme also wastes resources and consequently reduces network resource utilization when the anticipated high revenue yielding requests never arrive.

In [95], a homogeneous reinforcement learning (RL) based admission control scheme is proposed to maximize operator revenue. The scheme classifies slice requests into high priority (HP) and low priority (LP) and projects time-changing resource requirement of a slice such that a slice may need to be scaled up or down to meet its resource needs at a time if its resources exceed/fall-below a given threshold. Thus, whenever a slice requires a scale up but the InP cannot meet up, the InP is required to pay a penalty that is equivalent to the number of resources that could not be provided multiplied by a penalty factor that increases with the priority of the slice (i.e. the higher the priority, the higher the penalty factor). Its decision on whether or not to accept a slice depends on the potential revenue a slice can generate and the penalty it will pay when sufficient resources are not available to match the time varying resource requirements of the slice. Thus, it sometimes rejects LP slice requests even when resources are available by reserving the resources for future HP requests. The scheme maximizes revenue for operators and guarantees QoS. However, it reduces resource utilization and increases the blocking rate as well as reduces the throughput of LP requests as they can be rejected due to the reservation of resources for HP slices that may never require them.

To improve QoS, [106] proposed an overload cost and requirements based algorithm. The scheme considers three use cases (uRRLL, eMBB and mMTC). When a new slice request arrives, it first checks to determine if the requirements/SLA (this agreement comprises the reliability, availability and latency) can be met by any of the existing slices and selects candidate slices. Where reliability is the ability to deliver a requested service correctly and without interruption while the availability is the ability to deliver a service when requested in order to fulfill the required functions. Next, it selects among the candidate slices, the slice with the lowest total overloading cost, accepts and places it in that slice where the total overloading cost is a function of the overloading costs of the virtual machine (CPU and the memory utilization rates), links overloading cost (bandwidth of the link) and switches overloading cost (cost from forwarding nodes). If however, the requirements cannot be met by any of the existing slices, a new slice is created else, the slice request is rejected. The scheme improves QoS but reduces overall system throughput and resource utilization as some of the resources will be overprovisioned but left unutilized/underutilized even when new slice requests are rejected.

A utility-based algorithm (UBA) is proposed by [107] to improve QoS. First, the scheme sets a delay threshold and defines an increasing utility function (the variation of utility values is proportional to the changes in variable values) by computing the score of the network resource (between the source node and the destination node for different paths) which is based on three parameters: availability, reliability and latency. The score returns the values between 0 and 1 (the higher the score, the better). Next, it selects the path with the best value (non-null value/1) and consequently chooses the slice as the best to accept the new request in. The scheme assigns statics weight of 1/3 for each of the parameters and the sum of all the weights for reliability, availability and latency is 1. Whenever there is no path with non-null value, it indicates that the delay exceeds the delay threshold at which point, a new slice will be created to meet the requirement of the new slice request. The scheme improves QoS for all use cases. However, the static delay threshold set will lead to resource wastage and/ under-utilization and consequently reduced system throughput as the threshold is set without recourse to the varying delay requirement of different slices/tenants.

A utility-driven multi-queue admission control scheme is proposed by [102]. The scheme considers a homogenous tenancy in which each tenant caters only to a particular type of service. A binary admission decision is taken to accept or reject new slice requests depending on the slice type and network resources are close to saturation. High priority slice requests are accepted with low priority requests only considered when there are no more high priority requests. It also queues the various rejected slice requests into different queues depending on the slice type. Furthermore, rejected slice requests are resent for reconsideration after some predetermined delay period with hope that some resources would have been freed by departed connections with the queues served based on their priority. The scheme increases the acceptance rate due to the retrial introduced. However, the static nature of the assignment coupled with the priority-based consideration of the queues will lead to low resource utilization as low priority requests will not be considered until the high priority requests are exhausted even when the available resources are sufficient to accept the low priority requests but not the high priority requests.

In [103], a service admission control algorithm for radio access network (RAN) and core slicing is proposed to improve operators' profit. The scheme considers slicing of both RAN and core network (CN) resources. The virtual CN resources include computation, storage and networking while that of the RAN is bandwidth. The acceptance of slice requests is determined based on the profit to be accrued and resources to be consumed by the slice thus, improving the operator's revenue. However, by basing the admission decision on profit and resources to be consumed, it reduces resource utilization and throughput as low resource generating slices will be rejected even when the resources available can accommodate such slices.

A proactive network slices management algorithm based on fuzzy logic and support vector regression model (SVR) is proposed by [108] to improve QoS. The scheme predicts the state of slice resources and then applies a convenient management decision based on fuzzy logic system. First, information is gathered about network slices and virtual resources (CPU, memory and disk utilization rates). Secondly, resource usage is predicted using the SVR prediction model to forecast the stationarity of the slice state. Thirdly, the appropriate management action is triggered as determined by the fuzzy logic system based on the actual and predicted resources utilization rates. Lastly, the management decision is executed which can either be to scale-up (increase the resources), scale-down (decrease the resources) or scale-out (create new slices). The scheme reduces the network processing delay. However, just like in other prediction-based schemes, resources are wasted if the projected/predicted usage is greater than the actual quantity needed. Similarly, it will lead to underproivisioning of resources when the resource need is greater than predicted, which can have dare consequences for low latency slices/applications.

In [109], three CAC algorithms are proposed; Q-learning (Q-learning), deep Q-learning (DQL) and regret matching (RM). The schemes consider four parameters: the number of resources required, the slice hosting time, the price of the slice and the priority where the price of a slice per time unit is proportional to the priority. The reward for accepting a slice is equal to the penalty of accepting the slice without adequate resources to cater for its resource needs. The Q-learning approach which is an offline reinforcement learning algorithm, generates an optimal policy to maximize the expected total reward. QL enables the agent to select the best action based on the Q-value but the convergence time (the time needed by the agent to explore all the states to learn the best action to take in the future) is too high and it greatly increases the training time if the state space is large as the agent has to take more time to explore all the states. The DQL approach employs deep learning to represent the Q-values, with each state passing through several hidden layers of a neural network to get the Q-values after which the DQL calculates the loss function and the target Q-value. It employs two neural networks (one for predicted and the other for the target Q-values). In the RM algorithm, which is an online learning algorithm, the agent (network operator) looks for the right action based on the regrets of the previous actions. RM mainly consists of minimizing the regrets of its decisions at each step of the system by relying on past behavior of previous actions taken to guide its future decisions such that it favours the actions that it regrets not to have chosen before. An evaluation of the three proposed algorithms revealed that the RM outperforms the DQL and QL while the DQL outperforms the QL in terms of reduced penalty and increased reward. Generally, the schemes provide QoS guarantees to tenants. However, the schemes do not efficiently utilize network resources as they do not consider other network parameters. More so, by setting a penalty equal to the reward, the schemes

behave conservatively thereby reducing the number of accepted slices and consequently reducing operator revenue and system throughput.

In [104], a reinforcement-learning based admission control is proposed to maximize profit. The authors proposed two CAC algorithms: QL and R-Learning. The QL algorithm employs the concept of temporary difference and keeps track of values of each action in each state in a Q-table. At each state, the action with the maximum value determined by the values and exploration strategy is selected. The next state is observed, a reward is obtained from the environment and the state updated, after which it selects a random action from the action set. Similarly, the R-learning algorithm acts like the QL algorithm but differs in that in the RL, the state-action values are not the expected discounted reward. Rather, the reward is independent of the state and is learned over time. The RL algorithm improves reward/revenue but will not perform optimally when the state space grows. More so, the computational resources required will lead to increased cost for operators and consequently tenants and users.

A computer vision inspired reinforcement learning with DeepQ Networks based algorithm is proposed in [113] to maximize the InP's long term revenue . It considers three basic use cases and requests with attributes such as request-id, service class, duration, patience (how long the request is willing to wait in queue before been served) and urgency (how urgent a request needs its resources). Requests in the waiting queue are processed on a potential revenue basis with a request accepted if there are sufficient resources to meet the requests' needs else the request is rejected. A dynamic pricing model is employed that defines the revenue accrued from admitting a request as a product of the request's urgency, duration and sum of the product of the supply/demand function and base price fair. The scheme improves the InP's revenue but reduces acceptance rate. More so, the pricing model employed where the price of resources is fixed based on the forces of demand and supply is not realistic in real network scenario where prices changes do not happen on the fly.

In [114], a prioritized slice admission control (SAC) considering fairness (PSACCF) is proposed to improve fairness. The PSACCF is a multi-queue based online SAC algorithm based on two parameters: priority level and the threshold of acceptable fairness. It employs the concept of cumulative service acceptance ratio which enforces priority as a preference instead of the absolute privilege that usually characterizes priority.

A squatting and kicking (SKM) scheme is proposed in [98] to improve resource utilization where squatting is the process of occupying resources allocated to other classes/slices when their holders are not using them and kicking is the process of expelling a lower priority class/slice from its allocated resources, either partially or totally. The SKM scheme employs three key concepts: resource partitioning, reservation and borrowing such that

resources are partitioned and reserved according to the various priority classes as well as borrowed when other slices do not require them. That is, whenever there are no sufficient resources to accept a new slice, it checks the availability of unused resources, beginning with higher priority slices so as to admit the new slices using the unused resources of the high priority classes. Otherwise, it proceeds to check and use the resources belonging to low priority slices. If after these the resources are still not sufficient, it kicks (drops the lower priority slices) and if still not sufficient, the slice request is rejected. To eliminate squatting, the scheme sorts slice requests in descending order of priorities, thus ensuring that low priority slices are not admitted. The SKM scheme improves resource utilization and increases acceptance rate of high priority slices. However, the scheme's squatting and kicking of existing slices without recourse to the needed resources as well as the sufficiency of the kicked resources may reduce resource utilization and throughput of the network. It may also lead to the violation of SLAs when there is total kicking. More so, the assumed buffering of requests is based is not ideal for real time network scenarios as most times these requests may come in order of days, weeks, months or even years.

In [105], a DRL based admission control and slice placement algorithm is proposed to increase revenue and minimize cost of deployment. To maximize revenue, the scheme accepts slices with more profitability in the long run and minimizes the cost of deployment in open RAN (O-RAN) enabled network by efficient placement of the slices. The factors considered for profit maximization are: revenue generated from accepting more profitable slices and cost of idle power consumption of servers in various clouds for deploying the slices. The proposed DRL based solution employs proximal policy optimization. The optimization is subject to five constraints: first, the total processing placed in any edge/regional cloud server cannot exceed its capacity at any given time. Secondly, the traffic flow in the front-haul, mid-haul and back-haul cannot exceed their total capacity. Thirdly, the total delay imposed by the front-haul, mid-haul and back-haul links used for routing a slice cannot exceed the delay budget of that slice. To prevent the RL agent from taking unwanted action during training as is customary, invalid action masking is employed, leading to faster convergence with a smaller number of episodes. The scheme improves operator revenue and reduces cost. However, this approach is not suitable for deployment in heterogeneous 5G/beyond 5G (B5G) use cases scenario as some may never get accepted due to the profit-centric nature of the scheme. That is, in heterogeneous slice scenarios, mMTC slices for example with low revenue generating tendencies may greatly suffer rejection. Similarly, the high mobility tendencies and resource demanding nature of eMBB services will mean that in the event that they co-exist with other uses cases, they have higher chances of been accepted at the expense of even delay sensitive uRLLC use case slices.

A DRL-based slice admission solution, prediction and RL based overbooking (PRLOV)

is proposed in [110] to maximize revenue and improve resource utilization. The PRLOV considers two slice types: low priority and low revenue (elastic) and high priority and high revenue (inelastic). It maximizes revenue and improves resource utilization by employing the concept of overbooking, where the InP accepts more slices than the available capacity while considering slice priorities after predicting the future resource demands of elastic slices to accept more slices. PRLOV predicts the future resource needs (CPU, memory, bandwidth) of ongoing elastic slices by employing a slice forecasting block using a long short-term memory (LSTM) network. This prediction is used by an opportunistic overbooking heuristic to allocate resources and dynamically adjust the overbooking amount. Overbooking is activated when there are new requests without corresponding resources to accept them to determine the number of resources that can overbooked. The scheme increases revenue for InPs when the arrival rate of inelastic slices is high and improves resource utilization. However, the scheme's overbooking without recourse to the needed resources may lead to reduced revenue for the InP in the event that the required resources are less than the overbooked resources. It also lowers acceptance rate of elastic slice requests even at low arrival rate of requests due to the reservation used.

In [116], an online slice admission control (OSAC) algorithm is proposed to maximize revenue. The OSAC employs either linear or exponential reservation policies (LinRP and ExpRP) to admit requests into the network and determines the cost of admitting a slice by multiplying the current scarcity of each resource by the corresponding resource requirement of a request. It also discourages high utilization of a resource so as to increase the chances of admitting higher valued future requests and avoid performance degradation from resource sharing between co-located slices. Furthermore, the concept of willingness-to-pay ratio is employed to measure how willing a tenant is willing to pay. The willingness-to-pay ratio is defined as the ratio between the maximum and minimum price per resource unit per time slot that the tenants are willing to pay to reserve resources on the provider's infrastructure. For the LinRP policy, the higher the utilization of a given (single) resource, the higher the admission cost of admitting a new request requesting that resource. That is, when the overall network infrastructure has a higher load and consequently higher resource utilization, the admission cost increases too to reserve the resources for higher valued requests. For the ExpRP policy when there is exponential reservation. That is the sum of scarcities in each dimension is taken into consideration, making it a more conservative policy compared to the LinRP and making it suitable for very high resource utilization scenarios to ensure that only the highest valued requests are accepted. Overall, the LinRP policy is applied when the scarcity is based on the resource with the higher scarcity or normalized utilization while when the ExpRP is applied, the sum of scarcities in each dimension is summed up. A request is admitted if the value of the request is higher than the admission threshold for either policy and there is enough space in the

system to accommodate the resource requirements of the request else it is rejected. At the end of each time slot requests that have reached their completion time are removed from the system and their resources added to the resource pool. The scheme increases net revenue. However, it reduces acceptance rate and decreases resource utilization.

A summary of the various schemes reviewed, highlighting their objective (s), strengths and weaknesses is presented in Table 3.1.

Following the extensive review of literature, the following gaps have been identified:

- Low long-term revenue due to the fixed pricing employed that lead to the admission of requests that offer high short-term revenues but that may lead to low revenue in the long term against the low priority requests that offer low short-term revenues in the interim but have the potential to yield high revenue in the long term [105], [110], [114]. For instance, if an InP has requests from a hospital/school and a concert organizer, the organizer may be more willing and have to pay more given the type of services it will require. If the InP rejects slice requests of the other given that it will yield more revenue, this will lead to lower long term revenue as the concert may be for just a few days whereas the request from the school/hospital will be for a longer period of time even though the immediate revenue is not as much as the other

- Low resource utilization occasioned by the fixed priorities employed that consider only the price or revenue to be accrued from such slices [105], [110]. That is, when priorities are wholly based on price and resource reservations are made (mostly in excess of what is required to avoid SLA violation), this can lead to low resource utilization as most of the resources will be left unused.

- More so, the case of heterogeneous slices is seldom discussed in the literature whereas in reality, it is one of the most prevalent scenarios as most of the tenants will often comprise of mobile network operators (MNOs), requesting slices in order to serve customers with services cutting across diverse use cases. In the event where a tenant is considered to be able to request slices of only one type but can request multiple slices, managing these resources becomes a herculean task.

Table 3.1: A Recap of some literature.

Paper	Objective	Strength	weaknesses
[95]	Revenue maximization	-Maximizes operators' revenue -Guarantees QoS	- Low resource utilization - Increases the rejection rate of LP requests due to the reservation of resources for HP slices that may never require them
[98]	Improve resource utilization	-Increases acceptance rate of high priority slices	- Low resource utilization and throughput due to arbitrary squatting without recourse to what is needed
[101]	Revenue maximization and QoS	-Improved short term revenue	-Low resource utilization due to anticipatory reservation -Unfair to low priority slices
[102]	Revenue maximization	-Increased acceptance rate of high priority slices	-Low resource utilization rate
[103]	Revenue maximization	- Increases operator revenue	- Low resource utilization - Wastes resources due to the reservation
[105]	Increase revenue and minimize cost of deployment	-Improves operator revenue	-Low system resource utilization -High rejection rate as low priced slices will be rejected even in the presence of unused resources
[108]	QoS Control	-Reduces delay	-Lower resource utilization
[109]	QoS guarantee	-Guarantees QoS	-Low resources utilization -Low acceptance rate and revenue
[110]	Revenue maximization, resource utilization, QoS	-Increases revenue at high arrival rate of inelastic slices -Improves resource utilization	-Reduced revenue due to overbooking without recourse to the needed resources. -Low acceptance rate of low priority requests
[117]	Revenue maximization & QoS	Guarantees SLA & Increases revenue	Low resource utilization
[118]	Revenue maximization	-High long-term revenue	-Low resources utilization -Low acceptance rate and revenue due to the conservative nature that sets the penalty = to the revenue to be accrued
[119]	Revenue maximization	-Increases revenue	-Unfairness -Low resource utilization
[120]	Revenue maximization	-Increases revenue	-Low resources utilization due to spare resource reservation
[121]	Revenue maximization & Resource utilization	-Increases revenue -Improves resource utilization	-Low overall system resource utilization due to overprovisioning.

3.3 Slice Admission Control Scheme with Overbooking and Dynamic Buyback Process

This section presents an admission control algorithm that employs the concept of overbooking which allows the InP to admit slice requests beyond its nominal available resources. The proposed scheme also employs a dynamic queue adaption priority, step-wise pooling and dynamic buyback price mechanism to ensure efficient and profitable admission decision for the InP. The dynamic queue adaption priority mechanism is employed to determine the admission priority of the slice requests. It is employed to strike a balance between the revenue to be accrued from accepting a request and the total resources needed by the request. Similarly, the step-wise pooling mechanism is employed to reduce the buyback cost of the InP buy ensuring that the InP only buys back the amount of idle resources it needs as opposed to buying back every available idle resources. Furthermore,

the dynamic buyback price is a more practical and realistic price employed to ensure a win-win situation between the InP and slice tenants with idle resources to sell back to the InP. It determines the price an InP will pay to existing slice tenants to which the InP desires to buyback idle resources from.

We assess the performance of the proposed algorithm against SoA solution considering different priority schemes. The network model and system formulation are presented in subsection 3.3.1 and 3.3.2, the proposed algorithm in subsection 3.3.3 and the performance evaluation in subsection 3.3.4. The simulation settings are presented in subsection 3.3.5, the priority schemes in subsection 3.3.6 and comparison of the proposed solution against the SoA solutions in subsection 3.3.7.

3.3.1 System Model (P1)

We consider a scenario with a single InP having a set of resources, namely computing, storage and network resources, to serve slice requests of different types and diverse resource requirements. Indeed, we consider a set of predefined classes of slices with diverse priorities, denoted in the sequel as $\{C_1, \ldots, C_6\}$. The different classes C_1, C_2, C_3, C_4, C_5, C_6 correspond to homogeneous - uRLLC, heterogeneous - uRLLC, heterogeneous - eMBB, homogeneous - eMBB, heterogeneous - mMTC and homogeneous - mMTC slice classes, respectively. Homogeneous - uRLLC slices, homogeneous - eMBB slices and homogeneous - mMTC slices are the slices that support only uRLLC, eMBB and mMTC use cases, respectively. On the other hand, heterogeneous - uRLLC slices, heterogeneous - eMBB slices and heterogeneous - mMTC correspond to the slices that support multiple use cases but more of uRLLC, eMBB and mMTC, respectively. Moreover, we denote the computing, storage and network capacity of the InP as R_1, R_2 and R_3, respectively and the set of all the slice requests in queue, awaiting admission decision, as $\mathcal{N} = \{1, 2, ..., N\}$. For each slice request $i \in \mathcal{N}$, we define its duration d_i and set of requirements to be fulfilled as follows: computing $r_{i,1}$, storage $r_{i,2}$ and network resources $r_{i,3}$. The sketch of the system model is represented by Figure 3.2.

Furthermore, in the proposed model, we consider that the InP might buyback some resources from already accepted slices when there are requests but the resources in the InP's resource bank are not sufficient to admit any of them. We denote by $r'_{i,j}, \forall j \in \{1, 2, 3\}$ the amount of resources that slice i makes available, where $r'_{i,j} \leq r_{i,j}$. Indeed, this creates a win-win situation for the tenant and InP as the tenant will recoup some of its investment by selling unused resources, that it would have lost at the end and the InP makes some marginal profit while creating greater satisfaction for its clients. In addition, in contrast to existing solutions that unilaterally assume that all accepted slices are willing to sell the idle resources, we introduce the concept of willingness to sell probability to determine

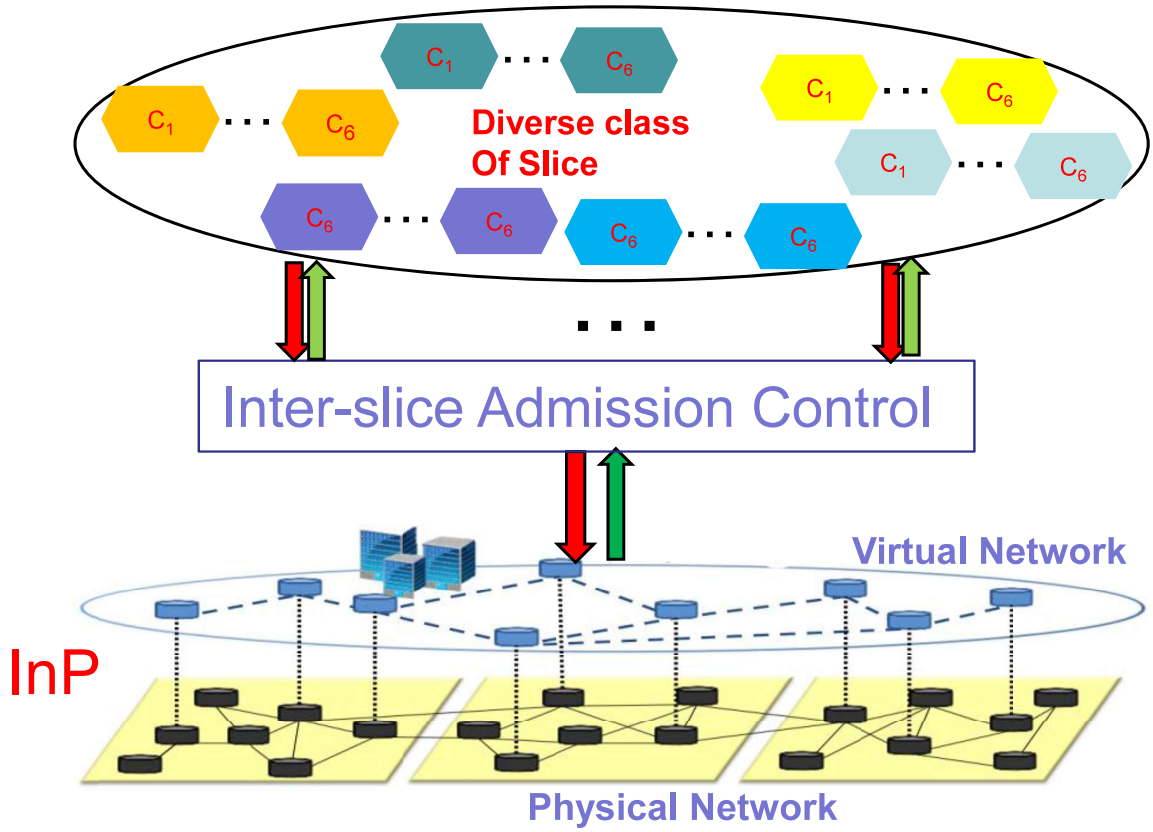


Figure 3.2: System Model (P1).

the likelihood of a slice being willing to sell its unused resources to the InP. Formally, we define a binary parameter δ_i as the indicator that an admitted request i is a candidate request for buyback (δ_i =1) or not (δ_i =0); δ_i can be defined according to relation (3.1).

$$\delta_i = \begin{cases} 1, & \text{if } P_i \leq \Gamma_i \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

Where: P_i and Γ_i are the willingness to sell probability and the minimum threshold probability of the admitted slice request i, respectively.

Therefore, the total resources R_j^s, $\forall j \in \{1,2,3\}$, that can be bought back is defined as:

$$R_j^s = \sum_{i=1}^{N} \delta_i r'_{i,j}. \tag{3.2}$$

3.3.2 Problem Formulation (P1)

In this subsection, we formulate the problem of admission control as a multi-objective problem. Indeed, the main objective is to simultaneously maximize the InP's revenue defined by U_1 and minimize the buyback cost defined by U_2, incurred for the recovered

resources from the admitted slices, given the request's requirements and the total capacities of the InP in terms of computing, storage and network resources. We introduce a binary variable x_i that indicates the successful admission of request i ($x_i = 1$) or not ($x_i = 0$). On the other hand, we define by $p_{i,j}$ and $p'_{i,j}$ the selling unit price and the buyback unit price per unit of the resource R_j, $\forall j \in \{1,2,3\}$ and request $i \in \mathcal{N}$, respectively. Note that, in this work, we do not consider additional costs such as the service fees. Indeed, we consider the gross earnings from admitting tenants, because we assume that an InP will not render services at a loss and as such all unit prices set are those that will be profitable to the InP.

Formally, the admission control problem including a buyback scheme is formulated as follows:

$$\textbf{[P1]} \qquad \begin{cases} U_1 = \max\limits_{x_i \in \{0,1\}} \sum_{i=1}^{N} \sum_{j=1}^{3} x_i r_{i,j} p_{i,j} d_i \\ U_2 = \min\limits_{x_i \in \{0,1\}} \sum_{i=1}^{N} \sum_{j=1}^{3} x_i \delta_i r'_{i,j} p'_{i,j} \end{cases} \tag{3.3}$$

Subject to:

$$\sum_{i=1}^{N} x_i r_{i,j} \leq R_j, \qquad \forall j \in \{1,2,3\}, \tag{3.4}$$

$$x_i \in \{0,1\}, \qquad \forall i \in \mathcal{N}. \tag{3.5}$$

The objective in (3.3) is to maximize total revenue and minimize the buyback cost that depends on the total admitted requests and the corresponding required resources, prices and duration. The constraints (3.4) are introduced to ensure that the total amount of resources distributed over the admitted slices cannot exceed the total capacity of the InP. In addition, the constraints (3.5) indicate the binary decision variables of the problem.

3.3.3 Proposed Algorithm for P1

The objective is to obtain the number of admitted slices which maximizes the total revenue and minimize the buyback price defined by (3.3) subject to the system capacity and the slices' requirements (3.4). To solve this problem, in this section, we propose a greedy heuristic algorithm to seek a near-optimal solution. The choice of a greedy heuristic algorithm is due to the ability of the greedy heuristic to always choose the best price to cost ratio in each iteration and provide the best polynomial-time approximation. The pseudo-code of the proposed algorithm is presented in Algorithm 1.

Initially, we fix all the parameters including the number of slice requests and their

requirements, the InP capacity in terms of computing, storage and network resources, the willingness to sell probability and the threshold and the different prices. The proposed algorithm sorts the slice requests according to a predefined priority, we denote α_i $\forall i \in \mathcal{N}$ and creates the new ordered set $\mathcal{L}^s$ (line 3). This metric, namely α_i, $\forall i \in \mathcal{N}$, is defined so that the InP's utility (3.3) is improved as much as possible. We search the requests from top to bottom according to α_i and admit the slice if this does not violate the capacity constraint of the InP (3.4). Once the slice is admitted the total number of available resources at the InP, R_j, $\forall j \in \{1,2,3\}$, and the accepted requests list, denoted by $\mathcal{L}^a$, are updated accordingly (line 5-8). However, if the units of resources in the InP's resource pool is not sufficient to meet the request's requirement, the request is added to the waiting list, denoted by $\mathcal{L}^w$ and the amount of needed resources to satisfy a maximum number of slice requests is updated R_j^N,$\forall j \in \{1,2,3\}$ (lines 9-12).

Next, the candidate requests are determined from the list of accepted requests. We denote the set of candidate requests by $\mathcal{L}^c$. Indeed, this allows the InP to satisfy waiting requests in $\mathcal{L}^w$ that have not been attended to due to insufficient resources. To determine which requests to buyback from, namely the set $\mathcal{L}^c$ and the corresponding resources R_j^s, $\forall j \in \{1,2,3\}$, we exploit the willingness to sell probability P_i (line 13-line 16). Afterwards, we introduce the incremental/step-wise pooling mechanism based on a sufficiency pre-check mechanism (line 22). Indeed, this is considered to ensure that even when there are much shareable/idle resources, the InPs do not buyback more than is necessary at a particular point in time as this can adversely affect their revenue and willingness to buyback.

Based on this mechanism, if the amount of resources required by all the requests of the waiting list, R_j^N, is less than the sum of the amount of resources remaining in the InP's resource bank and the resources that can be bought back from candidate requests, R_j^s, we employ partial pooling to buyback only the amount of resources needed to meet the demand of requests in the waiting list, accept them and update the resource pool accordingly (line 19- line 24). Otherwise, we employ full pooling procedure where the InP buys back all the idle resources R_j^s from candidate requests given by (3.2). Thereafter, the algorithm accepts the slices in the waiting list from top to down according to α_i and update the resources accordingly (line 27- line 31). At last, if a request is still not satisfied, it is considered rejected (line 33).

3.3.3.1 Time Complexity Analysis for Algorithm 1

The time complexity of the various steps of Algorithm 1 is as follows: the sorting procedure (line 3-4), admission without buyback and updating of resources (line 5-12), determination of willingness to sell and updating of idle resources (line 13-17), buyback and admission after buyback (line 18-33), each have complexity of $O(N\log(N))$, $O(N)$, $O(N)$

and $O(J2N)$ respectively. Thus, the time complexity is $O(N\log N + 2N + J(2N))$.

Algorithm 1: Proposed admission control algorithm for P1.

Input: $R_j, r_{i,j}, P_i, p_{i,j}, \Gamma_i$, $R_j^s = 0$, $R_j^N = 0$, $\mathcal{L}^a = \emptyset$, $\mathcal{L}^w = \emptyset$, $\mathcal{L}^c = \emptyset$, $\mathcal{L}^s = \emptyset$, α_i, $\forall i \in N$ and $\forall j \in \{1,2,3\}$.

Output: x_i, $\forall i \in \mathcal{N}$

```
1  begin
3      sort the slice requests in decreasing order of priority defined by α_i and add them to
       L^s.
5      for i ∈ L^s do
6          if r_{i,j} ≤ R_j, ∀j ∈ {1,2,3} then
7              L^a = L^a ∪ i and x_i = 1
8              R_j = R_j − r_{i,j}, ∀j ∈ {1,2,3}
9          else
10             L^w = L^w ∪ i
11             R_j^N = R_j^N + r_{i,j}, ∀j ∈ {1,2,3} /* update needed resource */
13     for i ∈ L^a do
14         if P_i ≤ Γ_i then
15             L^c = L^c ∪ i
16             R_j^s = R_j^s + r'_{i,j}, ∀j ∈ {1,2,3} /* add idle resources to shareable
                   resources */
18     if R_j^N < (R_j + R_j^s), ∀j ∈ {1,2,3} then
           /* Activate partial pooling */
19         for i ∈ L^c do
20             if (R_j + r'_{i,j}) ≤ R_j^N, ∀j ∈ {1,2,3} then
21                 R_j = R_j + r'_{i,j}, ∀j ∈ {1,2,3}
22         for i ∈ L^w do
23             L^a = L^a ∪ i and x_i = 1
24             R_j = R_j − r_{i,j}, ∀j ∈ {1,2,3}
26     else
           /* Activate full Pooling */
27         R_j = R_j + R_j^s, ∀j ∈ {1,2,3}
28         for i ∈ L^w do
29             if r_{i,j} ≤ R_j, ∀j ∈ {1,2,3} then
30                 L^a = L^a ∪ i and x_i = 1
31                 R_j = R_j − r_{i,j}, ∀j ∈ {1,2,3}
32             else
33                 the slice request i is rejected
```

3.3.4 Performance Evaluation of the Solution for P1

In this section, we assess the performance of the proposed admission control algorithm and compare it with a SoA algorithm [110]. We present the simulation settings in Section 3.3.5, the proposed priority schemes in Section 3.3.6 and some results in Section 3.3.7.

3.3.5 Simulation Settings for P1

The simulation scenario consists of an InP having fixed units of three resources : computing $R_1 = 10,000$, storage $R_2 = 30,000$ and network $R_3 = 50,000$. Moreover, the number of requests is generated randomly between 200 and 1, 300. The nominal priority of each class $C_i, i \in \{1, \ldots, 6\}$ correspond to $6, 5, 4, 3, 2, 1$, respectively; while the willingness to sell threshold Γ_i is set to 0.6, $\forall i \in \mathcal{N}$.

We introduce a dynamic buyback price that the InP have to pay to the tenant from which it buys back some resources. It is worthy to note that while existing solutions set the buyback price to the same price as initially paid by the tenant or higher [104], [113], [105], [114]. We propose a more realistic and consistent model defined as:

$$p'_{i,j} = \frac{d_i^r}{d_i} \times p_{i,j}, \tag{3.6}$$

where: d_i^r, d_i and $p_{i,j}$ are the remaining validity duration, duration and initial unit price paid by the tenant for that resource, respectively. Note that $0 < p'_{i,j} < p_{i,j}$ and $d_i^r < d_i$.

The willingness to sell probability of a slice is defined as:

$$P_i = \frac{r_{i,j}^u}{r_{i,j}} \times \frac{d_i^u}{d_i}, \tag{3.7}$$

where : $r_{i,j}^u$ is the amount of resources already used by the request from the total resources allocated to it $r_{i,j}$; d_i^u and d_i are the used duration and total duration of slice i, respectively. This probability indicates to what extent the tenant has used its resources and how long they have been used. This helps in creating a more realistic estimate of whether or not a tenant may be willing to sell part of it's unused resources as oppose to arbitrarily deciding that all or some users will sell their unused resources. The simulation parameters are captured in Table 3.2.

3.3.6 Proposed Priority Schemes

The proposed admission control algorithm 1 is executed given a priority metric α_i which is defined for each slice request $i \in \mathcal{N}$. Therefore, we propose four priority metrics, we describe in the following.

Table 3.2: Simulation Parameters for P1.

Parameter	**Value(s**
Nominal priority	{6,5,4,3,2,1}
$d_i \ \forall i \in \mathcal{N}$	random between 1 and 12 months
Willingness to sell threshold (Γ_i)	0.6
Number of requests N	random between 200 and 1300
InP's capacity	$R_1 = 10^3, R_2 = 3.10^3, R_3 = 5.10^3$
$r_{i,1} \ \forall i \in \mathcal{N}$	random between 1 and 50 units
$r_{i,2} \ \forall i \in \mathcal{N}$	random between 1 and 100 units
$r_{i,3} \ \forall i \in \mathcal{N}$	random between 1 and 500 units
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class C_1	$p_{1,1} = 10, p_{1,2} = 9, p_{1,3} = 8$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class C_2	$p_{1,1} = 9, p_{1,2} = 8, p_{1,3} = 7$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class C_3	$p_{1,1} = 8, p_{1,2} = 7, p_{1,3} = 7$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class C_4	$p_{1,1} = 7, p_{1,2} = 6, p_{1,3} = 5$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class C_5	$p_{1,1} = 6, p_{1,2} = 5, p_{1,3} = 5$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class C_6	$p_{1,1} = 5, p_{1,2} = 4, p_{1,3} = 3$

3.3.6.1 Revenue-Cost Ratio Based Scheme (RCRS)

In this scheme, the algorithm employs a dynamic priority defined as a ratio of the revenue to be accrued from accepting a request i to the total resources needed by this request.

$$\alpha_i = \frac{U_{1,i}}{\sum_{j=1}^{3} r_{i,j}}, \tag{3.8}$$

where: $U_{1,i}$ and $\sum_{j=1}^{3} r_{i,j}$ are the revenue to be accrued to the InP from admitting the slice request i and the total required resources (computing: $r_{i,1}$, storage: $r_{i,2}$ and network: $r_{i,3}$), respectively.

3.3.6.2 Revenue Based Scheme (RS)

The priority for the revenue based scheme corresponds to the revenue only. That is, this scheme prioritizes requests on the basis of the revenue to be accrued from the acceptance of such requests. The higher the revenue potential, the higher the priority of such request. It is defined as follows :

$$\alpha_i = U_{1,i}. \tag{3.9}$$

3.3.6.3 Priority-Duration Ratio Based Scheme (PDRS)

The priority in this scheme is calculated based on the nominal priority C_i and the requests duration as follows :

$$\alpha_i = \frac{C_i}{\frac{1}{N}\sum_{i=1}^{N} C_i} \times \frac{d_i}{\frac{1}{N}\sum_{i=1}^{N} d_i}. \tag{3.10}$$

This priority allows one to create a balance between the long term revenue of the InP and fairness to other low priority requests. Indeed, low priority requests with longer

duration will have their priority values increased, which will increase their chances of getting accepted as opposed to nominal priority based algorithms.

3.3.6.4 Class Based Scheme (CS)

In this scheme, we use the nominal priority inherited by requests from their class type $C_i, \forall i \in \{1, \ldots, 6\}$ like SoA schemes [96], [97].

$$\alpha_i = C_i, \tag{3.11}$$

where : $C_i \in \{1, 2, 3, 4, 5, 6\}$. That is, the CS scheme prioritize requests based on their class priority $C_i, \forall i \in \{1, \ldots, 6\}$, in decreasing order and deploys the full pooling mechanism by buying back all the idle resources from already admitted slices.

3.3.7 Comparison of Proposed Solution for P1 to SoA

In this subsection, we present the results showing the performance of the proposed algorithm, implemented with the different priority schemes, RCRS, RS, PDRS, CS, against the SoA [110] that accepts requests on the basis of their priority just like the CS scheme. However, the SoA only employs full pooling. Each value in the results is an average of 1000 simulation runs.

Figure 3.3 shows how the revenue earned by the InP from admitting the various slice requests scales with the number of requests N. The results indicate that at low slice requests rate, all implemented schemes have similar performance. However, as the number of requests increase, the provider exhausts the resources in its resource pool and has to buyback from willing already admitted tenants. The results indicate that from this point, the proposed algorithm, represented by RCRS, RS, PDRS, CS, outperforms the SoA algorithm. This improved performance is due to the dynamic priority introduced in the proposed algorithm and further demonstrates the impact of the priority on the revenue of the InP.

Figure 3.4 shows the performance in terms of buyback cost. We can see that the proposed algorithm, including RCRS, RS, PDRS, CS, incurs less buyback cost than the SoA algorithm right from the onset of the buyback process. The reason is that, all the proposed priority schemes employ step-wise pooling which enables the InP to only buyback as much as is required; thus, leading to lower buyback cost while with the SoA, the InP buys back without recourse to the needed resources.

Figure 3.5 shows the net revenue earned by the InP from admitting the various slice requests. It indicates that at a low slice requests rate, all the schemes have similar performance. However, when the number of requests increases, the provider runs out of

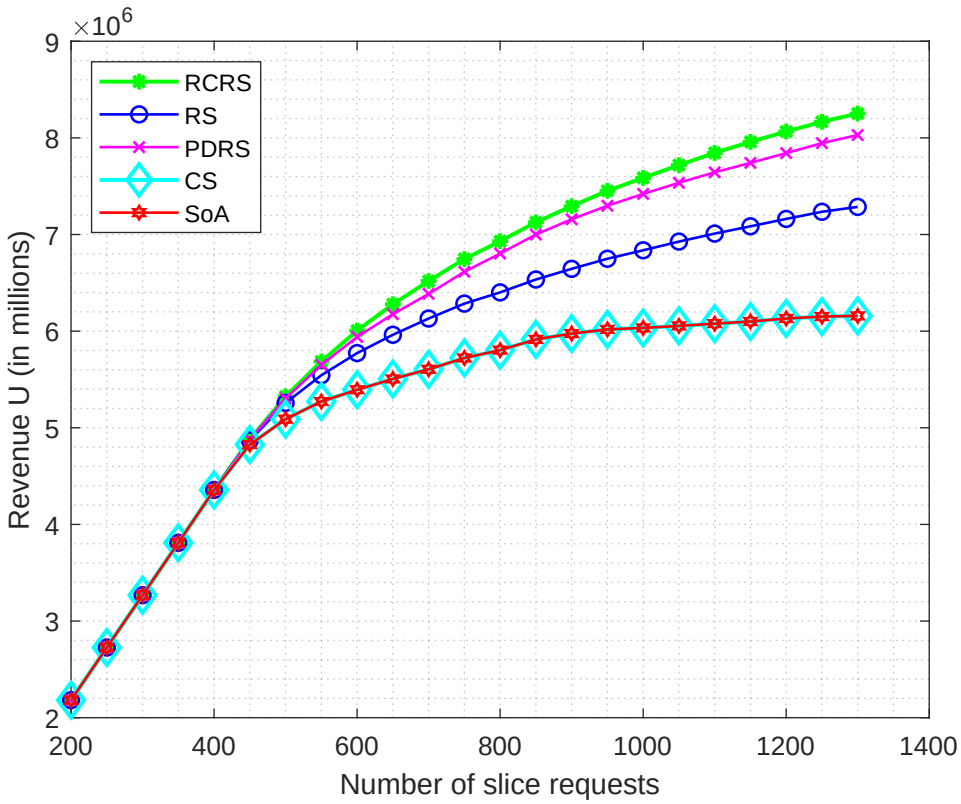


Figure 3.3: Comparison in terms of revenue between the proposed algorithm for P1 and the SoA algorithm, for different priority schemes.

resources and initiates a buyback process, resulting in a sharp contrast in the net revenue between the proposed algorithm and SoA solution. This is due to the dynamic priority and step-wise buyback mechanism introduced in the proposed algorithm, namely RCRS, RS, PDRS, CS, that ensures that the InP only buys back the units of resources needed as against buying back without recourse to the needed resources. Thus, reducing the provider's buyback cost and consequently increasing the net revenue.

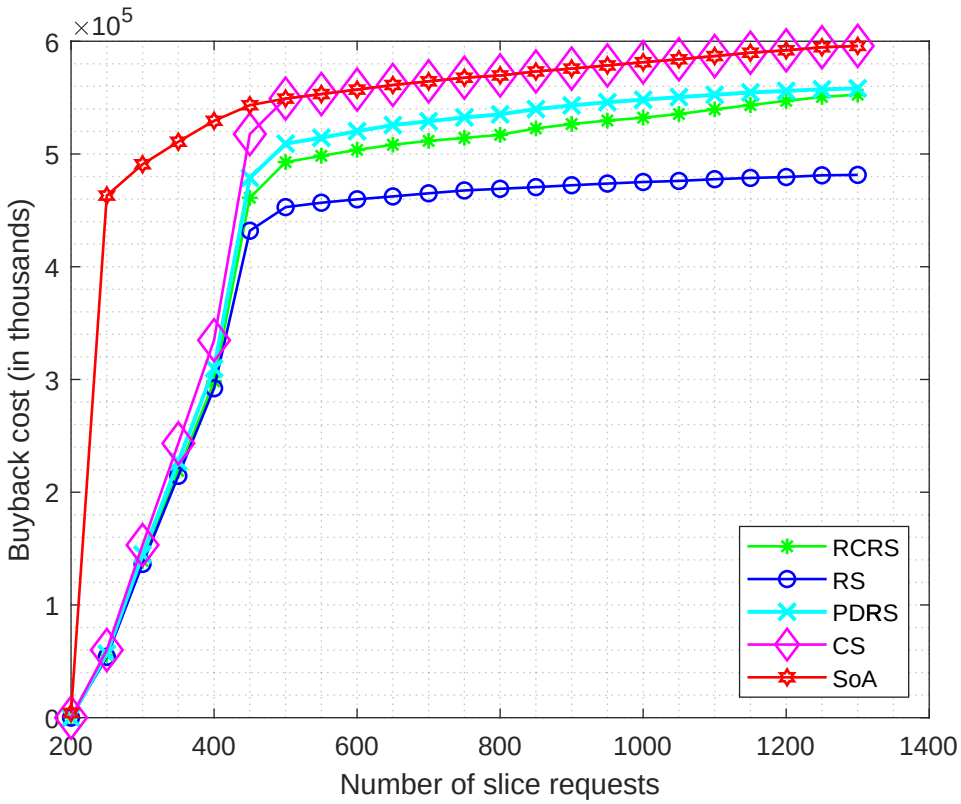


Figure 3.4: Comparison in terms of buyback cost between the proposed algorithm for P1 and the SoA algorithm.

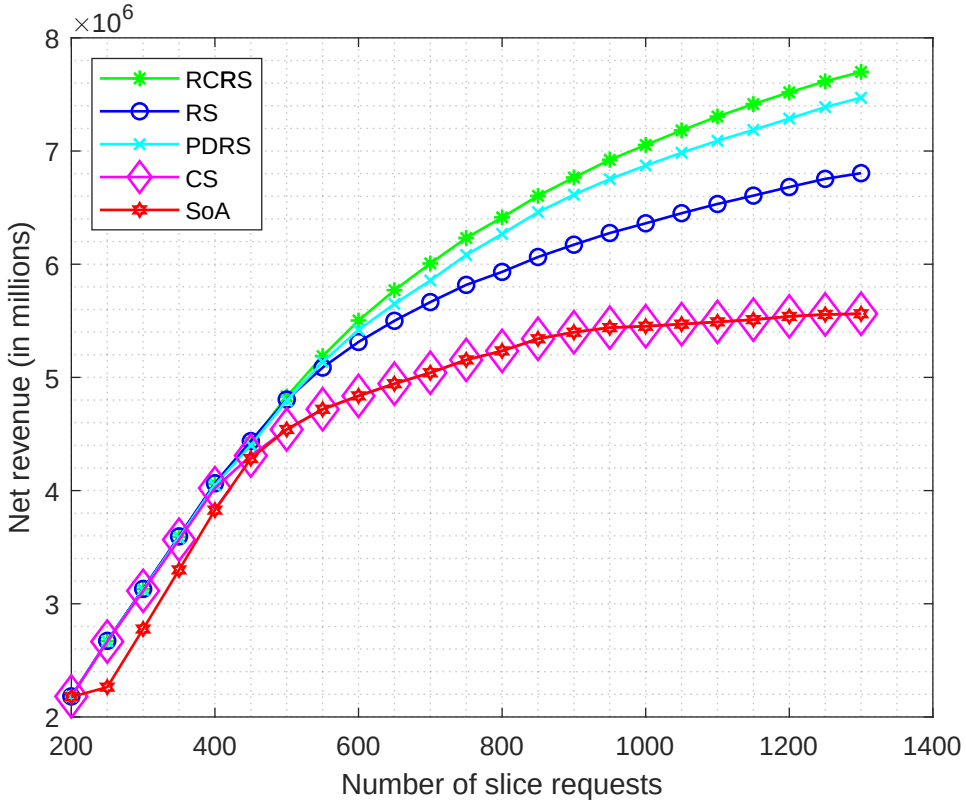


Figure 3.5: Comparison in terms of net revenue between the proposed algorithm for P1 and the SoA algorithm.

Figure 3.6 depicts the acceptance rate. The results show that the proposed algorithm slightly outperforms the SoA when it implements the priority schemes RCRS, PDRS, CS. However, implementing RS as a priority scheme provides the lowest acceptance rate. This can be attributed to the role played by the duration and resource units of the requests. That is, requests with higher duration and resource request, will yield higher revenues but also lead to the InP admitting less before running out of resources. Similarly, admitted requests with longer duration will invariably yield higher willingness to sell probability and be thus less willing to sell, making them admit less through the buyback process. This is reflected in the low buyback incurred by the RS.

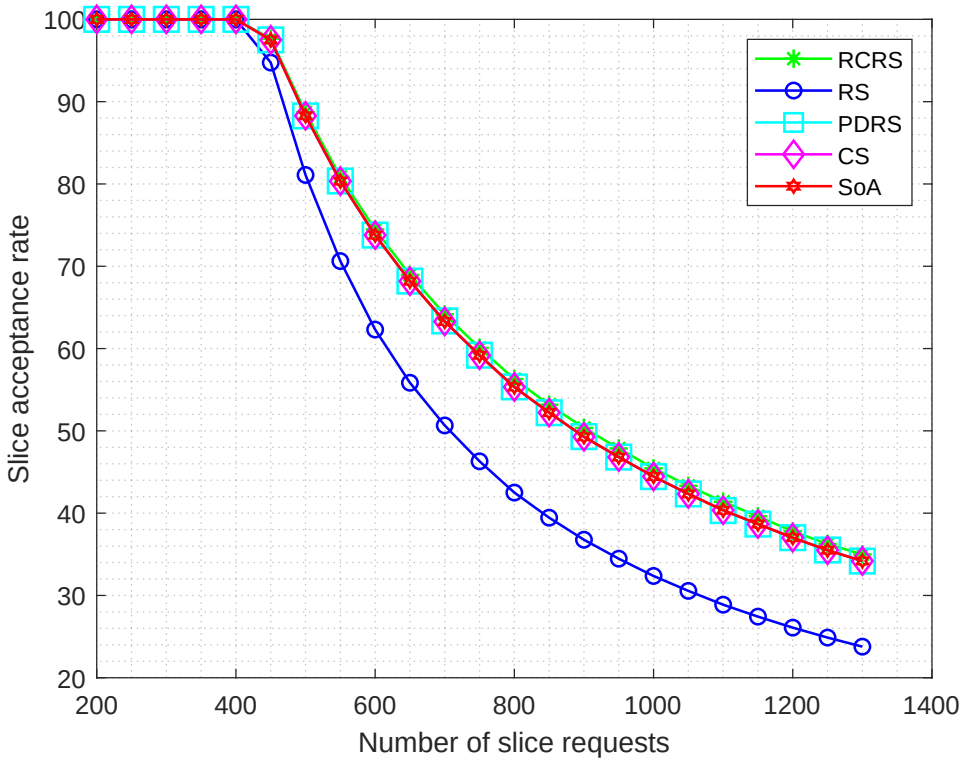


Figure 3.6: Comparison in terms of acceptance rate between the proposed algorithm for P1 and the SoA algorithm.

3.4 Efficient Online Slice Admission Control Scheme in Next Generation Networks

This section presents a second admission control algorithm that employs a dynamic priority and a capacity sharing mechanism to buy back idle resources from already accepted slices. It is an extension to the solution presented in section 3.3 where we considered an offline scenario assuming that the InP has all the requests at once and has to make admission decisions. This solution extends the former to a more realistic scenario where the InP has no knowledge of future network conditions and the AC is dynamic. Moreover, the requests due dates are taken into account in the decision process. We present the network model in subsection 3.4.1, problem formulation in subsection 3.4.2, the proposed algorithm in subsection 3.4.3 and the performance evaluation in subsection 3.4.4, with the simulation settings in subsection 3.4.5 and the comparison of the proposed solution against the SoA solutions in subsection 3.4.6.

3.4.1 System Model (P2)

The scenario we consider consists of an InP that deploys an AC algorithm, as depicted in Figure 3.7, to determine whether or not a slice request should be admitted based on the resource demands of the request and the InP's available resources. The InP has a set of resources that includes computing, storage and network, we denote in the sequel by R_1, R_2 and R_3, respectively. We consider a single AC period of time defined as $[T_{\text{start}}, T_{\text{end}}]$ that is subdivided into T slots of equal duration, T_{slot}, denoted by $\mathcal{T} = \{1, \ldots, T\}$. The InP has to serve many slice requests, denoted $\mathcal{N} = \{1, 2, ..., N\}$, of different types and diverse resource requirements. For each request $i \in \mathcal{N}$ we define its arrival time, duration (how long the slice needs resources for), maximum waiting time (the longest time it can wait before an admission decision) and a set of resource requirements in terms of computing, storage and network we denote as t_i, d_i, τ_i, $r_{i,1}$, $r_{i,2}$ and $r_{i,3}$, respectively. Furthermore, we define a set of predefined priority classes of slices as $\mathcal{S}=\{S_1, \ldots, S_6\}$ (same as earlier defined in section 3.4.1) that correspond to strictly uRLLC, heterogeneous but uRLLC dominant, heterogeneous but eMBB dominant, strictly eMBB, heterogeneous but mMTC dominant and strictly mMTC use case slices, respectively. Moreover, in this model, we consider that the requests have waiting time constraints to ensure a more realistic scenario (unlike in the first solution where we do not) where a request is not indefinite and a request is considered to be admitted only if its requirements are fulfilled within the required time given. To formalize this, we denote by $\gamma_{i,t}$ the indicator of a request i indicating whether

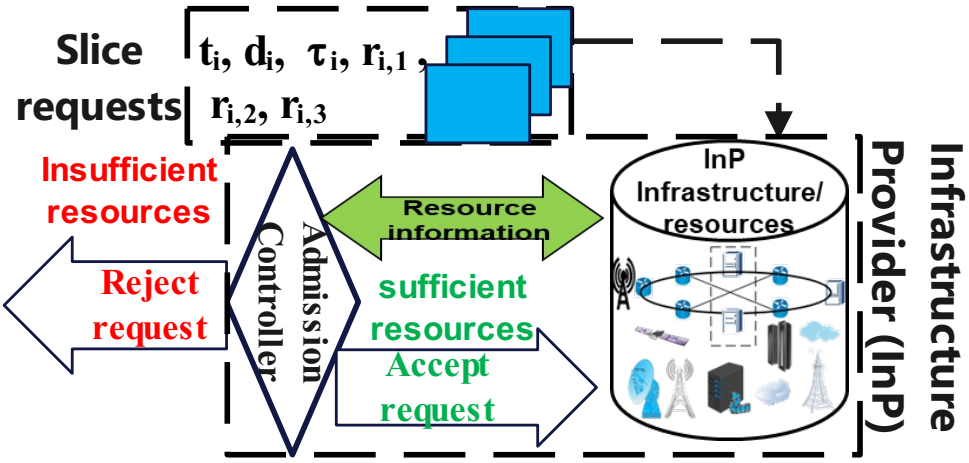


Figure 3.7: System Model (P2).

it is still valid at time t or not, which writes as:

$$\gamma_{i,t} = \begin{cases} 1, & \text{if } t \leq \tau_i; \\ 0, & \text{otherwise,} \end{cases} \tag{3.12}$$

where: $\tau_i = 1, ..., T$, is the maximum waiting time, which counts the number of slots relative to T_{start} until the request i becomes obsolete.

Furthermore, the proposed model considers an overbooking option, similar to the partial and full pooling employed in the former solution, which is activated when the InP has requests in queue but does not have sufficient resources to accept them. This process enables the InP to pay for the idle/unused resources of admitted slices and use these resources to admit as many of the waiting requests that can be admitted. Thus, allowing both parties to mutually benefit each other as the tenant reduces its cost by selling its idle resources while the InP is also able to satisfy more slice requests, increase its profit and improve resource utilization. We denote by $r^u_{i,j}$ the amount of resource j that slice i uses and the amount of resources that slice i sells back to the InP by $r'_{i,j}, \forall j \in \{1,2,3\}$ such that $r'_{i,j} = r_{i,j} - r^u_{i,j}$ and $r^u_{i,j} \leq r_{i,j}$. Note that in this solution, we do not include the willingness to sell probability as it does not provide any significant insights but adds to the complexity of the solution.

3.4.2 Problem Formulation (P2)

The AC problem is formulated as a multi-objective integer optimization problem that aims to maximize the profit defined by O_1 and resource utilization defined by O_2 of an InP with regard to the requests' requirements and the InP's available resource at each time slot, denoted by $R_{j,t}$. Let $x_{i,t}$ a binary variable that indicates that the request i is admitted in time slot t ($x_{i,t} = 1$) or not ($x_{i,t} = 0$). Similarly, we define a binary variable

to indicate the successful admission of request i ($y_i = 1$) or not ($y_i = 0$). Furthermore, we define by $p_{i,j}$ and $p'_{i,j}$ the selling and buyback prices per unit of resource $j \in \{1,2,3\}$ and request $i \in \mathcal{N}$, respectively. It is pertinent to note that we assume profitable-based prices and, as such, do not consider additional costs of service provisioning in our model. Formally, the AC problem aimed at maximizing InP profit and resource utilization is formulated as follows:

$$\textbf{[P2]} \qquad \begin{cases} O_1 : \max \sum_{i=1}^{N} \sum_{j=1}^{3} \left(y_i r_{i,j} p_{i,j} d_i - y_i r^{'}_{i,j} p^{'}_{i,j} \right), \\ O_2 : \max \sum_{i=1}^{N} \sum_{j=1}^{3} y_i r^{u}_{i,j}. \end{cases} \tag{3.13}$$

Subject to:

$$\sum_{i=1}^{N} x_{i,t} r_{i,j} \leq R_{j,t}, \qquad \forall j \in \{1,2,3\}, \forall t \in \mathcal{T}, \tag{3.14}$$

$$y_i \leq \sum_{t=1}^{T} x_{i,t} \gamma_{i,t}, \qquad \forall i \in \mathcal{N}, \tag{3.15}$$

$$\sum_{t=0}^{T} x_{i,t} \leq 1, \qquad \forall i \in \mathcal{N} \tag{3.16}$$

$$x_{i,t} \in \{0,1\}, y_i \in \{0,1\}, \qquad \forall i \in \mathcal{N}. \tag{3.17}$$

The objective in (3.13) aims at maximizing the profit, defined by the total revenue and the buyback cost, as well as the resource utilization. The constraints (3.14) are introduced to ensure that the total amount of resources assigned over the admitted slices, in each time slot t, cannot exceed the InP's total capacity. The constraints (3.15) ensure that a request is considered to be admitted only if its requirements are fulfilled within the required time given by γ, those in (3.16) guarantee that each request is admitted at most in one time slot while those in (3.17) indicate the binary decision variables of the problem.

3.4.3 Proposed Algorithm for P2

We attempt to solve the above problem and provide a solution that maximizes the total profit and resource utilization. The AC problem is variation of the budgeted maximum coverage problem, which is known to be NP-hard. Thus, the considered problem is also NP-hard. Therefore, we propose a greedy algorithm and the obtained solution follows the Algorithm 2. It is known that the greedy algorithm that always chooses the best price to cost ratio in each iteration is the best polynomial-time approximation algorithm. In our solution, the price is the revenue accrued by the request, while the cost is the total

resources required to satisfy that request. This is similar to the proposed solution RCRS in section 3.3.

For every time slot, we first initialize all the parameters, such as: the number of slice requests and their requirements, the InP resources as well as the unit prices of the various resources. Then, we introduce a metric, denoted α_i $\forall i \in \mathcal{N}$, to indicate whether admitting i at time slot t has a high performance-cost ratio (line 3). The algorithm sorts the slice requests according to α_i, $\forall i \in \mathcal{N}$ and creates the new ordered set $\mathcal{L}^s$ of sorted requests. For every slice request in the sorted list, started with the topmost request in the list, the algorithm accepts it if : *i)* the capacity constraint of the InP (4.42) is not violated and *ii)* the maximum waiting time is still not expired. If the request is admitted, the total number of available resources at the InP, R_j, $\forall j \in \{1,2,3\}$ and the accepted requests list we denote by $\mathcal{L}^a$, are updated (lines 7-8). Otherwise, the request is added to the waiting list, denoted by $\mathcal{L}^w$ and the needed resources R_j^N is updated.

Next, the algorithm determines the amount of shareable resources R_j^s from the accepted requests (line 14) and employs the incremental capacity pooling mechanism that ensures the InP buys back resources incrementally from existing tenants without having to buyback all resources at once except where absolutely necessary. The pooling mechanism relies on a sufficiency pre-check mechanism to determine whether to activate partial or full capacity pooling. The pre-check mechanism ensures that the InP only activates full capacity pooling, by buying back all the idle resources from already accepted slices R_j^s, if the amount of resources required by the waiting requests R_j^N is equal to or greater than the sum of resources remaining in the InP's resource bank and the resources that can be bought back R_j^s (line 30). After the buyback process, the algorithm goes through the waiting list and accepts the requests whose due dates are still valid, then updates the accepted requests list and the InP capacity. The valid requests that cannot be accepted due to insufficient resources even after the buyback process are rejected. Moreover, at each time slot, the accepted requests whose duration is expired release the allocated resources (lines 35-39).

3.4.3.1 Time Complexity Analysis for Algorithm 2

This subsection presents the time complexity analysis of our proposed solution. The complexity of the various steps of Algorithm 2 is as follows: the sorting procedure (line 2-4), admission without buyback (line 5-13), determination of idle resources (line 14), buyback and admission after buyback (line 15-39), release procedure (line 40-47) have complexity of $O(N + N\log(N))$, $O(N)$, $O(J)$, $O(J2N)$ and $O(N)$ respectively. In total, since each segment is executed once for each t in T, the time complexity is $O\left(T\left[N\log N + 3N + J(2N+1)\right]\right)$.

Algorithm 2: Proposed algorithm for P2.

Input: $R_j, r_{i,j}, p_{i,j}, p'_{i,j}$, γ_i, $R^s_j = 0$, $R^N_j = 0$, $\mathcal{N}$, $\mathcal{L}^a = \emptyset$, $\mathcal{L}^w = \emptyset$, $\mathcal{L}^s = \emptyset$
Output: y_i, $\forall i \in \mathcal{N}$

for $t = 1, \ldots, T$ **do**
  For each request whose $t_i \leq t$ determine α_i:
  $\alpha_i = \frac{\sum_{j=1}^{3} r_{i,j} p_{i,j} d_i}{\sum_{j=1}^{3} r_{i,j}}$
  Sort α_i descending, let $\mathcal{L}^s$ be the ordered set
  for $i \in \mathcal{L}^s$ **do**
    if $r_{i,j} \leq R_j$ ***and*** $\gamma_{i,t} = 1$, $\forall j \in \{1,2,3\}$ **then**
      $\mathcal{L}^a \leftarrow \mathcal{L}^a \cup \{i\}$ and $y_i = 1$
      $R_j \leftarrow R_j - r_{i,j}$, $\forall j \in \{1,2,3\}$
    else
      $\mathcal{L}^w \leftarrow \mathcal{L}^w \cup \{i\}$
      $R^N_j \leftarrow R^N_j + r_{i,j}$, $\forall j \in \{1,2,3\}$ `// update needed resource`
    end
  end
  Determine shareable resource: $R^s_j = \sum_{i \in \mathcal{L}^a} r'_{i,j}$, $\forall j \in \{1,2,3\}$
  `// where` $r'_{i,j}$ `is the idle resources of` i
  if $R^N_j < (R_j + R^s_j)$, $\forall j \in \{1,2,3\}$ **then**
    `// Activate partial pooling`
    Let $i = 1$ `//` i `indicates index in` $\mathcal{L}^a$
    while $(R_j + r'_{i,j}) \leq R^N_j$, $\forall j \in \{1,2,3\}$ **do**
      $R_j \leftarrow R_j + r'_{i,j}$, $\forall j \in \{1,2,3\}$
      $i \leftarrow i + 1$
    end
    for $i \in \mathcal{L}^w$ **do**
      if $\gamma_{i,t} = 1$ **then**
        $\mathcal{L}^a \leftarrow \mathcal{L}^a \cup \{i\}$ and $y_i = 1$
        $R_j \leftarrow R_j - r_{i,j}$, $\forall j \in \{1,2,3\}$
      else
        Reject request i
      end
    end
  else
    `// Activate full pooling`
    $R_j \leftarrow R_j + R^s_j$, $\forall j \in \{1,2,3\}$
    for $i \in \mathcal{L}^w$ **do**
      if $r_{i,j} \leq R_j$ ***and*** $\gamma_{i,t} = 1$, $\forall j \in \{1,2,3\}$ **then**
        $\mathcal{L}^a \leftarrow \mathcal{L}^a \cup \{i\}$ and $y_i = 1$
        $R_j \leftarrow R_j - r_{i,j}$, $\forall j \in \{1,2,3\}$
      else
        Reject request i
      end
    end
  end
  if $t > 1$ **then**
    for $i \in \mathcal{L}^a$ **do**
      if $t \geq t_i + d_i - 1$ **then**
        $R_j \leftarrow R_j + r_{i,j}$, $\forall j \in \{1,2,3\}$ `// release & add to pool`
        $\mathcal{L}^a \leftarrow \mathcal{L}^a \setminus \{i\}$
      end
    end
  end
end

Table 3.3: Simulation Parameters for P2.

Parameter	Value(s)
Duration d_i $\forall i \in \mathcal{N}$	random between 6 and 18 months
Arrival time t_i $\forall i \in \mathcal{N}$	random between 1 and 12 months
Total time slots T	12
T_{slot}	1
Waiting time τ_i	$t_i + \beta$, β random between 1 and 3
Number of requests N	$\{100, \ldots, 800\}$
InP's capacity	$R_1 = R_2 = R_3 = 3.10^3$
$r_{i,1}$, $r_{i,2}$,$r_{i,3}$ $\forall i \in \mathcal{N}$	random between 300 and 500 units
$r^u_{i,1}$, $r^u_{i,2}$,$r^u_{i,3}$ $\forall i \in \mathcal{N}$	random between 50 & $r_{i,1}$, $r_{i,2}$, $r_{i,3}$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class S_1	$p_{1,1} = 3, p_{1,2} = 4, p_{1,3} = 5$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class S_2	$p_{1,1} = 2.5, p_{1,2} = 3.5, p_{1,3} = 4.5$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class S_3	$p_{1,1} = 2, p_{1,2} = 3, p_{1,3} = 4$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class S_4	$p_{1,1} = 1.5, p_{1,2} = 2.5, p_{1,3} = 3.5$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class S_5	$p_{1,1} = 1, p_{1,2} = 2, p_{1,3} = 3$
$p_{i,j}$, $j \in \{1,2,3\}$, $i \in$ class S_6	$p_{1,1} = 1, p_{1,2} = 1.5, p_{1,3} = 2.5$

3.4.4 Performance Evaluation of the Proposed Solution for P2

In this section, we evaluate the performance of the proposed solution against two SoA algorithms [114] and [122]. We will first describe the simulation setup, in Section 3.4.5, then provide some results in Section 3.4.6.

3.4.5 Simulation Settings for P2

We consider a simulation scenario where the capacity of the InP is set to 30,000 units for each resource, namely computing (R_1), storage (R_2) and network (R_3) resources, with which it serves $N \in \{100, \ldots, 800\}$ generated requests. Each request belongs to only one class and the nominal priority of each class $S_1, \ldots, S_6$ corresponds to $6, 5, 4, 3, 2, 1$, respectively. The simulation settings are presented in Table 3.3.

Contrary to existing solutions that assume competitive bidding-based buyback prices [123] or higher than initial price buyback prices [114], [122], we adopt the dynamic buyback price proposed in our earlier work in section 3.3. This buyback price corresponds to the price paid by the InP to the tenants for units of resources bought back. Our dynamic buyback price is defined as :

$$p'_{i,j} = \frac{d^r_i}{d_i} \times p_{i,j}, \tag{3.18}$$

where: d^r_i, d_i and $p_{i,j}$ are the remaining validity duration, the initial duration and unit price paid by the tenant for that resource, respectively. Note that $0 < p^{'}_{i,j} < p_{i,j}$ and $d^r_i < d_i$.

3.4.6 Comparison of the Proposed Solution for P2 to SoA

In this subsection, we assess the performance of the proposed algorithm (PA) against two state-of-the-art schemes SoA1 and SoA2 considered in [114] and [122], respectively. To ensure a fair comparison, we augment them with the proposed buyback price and use the same simulation parameters for all the schemes (Table 3.3), namely the proposed and SoA schemes. SoA1 prioritizes requests based on their class priority $S_i, \forall i \in \{1,\ldots,6\}$ in decreasing order, while SoA2 prioritizes requests based on their revenue prospects. Furthermore, SoA1 and SoA2 buy back all the idle resources from already admitted slices of lower priority. Moreover, we consider two additional variants of the proposed algorithm, which we refer to by the acronyms PA-V1 and PA-V2. The PA-V1 variant does not consider the waiting time (provided by the constraints (4.43)), while PA-V2 does not include the buyback process. Each value in the presented results is the mean value taken for the entire slots for 10,000 simulation runs.

Figure 3.8, shows how the profit of the InP scales with the number of received requests. It shows that the proposed scheme with waiting time and buyback process, PA, outperforms all of its other variants, namely PA-V1 and PA-V2 and SoA schemes. This performance can be linked to the priority metric employed by the PA. That is, while SoA1 and SoA2 admit requests that pay higher based on their revenue prospects and class priority, eventually yielding higher short-term profit, the proposed schemes yield more profit in the long term. It is also worthy to note that the PA yields higher profit than its variants PA-V1 and PA-V2. This is because while PA-V1 outright rejects requests when there are insufficient resources after the buyback process, PA allows these requests to seek admission in ensuing time slots until their maximum wait time is exceeded. The results further reveal that the scheme without any buyback process, PA-V2, yields the least profit.

Figure 3.9 depicts the relative resource utilization. Note that, the values are normalized to fall within $[0,100]$, as the original values for both the proposed and compared schemes were over 100% due to the release process that allows a reuse of resources relinquished by slice tenants whose duration has elapsed and the unused resources the InP buys back. The resource utilization $U(T)$ is determined over the entire simulation including all the time slots as:

$$U(T) = \sum_{t=1}^{T} \frac{\left(\frac{\sum_{i=1}^{N} \sum_{j=1}^{3} x_{i,t} r_{i,j}}{R_j} \right)}{3} \times 100. \tag{3.19}$$

From Figure 3.9, we observe that the proposed scheme without buyback process, PA-V2, has the least resource utilization rate because it only accepts requests with the InP's nominal resources while the others employ the overbooking concept. On the other hand, the proposed scheme with buyback process and waiting time, PA outperforms all the

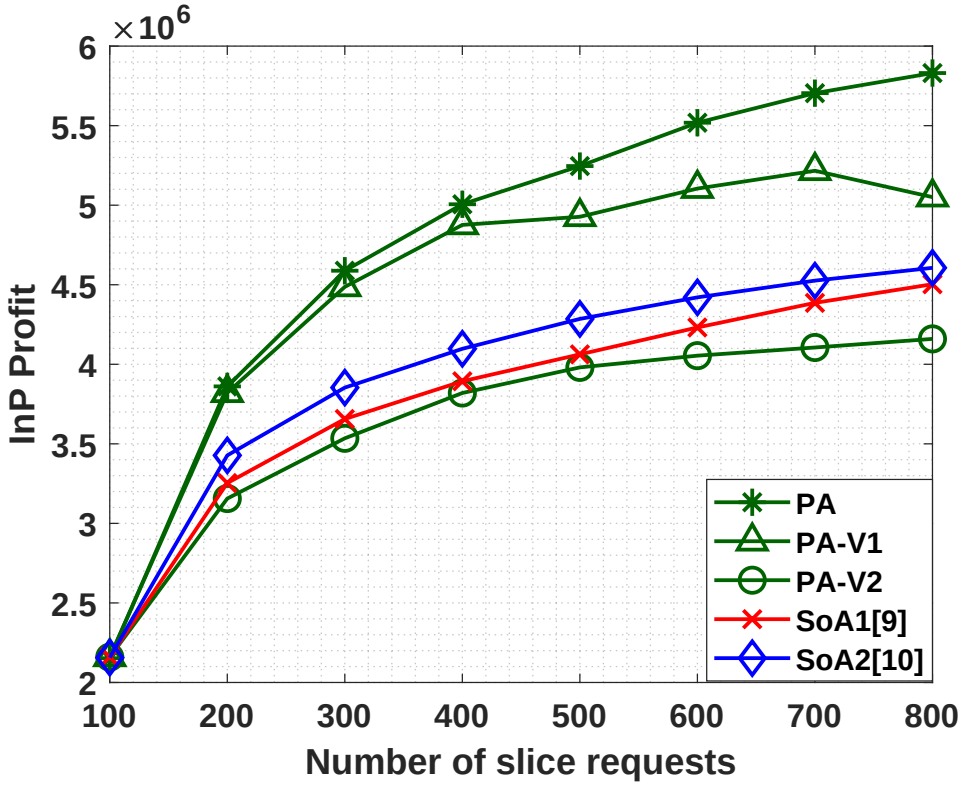


Figure 3.8: Comparison of the proposed solution for P2 against SoA algorithms in terms of profit.

Table 3.4: Acceptance rate for $N \in \{100, \ldots, 800\}$.

Scheme \ N	100	200	300	400	500	600	700	800
PA(%)	100	82	57	45	35	30	26	23
PA-V1(%)	100	81	55	42	32	27	23	19
PA-V2(%)	99	60	41	32	26	21	18	16
SoA1(%)	100	68	46	35	29	24	20	18
SoA2(%)	100	70	50	39	32	27	24	21

schemes including its variant without waiting time, PA-1, which hints that the buyback process helps to improve the performance.

We present more results in table 3.4 and 3.5 to further demonstrate the performance of our proposed solution in terms of total and per slice types acceptance rates. The results in Table 3.4 show that the proposed scheme with waiting time and buyback process, PA, has a higher acceptance rate compared to the other schemes. This superior performance resulted from the metric α we propose, which enables the algorithm to efficiently balance the profit and resource demands of a request, as well as the renege factor that allows to move the admission decision of requests to other time slots until their maximum wait time is exceeded, thus increasing the number of requests that are accepted.

Moreover, we compare the acceptance rate per slice type of the different schemes by

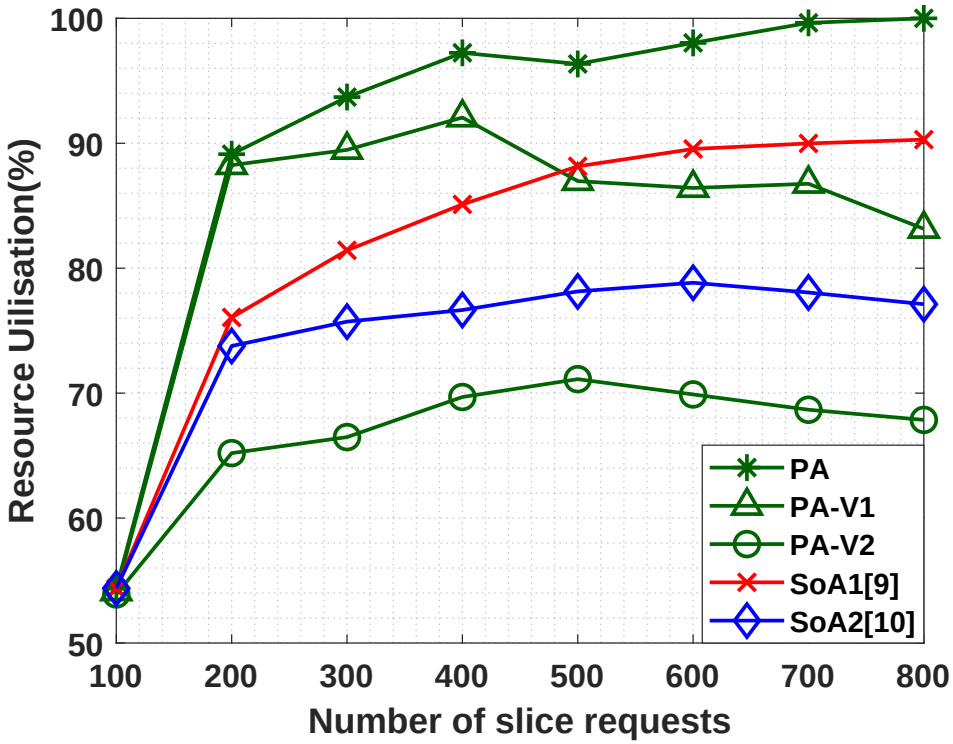


Figure 3.9: Comparison of the proposed solution for P2 against SoA algorithms in terms of resource utilization rate.

Table 3.5: The acceptance rate of requests by slice type.

Schemes \ Slice Type	S_1	S_2	S_3	S_4	S_5	S_6
PA(%)	64	58	52	46	42	39
PA-V1(%)	61	56	49	43	39	36
PA-V2(%)	52	46	40	36	32	30
SoA1(%)	81	53	44	38	32	29
SoA2(%)	55	49	43	39	35	33

presenting the average of the results for $N \in \{100, \ldots, 800\}$. The results are shown in Table 3.5, we observe that while SoA1 scheme admits about 23.4% more highest priority requests, the proposed scheme admits about 9%, 16.7%, 19%, 27% and 29.4% more of the S_2, S_3, S_4, S_5 and S_6 class of requests respectively. This is due to the robust metric employed by the proposed scheme PA that considers both the profit prospects and resource required by requests, making it to increase the likelihood of admitting low priority requests with profit to resource balance as opposed to SoA1 scheme, which only considers the class priority of requests. This consequently shows that in addition to performance improvement, the proposed scheme guarantees good fairness among the requests as well.

Further simulations have been performed to assess the impact of the price and buyback

cost. The new prices are defined as follows:

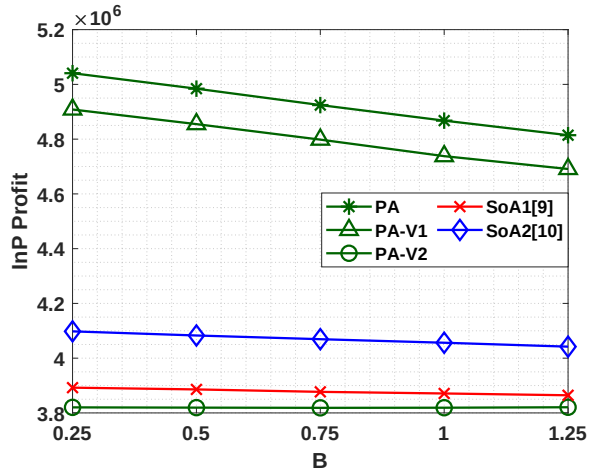
$$p^{k}_{i,j} = K \cdot p_{i,j}, \quad \text{where} K \in \mathbb{N}, \tag{3.20}$$

$$p'_{i,j} = B \cdot p_{i,j}, \quad \text{where} B \in \mathbb{R}. \tag{3.21}$$

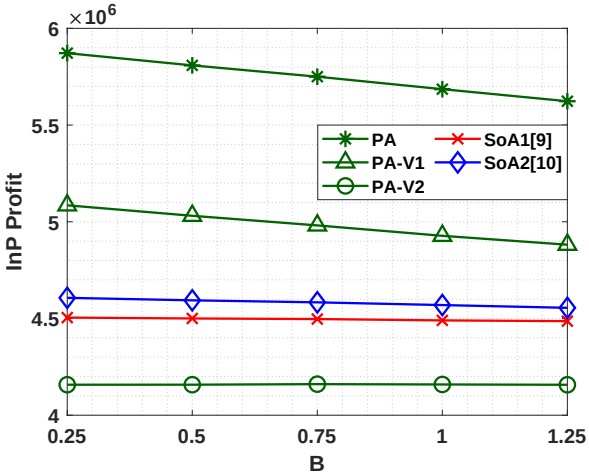
where $p_{i,j}$ and $p'_{i,j}$ denote the original price and new buyback costs respectively while $p^{k}_{i,j}$ is the adjusted price with the superscript k used solely to distinguish it from the original value and does not represent a variable or index.

Figure 3.10a and 3.10b show how the profit scales with different buyback costs, we use the default simulation settings and present for $N = 400$ and $N = 800$. The results indicates that for both buyback costs lower and higher than the initial selling price, the proposed scheme outperforms the other schemes. Interestingly, a further analysis of the results reveal that the average percentage difference between PA and others, namely PA-1, PA-2, SoA1 and SoA2, when we consider 400 requests is 2%, 15%, 11% and 14%, respectively, while for 800 requests, this difference is 9%, 20%, 14% and 16%, respectively. Thus, we can deduce that the proposed scheme scales well with increase in slice requests and is thus suitable for deployment in even changing network conditions.

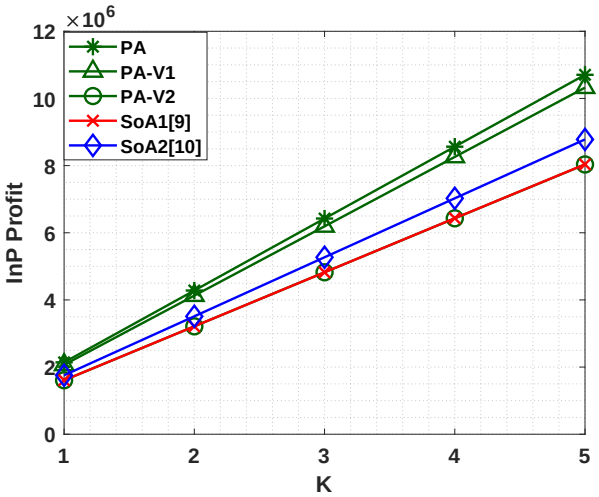
Similarly, Figure 3.10c and 3.10d show the profit for different prices determined based on K. We can clearly see that the proposed scheme always outperforms the compared schemes and provides a significant improvement around 2%-20%.



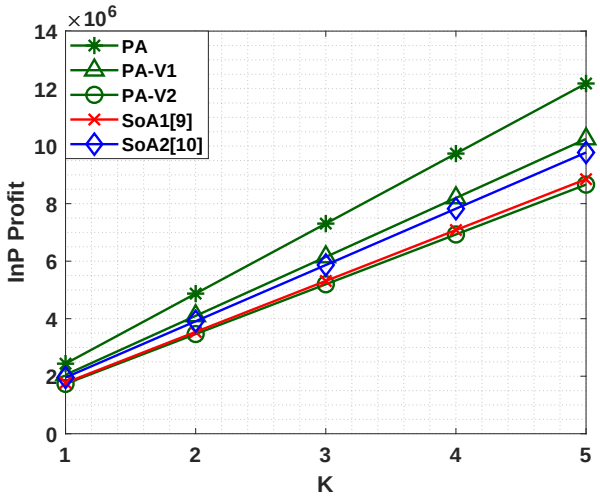
(a) Profit when $N = 400$ (for $p'_i, j = B \times p_i, j$)



(b) Profit when $N = 800$ (for $p'_i, j = B \times p_i, j$)



(c) Profit when $N = 400$ (for $p^k_{i,j} = K \times p_{i,j}$)



(d) Profit when $N = 800$ (for $p^k_{i,j} = K \times p_{i,j}$)

Figure 3.10: Comparison of the proposed solution for P2 against SoA algorithms in terms of profit, for different prices.

3.5 Chapter Summary

This chapter presents two slice admission control solutions. The first solution is a static version focusing on revenue and cost, while the second is a dynamic version optimizing for long-term profit and utilization under temporal constraints. First, we defined two different system models and derived two distinct problem formulations. To solve these NP-hard problems, we proposed novel greedy heuristic algorithms. The static algorithm leverages intelligent request prioritization and a sufficiency-check-driven buyback process. The dynamic algorithm extends this by operating over a timeline, managing request validity and releasing expired resources.

Simulation results demonstrate that both proposed algorithms significantly outperform

existing SoA solutions across key metrics: revenue/profit, buyback cost, resource utilization and acceptance rate. The dynamic algorithm also shows improved fairness across slice classes. This work provides InPs with practical and efficient strategies to maximize their operational efficiency and profitability in dynamic network slicing environments.

4

Delay-Guaranteed and Energy Efficient User Association Functions placement in O-RAN

Contents

4.1 Introduction 59

4.2 Review of Related Works 60

4.3 Energy Efficient and Delay-Guaranteed User Association for O-RAN 67

4.3.1 System Model (P3) 67

4.3.2 Access Network Delay Model 68

4.3.3 RU Power Consumption Model 68

4.3.4 Problem Formulation (P3) 69

4.3.5 Proposed Algorithm for P3 70

4.3.6 Performance Evaluation of the Proposed Solution for P3 72

4.4 End to End Delay-Guaranteed and Energy Efficient User Association with Functions Placement for O-RAN 77

4.4.1 System model (P4) 77

4.4.2 End to End Delay Model 78

4.4.3 Energy Consumption Model (P4) 81

4.4.4 Problem Formulation (P4) 82

4.4.5 Proposed Algorithm for P4 84

4.4.6 Performance Evaluation of the Solution Proposed for P4 86

4.5 Chapter 4 Summary 97

4.1 Introduction

5G and beyond of mobile networks support diverse services with diverse quality of service (QoS) and experience (QoE) requirements [124]. The need to meet these Qos and QoE requirements has necessitated the shift from the traditional radio access networks (RANs) to open RAN (O-RAN) which enables the disaggregation of RAN functionalities [125]. That is, the RAN functions are split into the following components: open-central unit (O-CU), open-distributed unit (O-DU) and open-radio unit (O-RU). Thus enabling the independent and simple implementation of the individual components and promoting multi-vendor implementation [12].

While the disaggregation of the RAN presents enormous benefits, it also presents some unique challenges as the interaction between the various components (UE to RU association, as well as placement of virtual DU/CU functions and their interaction with the RU) will need to be optimised, individually or jointly, in order to maximise these benefits. The key optimisation objectives revolve around minimizing delay, cost and energy or maximising reliability [126]. To this end, several works have considered energy consumption in user association and placement problem. However, the focus is mostly on energy consumption of DUs and CUs, with little or no consideration for energy consumption of the RUs which are also an integral part of the O-RAN architecture and account for about 88% of the energy consumption of the RAN [127].

Similarly, the user to RU association problem is often viewed as a delay minimization or allocated resource blocks (RBs) maximisation problem with attention seldom given to the energy consumption of the RUs despite the huge impact they have on overall RAN energy consumption [128]. It is pertinent to note that while modern RU, DU and CU models are equipped with advance sleep mode control capabilities that enable them to enter the sleep mode when there are no associated loads, the choice of a robust association schemes is critical to harnessing these sleep mode benefits which to the best of our knowledge has not been explored by current solutions. Therefore, it is necessary that efficient UE association and placement schemes be defined to enable reduction in total system energy consumption with a view to not only reducing operational cost (OPEX) but also contributing in the fight to save our planet.

This chapter addresses the critical problems of (UE) association and function placement in O-RAN. We present two solutions that tackle these problems at two levels of complexity:

1. UE-to-RU association: an approach is proposed that focuses on the access network, determining the optimal RU for each UE to minimize access network delay and system energy consumption by putting idle RUs to sleep. This solution is presented

in section 4.3

2. End-to-End association and placement: an extension is proposed to the problem to encompass the full O-RAN chain. The proposed novel approach involves jointly associating UEs to RUs, RUs to DUs and DUs to CUs, with the objective of minimizing total end-to-end delay and the combined energy consumption of RUs, DUs and CUs. This solution is presented in section 4.4

4.2 Review of Related Works

To tackle the challenge of CU and DU network functions placement in the regional and edge O-Cloud nodes as well as user association to RUs, a deep reinforcement learning (DRL)-based RU-UE association and CU-DU placement (DJRCD) solution is proposed in [129] to minimize the delay of UEs while reducing the deployment cost of DU and CU VNFs (the DU and CU are co-located) under node processing capacity and link bandwidth constraints. The DJRCD solution explores the best O-RAN cloud (O-Cloud) locations for DU and CU virtualized/cloud-native network functions (VNFs/CNFs), as well as the optimal UE to RU associations. Each node has limited computing capacity while each user generates packets of predefined lengths (in bits) and delay requirements. First, the problem is formulated as a multi-objective optimization problem to minimize the average end-to-end delay of the UE as well as the average cost of DU and CU deployment, that is computing and initiating costs under processing nodes computing capacity, fronthaul and midhaul links bandwidth capacity and UE delay constraints. The problem is decomposed into two sub-problems: finding the RB allocation values and user association and CU-DU placement with known resource allocation values. The resource allocation values are calculated using round robin for one millisecond time interval while the user association and CU-DU placement is also for 1ms time interval. Next, a Markov decision problem is developed and a RL-based deep Q-network (DQN) algorithm employed to solve the problem. In this work, the action specifies in each time step, for each RU, the nodes where the DU and CU functions of the RU are placed as well as which RU will serve each UE. The proposed novel solution finds the optimal O-Cloud nodes to run the VNFs/CNFs of DU and CU for each RU thus, reducing deployment cost and average user delay while assigning the best RU for the UEs. However, it does not consider the diverse service requirements of different users.

The work in [130] tackles the placement of the RAN slices while adopting flexible functional split options to maximize the profit of the infrastructure provider. They consider a RAN slice to be a set of virtual RAN functions O-RAN RU, O-RAN DU, O-RAN CU (O-RU, O-DU, O-CU) with three requirements (latency, midhaul and fronthaul capacity)

to be fulfilled if placement is made. The placement problem is formulated to maximize the revenue of the provider and the objective function is defined as a product of the admission decision variable, revenue for admitting the request and the centralization factor under bandwidth capacity and latency constraints. The centralization factor is such that the higher the functional split, the higher the centralization factor. Furthermore, it considers a network slice as a set of chained virtualized baseband functions (O-RU, O-DU and O-CU). It seeks to locate functions as central as possible with a view to lowering OPEX. However, it increases delay for low latency applications.

In [131], an optimization model is proposed that deploys O-RAN components within regional and edge clouds to maximize availability and consequently minimize the network outage (deployed O-RAN VNFs downtime) under delay constraints. These constraints ensure that VNF components are only placed on servers that can guarantee their delay threshold. In this model, CU and DU are placed at an edge server while the near real-time RAN intelligent controller (Near-RT RIC) is located on the regional cloud. It minimizes the probability of a failure by deploying VNFs and their redundant instances on servers with high mean time to failure (MTTF), that is, the lifespan of a node before it stops operating and low mean time to repair (MTTR) that is, the average time needed to repair a node upon its failure values. By prioritizing servers with MTTF, it favours the placement of VNFs on regional servers that usually have higher MTTF values thus, increasing the reliability but consequently increasing the latency. It also reduces the chances of recently added servers been used as they have zero or low MTTF.

The work of [132] proposes a framework to tackle the user to RU association and DU placement problem with a view to minimizing the overall deployment cost. A 2-phase integer programming and a time-efficient heuristic based on long-term network statistics are employed to optimise the number of instantiated RUs in a given area by using the statistics to associate RUs to open-access edge servers for hosting the corresponding DUs and CUs thus, installing the minimum number of RUs and open-access edge servers. That is, RUs in areas with limited past history are shutdown. The proposed framework lowers cost and latency. However, the deployment of RUs based on past statistics may lower potential for new users as users will only request services that are available. This can in turn lower the operator's revenue.

To solve the problem of dynamic placement of O-RAN components (allocation of O-CU and O-DU functions on either edge or regional clouds), [133] proposed an integer linear programming (ILP) model for VNF placement to maximize user admittance ratio. The scheme considers mid-haul delay and server capacity requirements as well as employs a recurrent neural network (RNN) heuristic while considering various constraints such as: capacity of cloud servers, link capacity, delay budget and diverse service requirements. It also dedicates 50%, 25% and 25% of the total RBs to eMBB, uRLLC and mMTC

users respectively. In addition, the scheme associates UEs to the nearest RU, increases admittance ratio, minimizes the deployment costs and increases throughput. However, its dedicated reservation of resources for the various class of users can lead to reduced resource utilisation and overall system acceptance rate if the demand of any of the classes is less than predicted.

A framework for elastic VNF orchestration for O-RAN is proposed in [134] to solve the challenge of intelligent/proactive management of O-RAN resources, in order to minimize delay. The framework considers an architecture where the near-RT RIC is hosted in the regional cloud while the O-CU and O-DU are hosted on the edge cloud. A machine learning (ML) based traffic forecasting-based dynamic scaling scheme that forecasts requests and uses the forecast to scale allocations is also employed. In every given time frame, it predicts a scaling decision (that is, the number of VNF instances that have to be provisioned or de-provisioned to handle the incoming traffic in that time-frame). Furthermore, a RL based VNF placement policy is employed to dynamically determine VNF placement based on the forecast. The traffic forecasting module predicts the incoming aggregated traffic using a ML model, after which the scaling module uses the prediction to scale by determining the number of VNF instances needed to handle incoming traffic. Next, the dynamic placement module employs a RL agent to execute the scaling decision and redeploys the VNF instances accordingly. This solution for the scaling decision employs an integrated ILP solver to generate the required dataset for training. The solver takes each request in the set of service function chain (SFC) requests and produces the optimal number and placement of VNFs required after each arrival by minimizing a cost objective function. This objective function comprises the deployment cost, resource uptime cost, forwarding cost and resource fragmentation cost. The solver then outputs a set of active VNFs in the pseudo-topology used in the ILP. The scheme increases resource utilization, consequently lowering OPEX. However, by hosting both the CU and DU on the edge, it significantly increases the cost for users and may increase energy consumption as well. Moreover, it does no consider the user association problem which is crucial in optimizing network performance.

The work in [135] proposed a two-timescale framework to solve the resource allocation problem aimed at optimizing users' long-term total QoS. The proposed solution uses congestion indicators to proactively modify user demands before allocation rules are enforced and employs an online learning algorithm for choosing the optimal control signals (OLAS) that uses bandit-feedback version of the combinatorial multi-armed bandit framework to deduce resource-specific signals. Furthermore, a gradient-based algorithm is employed to shrink the confidence ellipse (GASH) that gleans side information from live network traffic to refine predictions on user resource sensitivities. The solution improves application utility and throughput as well as reduce latency. However, the constant resource

monitoring and modification process it employs increases energy consumption.

In [136], an efficient ILP model to minimize DUs placement cost under delay and capacity constraints is proposed. The models analyzes the throughput between the O-RAN interfaces (compute the flow on the CU–DU interface and on the DU–RU interface) before using the ILP for placement. The model also comprises of CU co-located with the network core and several RUs at the edge, close to the users. This solution considers each demand to be a traffic from the CU to an RU that requires a certain amount of bandwidth while the DUs are implemented in VMs such that the cost of using a VM depends on the hardware with the cost of CUs being lower than that of RUs since CUs are centrally situated and use high-end servers. However, placement decisions are made based on only the cost without recourse to resource utilization and energy consumption. The consideration for only static load conditions can lead to poor performance when the load exceeds certain thresholds.

To overcome the challenge of scalability and portability in traffic steering (TS) solutions, a two-tiered ML based traffic steering solution is proposed in [137] to avoid and reduce queuing delay in a uRLLC-specific ORAN deployment scenario. Naive bayes classifier (NBC) is employed to predict congestion on each O-RAN VNF (by using the average number of packets processed and the average processing latency per-minute as an indicator for congestion) and uses the predictions as inputs to a RL (deep Q-learning) agent that issue TS decisions on any SFC projected to face congestion. That is, the RL agent makes sure that congested VNFs are avoided thus, optimizing new VNFs for latency and preserving reliability. This solution can be deployed as an rApp or xApp (third-party application hosted on the non-RT RIC or near-RT RIC respectively). This solution yields lower queuing delay. However, by basing the prediction of congestion on just the average number of packets processed, it reduces resource utilisation.

To minimize latency, a latency-aware and proactive service placement scheme is proposed in [138] to solve the service placement problem in edge computing such that, the decision service of IoT applications is strategically placed on edge nodes through a proactive (re) placement approach. This scheme employs a QoS metric prediction model that leverages exponential smoothing methods to identify suitable edge nodes (based on the predicted latency) for hosting the decision module of IoT applications. Moreover, a proactive service placement capability that uses the prediction model to detect QoS metric degradation is employed. That is, edge nodes with the lowest predicted latency are selected for service placement but services are relocated to a more appropriate node if the predicted latency show a degradation. While this scheme ensures QoS by minimizing delay, it increases energy consumption.

A DRL-based DU placement with carrier aggregation (DUPCA) scheme is proposed

in [139] for the placement of the DU of each component carrier (CC) (PCell or SCell) to minimize the number of used O-Cloud nodes and user end to end delay. The DUPCA selects the primary cell (Pcell) among available cells for a user in round robin fashion. A weighted sum scheme is employed to solve this multi-objective problem, where the weighted sum refers to the sum of the weight of delay to transmit one packet of a UE and the weight of the number of placed DUs at the current time. The scheme reduces the end to end delay and the number of employed O-Cloud nodes. However, it does not consider the overall energy consumption of the system.

The cost-aware move and swap cluster's assignment - repeated algorithm is proposed in [140] to solve the processing pool (PP) cost-aware DU placement and flow routing (PPC-DUP-FR) problem with a view to minimizing the overall cost of PPs activation and processing under processing and transmission resources as well as latency constraints. The scheme employs a mixed-integer programming (MIP) and heuristic approaches to formulate and solve the PPC-DUP-FR optimization problem and provide optimized solutions to larger PPC-DUP-FR problem instances respectively. The objective is defined to minimize activation and processing costs subject to DU processing, flow routing, resource capacity and flow latency constraints. It minimizes cost but increases delay of flows.

To solve the DU placement problem, an ILP model is proposed in [141] to minimize computing and routing cost under delay and capacity constraints. This solution considers dynamic load conditions during the day and defines an objective that seeks to minimize the processing and routing costs, both of which depend on the load of the network subject to computation capacity, link capacity and delay constraints. The computation cost comprises of upkeep and processing cost where the upkeep cost is a fixed cost paid each time a node is active while the processing cost depends on the load. Similarly, the routing cost depends on the load and the distance between the nodes with CUs placed on the core network and DUs anywhere in the network. The scheme reduces operating cost at off-peak periods. However, it increases the end to end delay.

In [142] a soft actor–critic energy-aware dynamic DU selection algorithm (SA2C-EADDUS) is proposed to minimize delay and energy consumption. The SA2C-EADDUS integrates two nested actor–critic agents in the O-RAN architecture and defines an objective that seeks to minimize energy consumption and mean delay under transport network latency and DU computation capacity constraints. The first SA2C dynamically transfers network functions (NFs) to a DU with higher processing power to save energy while the second SA2C employs a scheduler that allocates RB by considering user traffic type and delay budget. This solution optimally selects the number of DUs and switches off redundant DUs in order to minimize energy consumption. The scheme reduces energy consumption and delay.

In [143], NetMind is proposed to solve the placement problem and minimize energy consumption. An objective function is defined that seeks to maximize the ratio of minimal to actual power consumption subject to the request's delay (front, mid and end-to-end), node's computing resource and link's bandwidth capacity constraints. That is, minimizing the power consumption across multiple subnetworks by optimizing policies of both the baseband function placement and routing provisioning. NetMind first transforms the problem into a maze-solving scenario based on MDP which it resolves by employing a DQN-based approach with an objective function-based reward evaluator. To avoid multiple DRL training sessions and different neural networks due to the different subnetworks with diverse configurations, while also employing a novel feature extraction scheme that uses a graph convolutional network based (GCN-based) encoder and decoder to normalize network features and consequently create an adaptive DRL model for different networks. The DQN processes the diverse mazes by converting their different state information, into a uniform length feature that is used to train one general DRL model fit to different scenarios. The scheme improves power savings and increases service stability.

In order to tackle the placement and resource allocation problems, [144] proposed two nested actor-critic learning based techniques to reduce latency. That is, the first is used to assign RBs to UEs based on the UEs requirements (like traffic types, delay budget and priority) while the second employed to optimize function placement or resource allocation decisions (based on available processing power and end-to-end delay). The scheme provides latency and throughput gains. However, while it considers traffic requirements, it does not consider propagation delay between O-RAN components, energy consumption or DUs with different processing capacities.

In [145], a flexible user association and placement scheme is proposed to maximize UE admission rate. A joint association-placement optimization problem is formulated, subject to the available resources and QoS constraints. The problem is further framed into an ILP model where an approximation algorithm is employed. The optimization model determines the admission decision at every transmission time interval (TTI) considering CU-DU placement (define end-to-end, user-centric traffic forwarding in the network) and resource allocation (assigning sufficient RBs to different users). First, UEs request communication resources in the form of a slice with a slice having achieved data rate (throughput) and end-to-end (E2E) delay requirements. A UE's slice request is accepted and the UE allocated resources if there are sufficient resources to meet its request. Each admitted UE is associated to one RU. It increases admission rates, minimizes the deployment costs and increases fairness.

A mixed-integer linear program (MILP) and multi-agent deep deterministic policy gradient (MADDPG) scheme is proposed in [146] to minimize energy consumption under latency and resource constraints. In contrast with existing solutions, this scheme considers

user plane functions (UPFs) and activation time of multi-access edge computing (MEC) servers in addition to DU and CU functions considered by others. It categories MECs into three: the first category only provide DU service, the second provides DU, CU and UPF services, while the third supports the same services like the second but has more computing resources. To activate servers, it assigns MECs to multiple DRL agents and employs a heuristic for function placement and routing provisioning. First, the MILP solution is formulated to jointly optimize the MEC activation, DU/CU/UPF deployment and routing provisioning by minimizing the number of activated MECs and the switching times of MECs subject to traffic demand, resource limitations and latency requirement constraints. However, the MILP suffers from very high complexity. Thus, the MADDPG which employs a DRL algorithm in conjunction with a restricted function deployment heuristic (RFDH) to reduce the action space of intelligent agents. To select the server combination with the lowest energy consumption for baseband placement, the DRL agents collaborate while making individual decisions on activating MECs with knowledge of the choices of other agents after which the collective decisions of all agents are then fed as inputs to the RFDH, for further computation and bandwidth resource management. The RFDH determines the baseband function deployment and routing paths after which it calculates the corresponding reward and feed it back to the DRL agent.

To jointly tackle the DU and RB selection problem, an energy-aware optimization model is proposed in [147] to enhance energy efficiency and guarantees user delay. The model jointly performs RB allocation and DU selection. The objective function is defined as a summation of two main energy consumption factors in the DUs; fixed energy consumption, which can be reduced by switching off the DUs that do not serve any RUs in the network and dynamic energy consumption which depends on the DU selection and RB selection decisions under latency and user demand constraints. Where the delay is incurred from scheduling delay (the time difference between when a request arrives from a user traffic until the assignment of the last RB to it. This delay emanates from the RB selection problem) and propagation delay between RUs and DUs. That is, the scheduling and propagation delay must not exceed the user's delay which depend on the type of traffic. Futhermore, functions of traffics with higher delay budgets are migrated from edge to the DUs at the centre to reduce energy consumption. The scheme improves energy efficiency. However, given the delay constraint imposed, it is not clear what happens to functions whose delay requirements cannot be met upon migration. More so, the scheme is unfair to DUs with high delay budget and may increase the overall system delay.

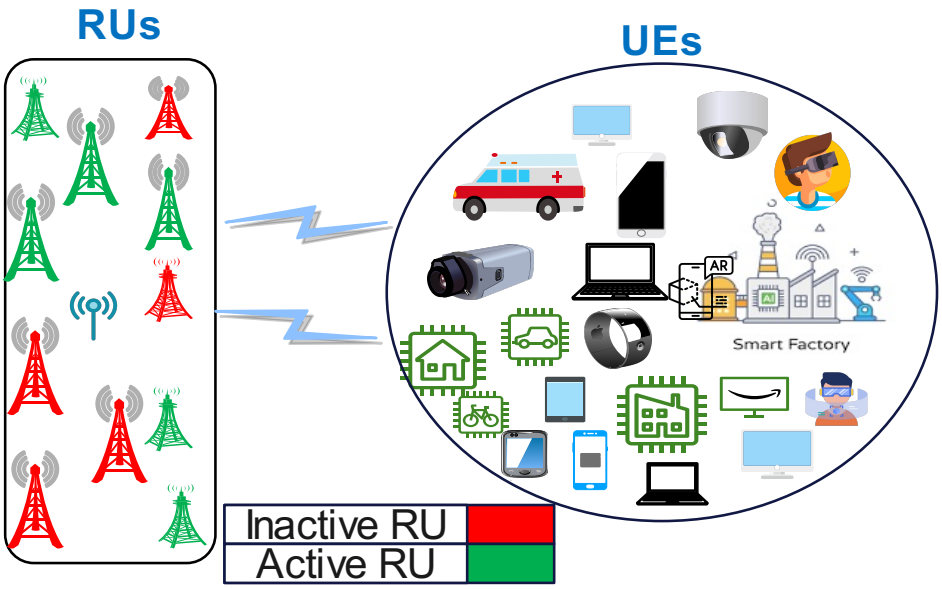


Figure 4.1: System Model (P3).

4.3 Energy Efficient and Delay-Guaranteed User Association for O-RAN

In this section, we present our solution for the UE-to-RU association problem. This solution is aimed at determining the optimal RU for each UE with a view to minimizing the access network delay and system energy. The proposed solution employs an optimal UE association metric that attempts to balance the conflicting objectives of delay and energy consumption. Furthermore, our proposed solution exploits the sleep mode capabilities of RUs to minimize power consumption by putting idle RUs to sleep.

4.3.1 System Model (P3)

The envisaged scenario consists of a set of O-RUs $\mathcal{R} = \{1, 2, \ldots, R\}$ and a set of UEs $\mathcal{U} = \{1, 2, \ldots, N\}$ of diverse types distributed randomly within a predefined area covered by the RUs as depicted in Figure 4.1. Each O-RU $j \in \mathcal{R}$ is equipped with a certain number of resource blocks (RBs) of the same size, $\mathcal{B}_j = \{1, 2, \ldots, B_j\}$ and can serve multiple UEs. However, each RB $k \in \mathcal{B}_j$ can be allocated to only one UE at a time and each UE $i \in \mathcal{U}$ can only be associated to one RU $j \in \mathcal{R}$. On the other hand, each user $i \in \mathcal{U}$ can either be of type uRLLC, eMBB or mMTC and has delay and data rate requirements, we denote in the sequel by d_i and r_i, respectively. Moreover, we denote by b_i the number of required resource blocks to meet the diverse QoS requirements of user i, namely r_i and d_i. An RU can be either active or inactive as depicted in Figure 4.1. An RU is said to be active if it has associated UE load and is not in sleep mode otherwise, it is said to be inactive.

4.3.2 Access Network Delay Model

Following the system model presented above, the delay in the access network between a UE i and a RU j, denoted by $d_{i,j}^{\text{acc}}$, can be expressed as a sum of the delays in the propagation, transmission and processing of the access network of the UE, we denote by $d_{i,j}^{\text{prop}}$, $d_{i,j}^{\text{trans}}$ and $d_{i,j}^{\text{proc}}$, respectively.

First, the access network propagation delay, defined as the time taken for a signal to travel from an RU to a UE associated to it, is expressed as a ratio of the distance between the UE and its associated RU to the propagation speed of the medium as defined by (4.1):

$$d_{i,j}^{\text{prop}} = \frac{\text{dist}_{i,j}}{s}, \tag{4.1}$$

where: $\text{dist}_{i,j}$ corresponds to the distance between the user i and RU j and s is the propagation speed of the medium (which is air in this instance).

Similarly, the access network transmission delay of user i with packet size α_i associated to RU j is defined by (4.2)[148]:

$$d_{i,j}^{\text{trans}} = \frac{\alpha_i}{r_{i,j}}, \tag{4.2}$$

where: $r_{i,j}$ and α_i are the data rate provided by RU j to UE i and mean packet size of the user, respectively.

Furthermore, the processing delay, $d_{i,j}^{\text{proc}}$, defined as the time taken to process a user's packet header and make forwarding decisions at the RU, is determined as defined by (4.3) [148]:

$$d_{i,j}^{\text{proc}} = \frac{\alpha_i \times c_j}{C_j}, \tag{4.3}$$

where: c_j, C_j and α_i are the number of CPU cycles required by RU j to process one bit per second of data, the total computing capacity of RU j (in CPU cycles per second) and the packet size of UE i, respectively.

Thus, access network delay between the user i and RU j, $d_{i,j}^{\text{acc}}$, can be expressed as:

$$d_{i,j}^{\text{acc}} = d_{i,j}^{\text{prop}} + d_{i,j}^{\text{tran}} + d_{i,j}^{\text{proc}}. \tag{4.4}$$

4.3.3 RU Power Consumption Model

In this section, we will introduce the power consumption model of RUs as in [147]. We define the power consumption P_j^{RU} of each RU $j \in \mathcal{R}$ as:

$$P_j^{\text{RU}} = P_j^{\text{RF}} + P^{\text{amp}} + P^{\text{DC}} + P^{\text{AC}}, \tag{4.5}$$

where: P_j^{RF}, P^{amp}, P^{DC} and P^{AC} are the power consumption of the RU's radio frequency (RF) unit, the power amplifier, the DC-AC and the AC-DC components, respectively, defined in the following.

$$P_j^{\text{RF}} = P_0^{\text{RF}} \sum_{i \in \mathcal{U}_j} b_i, \tag{4.6}$$

where: P_0^{RF} is the initial power consumed by the RF in idle mode and $\mathcal{U}_j$ is the set of users associated to RU j.

In addition, P^{DC} is defined as follows:

$$P^{\text{DC}} = \eta_1 P_j^0 \left(e^{\alpha B'_j} + \mathrm{e}^{\lambda.\text{MCS}} \right), \tag{4.7}$$

where: η_1, P_j^0, α, B'_j, λ and MCS are the efficiency of the DC, the initial power consumed by the RU in idle mode, an increasing constant, the number of used RBs by the RU ($B'_j \leq B_j$), a constant and the modulation and coding scheme (MCS), respectively.

Finally, we define P^{AC} as:

$$P^{\text{AC}} = \eta_2 P^{\text{DC}} \left(e^{\alpha B'_j} + \mathrm{e}^{\lambda.\text{MCS}} \right), \tag{4.8}$$

where: η_2 corresponds to the efficiency of the AC, P^{DC} is provided by (4.7) and all the other parameters remain the same as in (4.7).

4.3.4 Problem Formulation (P3)

To tackle the UE to RU association problem, we formulate it as a multi-objective integer optimization problem that seeks to minimize UE access network delay (4.4) and the total energy consumption of the RUs. Note that we consider energy consumption at the RU as opposed to existing solutions that only consider energy consumption at the DUs even though RUs account for 88% of RAN energy consumption [127]. Let $x_{i,j} \in \{0,1\}$ be a binary variable indicating whether UE i is associated to RU j ($x_{i,j} = 1$) or not ($x_{i,j} = 0$). Furthermore, we define another binary variables $y_j = 1$ that indicates if RU j is active or 0 otherwise. Thus, the multi-objective UE association problem is formulated as follows:

$$\textbf{[P3]} \qquad \begin{cases} O_1 : \min \sum_{i=1}^{U} \sum_{j=1}^{R} x_{i,j} \cdot d_{i,j}^{\text{acc}}, \\ O_2 : \min \sum_{j=1}^{R} y_j \cdot P_j. \end{cases} \tag{4.9}$$

Subject to:

$$\sum_{j=1}^{R} x_{i,j} = 1, \qquad \forall i \in \mathcal{U}, \tag{4.10}$$

$$x_{i,j} \leq y_j, \qquad \forall i \in \mathcal{U}, \forall j \in \mathcal{R} \tag{4.11}$$

$$\sum_{i=1}^{U} x_{i,j} \times b_i \leq B_j, \qquad \forall j \in \mathcal{R}, \tag{4.12}$$

$$\sum_{j=1}^{R} x_{i,j} \times d_{i,j}^{\text{acc}} \leq d_i, \qquad \forall i \in \mathcal{U}, \tag{4.13}$$

$$x_{i,j} \in \{0,1\}, y_j \in \{0,1\}, \quad \forall i \in \mathcal{U}, \forall j \in \mathcal{R}. \tag{4.14}$$

The objective defined in (4.9) minimizes the access network delay of UEs as well as the total energy consumption of the RUs. The constraint in (4.10) ensures that a UE is associated to only one RU at a time and (4.11) is introduced to guarantee that the associated RU is active. In addition, constraint (4.12) and the restriction B_j ensure that the total number of RBs allocated to users under each RU do not exceed the RU capacity. On the other hand, constraint (4.13) is the delay constraint that ensures that the maximum tolerable delay of a UE is not exceeded while constraint (4.14) indicate the problem's binary decision variables.

4.3.5 Proposed Algorithm for P3

In this section, we will introduce the proposed solution to solve the above optimization problem P3. To this end, we propose a heuristic to solve the problem in a reasonable time that is based on a delay guaranteed and energy efficient metric, we denote by $\gamma_{i,j}$. The metric is composed of two main components, namely the ratio of the access network delay between the UE i and the RU j to its maximum tolerable delay, which seeks to minimize the access network delay of the UE, while the second ratio is defined as the available RBs of the RU j to its total RBs whose objective is to minimize the number of active RUs by prioritizing potential RUs with more existing load (fewer available RBs), which consequently minimizes the total energy consumption. The metric is defined as:

$$\gamma_{i,j} = \frac{d_{i,j}^{\text{acc}}}{d_i} + \frac{B_j}{B_j^{\text{Int}}}, \tag{4.15}$$

where: $d_{i,j}^{\text{acc}}$, d_i, B_j and B_j^{Int} are the access network delay of UE i to RU j, maximum tolerable delay of UE i, number of available RBs at RU j and the initial total number of RBs of RU j. The potential RU that yields the minimum metric is considered the best

RU.

Moreover, the proposed scheme employs a power control mechanism to put to sleep mode RUs that have no UE associated to them. That is, the status of an RU without any associated UE is set to inactive ($y_j = 0$). Thus, reducing the energy consumption of the system as opposed to existing solutions where idle RUs remain active. This is in stack contrast to existing state-of-the-art (SoA) solutions that associate UEs to the nearest RU [133], or potential RU with the highest SNR value [148]. Indeed, the proposed energy-aware and delay guaranteed UE to RU association metric ensures that as fewer RUs as possible are active while the power control mechanism ensures that RUs without associated UEs are put to sleep. A combination of the association metric and power control mechanism minimizes the total energy consumption of the system while maintaining as low as possible access network delay for the UEs.

The full description of the algorithm follows the sequence depicted in Algorithm 3. For every UE in the UEs queue, we first determine the set of candidate RUs from the pool of available RUs (lines 7-11). Note that, we use the UEs queue in its default state without sorting to avoid increasing the complexity as an attempt at presorting the UEs based on closest RU distance did not yield any tangible delay improvements. An RU j is considered as a candidate for a UE i if its available RBs is at least equal to the UE's required RBs b_i and the delay offered by this RU is less than the UE's tolerable delay. Next, the algorithm computes the UE association metric $\gamma_{i,j}$ (as in line 15) using (4.15), which is used to determine the optimal RU that a UE should be associated to with a view to minimizing the access network delay of UEs while minimizing the overall energy consumption of the system. After determining the optimal RU $j^\star$ (line 16), the UE is associated to it ($x_{i,j^\star} = 1$) and added to its associated UEs list ($\mathcal{A}_{j^\star}$). Moreover, the status of the RU $j^\star$ is set to active ($y_{j^\star} = 1$) (line 19) and its available resources is updated (line 20). The same steps are repeated for all the UEs.

4.3.5.1 Time Complexity Analysis for Algorithm 3

This subsection presents the time complexity analysis of the proposed solution for P3. The time complexity of Algorithm 3 is $O(U \cdot R)$ and arises because for each UE (line 7-11) we check all RUs ($O(U \cdot R)$), then for each UE (lines 13-15) we evaluate all candidate RUs ($O(U \cdot R^c)$) and finally update the association (lines 17-24) in constant time per UE ($O(U)$), resulting in an overall complexity of $O(U \cdot (R + R^c))$.

Algorithm 3: Proposed UE association algorithm for P3.

Setting: $\mathcal{R}, \mathcal{U}, \mathcal{R}^c = \emptyset$, $\mathcal{A}_j = \emptyset$, $B_j^{\text{Int}}, B_j = B_j^{\text{Int}}, b_i, P_j$, $\forall i \in \mathcal{U}$ and $j \in \mathcal{R}$.

Output: $x_{i,j}$,y_j, $\forall i \in \mathcal{U}$ and $j \in \mathcal{R}$.

begin

  for $i \in \mathcal{U}$ **do**

    for $j \in \mathcal{R}$ **do**

      /* Determine candidate RUs */

      if $b_i \leq B_j$ *and* $\mathrm{d}_{i,j}^{\text{acc}} \leq d_i$, **then**

        /* Add candidate RUs to the candidate RUs list $\mathcal{R}^c$ */

        $\mathcal{R}^c = \mathcal{R}^c \cup j$

    for $j \in \mathcal{R}^c$ **do**

      determine $\gamma_{i,j}$ based on (4.15):

      $\gamma_{i,j} = \frac{d_{i,j}^{\text{acc}}}{d_i} + \frac{B_j}{B_j^{\text{Int}}}$.

    $j^\star = \arg\min_j \gamma_{i,j}$.

    /* Associate UE u to its best RU */

    $x_{i,j^\star} = 1$.

    /* Add the UE to the RU's associated UEs list */

    $\mathcal{A}_{j^\star} = \mathcal{A}_{j^\star} \cup i$.

    /* Set the best RU's status to active */

    $y_{j^\star} = 1$.

    /* Update the RU's available resources */

    $B_{j^\star} = B_{j^\star} - b_i$.

4.3.6 Performance Evaluation of the Proposed Solution for P3

This section presents an evaluation of the proposed solution's performance against two SoA algorithms [133] and [148]. First, we describe the simulation setup in Section 4.3.6.1 and present some results in Section 4.3.6.2.

4.3.6.1 Simulation Settings for P3

The simulation scenario consists of 15 RUs of 20MHz bandwidth, 100 RBs capacity and 200mw transmit power each, randomly placed in a rectangular region of 4Km×4Km size. We consider a network with path-loss exponent and noise variance set to 3.5 and -146.424 dBm, respectively as in [129], [149]. We also consider $U \in 50, \ldots, 950$ UEs, randomly generated in no specific ratio across the uRLLC, eMBB and mMTC traffic types and uniformly distributed across the network region. The tolerable delay of uRLLC, eMBB and mMTC UEs is set randomly to between 1ms to 2ms, 1ms to 4ms and 1ms to 4ms, respectively. Similarly, we set the required data rate of uRLLC, eMBB and mMTC UEs to 5Mb/s, 20Mb/s and 1Mb/s respectively while packet size is set randomly between 30-40, 450-500 and 5-30 bits, respectively. The simulation settings are summarized in Table 4.1.

Table 4.1: Simulation Parameters for P3.

Parameter	**Value(s)**
Bandwidth per RU	20 MHz
d_i for uRLLC, eMBB, mMTC	1ms, 4ms, 4ms
r_i for uRLLC, eMBB, mMTC	5, 20 and 1Mb/s
α_i for uRLLC	random 30 to 40 bits
α_i for eMBB	random 450 to 500 bits
α_i for mMTC	random 5 to 30 bits
Number of UEs U	between 50 and 950
Number of RUs	15
RBs per RU B_j	100 RBs
Noise variance σ^2	-146.424 dBm
P_j	200mW
η_1, η_2	0.008
λ	0.1
Path loss exponent	3.5
Processing capacity of RU j (C_j)	0.7 GHz
c_j	0.004 $\times$ 3.4 KHz
P_0^{RF}	12.9w
P^{amp}	6.1w
P_j^0	29.6w

4.3.6.2 Comparison of Proposed Solution for P3 to SoA

In this subsection, we assess the performance of the proposed solution (PPS) and two of its variants. The first variant (PPS1) defines its metric same as (4.15) but does not employ the power control mechanism, that is $y_j = 1$, $\forall j \in \mathcal{R}$, i.e., all the RUs remain active irrespective of whether UEs are associated to them or not. The second variant (PPS2) employs the power control mechanism but defines its metric $\gamma_{i,j}$ only in terms of the delay component as follows:

$$\gamma_{i,j} = \frac{d_{i,j}^{\text{acc}}}{d_i}. \tag{4.16}$$

In addition, the proposed solution is also evaluated against three SoA schemes.

1. **SoA1:** the SoA1 [133] associates UEs to the closest RU and does not employ the power control mechanism (i.e., $y_j = 1$, $\forall j \in \mathcal{R}$).

2. **SoA2:** SoA2 [148] defines its metric $\gamma_{i,j}$ as in equation (4.17) and chooses the highest value to make the association (line 16 of the Algorithm is replaced by arg max), but it does not include the power control mechanism either.

3. **SoA3:** the third SoA3 scheme works as SoA2 but employs the power control mechanism by putting to sleep RUs without any UEs associated to them.

The metric $\gamma_{i,j}$ of SoA2 and SoA3 is defined as:

$$\gamma_{i,j} = \mathrm{SNR}_{i,j} - \frac{\sum_{i \in \mathcal{U}_j} \alpha_i}{\sum_{i=1}^{U} r_{i,j}}, \tag{4.17}$$

where: $\mathrm{SNR}_{i,j}$, α_i and $r_{i,j}$ are the SNR between UE i and RU j (computed as in (4.17)), packet size of UE i associated to RU j and the data rate of UE i associated to RU j, respectively.

$$\mathrm{SNR}_{i,j} = \frac{P_j \times |h_{i,j}|^2}{\sigma^2}, \tag{4.18}$$

where: P_j is the transmit power of RU j, $h_{i,j}$ is the frequency flat fading channel coefficient from RU j to user j and σ^2 is the noise variance.

Figure 4.2 shows the total energy consumption of all RUs against the number of UEs U. It shows that when there are fewer UEs, the total energy consumed by the proposed scheme PPS is much lower than that consumed by all the SoA schemes and even its other variant with this difference gradually reducing as the number of UEs increases. In particular, the PPS decreases energy consumption by 92.4% and 28.0% under low and high number of UEs respectively. This improved performance is due to the UE association metric that enables the association of UEs to RUs with already associated UEs such that few RUs are active, setting the stage for the activation of the power control mechanism which enables idle RUs to be put to sleep. Thus, leading to lower energy consumption. Curiously, it can be observed that the PPS1, a variant of the PPS, which employs the same metric as the PPS but without the power control mechanism and other compared schemes including the SoA3, which employs the power control mechanism, all consume higher and almost the same amount of energy. This further buttresses our earlier assertion that the power control mechanism (ability to turn off or sleep idle RUs) alone without an efficient UE association metric is not sufficient in itself to deliver the required energy gains.

Figure 4.3 presents the total access network delay incurred by all associated UEs. We can observe that all the schemes incur similar total access network delay when there are few UEs. However, as the number of UEs increases, the proposed schemes incur marginally higher delays (3.2%) than the SoA. This marginal increment can be attributed to the UE association metric which for the proposed scheme (PPS) and its variant (PPS1) seeks to balance delay and energy consumption factor as against the SoA schemes that rely on the distance to the RU as the selection metric. Thus, leading to slightly higher access network delay for the PPS and PPS1, however this is well compensated for by the energy consumption which is greatly reduced (about 92.4% and 28.0% under low and high number

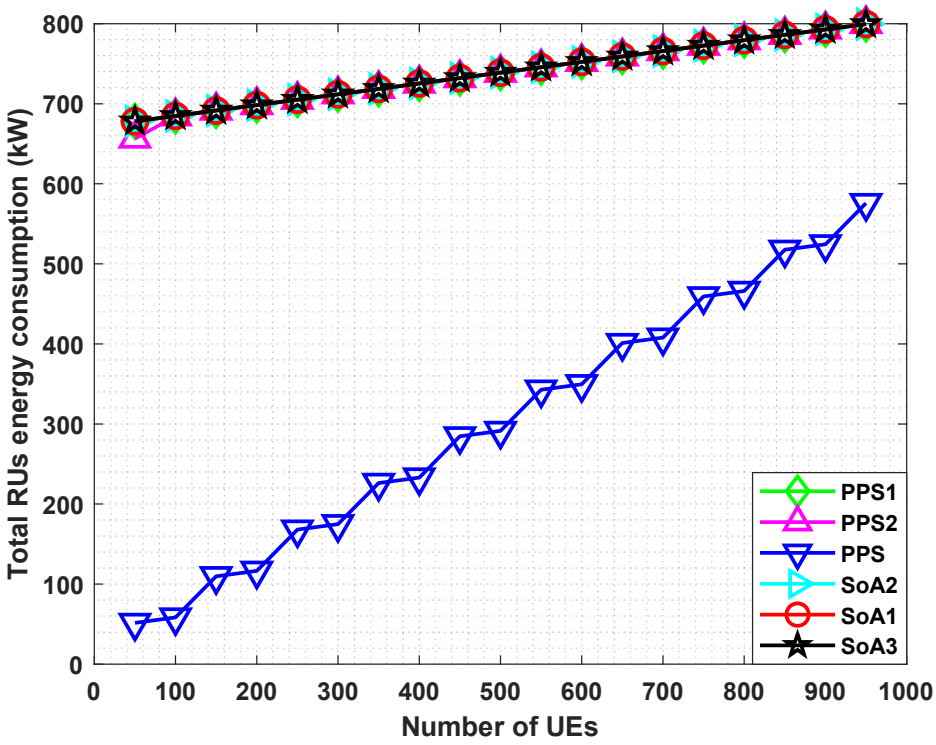


Figure 4.2: Total energy consumption of proposed algorithm for P3 and the SoA algorithms.

of UEs respectively).

The results in Figure 4.4 show how the average access network delay incurred per UE scales with different number of UEs. The results indicate that the distance based scheme SoA1 and the PPS2 incur lower per UE average access network delay than the PPS, PPS1, SoA2 and SoA3 schemes. This is due to the fact that the former, associates UEs to the RUs that are closest and offer the least delay. It is pertinent to note that while the PPS scheme incurs slightly higher delay when compared to the SoA schemes for both the total and per UE delays, it does not violate the delay requirements of the UEs.

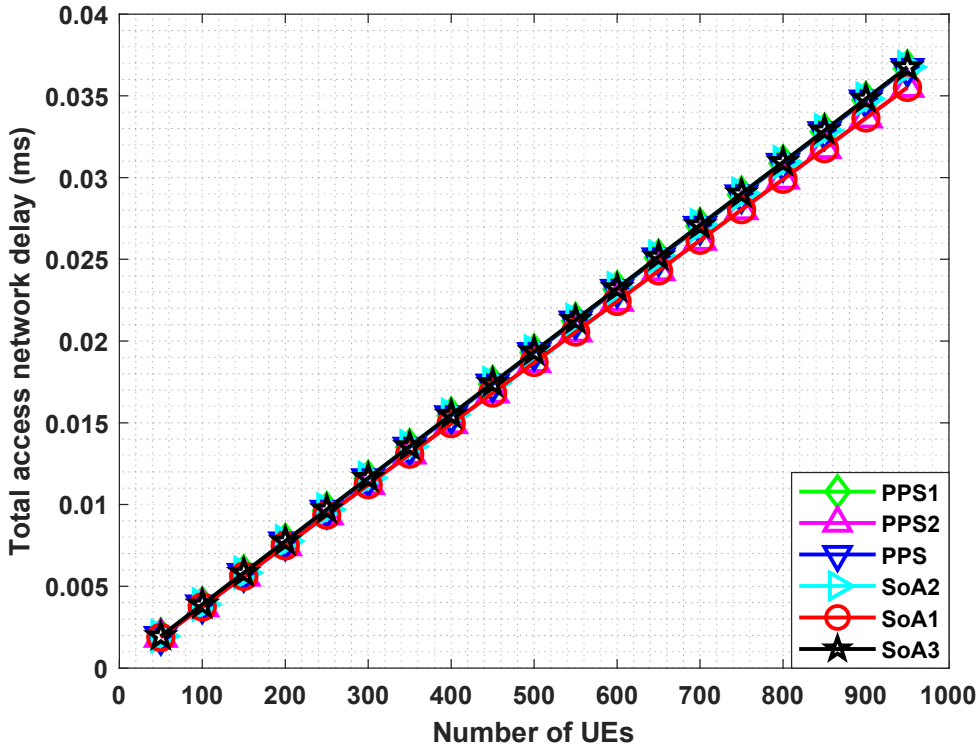


Figure 4.3: Comparison in terms of total access network delay for the proposed algorithm for P3 and SoA algorithms.

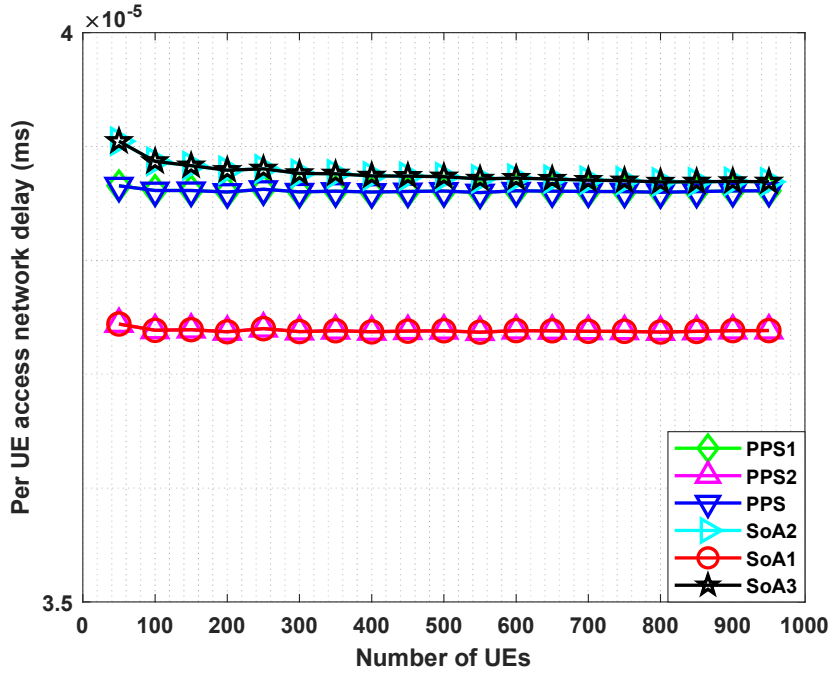


Figure 4.4: Comparison in terms of delay per UE between the proposed algorithm for P3 and the SoA algorithms.

4.4 End to End Delay-Guaranteed and Energy Efficient User Association with Functions Placement for O-RAN

In this section, we present our solution for the end-to-end user association and placement problem. This solution extends the user association solution presented in Section 4.3 and is aimed at determining the optimal RU, DU and CU for each UE, RU and DU respectively. It employs optimal UE, RU and DU association metric that attempts to balance the conflicting objectives of delay and energy consumption. The solution exploits the sleep mode capabilities of RUs, DUs and CUs to minimize power consumption by putting idle units to sleep.

4.4.1 System model (P4)

In this section, we consider an end to end scenario captured in Figure 4.5 as opposed to the access network scenario captured in the previous contribution presented in Section 4.3.1. We consider a network scenario comprising a set of UEs $\mathcal{U} = \{1, 2, \ldots, N\}$, a set of RUs $\mathcal{R} = \{1, 2, \ldots, R\}$, DUs $\mathcal{K} = \{1, 2, \ldots, K\}$ placed at the edge node and a set of CUs $\mathcal{M} = \{1, 2, \ldots, M\}$ placed at the regional node. The UEs are randomly distributed within a predefined area around the RUs and each RU $j \in \mathcal{R}$ is equipped with a certain number of resource blocks (RBs) of equal size, $\mathcal{B}_j = \{1, 2, \ldots, B_j\}$ with which it can serve multiple UEs. Each of these RB $z \in \mathcal{B}_j$ can be allocated to only one UE at a time and each UE $i \in \mathcal{U}$ can only be associated to one RU $j \in \mathcal{R}$. Similarly, each user $i \in \mathcal{U}$ can either be of type uRLLC, eMBB or mMTC with delay and data rate requirements we denote by d_i and r_i, respectively. Moreover, let b_i be the number of required resources blocks to meet the diverse QoS requirements of user i (r_i and d_i).

Similarly, each RU j with total load L_j^{RU} (defined in terms of the RBs it has allocated to the UEs associated to it) is associated to only one DU $k \in \mathcal{K}$ and each DU k with total load L_k^{DU} (defined in terms of the total bits processed by all the RUs associated to it) is in turn connected to only one CU $m \in \mathcal{M}$. Each DU k and each CU m have capacity C_k^{DU} and C_m^{CU} in CPU cycles and requires c_k^{DU} and c_m^{CU} CPU cycles to process one bit per second of data from its associated RUs and DUs, respectively.

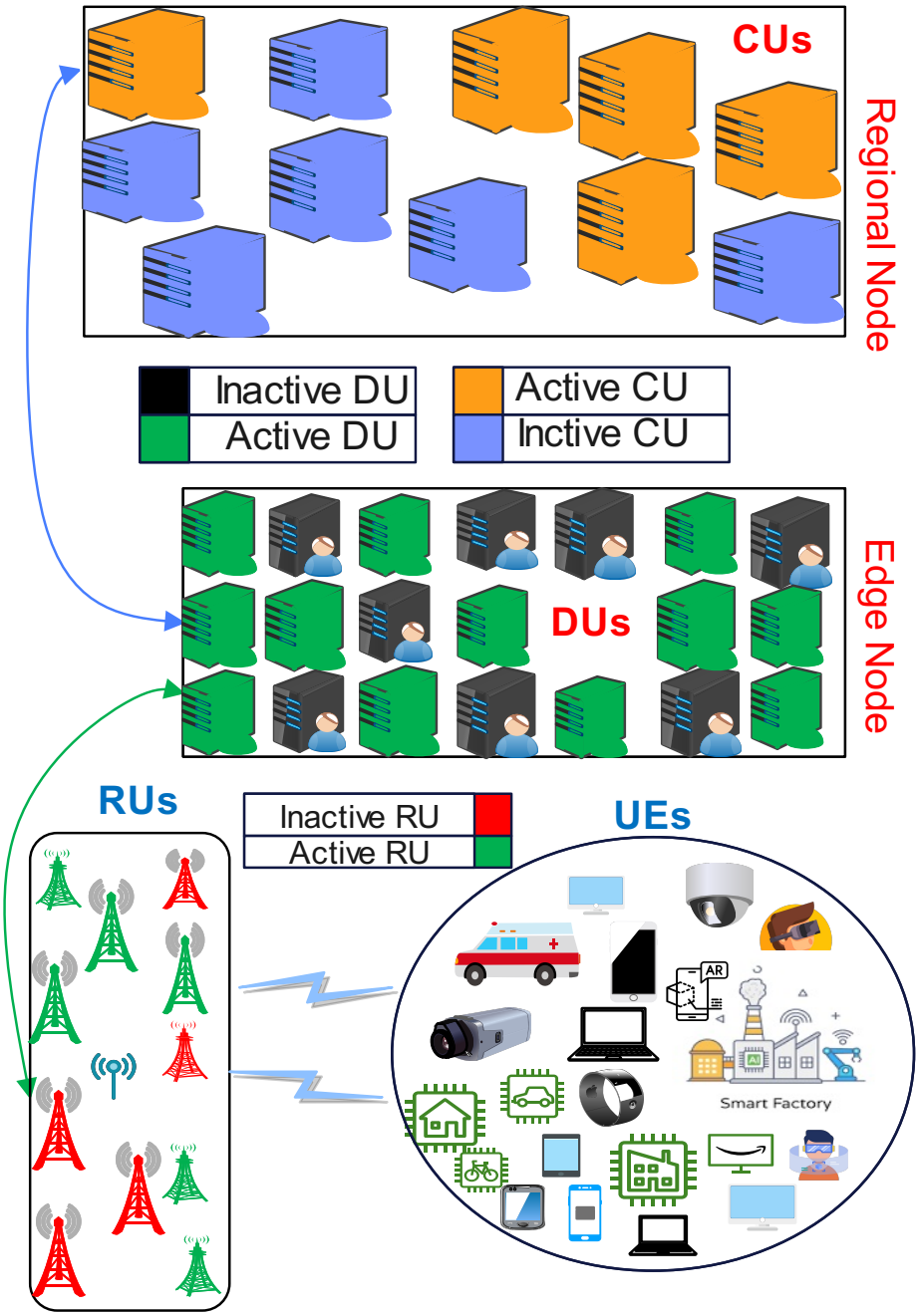


Figure 4.5: System Model (P4).

4.4.2 End to End Delay Model

Following the system model presented above, the end to end delay of UE i (d_i^{e2e}) can be expressed as a sum of the total propagation, transmission and processing delays of the UE, we denote by d_i^{prop}, d_i^{trans} and d_i^{proc}, respectively.

First, the total propagation delay, defined as the time taken for a signal to travel from the various links to a UE. That is, travel from a CU to the DU, through the midhaul (MH), from a DU to RU, through the fronthaul (FH) and access network (AN) from RU

to a UE is expressed as (4.1):

$$d_i^{\text{prop}} = d_{i,j}^{\text{prop,AN}} + d_{j,k}^{\text{prop,FH}} + d_{k,m}^{\text{prop,MH}}, \tag{4.19}$$

where: $d_{i,j}^{\text{prop,AN}}$, $d_{j,k}^{\text{prop,FH}}$ and $d_{k,m}^{\text{prop,MH}}$ are the propagation delay in the access network, FH and MH respectively. The access network represent the communication between user i and its associated RU j, the FH is the communication between user $i's$ associated RU j and the DU k that this associated RU to. Furthermore, the communication in the MH is that between user $i's$ associated RU's DU k and the DU's associated CU m. The propagation delay in the various communication links are determined as follows:

$$d_{i,j}^{\text{prop,AN}} = \frac{\text{dist}_{i,j}^{\text{AN}}}{S_{\text{AN}}}, \tag{4.20}$$

$$d_{j,k}^{\text{prop,FH}} = \frac{\text{dist}_{j,k}^{\text{FH}}}{S_{\text{FH}}}, \tag{4.21}$$

$$d_{k,m}^{\text{prop,MH}} = \frac{\text{dist}_{k,m}^{\text{MH}}}{S_{\text{MH}}}, \tag{4.22}$$

where: $\text{dist}_{i,j}$, $\text{dist}_{j,k}$ and $\text{dist}_{k,m}$ corresponds to the distance that the data of the user i needs to travel in the various links (AN, FH and MH); j is the RU serving UE i, k is the DU serving RU j and m is the CU serving DU k. Furthermore, S_{AN}, S_{FH} and S_{MH} correspond to the propagation speed of the AN, FH and MH medium.

Similarly, the total transmission delay, d_i^{trans} of user i with packet size α_i is defined as the sum of the transmission delay incurred in all the links (AN, FH and MH), given as follows (4.23) [139]:

$$d_i^{\text{trans}} = d_{i,j}^{\text{trans,AN}} + d_{j,k}^{\text{trans,FH}} + d_{k,m}^{\text{trans,MH}}, \tag{4.23}$$

where: $d_{i,j}^{\text{trans,AN}}$, $d_{j,k}^{\text{trans,FH}}$ and $d_{k,m}^{\text{trans,MH}}$, are the transmission delay in the access network between user i and RU j, in the FH between RU j and DU k and in the MH between DU k and CU m, respectively and are determined as follows:

$$d_{i,j}^{\text{trans,AN}} = \frac{\alpha_i}{r_i}, \tag{4.24}$$

$$d_{j,k}^{\text{trans,FH}} = \frac{w_{\text{FH}} \times r_i}{W_{\text{FH}}}, \tag{4.25}$$

$$d_{k,m}^{\text{trans,MH}} = \frac{w_{\text{MH}} \times r_i}{W_{\text{MH}}}, \tag{4.26}$$

where: r_i, α_i, w_{FH}, w_{FH}, W_{FH}, W_{MH} are the data rate of user i, mean packet size of the user i, bandwidth required to transmit one bit per second of data across the FH link, bandwidth required to transmit one bit per second of data across the MH link, bandwdith of the FH and MH links, respectively.

Furthermore, the total processing delay, d_i^{proc}, of user i with packet size α_i is defined as the sum of the processing delay incurred in all the links (AN, FH and MH), given as follows (4.27) [148]:

$$d_i^{\text{proc}} = d_{i,j}^{\text{proc,AN}} + d_{j,k}^{\text{proc,FH}} + d_{k,m}^{\text{proc,MH}}, \tag{4.27}$$

where: the processing delay in the access network, fronthaul and midhaul are determined as follows:

$$d_{i,j}^{\text{proc,AN}} = \frac{\alpha_i \times c_j^{\text{RU}}}{C_j^{\text{RU}}}, \tag{4.28}$$

$$d_{j,k}^{\text{proc,FH}} = \frac{\alpha_i \times c_k^{\text{DU}}}{C_k^{\text{DU}}}, \tag{4.29}$$

$$d_{k,m}^{\text{proc,MH}} = \frac{\alpha_i \times c_m^{\text{CU}}}{C_m^{\text{CU}}}, \tag{4.30}$$

where: α_i, c_j^{RU}, c_k^{DU}, c_m^{CU}, C_j^{RU}, C_k^{DU}, C_m^{CU}, are the mean packet size of user i, CPU cycles required by RU j to process one bit per second of data from its associated UEs, CPU cycles required by DU k to process one bit per second of data from its associated RUs, CPU cycles required by CU m to process one bit per second of data from its associated DUs, processing capacities of the RU, DU and CU, respectively.

Thus, the end to end delay of user i, d_i^{e2e}, is expressed as:

$$d_i^{\text{e2e}} = d_i^{\text{prop}} + d_i^{\text{proc}} + d_i^{\text{trans}}. \tag{4.31}$$

Similarly, as opposed to the above where we defined the delays by the various types, we can define the delays in the various links (AN, FH, MH) as:

$$d_{i,j}^{\text{AN}} = d_{i,j}^{\text{prop,AN}} + d_{i,j}^{\text{proc,AN}} + d_{i,j}^{\text{trans,AN}}, \tag{4.32}$$

$$d_{j,k}^{\text{FH}} = d_{j,k}^{\text{prop,FH}} + d_{j,k}^{\text{proc,FH}} + d_{j,k}^{\text{trans,FH}}, \tag{4.33}$$

$$d_{k,m}^{\text{MH}} = d_{k,m}^{\text{prop,MH}} + d_{k,m}^{\text{proc,MH}} + d_{k,m}^{\text{trans,MH}}. \tag{4.34}$$

4.4.3 Energy Consumption Model (P4)

In this section, we introduce the power consumption model of DUs and CUs as in [149] as a followup to the RUs energy consumption model defined by equation (4.6) in subsection 4.3.3.

4.4.3.1 CU Power Consumption Model

We define the power consumption of each CU $m \in \mathcal{M}$, P_m^{CU}, as follows:

$$P_m^{\mathrm{CU}} = P_m^{\mathrm{T}} + P_m^{\mathrm{DC}} + P_m^{\mathrm{AC}} + P_m^{\mathrm{cool}}, \tag{4.35}$$

where: P_m^{T}, P_m^{DC}, P_m^{AC} and P_m^{cool}, are the total fixed power consumption of the CU, the direct current to direct current (DC-DC) converter and the alternating current to direct current (AC-DC) converter or inverter module power consumption and cooling power consumption, respectively, defined in the following.

$$P_m^{\mathrm{T}} = P_m^0 \left(e^{L_m^{\mathrm{CU}} + e^{\lambda.\mathrm{MCS}} + e^{\lambda.V}} \right), \tag{4.36}$$

where: P_m^0 is the initial power consumed by the CU in idle mode, L_m^{CU} is the total load of the CU (resource blocks processed), λ is a constant, MCS is the modulation and coding scheme and V is the number of virtual machines.

In addition, P_m^{DC} is defined as follows:

$$P_m^{\mathrm{DC}} = \ell^{\mathrm{DC}}(\eta^{\mathrm{DC}}) \times P_m^T, \tag{4.37}$$

where: ℓ^{DC} corresponds to the CU's DC-DC conversion losses, defined as a function of the efficiency of the DC conversion (η^{DC}). We also define P_m^{AC} as:

$$P_m^{\mathrm{AC}} = \ell^{\mathrm{AC}}(\eta^{\mathrm{AC}}) \times P_m^{\mathrm{DC}} \times P_m^{\mathrm{T}}, \tag{4.38}$$

where: ℓ^{AC} and η^{AC} correspond to the losses and efficiency of the CU AC-DC converters. Furthermore, we present the cooling power P_m^{cool}, modeled as being proportional to the power consumed by other components and the consumption of cooling loss in the CU $\ell_{\mathrm{cool}}^{\mathrm{CU}}$ as:

$$P_m^{\mathrm{cool}} = \ell^{\mathrm{CU,cool}} \left(P_m^{\mathrm{T}} + P_m^{\mathrm{DC}} + P_m^{\mathrm{AC}} \right). \tag{4.39}$$

where: $\ell^{\mathrm{CU,cool}}$ is the cooling loss of the CU.

4.4.3.2 DU Power Consumption Model

Similarly the energy consumption of the CU, we present the power consumption of each DU $k \in \mathcal{K}$, P_k as:

$$P_k^{\mathrm{DU}} = P_k^{\mathrm{T}} + P_k^{\mathrm{DC}} + P_k^{\mathrm{AC}} + P_k^{\mathrm{cool}}. \tag{4.40}$$

where: P_k^{T}, P_k^{DC}, P_k^{AC} and P_k^{cool}, are the total fixed power consumption of the DU, the direct current to direct current (DC-DC) converter and the alternating current to direct current (AC-DC) converter or inverter module power consumption and cooling power consumption, respectively.

4.4.4 Problem Formulation (P4)

We formulate the joint UE to RU association with DU and CU placement problem as an integer optimization problem that seeks to minimize a weighted utility defined as the UE end to end delay (d_i^{e2e}) and number of active RUs, DUs and CUs. Let $x_{i,j} \in \{0,1\}$, $y_{j,k} \in \{0,1\}$ and $z_{k,m} \in \{0,1\}$ be the binary variables that indicate whether or not UE $i \in \mathcal{U}$ is associated with RU $j \in \mathcal{R}$, RU $j \in \mathcal{R}$ is associated with DU $k \in \mathcal{D}$ and DU $k \in \mathcal{D}$ is associated with CU $m \in \mathcal{M}$, respectively, with 1 indicating the successful association and 0 otherwise.

We also introduce the binary variables $a_j \in \{0,1\}$, $p_k \in \{0,1\}$ and $q_m \in \{0,1\}$ that indicate whether or not an RU, DU or CU is active, respectively, with 0 indicating inactive status and 1 indicating active status.

Thus, the joint UE, RU and DU association problem is formulated as follows:

$$\begin{aligned}\textbf{[P4]} \quad & \min\ \omega_d \left(\sum_{i\in\mathcal{U}} \sum_{j\in\mathcal{R}} x_{i,j}.d_{i,j}^{\mathrm{AN}} + \sum_{j\in\mathcal{R}} \sum_{k\in\mathcal{K}} y_{j,k}.d_{j,k}^{\mathrm{FH}} + \sum_{k\in\mathcal{K}} \sum_{m\in\mathcal{M}} z_{k,m}.d_{k,m}^{\mathrm{MH}} \right) + \\ & \omega_e \left(\sum_{j\in\mathcal{R}} a_j.P_j^{\mathrm{RU}} + \sum_{k\in\mathcal{K}} p_k.P_k^{\mathrm{DU}} + \sum_{m\in\mathcal{M}} q_m.P_m^{\mathrm{CU}} \right),\end{aligned} \tag{4.41}$$

Subject to:

$$\sum_{j\in\mathcal{R}} x_{i,j} = 1, \quad \forall i \in \mathcal{U}, \tag{4.42}$$

$$x_{i,j} \leq a_j, \qquad \forall i \in \mathcal{U},\ \forall j \in \mathcal{R}, \tag{4.43}$$

$$\sum_{k\in\mathcal{K}} y_{j,k} = 1, \quad \forall j \in \mathcal{R}, \tag{4.44}$$

$$y_{j,k} \leq p_k, \qquad \forall j \in \mathcal{R},\ \forall k \in \mathcal{K}, \tag{4.45}$$

$$\sum_{m \in \mathcal{M}} z_{k,m} = 1, \quad \forall k \in \mathcal{K}, \tag{4.46}$$

$$z_{k,m} \leq q_m, \qquad \forall k \in \mathcal{K}, \forall m \in \mathcal{M}, \tag{4.47}$$

$$\sum_{i \in \mathcal{U}} x_{i,j} \times b_i \leq B_j, \quad \forall j \in \mathcal{R}, \tag{4.48}$$

$$\sum_{i \in \mathcal{U}} \sum_{j \in \mathcal{R}} x_{i,j} \times y_{j,k} \times c_k^{\mathrm{DU}} \times \alpha_i \leq C_k^{\mathrm{DU}}, \quad \forall k \in \mathcal{K}, \tag{4.49}$$

$$\sum_{i \in \mathcal{U}} \sum_{j \in \mathcal{R}} \sum_{k \in \mathcal{K}} x_{i,j} \times y_{j,k} \times z_{k,m} \times c_m^{\mathrm{CU}} \times \alpha_i \leq C_m^{\mathrm{CU}}, \quad \forall m \in \mathcal{M}, \tag{4.50}$$

$$\begin{aligned} \sum_{j \in \mathcal{R}} x_{i,j} d_{i,j}^{\mathrm{AN}} + \sum_{j \in \mathcal{R}} \sum_{k \in \mathcal{K}} x_{i,j}.y_{j,k}.d_{j,k}^{\mathrm{FH}} + \\ \sum_{j \in \mathcal{R}} \sum_{k \in \mathcal{K}} \sum_{m \in \mathcal{M}} x_{i,j}.y_{j,k}.z_{k,m}.d_{k,m}^{\mathrm{MH}} \leq d_i, \quad \forall i \in \mathcal{U} \end{aligned} \tag{4.51}$$

$$\begin{aligned} x_{i,j}, y_{j,k}, z_{k,m}, a_j, p_k, q_m \in \{0,1\}, \forall i \in \mathcal{U}, \\ \forall j \in \mathcal{R}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}. \end{aligned} \tag{4.52}$$

The objective defined in (4.41) seeks to minimize the end to end delay of UEs as well as the total energy consumption of the RUs, DUs and CUs, that are normalized to express them in the same scale, while ω_d and ω_e are the delay and energy weights, respectively. The constraints defined by (4.42) correspond to the user association constraints that ensure that a UE is associated to only one RU at a time and the constraint (4.43) ensure that the associated RU is active. In addition, (4.44) are the RU association constraint that ensure that an RU is associated to only one DU at a time and the constraints (4.45) are introduced to guarantee that the associated DU is active. On the other hand, (4.46) are the DU association constraint that ensure that a DU is associated to only one CU at a time while constraints (4.47) guarantee that the associated CU is active. Furthermore, constraints (4.48), (4.49) and (4.50) are the RU, DU and CU capacity constraints, respectively. Indeed, they ensure that the total number of RBs allocated to users under each RU does not exceed the RU capacity and the total load under each DU

and CU do not exceed the DU and CU processing capacity. Constraint (4.51) are the delay constraints that guarantee that the maximum tolerable delay of each UE is not exceeded while constraints (4.52) indicate the problem's binary decision variables.

4.4.5 Proposed Algorithm for P4

In this section, we propose an algorithm to solve the above optimization problem in a reasonable time.

To address the association problem in the proposed architecture, we introduce three association metrics that govern the selection of RUs, DUs and CUs for UEs, RUs and DUs, respectively. These metrics, denoted as $\gamma_{i,j}^{\mathrm{AN}}$, $\gamma_{j,k}^{\mathrm{FH}}$ and $\gamma_{k,m}^{\mathrm{MH}}$, are designed to optimize both network performance and energy efficiency through a dual-objective approach.

Each association metric incorporates two fundamental components. The first component captures the normalized delay contribution, where delays are expressed as ratios relative to maximum tolerable thresholds. Specifically, this includes the access network delay for UE-to-RU associations, the fronthaul delay for RU-to-DU associations and the midhaul delay for DU-to-CU associations. The normalization ensures fair comparison across heterogeneous delay requirements and network conditions.

The second component represents resource utilization, quantified as the ratio of occupied resources to total available capacity. For RUs, this corresponds to the ratio of occupied resource blocks to total resource blocks. For DUs and CUs, it represents the ratio of allocated processing capacity to total processing capacity, measured in CPU cycles. This component promotes load balancing by favoring network elements with existing resource commitments, thereby minimizing the total number of active components and consequently reducing overall energy consumption.

The association metrics are formally defined as follows:

$$\gamma_{i,j}^{\mathrm{AN}} = \frac{d_{i,j}^{\mathrm{AN}}}{d_i} + \frac{B_j}{B_j^{\mathrm{Int}}}, \tag{4.53}$$

$$\gamma_{j,k}^{\mathrm{FH}} = \frac{d_{j,k}^{\mathrm{FH}}}{d_{\max}} + \frac{C_k^{\mathrm{DU}*}}{C_k^{\mathrm{DU}}}, \tag{4.54}$$

$$\gamma_{k,m}^{\mathrm{MH}} = \frac{d_{k,m}^{\mathrm{MH}}}{d_{\max}} + \frac{C_m^{\mathrm{CU}*}}{C_m^{\mathrm{CU}}}. \tag{4.55}$$

The parameters in these equations are defined as follows:

- $d_{i,j}^{\mathrm{AN}}$ denotes the access network delay between UE i and RU j;
- d_i represents the maximum tolerable delay for UE i;
- B_j and B_j^{Int} denote the number of available resource blocks and the initial total number of resource blocks at RU j, respectively;
- $d_{j,k}^{\mathrm{FH}}$ represents the fronthaul delay between RU j and DU k;
- $d_{\max} = \max_{i \in U} d_i$ denotes the maximum possible tolerable delay across all users;
- C_k^{DU} and $C_k^{\mathrm{DU}*}$ represent the total and available processing capacity of DU k, respectively, measured in CPU cycles;
- $d_{k,m}^{\mathrm{MH}}$ denotes the midhaul delay between DU k and CU m;
- C_m^{CU} and $C_m^{\mathrm{CU}*}$ represent the total and available processing capacity of CU m, respectively, measured in CPU cycles.

The association decisions are then made by selecting the network element that minimizes the respective association metric, thereby achieving the desired balance between delay minimization and energy efficiency. Note that we used the maximum possible delay $d_{\max}$ as the denominators of the first components of the RU and DU association metrics as opposed to the UE association metric where the actual delay of the UE is used. This is because unlike the UE association metric where UEs are directly associated to the RUs, the UEs are not directly associated to DUs and CUs. Thus, the use of the maximum possible delay for the RU and DU association metrics enables an estimate that works for every UE under consideration given that the RU and DU is yet to be performed at this point.

Moreover, the proposed scheme employs a power control mechanism to put to sleep mode RUs, DUs and CUs that have no UE, RU and DU associated to them, respectively. The proposed algorithm follows the sequence depicted in Algorithm 4. First the simulation parameters are initialized and set (lines 1-2) after which for every UE in the UEs queue, we determine the set of candidate RUs from the pool of available RUs (lines 4-9). An RU j is considered as a candidate for a UE i if its available RBs is at least equal to the UE's required RBs b_i and the delay offered by this RU is less than the UE's tolerable delay d_i. Next, the algorithm computes the UE association metric $\gamma_{i,j}^{\mathrm{AN}}$ (as in line 12) using (4.53), which is used to determine the optimal RU that a UE should be associated to with a view to minimizing the access network delay of UEs while minimizing the overall energy consumption of the system. After determining the optimal RU $j^\star$ (line 13), the UE is associated to it ($x_{i,j^\star} = 1$) and added to its associated UEs list ($\mathcal{A}_{j^\star}^{\mathrm{RU}}$). Moreover,

the status of the RU $j^\star$ is set to active ($a_{j^\star} = 1$) (line 16) and its available resources is updated (line 17). The same steps are repeated for all the UEs.

Similarly, for every RU in RUs queue, the proposed algorithm determines candidate DUs (lines 20-25) similar to the procedure for RUs after which it employs the RU association metric $\gamma^{\text{FH}}_{j,k}$ (as in line 28) using (4.54) to determine the optimal DU that an RU should be associated to. This metric is designed to minimize the fronthaul delay while minimizing the overall energy consumption of the system by ensuring that as fewer as possible DUs are active. After determining the best DU (as in line 29), the RU is associated to the DU, with the DU's status set to active and capacity updated (as in line 32 and 33, respectively). A similar procedure is repeated to associate DUs to CUs (lines 36-51).

4.4.5.1 Time Complexity Analysis for Algorithm 4

In this subsection, we present the time complexity analysis of our proposed solution to P4. The time complexity of Algorithm 4 is $O(U \cdot (R + R^c) + R \cdot (K + K^c) + K \cdot (M + M^c))$. This is because for each UE (lines 4-9) we check all RUs to build the candidate list ($O(U \cdot R)$), after which we evaluate all candidate RUs, determine the best RU and perform the association (lines 10-19) all in ($O(U \cdot R^c)$). Similarly, for each RU we check all DUs and for each DU we check all CUs to build candidate lists ($O(R \cdot K + K \cdot M)$), after which we go over the various candidate lists to select the best unit in ($O(R \cdot K^c + K \cdot M^c)$) with the respective associations and resource updates performed in linear time, resulting in an overall complexity of $O(U \cdot (R + R^c) + R \cdot (K + K^c) + K \cdot (M + M^c))$.

4.4.6 Performance Evaluation of the Solution Proposed for P4

This section presents an evaluation of the performance of the proposed solution against the exact solution of the optimization problem and a SoA solution [145]. We first present the simulation setup in Section 4.4.6.1, then some results in Section 4.4.6.2.

4.4.6.1 Simulation Settings for P4

The simulation scenario consists of a rectangular region of 4Km×4Km size, with $R \in \{25, \ldots, 60\}$ RUs randomly placed in the area, each of 20MHz bandwidth, 100 RBs capacity and 200mw transmit power and processing capacity of 0.7GHz each (in CPU cycles per seconds). It also consists of 7 DUs and 2 CUs with processing capacity of 0.7GHz and 3.4GHz requiring 0.04×3.4 KHz and 0.00175×3.4 KHz units of computing resource (in CPU cycles per second) to process 1 bit per second data each, respectively. The path-loss exponent and noise variance are set to 3.5 and -146.424 dBm, respectively, as in

Algorithm 4: Proposed UE association and functions placement algorithm for P4.

Setting: $\mathcal{R}$, $\mathcal{U}$, $\mathcal{R}^c$,$\mathcal{K}$, $\mathcal{M}$, $\mathcal{K}^c$, $\mathcal{R}^c = \emptyset$, $\mathcal{K}^c = \emptyset$, $\mathcal{M}^c = \emptyset$, $\mathcal{A}_j$, $\mathcal{A}_j^{\text{RU}}$, $\mathcal{A}_k^{\text{DU}}$, $\mathcal{A}_m^{\text{CU}} = \emptyset$, B_j^{Int}, $B_j = B_j^{\text{Int}}$, b_i, P_j,
  $\forall i \in \mathcal{U}$, $j \in \mathcal{R}$, $k \in \mathcal{K}$ and $m \in \mathcal{M}$
Output: $x_{i,j}$, a_j, $y_{j,k}$, p_k, $z_{k,m}$, q_m, $\forall i \in \mathcal{U}$, $j \in \mathcal{R}$, $k \in \mathcal{K}$ and $m \in \mathcal{M}$
for $i \in \mathcal{U}$ **do**
  for $j \in \mathcal{R}$ **do**
    if $b_i \leq B_j$ and $d_{i,j}^{\text{acc}} \leq d_i$ **then**
      $\mathcal{R}^c = \mathcal{R}^c \cup j$
    end if
  end for
  for $j \in \mathcal{R}^c$ **do**
    determine $\gamma_{i,j}^{\text{AN}}$ based on (4.53):
    $\gamma_{i,j}^{\text{AN}} = \frac{d_{i,j}^{\text{acc}}}{d_i} + \frac{B_j}{B_j^{\text{Int}}}$
    $j^\star = \arg\min_j \gamma_{i,j}^{\text{AN}}$
    $x_{i,j^\star} = 1$
    $\mathcal{A}_{j^\star}^{\text{RU}} = \mathcal{A}_{j^\star}^{\text{RU}} \cup i$
    $a_{j^\star} = 1$
    $B_{j^\star} = B_{j^\star} - b_i$
  end for
end for
for $j \in \mathcal{R}$ **do**
  for $k \in \mathcal{K}$ **do**
    if $L_j \leq C_k$ **and** $d_{j,k} \leq d_{\max}$ **then**
      $\mathcal{K}^c = \mathcal{K}^c \cup k$
    end if
  end for
  for $k \in \mathcal{K}^c$ **do**
    determine $\gamma_{j,k}^{\text{FH}}$ based on (4.54):
    $\gamma_{j,k}^{\text{FH}} = \frac{d_{j,k}}{d_{\max}} + \frac{C_k^{\text{DU}*}}{C_k^{\text{DU}}}$
    $k^\star = \arg\min_k \gamma_{j,k}^{\text{FH}}$
    $y_{j,k^\star} = 1$
    $\mathcal{A}_{k^\star}^{\text{DU}} = \mathcal{A}_{k^\star}^{\text{DU}} \cup j$
    $p_{k^\star} = 1$
    $C_{k^\star} = C_{k^\star} - L_j$
  end for
end for
for $k \in \mathcal{K}$ **do**
  for $m \in \mathcal{M}$ **do**
    if $L_k \leq C_m$ and $d_{k,m} \leq d_{\max}$ **then**
      $\mathcal{M}^c = \mathcal{M}^c \cup m$
    end if
  end for
  for $m \in \mathcal{M}^c$ **do**
    determine $\gamma_{k,m}^{\text{MH}}$ based on (4.55):
    $\gamma_{k,m}^{\text{MH}} = \frac{d_{k,m}}{d_{\max}} + \frac{C_m^{\text{CU}*}}{C_m^{\text{CU}}}$
    $m^\star = \arg\min_m \gamma_{k,m}^{\text{MH}}$
    $z_{k,m^\star} = 1$
    $\mathcal{A}_{m^\star}^{\text{CU}} = \mathcal{A}_{m^\star}^{\text{CU}} \cup k$
    $q_{m^\star} = 1$
    $C_{m^\star} = C_{m^\star} - L_k$
  end for
end for

[132] and [146]. Furthermore, we consider $U \in \{20, \ldots, 250\}$ UEs, where 20%, 30%, and 50% are randomly assigned to uRLLC, eMBB, and mMTC traffic types, respectively, and uniformly distributed across the network region. The tolerable delay of uRLLC, eMBB and mMTC UEs is set randomly to between 1ms to 2ms, 1ms to 4ms and 1ms to 4ms, respectively, while the required data rate of uRLLC, eMBB and mMTC UEs is set to 5Mb/s, 20Mb/s and 1Mb/s, respectively and packet size randomly between 30-40, 450-500 and 5-30 bits, respectively. The simulation settings are summarized in Table 4.2.

Table 4.2: Simulation Parameters for P4.

Parameter	**Value(s)**
Bandwidth per RU	20 MHz
d_i for uRLLC, eMBB, mMTC	1ms, 4ms, 4ms
r_i for uRLLC, eMBB, mMTC	5, 20 and 1Mb/s
α_i for uRLLC	random 30 to 40 bits
α_i for eMBB	random 450 to 500 bits
α_i for mMTC	random 5 to 30 bits
Number of UEs U	between 50 and 250
Number of RUs	15
RBs per RU B_j	100 RBs
Noise variance σ^2	-146.424 dBm
P_j	200mW
ℓ^{DC}, ℓ_{ru}^{AC}, ℓ^{DC}, ℓ^{AC}, ℓ^{DC}, ℓ^{AC}	0.008
$\ell^{\text{CU,cool}} = \ell^{\text{DU,cool}}$	0.1
η^{DC}, η^{AC}, η^{DC}, η^{AC}, η^{DC}, η^{AC}	0.50, 0.38, 0.40, 0.36, 0.50, 0.45
λ	0.1
Path loss exponent	3.5
Processing capacity of RU j (C_j^{RU})	0.7 GHz
Processing capacity of DU k (C_k^{DU})	0.7 GHz
Processing capacity of CU m (C_m^{CU})	3.4 GHz
c_j^{RU}	0.004 × 3.4 KHz
c_k^{DU}	0.04 × 3.4 KHz
c_m^{CU}	0.00175 × 3.4 KHz
P_0^{RF}	12.9w
P_j^{amp}	6.1w
P_j^0	29.6w

4.4.6.2 Comparison of the Proposed Solution for P4 to the Solver Solution

In this subsection, we compare the results provided by the proposed algorithm (PA) and the results obtained by solving the optimization problem using Gurobi optimization solver, which provides an exact solution and consequently makes the comparison more general. We present the average results of 50 runs.

The total energy consumption of the system (comprising those of the RUs, DUs and

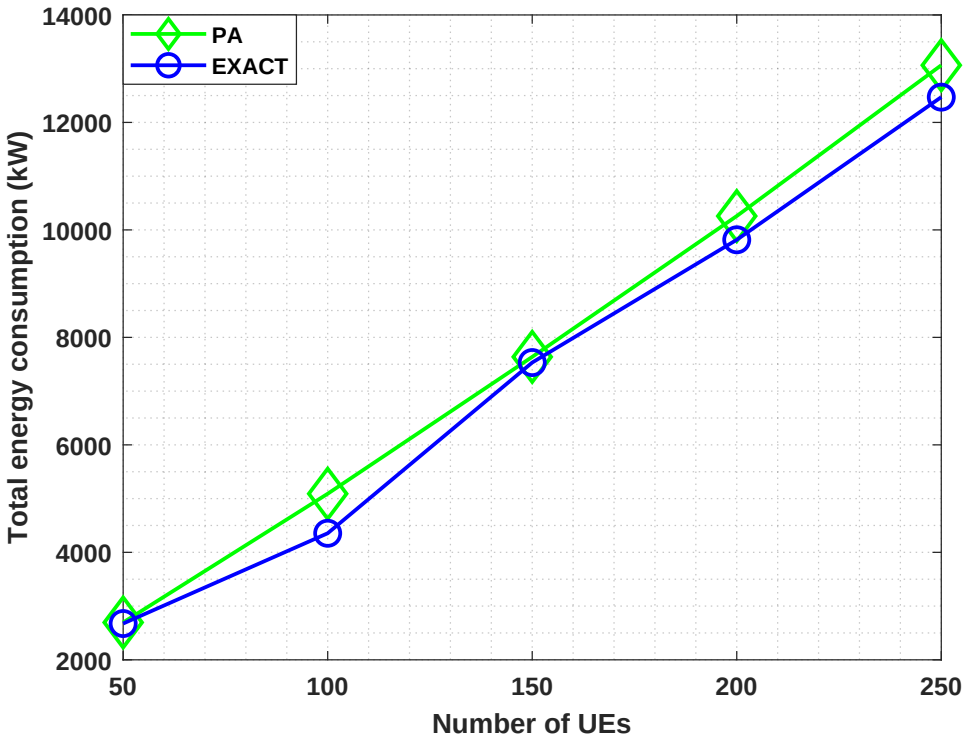


Figure 4.6: Comparison of the proposed algorithm for P4 against the exact solution in terms of total energy consumption.

CUs) against the number of UEs U is captured in Figure 4.6. It shows that the proposed algorithm and the exact solution consume almost the same amount of energy. This is because both solutions employ the same association metric and power control mechanism.

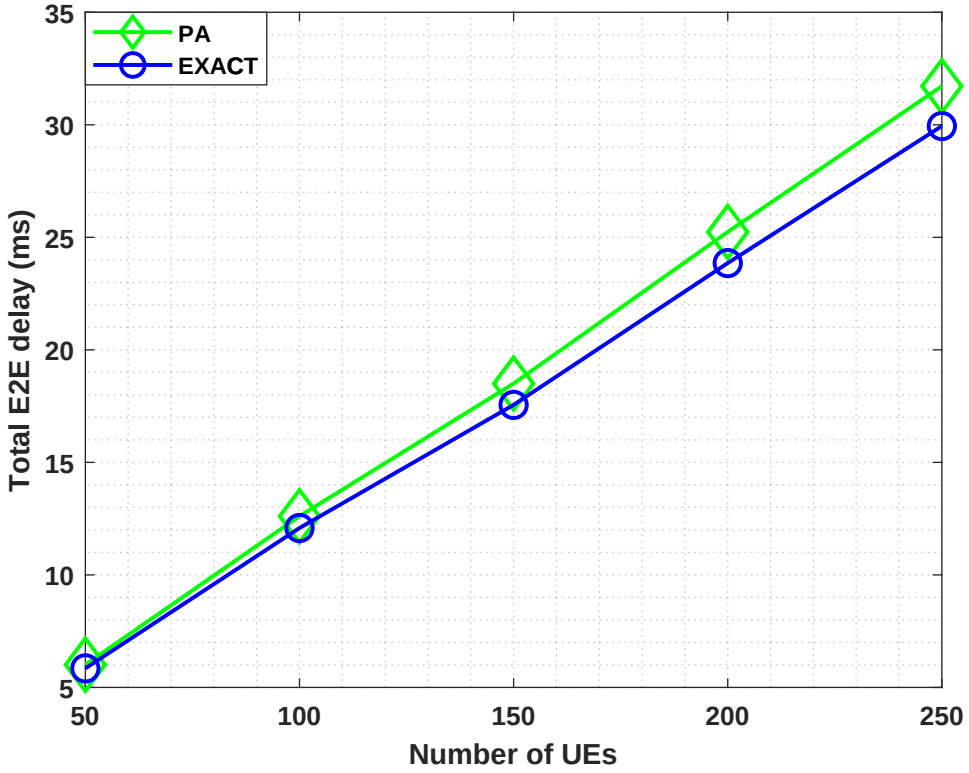


Figure 4.7: Comparison of the proposed algorithm for P4 against the exact solution in terms of total UEs delay.

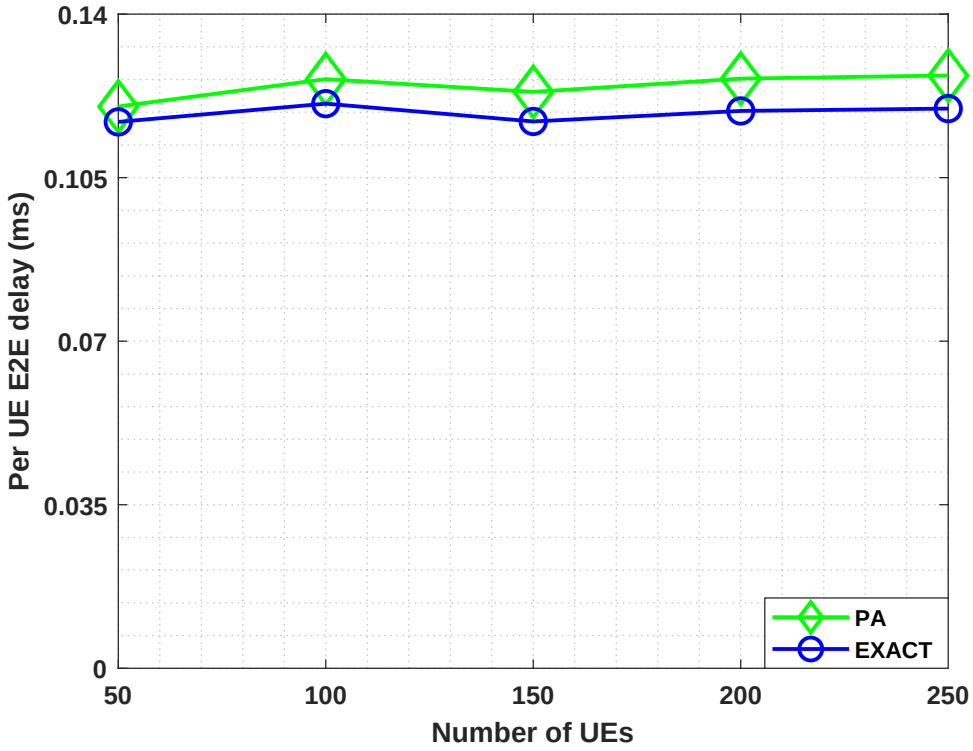


Figure 4.8: Comparison of the proposed algorithm for P4 against the exact solution in terms of per UE delay.

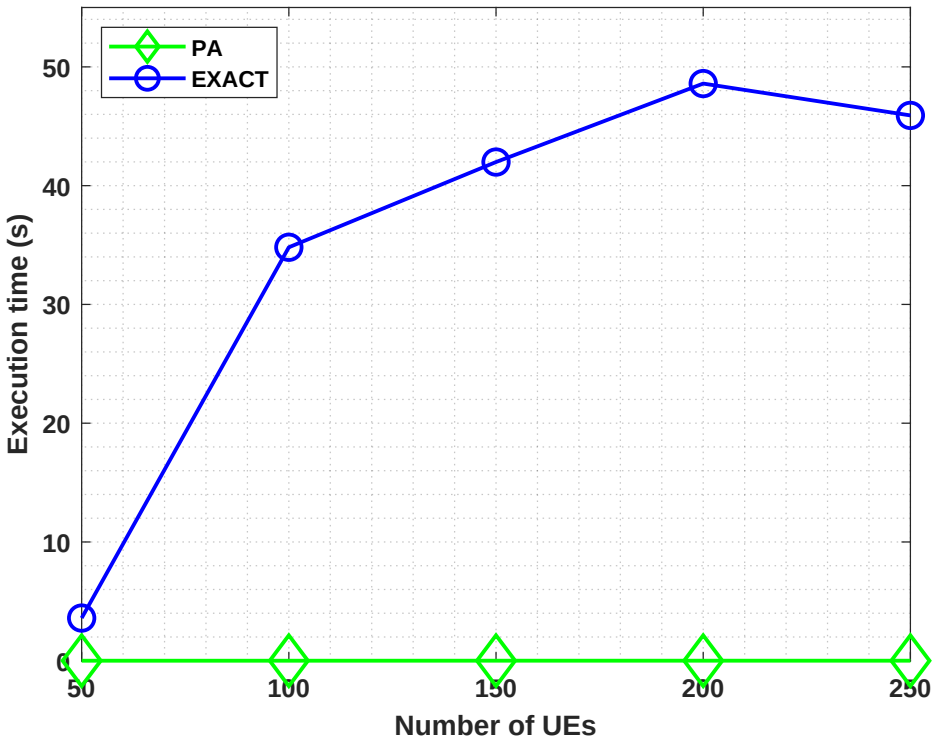


Figure 4.9: Comparison of the proposed algorithm for P4 against the exact solution in terms of execution time.

Figure 4.7 presents the total end to end delay incurred by all associated UEs. We can observe that both solutions incur similar total delay when there are few UEs. However, as the number of UEs increases, the proposed algorithm incurs about 4.5% than the solution provided by the solver without violating the delay requirement of the UEs.

The results in Figure 4.8 show how the average end to end delay incurred per UE scales with different number of UEs. The results show that similar to the total delay, the solution provided by the solver incurs the lowest average per UE delay.

Figure 4.9 depicts the execution time of both solutions for different number of UEs in the system. It shows that the exact solution takes much longer time than the proposed algorithm.

Haven established that the performance of the proposed algorithm is very close to that of the exact solution in terms of energy consumption and delay, in the next section, we will present an evaluation of the proposed algorithm against the SoA solution [145].

4.4.6.3 Comparison of the Proposed Solution for P4 to SoA

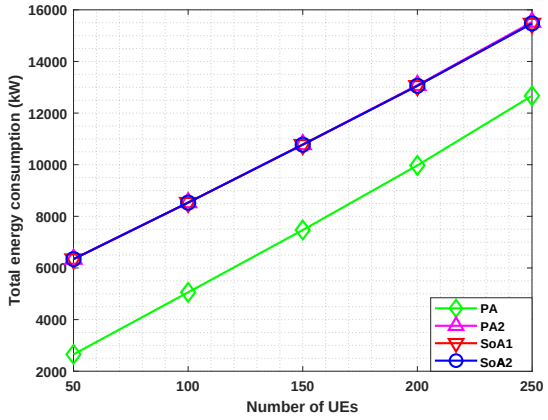
This subsection presents an evaluation of the performance of the proposed algorithm (PA) and its variant PA2 which employs the same association metrics without the power control mechanism against two distance based state-of-the-art schemes denoted by SoA1

and SoA2. Both SoA1 [145] and SoA2 associate the UEs, RUs and DUs to the closest RU, DU and CU, respectively. However, the former does not include the power control mechanism (i.e., $y_j = 1$, $\forall j \in \mathcal{R}$) while the later includes the power control mechanism. We will present the average results for 10000 simulation runs.

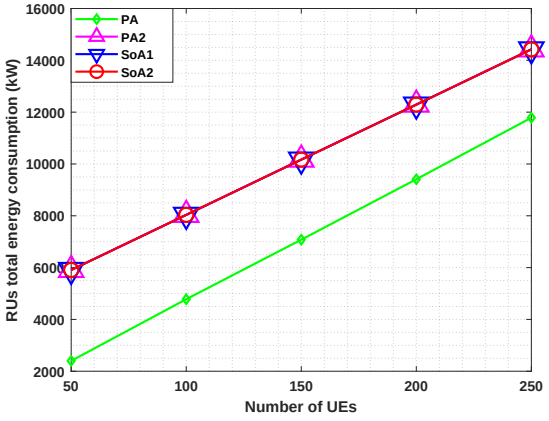
Figure 4.10a shows the total energy consumption of all the components (RUs, DUs and CUs) against the number of UEs U. It shows that when there are fewer UEs, the total energy consumption of the proposed algorithm PA is much lower than that of the SoA schemes (SoA1 and SoA2) and the variant of the proposed scheme (PA2) with this difference gradually reducing as the number of UEs increases. This improved performance is due to the employed association metrics that favour the association of UEs, RUs and DUs to RUs, DUs and CUs, with higher resource utilisation and achievable delay ratio, respectively. These metrics help in minimizing the number of active RUs, DUs and CUs, which when combined with the power control mechanism, that enables idle units to be put to sleep, lead to the achieved lower energy consumption. It can also be observed that the PA2, a variant of the PA without power control mechanism, together with the SoA scheme without power control (SoA1) and with power control (SoA2), consume the same and higher amount of energy than the PA. The percentage increase in energy consumption by the SoA solutions to the proposed solution at the highest and lowest considered UE traffic is respectively 62.4% and 21.3%. A more detail analysis of the total energy consumption for varying number of RUs and network area is presented in Figure 4.13

Similarly, Figure 4.10b, 4.10c and 4.10d show the RUs, DUs and CUs energy consumption, respectively. The results show that for all the components (RUs, DUs and CUs), the proposed algorithm PA outperforms the SoA schemes and its variant PA2.

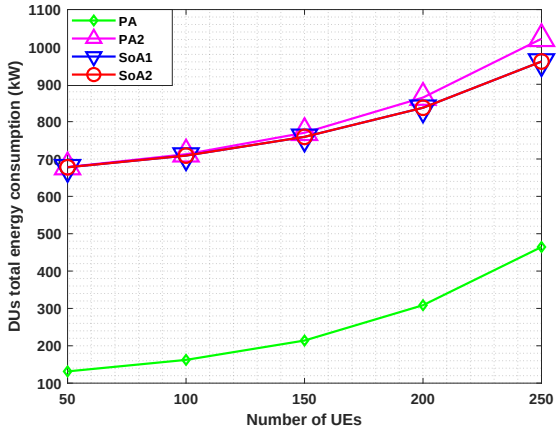
Figures 4.11 and 4.12 presents the total and per UE end to end delay incurred by all associated UEs. We can observe that while all the schemes incur nearly similar total delay when there are few UEs, the SoA schemes incur the same and lowest average per UE delay, with the proposed schemes (PA and PA2) incurring slightly higher average per UE delays both at low and high number of UEs. However, as the number of UEs increases, the PA and PA2 incur about 7.8% higher total delay than the SoA schemes. This increment in both the total and per UE delays can be attributed to the association metrics employed by the proposed schemes that seek to balance delay and energy consumption factor as against the SoA schemes whose metrics are strictly based on the shortest distance between the associating components. It is pertinent to reiterate that the delay incurred by the proposed schemes though slightly higher than the SoA, is still less than the maximum tolerable delay of the UEs.



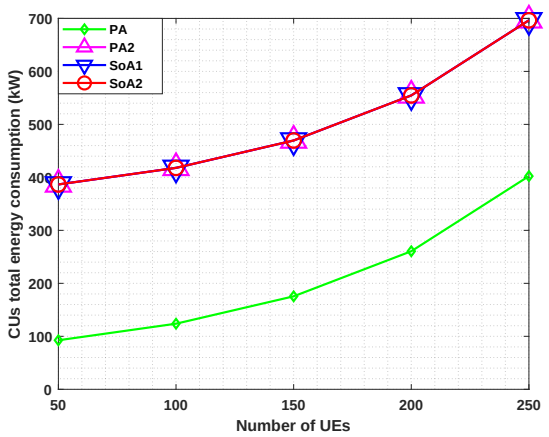
(a) Total energy consumption.



(b) RUs energy consumption



(c) DUs energy consumption.



(d) CUs energy consumption.

Figure 4.10: Total and component-wise energy consumption of the proposed algorithm for P4 and the SoA algorithms.

4.4.6.4 Impact of the Number of RUs

In this subsection, we we further evaluate the performance of the proposed algorithm (PA) and its variant PA2 the two SoA solutions under different number of RUs.

Figure 4.13 shows how the total energy consumption of the proposed and SoA schemes scale with respect to the number of RUs in the network for 250 UEs and different network area settings. The results show that across all the network areas ($1km^2$ to $4km^2$), the proposed scheme shows about 16.3% lower energy consumption against the SoA at the lowest considered number of RUs (Figure 4.13a), with this difference climbing up to 44.2% under high number of RUs (Figure 4.13c). Moreover, we observe that scaling the network area has no significant impact on the energy consumption of all the compared schemes. On the contrary, scaling up the number of RUs significantly impacts the total energy consumption. This is because the change in area size does not affect the total energy consumption given that the number of RUs is constant whereas, increasing the number of RUs allows more RUs to be idle and put to sleep mode when not in use.

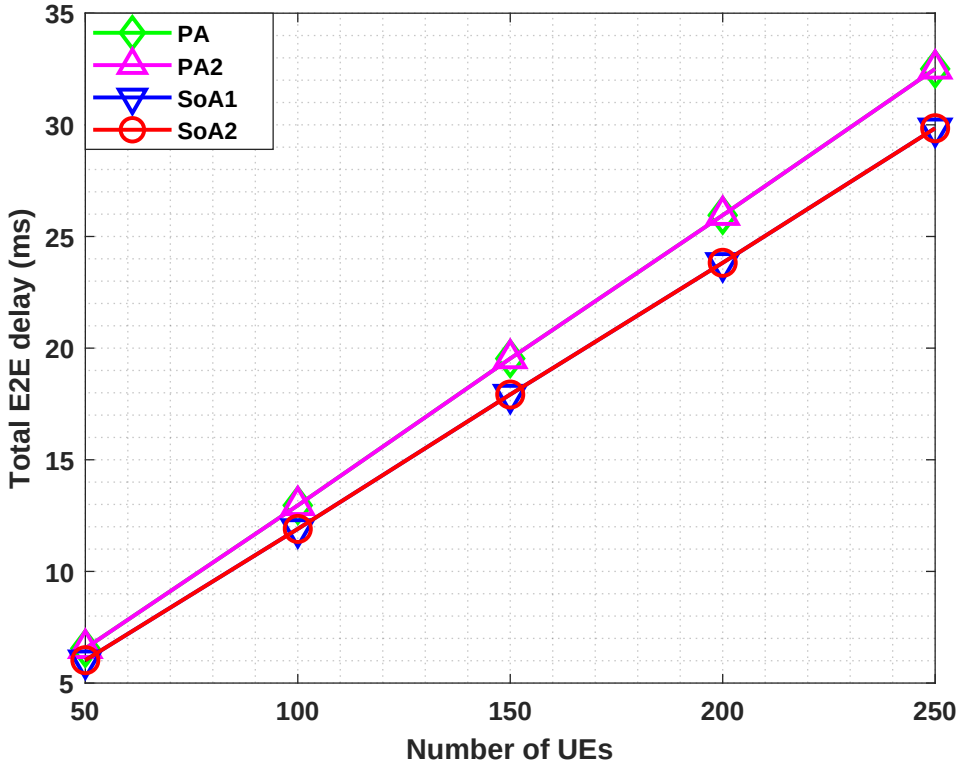


Figure 4.11: Comparison in terms of total end to end delay for the proposed algorithm for P4 and SoA algorithms.

In Figure 4.14, we present the total delay of the proposed and SoA schemes for different values of the number of RUs in the network and different network area settings. We present for 250 UEs. The results show that the proposed scheme incurs about 4.7%, 5.6%, 6.2% and 6.7% more delay than the SoA when we consider 1km^2, 2km^2, 3km^2 and 4km^2 network areas, respectively, under low number of RUs (Figure 4.14a), with this difference climbing up to 7.1%, 8.2%, 9.1% and 9.7% under high number of RUs (Figure 4.14c). That is, contrary to what we observed with the total energy consumption, both parameters impact the total delay experienced by the UEs. This is because, the smaller the number of RUs and area size, the lower the inter-RU distance and the higher the likelihood of choosing the same RUs and DUs for association, consequently reducing the difference in the delay that can be incurred for both schemes and vice versa. These observations hold with the average delay per UE as shown by Figure 4.15.

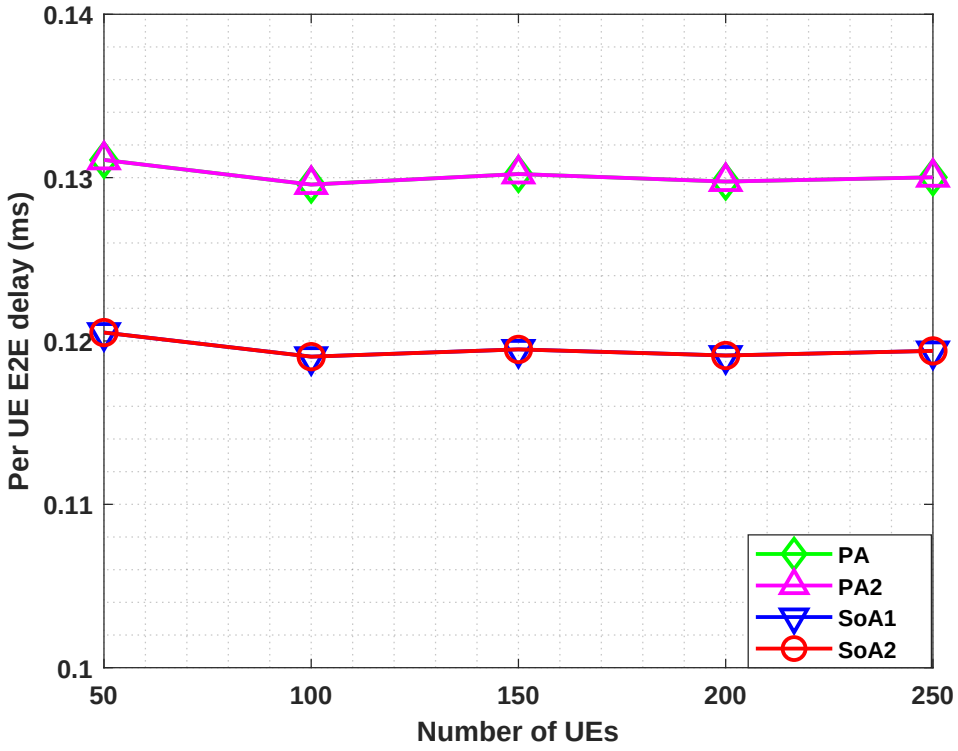
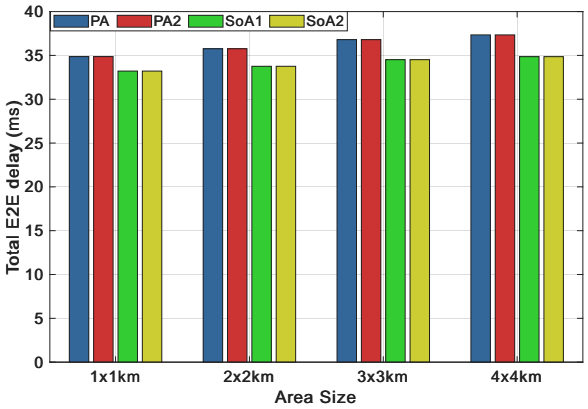
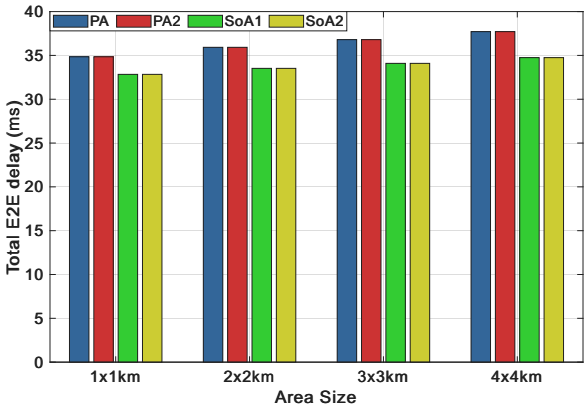


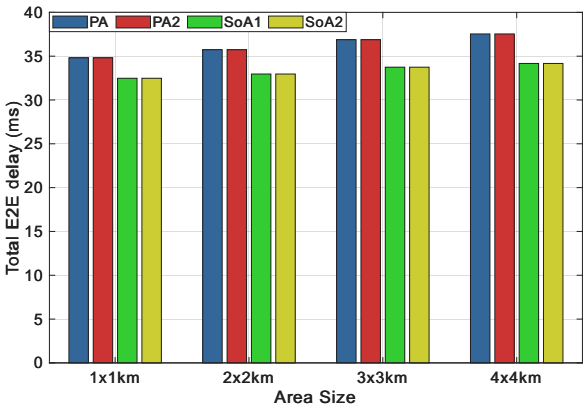
Figure 4.12: Comparison in terms of per UE end to end delay for the proposed algorithm for P4 and SoA algorithms.



(a) Total UEs delay for 15 RUs.

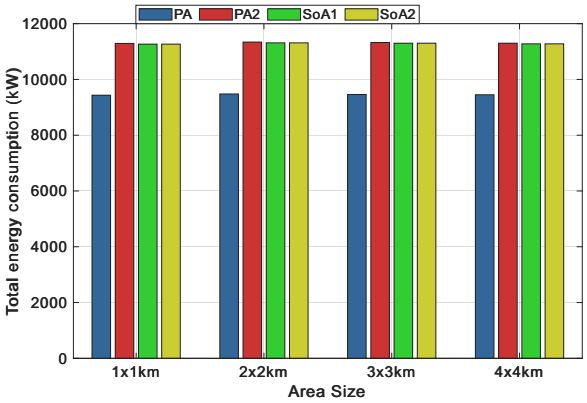


(b) Total UEs delay for 30 RUs.

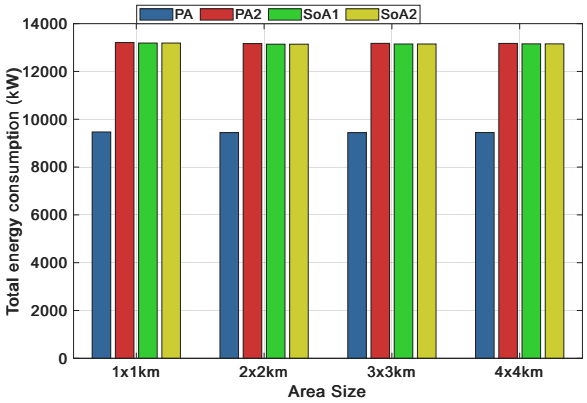


(c) Total UEs delay for 45 RUs.

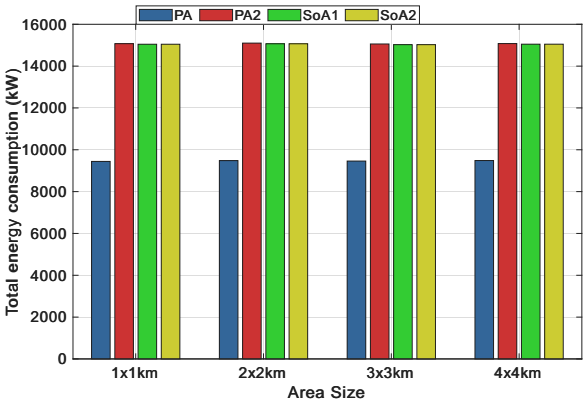
Figure 4.14: Proposed algorithm for P4 vs SoA algorithms in terms of total UEs delay for different network area and number of RUs setting.



(a) Total energy consumption for 15 RUs.

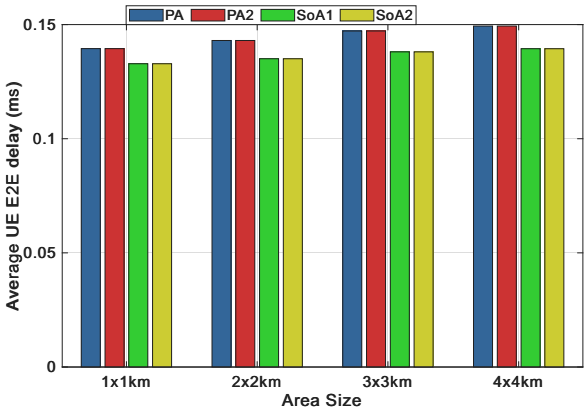


(b) Total energy consumption for 30 RUs.

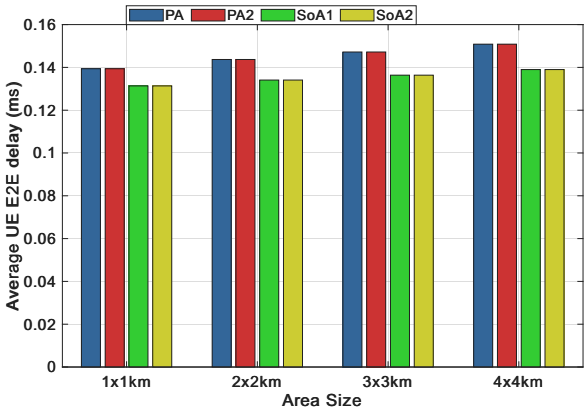


(c) Total energy consumption for 45 RUs.

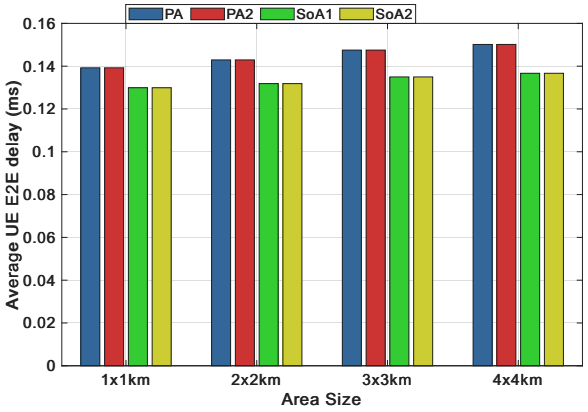
Figure 4.13: Proposed algorithm for P4 vs SoA algorithms in terms of total energy consumption for different network area and number of RUs setting.



(a) Average UE delay for 15 RUs.



(b) Average UE delay for 30 RUs.



(c) Average UE delay for 45 RUs.

Figure 4.15: Proposed algorithm for P4 vs SoA algorithms in terms of average delay per UE for different network area and number of RUs.

4.5 Chapter 4 Summary

This chapter presents a framework for addressing the user association and functions placement problem in O-RAN. We formulate and solve two open research problems: 1) energy-efficient and delay-aware UE association to RUs and 2) joint end-to-end association and function placement across RUs, DUs and CUs.

The core of our solutions is a novel user association metric for the UE to RU association and a multi-component association metric for the second end-to-end scenario that efficiently balances the conflicting goals of low latency and low energy consumption. We define a UE to RU association metric for the first solution and three associations metrics for UE to RU, RU to DU and DU to CU association respectively for the second solution. These metrics combined with a simple yet effective power control mechanism that enables RUs, DUs and CUs to enter the sleep mode when they have no load to process, allows our algorithms to significantly reduce the total energy consumption of the O-RAN system.

Extensive simulations demonstrate that both of our proposed solutions significantly

outperform the SoA solutions in terms of reduced energy consumption while slightly incurring higher delay but without violating the delay requirements of the UEs. Moreover, our proposed heuristic for the end to end scenario achieves performance close to the optimal solution while being computationally efficient. More importantly, both of the proposed solutions consistently outperform standard SoA approaches, offering substantial energy savings with only a marginal increase in delay. These solutions provide practical strategies for network operators to manage their O-RAN infrastructure efficiently, meeting the dual challenges of performance and sustainability.

5 Multi-Objective Deep Reinforcement Learning Based RAT Selection Solution for Vehicle-to-Infrastructure Communication

Contents

5.1 Introduction 99

5.2 Related Works 101

5.3 RAT Selection in Vehicular Communication 104

5.3.1 System Model (P5) 104

5.3.2 Problem Formulation (P5) 106

5.4 Proposed Algorithm for P5 107

5.5 Performance Evaluation of the Proposed Solution for P5 . . 111

5.5.1 Simulation Settings for P5 111

5.5.2 Comparison of Proposed Solution of P5 to SoA 113

5.5.3 Impact of Reward Function Weighting 114

5.6 Chapter Summary 117

5.1 Introduction

The rapid evolution of vehicular communication systems has greatly transformed the landscape of intelligent transportation systems (ITS). As vehicular networks mature, the complexity of communication requirements has grown exponentially, with diverse and stringent quality of service (QoS) requirements. This heterogeneous nature of vehicular-

to-everything (V2X) applications, coupled with the dynamic and highly mobile environment in which vehicles operate, has necessitated the development of multi-radio access technology (RAT) communication frameworks to meet these requirements [150].

Contemporary V2X ecosystems encompasses multiple RATs with distinct characteristics in terms of coverage, latency, throughput, reliability and energy consumption, including dedicated short range communications (DSRC), cellular V2X (C-V2X), fifth generation (5G) new radio (NR), WiFi and emerging satellite communication systems. However, the coexistence of multiple RATs presents unprecedented challenges in terms of coordination between the multiple technologies, optimal RAT selection, resource allocation and seamless handover management. The RAT selection problem in multi-RAT V2X networks is inherently complex due to several interdependent factors such as dynamic channel conditions and network topologies, heterogeneous QoS requirements, limited and variable availability of radio resources across different RATs [150].

While existing traditional approaches to RAT selection have provided foundational insights, they fail to capture the multi-dimensional nature of the RAT selection problem in modern V2X environments. Recent advances in machine learning (ML), particularly reinforcement learning (RL) and deep neural networks, have opened new avenues for intelligent RAT selection. However, the integration of learning-based approaches with real-time V2X requirements, the handling of incomplete information scenarios and the provision of performance guarantees remain significant research challenges.

This chapter of the thesis addresses the research questions in multi-RAT V2X communication networks by proposing a comprehensive framework for intelligent RAT selection that simultaneously considers multiple QoS requirements while minimizing the number of handovers with the following key contributions:

- We formulate the multi-RAT V2X selection problem as a multi-objective optimization problem to jointly minimize vehicle delay and number of handovers while maximising reliability.
- We propose a novel adaptive learning algorithm employs deep reinforcement learning (DRL) to enable real-time RAT selection decisions under uncertainty.
- We evaluate the performance of the proposed scheme against a state-of-the-art solution in terms of vehicle delay, reliability and number of successful handovers.

The rest of the chapter is organised as follows. Section 5.2 presents the related works, Section 5.3.1 presents the network model and problem formulation. Section 5.4 introduces the proposed algorithm while Section 5.5 presents the performance evaluation of the proposed algorithm against the state-of-the-art (SoA). Finally, Section 5.6 summarizes the chapter.

5.2 Related Works

In this section, we present a review of some related works on RAT selection in vehicular communications.

In [150], an intelligent rule-based algorithm for hybrid V2X wireless communication technology selection. The proposed system operates in two phases: a warm-up phase for environment scanning and an intelligent transmission phase for adaptive technology selection. The scheme leverages a decision tree to dynamically choose between three technologies—DSRC, C-V2X device-to-device direct communication interface (PC5) also referred to as sidelink communication and 5G user equipment (UE) to infrastructure cellular network interface (Uu)—based on real-time selection criteria such as latency, inter-vehicular distance, cooperative awareness message (CAM) density. Others are C-ITS message type (CAM only or CAM with decentralized environmental notification message, DENM) and vehicle speed. The first three criteria are based on the received CAMs and have to do with the connecting device while the last two are relevant to the V2X node on which the intelligent V2X technology selection decision is to be made. It is based on the three key assumptions: first that all V2X nodes support C-ITS services and transmit periodic CAMs. Second that short-range is only considered for low throughput data transfer of certain C-ITS messages while the third assumes that high throughput connections like infotainment and remote driving using video streaming use long-range Uu link. The hybrid scheme optimizes CAM transmission latency, packet delivery ratio (PDR) and reliability.

A service-oriented joint long short-term memory (LSTM) multi-criteria (SOLMC) RAT selection scheme is proposed in [151] to maximize overall network throughput. SOLMC employs an LSTM and RAT selection scheme. The LSTM prediction technique is employed as a network filter to reject RATs that offer the worst channel quality before initial RAT selection scheme is used for RAT selection. SOLMC also uses transmission capacity, connectivity time, data delivery cost and queue backlog as decision criteria RAT selection. The scheme maximizes average throughput and packet delivery rate.

The work in [152] introduces a dynamic coordination framework between IEEE 802.11p (DSRC) and LTE-V2X (including both PC5 sidelink and Uu interfaces). The framework also proposes communication profiles (CPs) that bundle RATs with specific operational modes, enabling adaptive selection based on real-time channel conditions and application requirements. Furthermore, this framework analyzes static and dynamic selection strategies using metrics like signal-to-noise-plus-interference ratio (SNIR) and channel busy ratio, providing a theoretical framework without extensive algorithmic validation or performance quantification.

A practical DRL solution for RAT selection is proposed in [153] for DSRC/LTE-V2X hybrid networks. The employed double deep Q-learning (DQN) algorithm defines a state space comprising six states defined as the measured SNIR of each RAT network, the calculated packet reception ratio of each RAT, the service's latency and reliability requirements. The DQN optimizes communication modes—single, redundant, or division—based on real-time channel quality metrics (SNIR and packet reception rate), application requirements (latency and reliability) and network state observations. The single communication mode uses only one communication link to transmit the message, as in legacy vehicular communication networks, the second communication mode (redundant mode) duplicates the message to achieve higher reliability while the third communication mode (division mode) sends the message through two RATs as two independent transmissions to enhance the transmission throughput. The scheme increased packet reception ratio (PRR) and resource utilization in platooning scenarios.

In [154], a context-aware task offloading mechanism (CAVTOMEC) is proposed to improve QoS. The CAVTOMEC proposes a framework that combines DSRC and LTE-V2X with multi-access edge computing (MEC) is proposed. CAVTOMEC also employs a three-tier priority system (high, indifferent, low priority) for task offloading that integrates: traffic classification, CAMs beacon-based location awareness and long-range vehicle-to-network (V2N) resource polling. Furthermore, this solution proposes two heuristic algorithms: one for real-time offloading decisions based on current resource availability and another incorporating queuing mechanisms for future resource estimation. The scheme reduces offloading delay and increases success rates for high-priority tasks.

The work in [155] proposes a slice-aware RAT selection mechanism that employs stochastic geometry, specifically Poisson line processes (PLP) and Poisson line Cox processes (PLCP), to model road-based small base statio (SBS) deployments and pedestrian UE performance. The proposed solution also employs a tractable mm-wave interference model that accounts for dominant interferers and vehicular blockages, as well as two-step association policy—where UEs first select a tier based on sub-6GHz power and then a RAT using a bias factor—enables dynamic resource allocation. The bias enables operators to balance reliability, coverage and throughput. The optimal RAT bias values vary with service requirements: low bias for ultra reliable low latency communication (uRLLC), prioritizing reliability and higher bias for enhanced mobile broadband (eMBB), maximizing throughput.

An adaptive network selection framework is proposed in [156] to dynamically choose between DSRC, LTE, or 5G-NR based on application requirements. The proposed scheme prioritizes low-latency connections for safety applications (e.g., collision avoidance) and high-bandwidth networks for infotainment services (e.g., video streaming). This solution considers bandwidth, power and SINR thresholds constraints. Thus, reducing the delay

for safety applications and throughput for infotainment services.

The work in [157] proposed a decentralized context-aware algorithm for heterogeneous V2V communications that dynamically selects the optimal RAT based on application requirements (e.g., throughput, latency) and real-time channel conditions, leveraging periodic context-sharing packets to balance load across RATs. This solution increases higher packet delivery ratios and lower latency while minimizing computational overhead and communication costs.

In [158], a QoS-aware RAT selection algorithm for hybrid vehicular networks (HVNs) is proposed that dynamically selects between IEEE 802.11p and LTE networks based on QoS requirements and network load while minimizing vertical handovers (VHOs). This solution dynamically switches between the RATs based on network load and application QoS requirements by employing a distributed radio resource management system that monitors network conditions and dynamically adjusts beaconing frequencies via a beaconing frequency adaptation (BFA) mechanism that uses predefined QoS parameters like initial beaconing frequency (bFreqInitial), frequency reduction factor (rTolerance) and beaconing frequency reduction factor (rFactor) to reduce network load and unnecessary VHOs. The scheme achieves fewer handovers, higher PDRs, lower latency and better throughput. However, the BFA mechanism's dependency on predefined QoS parameters may limit adaptability in highly dynamic environments.

The work in [159] investigates the performance of multi-RAT (LTE-Uu and PC5) solutions for V2X communication, focusing on improving reliability and reducing end-to-end latency for vehicular networks. This investigation compares traditional LTE-Uu-based V2X communication, which relies on cellular infrastructure, with direct PC5-based sidelink communication, highlighting the trade-offs between latency, reliability and resource efficiency. Furthermore, a multi-RAT diversity scheme is proposed that combines both LTE-Uu and PC5 transmissions to enhance reliability, particularly in challenging scenarios such as high vehicle density or extended communication ranges. The result indicates that while direct PC5 communication offers lower latency, LTE-Uu multicast improves reliability at longer distances. The multi-RAT approach improves reliability.

In [160], a comprehensive study on the application of machine learning algorithms for RAT selection in 5G non-standalone (NSA) networks, focusing on travelling users in public buses is presented. This solution attempts to efficiently manage network resources for mobile users traversing multiple base stations and leverages live signal measurements from 47 base stations, incorporating parameters such as geographical location, data rate, received signal strength indicator (RSSI) and bandwidth to train and evaluate three ML models—support vector machine (SVM), deep neural network (DNN) and eXtreme gradient boosting (XGBoost). The results demonstrate XGBoost achieves higher accuracy,

outperforming SVM and DNN.

A service aware RAT selection solution is proposed in [161] to improve reliability for traffic-efficiency service vehicles, delay for safety-critical service vehicles and throughput for infotainment service vehicles. This solution selects the DSRC RAT for safety-critical services, C-V2X for infotainment and chooses the RAT that offers the highest reliability for traffic-efficiency services. This solution minimizes delay, throughput and reliability for safety-critical, infotainment and traffic-efficiency services vehicles respectively. However, It may increase number of handovers as it will have to change RATs each time a vehicle's service changes.

5.3 RAT Selection in Vehicular Communication

In this section, we describe the system model and formulate the RAT selection problem as a multi-objective problem that seeks to jointly minimize the delay and number of handovers while maximizing the reliability for vehicles.

5.3.1 System Model (P5)

We consider a geographical area with vehicular communication infrastructure operating in the 5.9 GHz ITS band. The area is represented with coordinates (x, y) and operates over a time horizon divided into discrete time slots $\mathcal{T} = \{1, 2, \ldots, T\}$, where each slot $t \in \mathcal{T}$ has duration Δt (ms).

The communication infrastructure consists of two RATs: $\mathcal{J} = \{\text{DSRC}, \text{C-V2X}\}$, whose coverage area is noted as $C_j, \forall j \in \mathcal{J}$. DSRC is deployed through multiple road side units (RSUs) distributed across the geographical area and C-V2X is deployed through a single 5G next generation node B (gNodeB) providing area-wide coverage.

Let $\mathcal{R} = \{1, 2, \ldots, R\}$ denote the set of DSRC RSUs distributed across the area, where each RSU $r \in \mathcal{R}$ is positioned at coordinates (x_r, y_r) with coverage radius C_r (m). The C-V2X BS is positioned at coordinates $(x_{\text{BS}}, y_{\text{BS}})$ with coverage radius $C_{\text{C-V2X}}$ (m).

Each RAT $j \in \mathcal{J}$ is characterized by its bandwidth capacity B_j (Hz), maximum number of supported connections N_j and transmission power Q_j^{tx} (dBm). For DSRC, the total number of connections is $N_{\text{DSRC}} = \sum_{r \in \mathcal{R}} N_r$, where N_r is maximum connections that can be supported by RSU r. Let $\mathcal{V} = \{1, 2, \ldots, V\}$ denote the set of all vehicles in the considered area such that each vehicle $i \in \mathcal{V}$ is characterized by time-varying and time-invariant parameters. The time-invariant characteristics include the vehicle class $\kappa_i \in \mathcal{K} = \{\text{emergency}, \text{public transport}, \text{commercial}, \text{private}\}$. The time-varying characteristics at each time slot $t \in \mathcal{T}$ include the current position/coordinates $(x_i(t), y_i(t))$,

speed $v_i(t)$ (m/s), direction vector $\vec{d_i}(t) = (d_x(t), d_y(t))$ and active service type $\tau_i(t) \in \mathcal{S} = \{\text{safety-critical}, \text{traffic-efficiency}, \text{infotainment}\}$. The dynamic priority level $p_i(t) \in \{1, 2, \ldots, P\}$ (where 1 represents the highest priority) is determined as a function of both the vehicle class and the active service type.

Each vehicle's QoS requirements are service-dependent and vary with time slots based on the active service type. These requirements are defined in terms of maximum tolerable latency $D_i^{max}(t)$ (ms), minimum reliability $\lambda_i^{\min}(t)$ and minimum data rate $\nu_i^{\min}(t)$ (bps). Similarly, $D_{i,j}(t)$ and $\lambda_{i,j}(t)$ and $\nu_{i,j}(t)$, are the delay, reliability and data rate for vehicle i when connected to RAT j at time slot t, respectively. We define the estimated reliability for vehicle i on RAT j at time slot t as:

$$\lambda_{i,j}(t) = \lambda_j^{\text{base}} \cdot \left(1 - \xi_j \cdot \frac{N_j(t)}{N_j}\right), \quad \forall i \in \mathcal{V}, \forall j \in \mathcal{J}, \forall t \in \mathcal{T}, \tag{5.1}$$

where: ξ_j and λ_j^{base} are the degradation factor of RAT j with respect to its load and the RAT's base reliability while $N_j(t)$ and N_j are the current number of vehicles connected to RAT j at time slot t and the maximum possible connections for RAT j, respectively.

Similarly, we define the estimated delay for vehicle i on RAT j at time slot t as:

$$D_{i,j}(t) = \frac{\text{distance}_{i,j}(t)}{c} + D_{j,\text{proc}}^{\text{base}} + \left(\frac{N_j(t)}{N_j}\right) \times Q_j, \tag{5.2}$$

where: the first, second and third term on the right hand side account for the propagation, processing and queuing delays respectively. Similarly, $\text{distance}_{i,j}(t)$, c, and $D_{j,\text{proc}}^{\text{base}}$ denote the euclidean distance between vehicle i and serving node of RAT j at time t, speed of light and base processing delay of RAT j, respectively, while $N_j(t)$, N_j and Q_j are the current number of vehicles connected to RAT j at time t, the maximum possible connections for RAT j and the queuing sensitivity coefficient of RAT j, respectively.

The data rate $\nu_{i,j}(t)$ for vehicle $i \in \mathcal{V}$ connected to RAT $j \in \mathcal{J} = \{\text{DSRC}, \text{C-V2X}\}$ at time slot $t \in \mathcal{T}$ is given by:

$$\nu_{i,j}(t) = B_j \cdot \max\left(0, 1 - \frac{\text{distance}_{i,j}(t)}{C_j}\right) \cdot \max\left(0, 1 - \frac{N_j(t)}{N_j}\right), \tag{5.3}$$

where: B_j, $\text{distance}_{i,j}(t)$ and C_j denote the bandwidth capacity of RAT j, euclidean distance between vehicle i and serving node of RAT j at time t and coverage radius of RAT j, respectively; while $N_j(t)$ and N_j are the current number of vehicles connected to RAT j at time t and the maximum possible connections for RAT j, respectively.

Furthermore, we assume that in every time slot, each vehicle initiates the RAT switching procedure by sending a message containing its current profile to the selection orches-

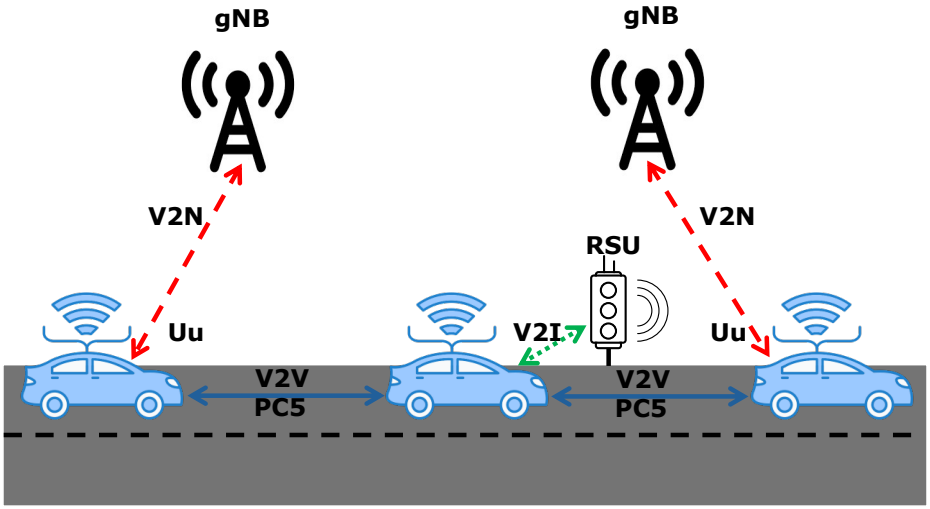


Figure 5.1: System Model (P5).

trator. The orchestrator then sends the profile information of all the available RATs that the vehicle is within coverage. A high level diagram of the scenario is captured in Figure 5.1.

5.3.2 Problem Formulation (P5)

In this section, we formulate the RAT selection problem as an optimization problem for the two-RAT scenario with DSRC and C-V2X.

Let $s_{i,j}(t) \in \{0,1\}$ denote the binary decision variable indicating whether vehicle i is assigned to RAT $j \in \mathcal{J}$ at time slot t. Let $h_i(t) \in \{0,1\}$ denote the binary variable indicating whether vehicle i undergoes a handover at time slot t. Next, we formulate our multi-objective function (P) that seeks to minimize the delay and handovers and maximize reliability.

$$\textbf{[P5]} \quad \min \sum_{t\in\mathcal{T}} \sum_{i\in\mathcal{V}} \Bigg(\alpha \sum_{j\in\mathcal{J}} s_{i,j}(t) D_{i,j}(t) - \beta \sum_{j\in\mathcal{J}} s_{i,j}(t) \lambda_{i,j}(t) + \gamma h_i(t) \Bigg) \tag{5.4}$$

Subject to:

$$\sum_{j\in\mathcal{J}} s_{i,j}(t) = 1, \quad \forall i \in \mathcal{V}, \forall t \in \mathcal{T} \tag{5.5}$$

$$\sum_{i\in\mathcal{V}} s_{i,j}(t) \leq N_j, \quad \forall t \in \mathcal{T}, \forall j \in \mathcal{J} \tag{5.6}$$

$$\sum_{j\in\mathcal{J}} s_{i,j}(t)D_{i,j}(t) \leq D_i^{max}(t), \quad \forall i \in \mathcal{V}, \forall t \in \mathcal{T} \tag{5.7}$$

$$\sum_{j\in\mathcal{J}} s_{i,j}(t)\lambda_{i,j}(t) \geq \lambda_i^{\min}(t), \quad \forall i \in \mathcal{V}, \forall t \in \mathcal{T} \tag{5.8}$$

$$\sum_{j\in\mathcal{J}} s_{i,j}(t)\nu_{i,j}(t) \geq \nu_i^{\min}(t), \quad \forall i \in \mathcal{V}, \forall t \in \mathcal{T} \tag{5.9}$$

$$s_{i,j}(t) \leq \mathbf{1}((x_i(t), y_i(t)) \in \mathcal{C}_j), \quad \forall i \in \mathcal{V}, \forall t \in \mathcal{T}, \forall j \in \mathcal{J} \tag{5.10}$$

$$h_i(t) \geq \sum_{j\in J} \left(s_{i,j}(t-1) \cdot (1 - s_{i,j}(t))\right), \quad \forall i \in \mathcal{V}, \forall t \in \mathcal{T}, t \geq 2 \tag{5.11}$$

$$h_i(1) = 0, \quad \forall i \in \mathcal{V} \tag{5.12}$$

$$s_{i,j}(t), h_i(t) \in \{0,1\}, \quad \forall i \in \mathcal{V}, \forall j \in \mathcal{J}, \forall t \in \mathcal{T}, t \geq 2 \tag{5.13}$$

The objective in (5.4) ensures a balance between vehicles' delay, reliability and the number of handovers, where α, β and γ are the delay, reliability and handover weights, respectively. The constraints in (5.5) ensure that each vehicle is assigned to exactly one RAT, (5.6) ensure that the number of vehicles assigned to each RAT cannot exceed its maximum number of connections. Constraints (5.7), (5.8) and (5.9) are the QoS (delay, reliability and data rate) constraints that ensure that the assigned RAT must meet the vehicle's QoS requirements. Similarly, (5.10) are the coverage constraints that ensure that a vehicle i is only assigned to a RAT if it is within its coverage area. Constraints (5.11) are the handover constraint that track handovers. Constraints (5.12) ensure that there is no handover in the first time slot. Constraints (5.13) define the binary decision variables for RAT assignment.

5.4 Proposed Algorithm for P5

In this section, we introduce a DRL based solution to address the multi-objective optimization problem of RAT selection in vehicular networks, leveraging the learning capabilities of DRL to automatically discover optimal RAT assignment policies that adapt to the dynamic and complex nature of vehicular communication environments while simultaneously minimizing delay, maximizing reliability and reducing handovers.

The core of our proposed solution is formulated as a Markov decision process (MDP) where an intelligent infrastructure agent learns to make optimal vehicle-RAT association decisions through continuous interaction with the vehicular network environment. Our DRL-based solution automatically adapts to varying network conditions and traffic patterns by learning from experience. We define in the following the set of possible states, actions and the reward.

States: the state in our approach encompasses comprehensive information about both individual vehicle characteristics and the overall network condition. That is, for each decision-making instance, the state vector captures vehicle-specific features including normalized (ratio of the parameter to its maximum possible value) position ($x_i^{\text{norm}}(t)$, $y_i^{\text{norm}}(t)$), speed ($v_i^{\text{norm}}(t)$), vehicle service type ($\tau_i^{\text{norm}}(t)$), priority level ($p_i^{\text{norm}}(t)$), direction ($\vec{d}_i^{\text{norm}}(t)$) and QoS requirements such as maximum tolerable latency ($D_i^{max,\text{norm}}(t)$), minimum data rate ($\nu_i^{\text{min,norm}}(t)$) and minimum reliability ($\lambda_i^{\text{min,norm}}(t)$) threshold requirements. Additionally, the state incorporates network-wide information including current RAT load ratios for DSRC ($N_{\text{DSRC}}^{\text{norm}}(t)$) and C-V2X ($N_{\text{C-V2X}}^{\text{norm}}(t)$) technologies, estimated network delays for DSRC ($D_{i,\text{DSRC}}^{\text{norm}}(t)$) and C-V2X ($D_{i,\text{C-V2X}}^{\text{norm}}(t)$) technologies, reliability for DSRC ($\lambda_{i,\text{DSRC}}^{\text{norm}}(t)$) and C-V2X ($\lambda_{i,\text{C-V2X}}^{\text{norm}}(t)$) technologies and data rates for both DSRC ($\nu_{i,\text{DSRC}}^{\text{norm}}(t)$) and C-V2X ($\nu_{i,\text{C-V2X}}^{\text{norm}}(t)$) technologies. The state vector $s_i(t) = [x_i^{\text{norm}}(t), v_i^{\text{norm}}(t), \tau_i^{\text{norm}}(t), p_i^{\text{norm}}(t), \vec{d}_i^{\text{norm}}(t), D_i^{max,\text{norm}}(t), \nu_i^{\text{min,norm}}(t), \lambda_i^{\text{min,norm}}(t), N_{\text{DSRC}}^{\text{norm}}(t), N_{\text{C-V2X}}^{\text{norm}}(t), D_{i,\text{DSRC}}^{\text{norm}}(t), D_{i,\text{C-V2X}}^{\text{norm}}(t), \lambda_{i,\text{DSRC}}^{\text{norm}}(t), \lambda_{i,\text{C-V2X}}^{\text{norm}}(t), \nu_{i,\text{DSRC}}^{\text{norm}}(t), \nu_{i,\text{C-V2X}}^{\text{norm}}(t)]$.

Actions: The action for vehicle i at time slot t is represented as $a_i(t) \in \mathcal{A} = \{0, 1\}$, where $a_i(t) = 0$ denotes assignment to DSRC and $a_i(t) = 1$ denotes assignment to C-V2X. The collective action vector for all vehicles at time t is given by $\mathbf{a}(t) = [a_1(t), a_2(t), \ldots, a_V(t)]^T \in \mathcal{A}^V$. To ensure feasibility, action masking is applied such that the valid action space for vehicle i is defined as $\mathcal{A}_i^{\text{valid}}(t) = \{a \in \mathcal{A} : (x_i(t), y_i(t)) \in \mathcal{C}_j, N_j(t) < N_j\}$, where j corresponds to the RAT selected by action a.

Reward ($r_i(t)$): The reward function for vehicle i at time slot t is derived from the objective function in (5.4) which is aimed at jointly minimizing delay, maximizing reliability and minimizing handovers. We achieve this reward by negating the objective to convert the minimization problem into a reward maximization problem suitable for reinforcement learning. It is formulated as:

$$r_i(t) = -\alpha \sum_{j \in \mathcal{J}} s_{i,j}(t) D_{i,j}(t) + \beta \sum_{j \in \mathcal{J}} s_{i,j}(t) \lambda_{i,j}(t) - \gamma h_i(t) \tag{5.14}$$

where α, β and γ are weighting coefficients for delay minimization, reliability maximization and handover minimization respectively. The assignment variable $s_{i,j}(t) \in \{0, 1\}$ indicates whether vehicle i is assigned to RAT j at time t based on the agent's action $a_i(t)$ while $h_i(t) \in \{0, 1\}$ indicates whether a handover occurred as a result of the action

differing from the previous assignment. The total system reward at time t is given by $R(t) = \sum_{i=1}^{V} r_i(t)$.

The proposed approach employs a priority-based sequential decision-making strategy that processes vehicles in order of their priority levels, with highest-priority vehicles making RAT selections first to ensures that critical vehicular services such as emergency and safety applications are prioritized compared to global optimization methods that attempt to solve for all vehicles simultaneously.

The DRL architecture utilizes a deep Q-network (DQN) enhanced with experience replay and target networks to handle the temporal correlations and high-dimensional nature of the vehicular network environment. The experience replay mechanism stores past decisions and their outcomes, enabling the agent to learn from historical patterns and break temporal correlations that could lead to unstable learning. Training incorporates an exploration-exploitation strategy that balances learning new policies with utilizing existing knowledge.

The algorithm workflow, as depicted in Algorithm 5, follows a systematic approach for each time slot. First, vehicles are sorted by priority (line 5). For each vehicle, the algorithm computes the comprehensive state vector incorporating both vehicle characteristics and current network conditions (line 8). The set of valid actions is determined based on coverage and capacity constraints (lines 9-14). If the set of valid actions is empty, no RAT selection is performed for that vehicle and it continues execution (lines 15-17). The action selection mechanism employs an epsilon-greedy policy that balances exploration and exploitation (lines 18-27), where the agent either explores random valid actions or selects the action with the highest predicted Q-value based on current policy.

After action execution and RAT assignment (lines 28-35), the algorithm evaluates the decision quality by computing individual rewards based on QoS satisfaction and handover occurrence using (5.4) (line 38). The experience tuple consisting of state, action, reward and next state is stored in the replay buffer for future learning (line 39). Model updates occur periodically when sufficient experiences are available, using mini-batch training to update the neural network parameters (lines 40-46). The target network is updated through soft updates to maintain training stability (lines 48-51).

The proposed DRL solution demonstrates superior adaptability compared to traditional approaches by continuously learning from network dynamics and automatically adjusting decision-making strategies based on observed outcomes. The infrastructure-centric design enables global optimization while the priority-based processing ensures that critical vehicular communications receive appropriate resource allocation, making it particularly suitable for realistic V2X communication scenarios.

Algorithm 5: Deep Reinforcement Learning based RAT Selection Algorithm for P5.

Input: $\mathcal{V}$, $\mathcal{J}$, $Q(s,a;\theta)$, $Q(s,a;\theta^-)$, ϵ, α, γ, N_j
Output: $s_{i,j}(t)$, $h_i(t)$, $\forall i \in \mathcal{V}, j \in \mathcal{J}$

Init θ randomly, $\theta^- = \theta$, $\mathcal{D} = \emptyset$, $handovers = 0$
for *episode* $e = 1$ *to* E **do**
  Init network state and vehicle positions
  for *time slot* $t = 1$ *to* T **do**
    $\mathcal{V}_{sorted} = \text{sort}(\mathcal{V}, \text{key} = p_i)$, $handovers = 0$
    for $i \in \mathcal{V}_{sorted}$ **do**
      $prev_assign = s_{i,j}(t-1)$
      Compute state
        $s_i(t) = [(x_i^{\text{norm}}(t), y_i^{\text{norm}}(t)), v_i^{\text{norm}}(t), \tau_i^{\text{norm}}(t), p_i^{\text{norm}}(t), \vec{d}_i^{\text{norm}}(t), D_i^{\text{max,norm}}(t), \nu_i^{\text{min,norm}}(t),$
        $\lambda_i^{\text{min,norm}}(t), N_{\text{DSRC}}^{\text{norm}}(t), N_{\text{C-V2X}}^{\text{norm}}(t), \vec{d}_i^{\text{norm}}(t) D_{i,\text{DSRC}}^{\text{norm}}(t),$
        $D_{i,\text{C-V2X}}^{\text{norm}}(t), \lambda_{i,\text{DSRC}}^{\text{norm}}(t), \lambda_{i,\text{C-V2X}}^{\text{norm}}(t), \nu_{i,\text{DSRC}}^{\text{norm}}(t), \nu_{i,\text{C-V2X}}^{\text{norm}}(t)]$
      $\mathcal{A}_i^{\text{valid}} = \emptyset$
      for $j \in \mathcal{J}$ **do**
        if $(x_i(t), y_i(t)) \in C_j \wedge N_j(t) < N_j$ **then**
          $\mathcal{A}_i^{\text{valid}} = \mathcal{A}_i^{\text{valid}} \cup \{j\}$
        end
      end
      if $\mathcal{A}_i^{valid} = \emptyset$ **then**
        $s_{i,j}(t) = 0, \forall j \in \mathcal{J}$; **continue**
      end
      if $random() < \epsilon$ **then**
        $a_i(t) = \text{random_choice}(\mathcal{A}_i^{valid})$
      else
        $q_vals = Q(s_i(t);\theta)$, $masked_q = -\infty \cdot \mathbf{1}$ `// otherwise selects optimal action based on Q-values for state` $s_i(t)$ `with policy` θ`. The mask ensures infeasible RATs (e.g., beyond` C_j`) are excluded`
        for $a \in \mathcal{A}_i^{valid}$ **do**
          $masked_q[a] = q_vals[a]$
        end
        $a_i(t) = \arg\max masked_q$
      end
      $j^* = a_i(t) + 1$
      $s_{i,j^*}(t) = 1$, $s_{i,k}(t) = 0, \forall k \neq j^*$
      if $prev_assign \neq NULL \wedge prev_assign \neq j^*$ **then**
        $h_i(t) = 1$, $handovers = handovers + 1$
      else
        $h_i(t) = 0$
      end
      $N_{j^*}(t) = N_{j^*}(t) + 1$
      Compute $D_{i,j^*}(t)$, $\lambda_{i,j^*}(t)$, $\nu_{i,j^*}(t)$, check against (5.7),(5.8) and (5.9) `// where` $D_{i,j^*}(t)$, $\lambda_{i,j^*}(t)$ `and` $\nu_{i,j^*}(t)$ `are defined as` (5.2), (5.1) `and` (5.3)
      Compute the reward ($r_i(t)$) based on (5.4)
      Compute $s_i(t+1)$; Store $(s_i(t), a_i(t), r_i(t), s_i(t+1), done) \rightarrow \mathcal{D}$
      if $|\mathcal{D}| \geq batch_size \wedge steps_done \bmod update_every = 0$ **then**
        Sample mini-batch from $\mathcal{D}$
        $m = r + \Gamma \max_{a'} Q(s', a'; \theta^-)$
        Minimize $L = \text{MSE}(Q(s,a;\theta), y)$ and update θ
        Apply gradient clipping with max norm 1.0
      end
      $\epsilon = \max(\epsilon_{min}, \epsilon \times \epsilon_{decay})$
    end
    if $steps_done \bmod target_update = 0$ **then**
      $\theta^- \leftarrow \tau\theta + (1-\tau)\theta^-$
    end
    $steps_done = steps_done + 1$
  end
end

5.5 Performance Evaluation of the Proposed Solution for P5

In this section, we present an evaluation of the performance of the proposed solution against a service-aware state-of-the-art algorithm [161]. First, we describe the simulation setup in Section 5.5.1 and present some results in Section 5.5.2.

5.5.1 Simulation Settings for P5

The simulation scenario consists of one C-V2X base station with a capacity of 100 MHz bandwidth and 500 maximum connections as well as 3 RSUs of 10MHz bandwidth and 200 maximum number of connections each. The RSUs are positioned at coordinates $(250, 0)$, $(500, 0)$ and $(750, 0)$ meters along a straight highway segment of length 1000 m, while the C-V2X station is centrally located at $(500, 0)$. The coverage radius of each DSRC mounted on a RSU is set to 400m while that of the 5G gNodeB (C-V2X) is 1000m/1km. Similarly, transmit power is set to $Q^{\text{tx}}_{\text{DSRC}} = 23$ dBm for DSRC and $Q^{\text{tx}}_{\text{C-V2X}} = 46$ dBm for C-V2X while the reliability degradation factors are $\xi_{\text{DSRC}} = 0.15$ and $\xi_{\text{C-V2X}} = 0.08$, with base reliability values $\lambda^{\text{base}}_{\text{DSRC}} = 0.99$ and $\lambda^{\text{base}}_{\text{C-V2X}} = 0.95$ while the base processing delay values are set to $D^{\text{base}}_{\text{DSRC,proc}}$ =2ms and $D^{\text{base}}_{\text{C-V2X,proc}}$ = 1ms.

Similarly, we consider $V \in 100, 200, 300, 400, 500$ vehicles of types emergency, public transport, commercial and private comprising 5%, 10%, 20% and 65% respectively, moving bidirectionally along the highway with velocities $v \in [60, 120]$ km/h and random velocity perturbations ($\pm$5% each time slot). These vehicles request services of type safety-critical, traffic-efficiency and infotainment with diverse quality of service requirements. The latency requirements for the safety traffic efficiency and infotainment services is set to 100, 200 and 300 ms, respectively, while their reliability 0.99, 0.95 and 0.90 and their data rate to 1, 2 and 5Mbps, respectively. Furthermore, we set the priorities for the different types of vehicles to $1, 2, 3, 4$ and that of the services (safety-critical, traffic-efficiency, infotainment) to $1, 2, 3$ in that order. Thus at time slot t, the priority of the vehicle i ($p_i(t)$) is set to the sum of its type and service class priorities.

The simulation runs for $T = 40$ time slots with duration $\Delta t = 100$ ms each. RAT selection is triggered every $\Delta d = 25$ m (25meters) of vehicle movement. The objective function weights are set to $\alpha = 0.5$ for delay, $\beta = 0.3$ for reliability and $\gamma = 0.2$ for handover. The DQN was implemented using the PyTorch framework. The network architecture is a multi-layer perceptron (MLP) consisting of two hidden layers with 256 and 128 neurons, respectively, with rectified linear unit (ReLU) activation functions and dropout layers ($\Psi = 0.2$) for regularization. The agent was trained online using an experience replay

buffer and the Adam optimizer. The simulation settings used are similar to those in [150], [151], [157] and summarized in Table 5.1 and 5.2.

Table 5.1: Simulation Parameters for P5.

Parameter	**Value(s)**
Highway length	1000 meters
Number of RSUs	3
RSU coverage radius	400 meters
C-V2X coverage radius	1000 meters
C-V2X position	Center (500m, 0)
RSU positions	((250m, 500m, 750m), 0)
Max DSRC connections per RSU	200
Max C-V2X connections	500
DSRC bandwidth	10 MHz
C-V2X bandwidth	100 MHz
DSRC transmit power	23 dBm
C-V2X transmit power	46 dBm
DSRC degradation factor	0.15
C-V2X degradation factor	0.08
Base DSRC reliability	0.99
Base C-V2X reliability	0.95
Vehicle speed range	60-120 km/h
Time slots	40
Time slot duration	100 ms
Trigger distance	25 meters
Number of vehicles tested	50, 75, 100, 125, 150
Delay weight (α)	0.5
Reliability weight (β)	0.3
Handover weight (γ)	0.2
Safety-critical latency requirement	100 ms
Traffic-efficiency latency requirement	200 ms
Infotainment latency requirement	500 ms
Safety-critical reliability requirement	0.99
Traffic-efficiency reliability requirement	0.95
Infotainment reliability requirement	0.90
Safety-critical data rate requirement	1 Mbps
Traffic-efficiency data rate requirement	2 Mbps
Infotainment data rate requirement	5 Mbps
$D^{\text{base}}_{\text{DSRC,proc}}$	2ms
$D^{\text{base}}_{\text{C-V2X,proc}}$	1ms

Table 5.2: DRL specific Parameters for P5.

Parameter	Value(s)
DRL hidden layer Size	128
DRL learning rate (η)	0.001
DRL batch size	32
DRL buffer size	10,000
DRL discount factor (Γ)	0.99
Final exploration rate (ϵ_{end})	0.05
ϵ_{decay}	0.98
$\epsilon_{\min}$	0.005
Target network update (τ)	0.005
Training update frequency	Every 4 steps

5.5.2 Comparison of Proposed Solution of P5 to SoA

In this subsection, we evaluate the performance of the proposed solution against the SoA solution [161]. The SoA solution selects the DSRC RAT for safety-critical services, C-V2X for infotainment and chooses the RAT that offers the highest reliability for traffic-efficiency services. In addition to the proposed DRL solution we refer to as DRL-DRH that aims to optimize delay, reliability and handovers, we also solve our problem including delay, reliability and handovers optimization using the Gurobi optimization solver, we denote the obtained solution by PP-EXACT. We evaluate the proposed DRL solution and its exact solution based on the objective defined in (5.4) against the SoA. The results presented are averages taken over 150 simulation runs.

Fig. 5.2 shows the total end-to-end delay incurred by vehicles against the number of vehicles V. It shows that the proposed DRL and its exact solution outperform the SoA solution, incurring the least delay. This is likely because while the SoA selects the DSRC RAT for vehicles with traffic-efficiency services to presumably minimize their delay, it does not consider current network conditions except for traffic-efficiency services for which it determines the best RAT based on current reliability conditions. On the contrary, the proposed DRL solution jointly considers delay, reliability and number of handovers in its decision making.

The results in Fig. 5.3 shows the average reliability achieved by all vehicles. It can be observed that the proposed solution and its exact solution achieve higher reliability compared to the SoA solution. This improved performance by the proposed solution is due to their attempt to jointly minimize delay and number of handovers while maximizing reliability as opposed to the SoA solution that only considers the reliability of one type of service in its decision making.

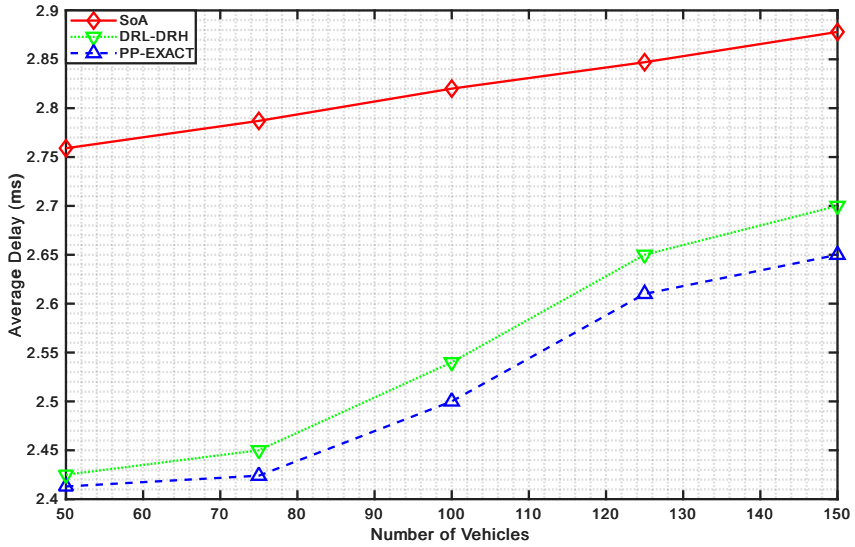


Figure 5.2: Proposed algorithm for P5 vs SoA algorithm in terms of total vehicle delay.

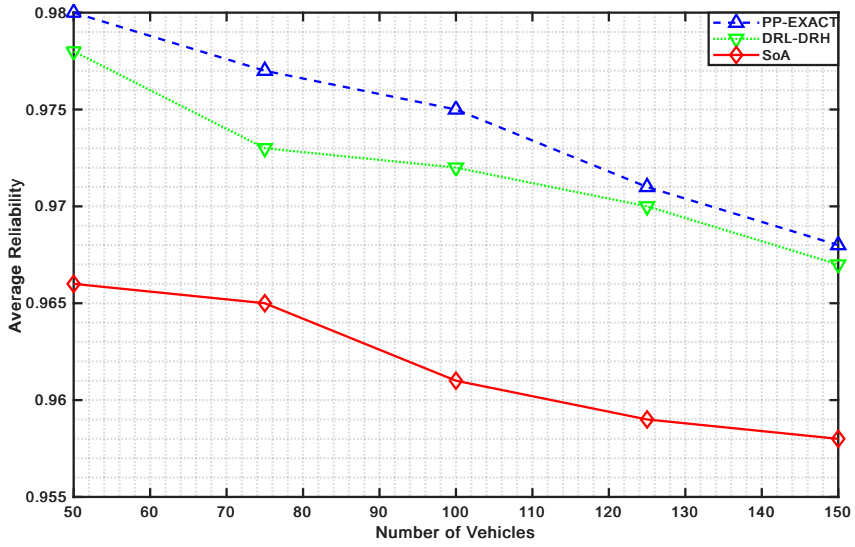


Figure 5.3: Proposed algorithm for P5 vs SoA algorithm in terms of achieved reliability.

Fig. 5.4 captures the total number of handovers performed by all vehicles. The result shows that the SoA incurs the highest number of handovers which is due to the proposed solutions jointly optimizing handover, delay and reliability consideration as opposed to the SoA solution that associates vehicles based on service class, increasing the chances of handovers from one decision point to another due to the changing service class of vehicles.

5.5.3 Impact of Reward Function Weighting

In this subsection, we evaluate the performance of the proposed solution by solving our problem using three DRL variants of the proposed DRL-DRH solution. The first variant DRL-DR unlike the DRL-DRH considers only the delay and reliability by setting the delay, reliability and handover weights to 0.5, 0.5 and 0 respectively. The second variant DRL-DH, considers only the delay and handovers by setting the delay, reliability and handover

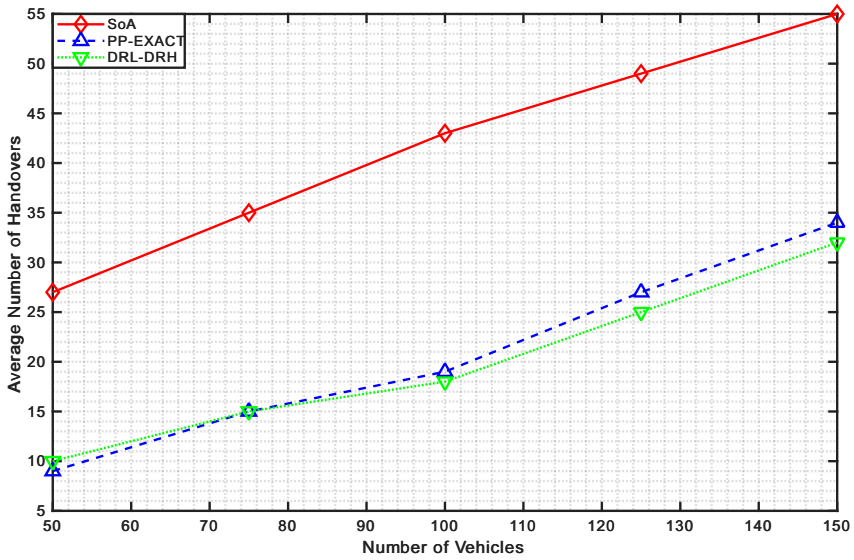


Figure 5.4: Proposed algorithm for P5 vs SoA algorithm in terms of number of handovers.

weights to 0.5, 0 and 0.5 respectively, while the third variant DRL-RH considers only the reliability and handovers by setting the delay, reliability and handover weights to 0, 0.5 and 0.5 respectively. Thus, we evaluate the proposed DRL solution based on the objective defined in (5.4) and its three variants. We present the average of the results obtained over 150 simulation runs.

Fig. 5.5 shows the total end-to-end delay incurred by vehicles against the number of vehicles V. We can observe that the variant of the proposed DRL solution (DRL-DR) achieves the minimum delay closely followed by the DRL-DH variants. This is due to the DRL-DR solution intelligently selecting RATs based on delay and reliability considerations unlike the proposed DRL-DRH solution that attempts to jointly optimize all three objectives. The DRL-RH variant incurs the highest delay among the proposed solutions, followed by the DRL-DH. This is because the DRL-RH assigns a weight of 0 to the delay component of the objective as opposed to the the other variants that assign non zero weights to the delay component.

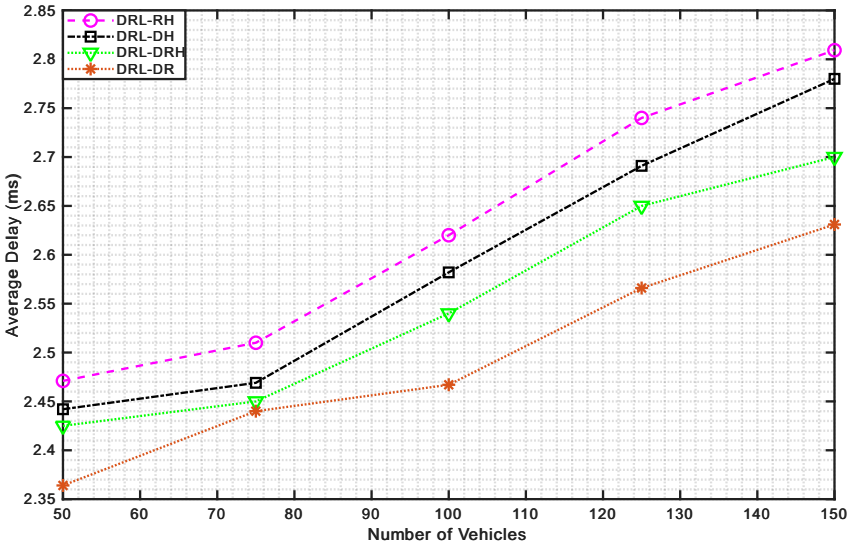


Figure 5.5: Total vehicle delay.

The results in Fig. 5.6 shows the average reliability achieved by all vehicles. It can be observed that the DRL-DR variant achieves the highest reliability with the DRL-DH variant that selects RATs based on delay and handover consideration without accounting for reliability achieving the lowest reliability among the compared variants. This performance by the DRL variant (DRL-DR) is because by considering only two of the three aspects of the objective, it is able to assign higher weights to the reliability component thereby leading to improved reliability performance.

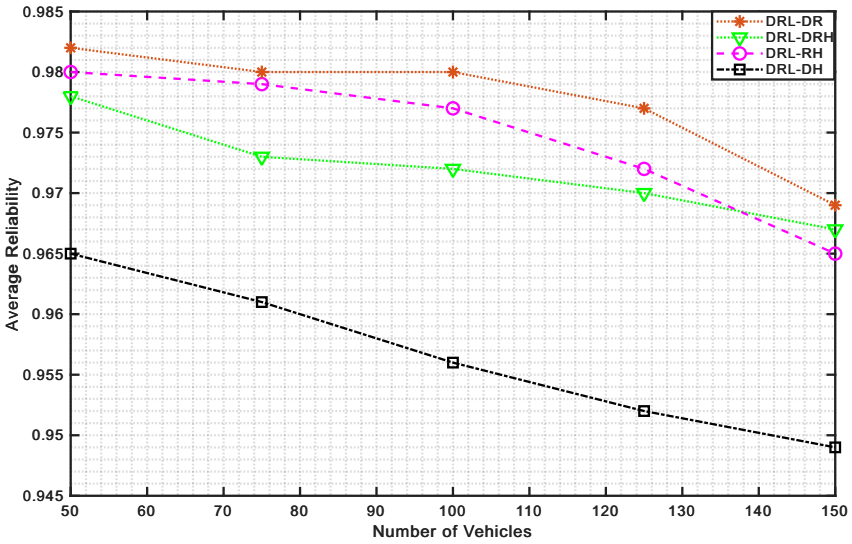


Figure 5.6: Achieved reliability.

Fig. 5.7 captures the total number of handovers performed by all vehicles. It can be observed that the DRL-DR variant incurs the highest number of handovers with the DRL-RH incurring the least number of handovers followed by the DRL-DH. This performance by the DRL variants (DRL-RH and DRL-DH) is because by considering only two of the three aspects of the objective, they are able to assign higher weights to the handover

component thereby leading to improved performance in this regard. In comparison, the DRL-DR and DRL-DRH that assign 0 and 0.2 weights to the handover components of the objective, respectively.

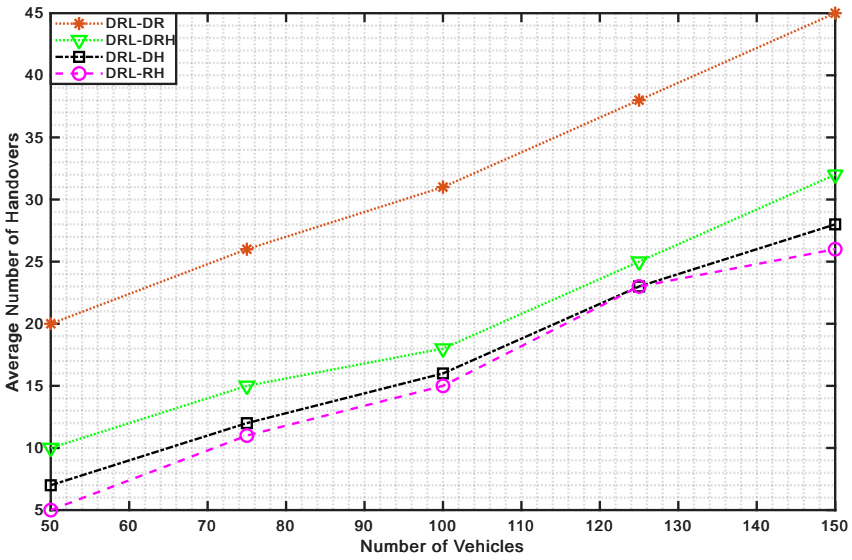


Figure 5.7: Comparison in terms of number of handovers.

5.6 Chapter Summary

In this chapter, we proposed a DRL-based RAT selection algorithm to jointly minimise delay and number of handovers while improving reliability. Our solution addresses the open research challenge of optimal RAT selection in multi-RAT vehicular communication networks, presenting a comprehensive framework that significantly advances the SoA in vehicular network optimization, consistently outperforming the SoA in terms of delay, reliability and number of handovers.

6 Conclusions and Perspectives

Contents

6.1 Conclusions 118

6.2 Perspectives 119

6.2.1 Towards Multi-level Network Slicing 120

6.2.2 Holistic O-RAN Optimization 120

6.2.3 Intelligent Vehicular Networking 120

6.2.4 Further Evaluation 120

6.3 Chapter Summary 121

6.1 Conclusions

In this thesis, we have investigated novel frameworks and algorithms for resource management in next-generation networks, focusing on three interrelated domains: network slicing admission control, delay- and energy-aware user association in O-RAN and multi-objective RAT selection for vehicular communications. Across these domains, the unifying theme has been the design of intelligent and adaptive solutions that reconcile competing objectives such as profit maximization, energy efficiency, delay minimization and reliability assurance under diverse and dynamic network conditions.

In Chapter 3, we introduced innovative slice admission control mechanisms for 5G and beyond systems. The first contribution employed overbooking and dynamic buyback processes, enabling infrastructure providers to increase revenue while reducing buyback costs without compromising service-level agreements. The second contribution extended the admission control framework to online scenarios, where slice requests arrive stochas-

tically and may renege if delayed. By leveraging dynamic priority and capacity pooling, the proposed solution achieved higher profit, improved utilization and a marginally better acceptance rate compared to state-of-the-art baselines.

Chapter 4 presented two O-RAN related contributions. The first contribution was a user association algorithm that simultaneously minimized system energy consumption and access network delay. This solution introduced a power control mechanism that dynamically puts idle RUs into sleep mode, thus, significantly reducing energy consumption at only marginal delay costs but without violating the delay requirements of the UEs. Building on this, the second contribution extended the problem to include end-to-end delay guarantees and functions placement across CU, DU and RU components. The proposed solution also yielded substantial energy savings across all network units (RUs, DUs and CUs) with a patery increase in delay compared to the state-of-the-art but without violating the delay requirements of users. These proposed solutions demonstrated the feasibility of energy-efficient and delay-guaranteed O-RAN operation, paving the way for greener and more sustainable next-generation RAN deployments.

Finally, Chapter 5 explored multi-RAT vehicular communications, a highly dynamic environment where vehicles must seamlessly select the most suitable RAT under stringent quality of service requirements. We formulated the RAT selection problem as a multi-objective optimization targeting delay, reliability and handover frequency. Simulation results showed that the proposed DRL-based approach outperformed state-of-the-art baselines, reducing handovers and improving both reliability and latency performance. This contribution marked a novel step towards intelligent vehicular connectivity management in heterogeneous wireless environments.

Overall, the thesis provided a comprehensive set of solutions for resource management in beyond 5G and future 6G networks. The contributions have not only advanced the state-of-the-art across multiple domains but also demonstrated the importance of multi-objective and adaptive approaches in balancing the conflicting demands.

6.2 Perspectives

While the contributions of this thesis have addressed open research challenges in slice admission control, O-RAN optimization and vehicular RAT selection, several open research directions can be pursued, some of which are discussed below.

6.2.1 Towards Multi-level Network Slicing

The slice admission control solutions presented in this thesis focus primarily on inter-slice resource allocation at the infrastructure provider level. A promising avenue for future work is the consideration for intra-slice as well joint optimization of inter-slice and intra-slice admission control, where not only acceptance decisions but also internal slice-level resource orchestration are coordinated. More so, the deployment of game theory to solve the multi-tenant and multi-stakeholder problem as well as intelligent resource management through the use of reinforcement learning and other learning solutions will be a great direction to consider.

6.2.2 Holistic O-RAN Optimization

The O-RAN research in this thesis has shown that energy savings and delay guarantees can be achieved via intelligent UE-to-RU association and functions placement. However, it can benefit from more holistic frameworks that integrate real-time traffic variability. Furthermore, online algorithms that make decisions under uncertainty, possibly by incorporating reinforcement learning or federated learning, can further enhance scalability and adaptability. The work can also benefit from the integration of multi-sleep mode operations.

6.2.3 Intelligent Vehicular Networking

The proposed DRL-based RAT selection framework marks an important step, but vehicular networks encompass multiple communication modes beyond vehicle-to-infrastructure (V2I), including vehicle-to-vehicle (V2V), vehicle-to-pedestrian (V2P) and vehicle-to-network (V2N). Extending the solution to support these scenarios in an integrated manner, while accounting for large-scale mobility is a fertile research direction. Furthermore, the integration of predictive mobility models with federated and distributed learning can further reduce decision latency and improve robustness under high mobility. Furthermore, given that standardization of RATs coexistence mechanisms is yet to be achieved, exploring this and interference management in such scenarios is worth exploring.

6.2.4 Further Evaluation

In this thesis, we largely evaluate our proposed solutions with simulations. The work can benefit from other evaluation methods. This can range from implementation on test beds to implementation on actual hardware.

6.3 Chapter Summary

The results and insights presented in this thesis contribute to the ongoing evolution of wireless networks towards intelligent, adaptive and sustainable systems. By addressing open research challenges in network slicing, O-RAN and vehicular communications, this work lays a foundation for future research at the intersection of optimization, artificial intelligence and next-generation communication architectures.

Bibliography

[1] Ibrahim Afolabi, Miloud Bagaa, Tarik Taleb, Marius Corici, and Adlen Ksentini. Network slicing and softwarization: A survey on principles, enabling technologies, and solutions. *IEEE Communications Surveys & Tutorials*, 20(3):2429–2453, 2018. 1, 13

[2] Jose Ordonez-Lucena, Pablo Ameigeiras, Diego Lopez, Juan J Ramos-Munoz, Javier Lorca, and Jesus Folgueira. Network slicing for 5g with sdn/nfv: Concepts, architectures, and challenges. *IEEE Communications Magazine*, 55(5):80–87, 2017. 1, 13

[3] Cisco Systems. Cisco annual internet report (2018–2023) white paper. Technical report, Cisco Systems, 2020. 1

[4] Petar Popovski, Jimmy J. Nielsen, Cedomir Stefanovic, Elisabeth de Carvalho, Erik Strom, Kasper F. Trillingsgaard, Alexandru-Sabin Bana, Dong Min Kim, Radoslaw Kotaba, Jihong Park, and Rene B. Sorensen. Wireless access for ultra-reliable low-latency communication: Principles and building blocks. *IEEE Network*, 32(2): 16–23, 2018. 2

[5] Shancang Li, Li Da Xu, and Shanshan Zhao. 5g internet of things: A survey. *Journal of Industrial Information Integration*, 10:1–9, 2018. 2

[6] Akihiro Nakao. Network slicing for customized mobile network services. *IEEE Communications Magazine*, 55(10):15–21, 2017. 2

[7] NM Mosharaf Kabir Chowdhury and Raouf Boutaba. Network virtualization: state of the art and research challenges. *IEEE Communications magazine*, 47(7):20–26, 2009. 2, 11

[8] Diego Kreutz, Fernando MV Ramos, Paulo Esteves Verissimo, Christian Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76, 2015. 2, 12

[9] Xenofon Foukas, Georgios Patounas, Ahmed Elmokashfi, and Mahesh K Marina. Network slicing in 5G: Survey and Challenges. *IEEE communications magazine*, 55 (5):94–100, 2017. 2, 12, 13

[10] Shunliang Zhang. Network slicing based 5g and future mobile networks: mobility, resource management, and challenges. *IEEE communications magazine*, 55(8):138–145, 2017. 2, 13

[11] Dario Bega, Marco Gramaglia, Albert Banchs, Vincenzo Sciancalepore, Konstantinos Samdanis, and Xavier Costa-Perez. Deepcog: Cognitive network management in sliced 5g networks with deep learning. *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 280–288, 2019. 3, 17

[12] Michele Polese, Leonardo Bonati, Salvatore D'oro, Stefano Basagni, and Tommaso Melodia. Understanding o-ran: Architecture, interfaces, algorithms, security, and research challenges. *IEEE Communications Surveys & Tutorials*, 25(2):1376–1411, 2023. 3, 18, 59

[13] Leonardo Bonati, Michele Polese, Salvatore D'oro, Stefano Basagni, and Tommaso Melodia. Open, programmable, and virtualized 5g radio access networks: State-of-the-art and the road ahead. *Computer Networks*, 182:107516, 2020. 3, 18

[14] Aleksandra Checko, Henrik L Christiansen, Ying Yan, Lara Scolari, Georgios Kardaras, Michael S Berger, and Lars Dittmann. Cloud ran for mobile networks—a technology overview. *IEEE Communications surveys & tutorials*, 17(1):405–426, 2015. 3, 19

[15] Lars MP Larsen, Aleksandra Checko, and Henrik L Christiansen. A survey of the functional splits proposed for 5g mobile crosshaul networks. *IEEE Communications Surveys & Tutorials*, 21(1):146–172, 2019. 3, 19, 20

[16] Congzheng Han, Tim Harrold, Simon Armour, Ioannis Krikidis, Stefan Videv, Peter M Grant, Harald Haas, John S Thompson, Izzat Ku, Cheng-Xiang Wang, et al. Green radio: radio techniques to enable energy-efficient wireless networks. *IEEE communications magazine*, 49(6):46–54, 2011. 3, 20

[17] Fabio Arena and Giovanni Pau. An overview of vehicular communications. *Future Internet*, 11(2):27, 2019. 4, 15

[18] Syed Adeel Ali Shah, Ejaz Ahmed, Muhammad Imran, and Sherali Zeadally. Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions. *IEEE communications surveys & tutorials*, 20(4):4075–4111, 2018. 4, 16, 21

[19] David Garcia-Roger, Edgar E. González, David Martín-Sacristán, and Jose F. Monserrat. V2x support in 3gpp specifications: From 4g to 5g and beyond. *IEEE Access*, 8:190946–190963, 2020. doi: 10.1109/ACCESS.2020.3028621. 4, 15

[20] Ejaz Ahmed, Abdullah Gani, Saeid Abolfazli, Liu Jie Yao, and Samee U Khan. A survey on vehicular task offloading: Classification, issues, and challenges. *Journal of Network and Computer Applications*, 58:333–349, 2015. 4, 17

[21] Alessandro Bazzi, Barbara M Masini, Alberto Zanella, and Ioannis Thibault. Beaconing from connected vehicles: Ieee 802.11 p vs lte-v2v. *IEEE Wireless Communications Letters*, 8(6):1647–1650, 2019. 4, 15

[22] Wei Yang Bryan Lim, Nguyen Cong Luong, Dinh Thai Hoang, Yutao Jiao, Ying-Chang Liang, Qiang Yang, Dusit Niyato, and Chunyan Miao. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(3):2031–2063, 2020. 4

[23] A. F. M. Shahen Shah. A survey from 1g to 5g including the advent of 6g: Architectures, multiple access techniques, and emerging technologies. In *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 1117–1123, 2022. doi: 10.1109/CCWC54503.2022.9720781. 8, 9

[24] Erik Dahlman, Stefan Parkvall, and Johan Skold. *5G NR: The next generation wireless access technology*. Academic Press, 2020. 8

[25] Christopher Cox. An introduction to lte: Lte, lte-advanced, sae and 4g mobile communications. *John Wiley & Sons*, 2012. 9

[26] Qazi Kamal Ud Din Arshad, Ahsan Ullah Kashif, and Ijaz Mansoor Quershi. A review on the evolution of cellular technologies. In *2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, pages 989–993, 2019. doi: 10.1109/IBCAST.2019.8667173. 9

[27] Harri Holma and Antti Toskala. *HSDPA/HSUPA for UMTS: High speed radio access for mobile communications*. John Wiley & Sons, 2007. 9

[28] Chetan Sambhajirao More and Navnath Tatyaba Markad. A systematic overview of high speed communication. In *2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, pages 1–9, 2023. doi: 10.1109/ACCAI58221.2023.10199876. 9

[29] Amitabh Ghosh, Nitin Mangalvedhe, Rapeepat Ratasuk, Bishwarup Mondal, Mark Cudak, Eugene Visotsky, Timothy A Thomas, Jeffrey G Andrews, Pengfei Xia, Han Shin Jo, et al. Lte-advanced: next-generation wireless broadband technology. *IEEE wireless communications*, 17(3):10–22, 2010. 9

[30] Balazs Bertenyi. 5g evolution: What's next? *IEEE Wireless Communications*, 28 (1):4–8, 2021. doi: 10.1109/MWC.2021.9363048. 9

[31] Jeffrey G Andrews, Stefano Buzzi, Wan Choi, Stephen V Hanly, Angel Lozano, Anthony CK Soong, and Jianzhong Charlie Zhang. What will 5g be? *IEEE Journal on selected areas in communications*, 32(6):1065–1082, 2014. 9

[32] Theodore S Rappaport, Shu Sun, Rimma Mayzus, Hang Zhao, Yaniv Azar, Kevin Wang, George N Wong, Jocelyn K Schulz, Mathew Samimi, and Felix Gutierrez Jr. Millimeter wave mobile communications for 5g cellular: It will work! *IEEE access*, 1:335–349, 2013. 9

[33] Haesik Kim. 5g vertical trials, use cases and scenarios. In *2022 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS)*, pages 81–86, 2022. doi: 10.1109/IoTaIS56727.2022.9975928. 9

[34] Mehdi Bennis, Merouane Debbah, and H Vincent Poor. Ultrareliable and low-latency wireless communication: Tail, risk, and scale. *Proceedings of the IEEE*, 106 (10):1834–1853, 2018. 10

[35] Olaonipekun Oluwafemi Erunkulu, Adamu Murtala Zungeru, Caspar K. Lebekwe, Modisa Mosalaosi, and Joseph M. Chuma. 5g mobile communication applications: A survey and comparison of use cases. *IEEE Access*, 9:97251–97295, 2021. doi: 10.1109/ACCESS.2021.3093213. 10

[36] Rapeepat Ratasuk, Benny Vejlgaard, Nitin Mangalvedhe, and Amitabh Ghosh. Nb-iot system for m2m communication. *IEEE Wireless Communications and Networking Conference*, pages 1–5, 2016. 10

[37] Salah Eddine Elayoubi, Mikael Fallgren, Panagiotis Spapis, Gerd Zimmermann, David Martín-Sacristán, Changqing Yang, Sébastien Jeux, Patrick Agyapong, Luis Campoy, Yinan Qi, and Shubhranshu Singh. 5g service requirements and operational use cases: Analysis and metis ii vision. In *2016 European Conference on Networks and Communications (EuCNC)*, pages 158–162, 2016. doi: 10.1109/EuCNC.2016.7561024. 10

[38] Shuping Dang, Omid Amin, Basem Shihada, and Mohamed-Slim Alouini. What should 6g be? *Nature Electronics*, 3(1):20–29, 2020. 10

[39] Faisal Tariq, Muhammad RA Khandaker, Kai-Kit Wong, Muhammad A Imran, Mehdi Bennis, and Mérouane Debbah. A speculative study on 6g. *IEEE Wireless Communications*, 27(4):118–125, 2020. 10

[40] Khaled B Letaief, Wei Chen, Yuming Shi, Jun Zhang, and Yi-Jun Angela Zhang. The roadmap to 6g: Ai empowered wireless networks. *IEEE Communications Magazine*, 57(8):84–90, 2019. 10

[41] Walid Saad, Mehdi Bennis, and Mingzhe Chen. A vision of 6g wireless systems: Applications, trends, technologies, and open research problems. *IEEE Network*, 34 (3):134–142, 2019. 10

[42] Haijian Yang, Arokiaswami Alphones, Zehui Xiong, Dusit Niyato, Jun Zhao, and Kaishun Wu. Artificial intelligence-enabled intelligent 6g networks. *IEEE Network*, 34(6):272–280, 2020. 10

[43] Emilio Calvanese Strinati, Sergio Barbarossa, Jose Luis Gonzalez-Jimenez, Dimitri Ktenas, Nicolas Cassiau, Laurent Maret, and Cedric Dehos. 6g: The next frontier: From holographic messaging to artificial intelligence using subterahertz and visible light communication. *IEEE Vehicular Technology Magazine*, 14(3):42–50, 2019. 10

[44] Chunxiao Zhang, Paul Patras, and Hamed Haddadi. Deep learning empowered radio resource management for 6g networks. *IEEE Wireless Communications*, 27 (6):114–120, 2020. 10

[45] Ian F Akyildiz, Josep Miquel Jornet, and Chong Han. Terahertz band: Next frontier for wireless communications. *Physical Communication*, 12:16–32, 2014. 10

[46] Theodore S Rappaport, Yunchou Xing, Ojas Kanhere, Shihao Ju, Arjuna Madanayake, Soumyajit Mandal, Ahmed Alkhateeb, and Georgios C Trichopoulos. Wireless communications and applications above 100 ghz: Opportunities and challenges for 6g and beyond. *IEEE Access*, 7:78729–78757, 2019. 10

[47] Qingqing Wu and Rui Zhang. Intelligent reflecting surface-aided wireless communications: A tutorial. *IEEE Transactions on Communications*, 69(5):3313–3351, 2021. 10

[48] Marco Di Renzo, Alessio Zappone, Mérouane Debbah, Mohamed-Slim Alouini, Chau Yuen, Julien de Rosny, and Sergei Tretyakov. Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and the road ahead. *IEEE Journal on Selected Areas in Communications*, 38(11): 2450–2525, 2020. 10

[49] Marco Giordani and Michele Zorzi. Non-terrestrial networks in the 6g era: Challenges and opportunities. *IEEE Network*, 35(2):244–251, 2021. 10

[50] Mostafa Zaman Chowdhury, Md Shahjalal, Shakil Ahmed, and Yeong Min Jang. 6g wireless communication systems: Applications, requirements, technologies, challenges, and research directions. *IEEE Open Journal of the Communications Society*, 1:957–975, 2020. 10

[51] Bo Yi, Xingwei Wang, Keqin Li, Sajal k. Das, and Min Huang. A comprehensive survey of network function virtualization. *Computer Networks*, 133:212–262, 2018. ISSN 1389-1286. doi: 10.1016/j.comnet.2018.01.021. 10

[52] Rashid Mijumbi, Joan Serrat, Juan-Luis Gorricho, Niels Bouten, Filip De Turck, and Raouf Boutaba. Network function virtualization: State-of-the-art and research challenges. *IEEE Communications Surveys & Tutorials*, 18(1):236–262, 2016. 11, 12

[53] Xin Li and Chen Qian. A survey of network function placement. In *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pages 948–953, 2016. doi: 10.1109/CCNC.2016.7444915. 11

[54] Thomas Anderson, Larry Peterson, Scott Shenker, and Jonathan Turner. Overcoming the internet impasse through virtualization. *Computer*, 38(4):34–41, 2005. 12

[55] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74, 2008. 12

[56] Yong Li and Min Chen. Software-defined network function virtualization: A survey. *IEEE Access*, 3:2542–2553, 2015. doi: 10.1109/ACCESS.2015.2499271. 12

[57] Ying Zhang. *Network Function Virtualization*, pages 37–65. John Wiley & Sons, 2018. doi: 10.1002/9781119390633.ch3. 12

[58] Hussein Hawilo, Abdallah Shami, Maysam Mirahmadi, and Rasool Asal. Nfv: state of the art, challenges, and implementation in next generation mobile networks (vepc). *IEEE Network*, 28(6):18–26, 2014. 12

[59] Michel S. Bonfim, Kelvin L. Dias, and Stenio F. L. Fernandes. Integrated nfv/sdn architectures: A systematic literature review. *ACM Comput. Surv.*, 51(6), February 2019. doi: 10.1145/3172866. 12

[60] Ahmed M Medhat, Tarik Taleb, Asma Elmangoush, Giuseppe A Carella, Stefan Covaci, and Thomas Magedanz. Service function chaining in next generation networks: State of the art and research challenges. *IEEE Communications Magazine*, 55(2):216–223, 2017. 12

[61] Prashant Subedi, Abeer Alsadoon, PWC Prasad, Sabih Rehman, Nabil Giweli, Muhammad Imran, and Samrah Arif. Network slicing: A next generation 5g perspective. *EURASIP Journal on Wireless Communications and Networking*, 2021 (1):102, 2021. 12

[62] Alcardo Alex Barakabitze, Arslan Ahmad, Rashid Mijumbi, and Andrew Hines. 5g network slicing using sdn and nfv: A survey of taxonomy, architectures and future challenges. *Comput. Netw.*, 167(C), February 2020. doi: 10.1016/j.comnet.2019.106984. 13

[63] Alexandros Kaloxylos. A survey and an analysis of network slicing in 5g networks. *IEEE Communications Standards Magazine*, 2(1):60–65, 2018. 13

[64] Merim Ashwell Marotta, Nicolas Kaminski, Ismael Gomez-Miguelez, Fabrizio Granelli, and Luiz A DaSilva. Business models for network slicing: A survey. *IEEE Communications Surveys & Tutorials*, 21(4):3613–3642, 2019. 13

[65] Peter Rost, Christian Mannweiler, Diomidis S Michalopoulos, Cinzia Sartori, Vincenzo Sciancalepore, Navid Sastry, Oliver Holland, Shreya Tayade, Binqiang Han, Dario Bega, et al. Network slicing to enable scalability and flexibility in 5g mobile networks. *IEEE Communications magazine*, 55(5):72–79, 2017. 14

[66] TV 3GPP. 3rd generation partnership project; technical specification group services and system aspects; study on lte support for vehicle to everything (v2x) services (release 14), 2015. 15

[67] Alfred Daniel, Anand Paul, Awais Ahmad, and Seungmin Rho. Cooperative intelligence of vehicles for intelligent transportation systems (its). *Wireless personal communications*, 87(2):461–484, 2016. 15

[68] John B Kenney. Dedicated short-range communications (dsrc) standards in the united states. *Proceedings of the IEEE*, 99(7):1162–1182, 2011. 15

[69] Waleed Albattah, Shabana Habib, Mohammed F Alsharekh, Muhammad Islam, Saleh Albahli, and Deshinta Arrova Dewi. An overview of the current challenges, trends, and protocols in the field of vehicular communication. *Electronics*, 11(21): 3581, 2022. 15

[70] Nurul H Mahmood, Okan Karatalay, Gilberto Berardinelli, Guillermo Pocovi, Klaus I Pedersen, and Preben Mogensen. Reliable wireless communications for cooperative intelligent transport systems. *IEEE Wireless Communications*, 25(6): 32–38, 2018. 15

[71] Umer Saeed, Khalid Haseeb, Syed Aziz Shah, and Adeel Ahmad. Pedestrian detection in vehicular networks: A literature survey. *Vehicular Communications*, 19: 100265, 2019. 15

[72] Mario H. Castañeda Garcia, Alejandro Molina-Galan, Mate Boban, Javier Gozalvez, Baldomero Coll-Perales, Taylan Şahin, and Apostolos Kousaridas. A tutorial on 5g nr v2x communications. *IEEE Communications Surveys & Tutorials*, 23(3):1972–2026, 2021. doi: 10.1109/COMST.2021.3057017. 16

[73] Sugih Jamin, Peter B Danzig, Scott J Shenker, and Lixia Zhang. A measurement-based admission control algorithm for integrated services packet networks. *IEEE/ACM Transactions on networking*, 5(1):56–70, 1997. 16

[74] Keith W Ross. *Multiservice loss models for broadband telecommunication networks*. Springer Science & Business Media, 1995. 16

[75] Frank P Kelly. Notes on effective bandwidths. *Stochastic networks: theory and applications*, 4:141–168, 1996. 16

[76] Steven Blake, David Black, Mark Carlson, Elwyn Davies, Zheng Wang, and Walter Weiss. *An architecture for differentiated services*. RFC 2475, 1998. 16

[77] Dong-Ho Kim, Vinod Sridhara, and Stephan Bohacek. Call admission control for cdma mobile communications systems supporting multimedia services. *IEEE Transactions on Wireless Communications*, 2(4):795–808, 2003. 17

[78] Sandeep Chinchali, Pan Hu, Tianshu Chu, Manu Sharma, Manu Bansal, Rakesh Misra, Marco Pavone, and Sachin Katti. Cellular network traffic scheduling with deep reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2018. 17

[79] Dusit Niyato and Ekram Hossain. Competitive spectrum sharing in cognitive radio networks: a dynamic game approach. *IEEE Transactions on Wireless Communications*, 7(7):2651–2660, 2008. 17

[80] Jeffrey G Andrews, Stefano Buzzi, Wan Choi, Stephen V Hanly, Angel Lozano, Anthony CK Soong, and Jianzhong Charlie Zhang. What will 5g be? *IEEE Journal on selected areas in communications*, 32(6):1065–1082, 2014. 17

[81] Enrique Stevens-Navarro, Yu Lin, and Vincent WS Wong. Efficient radio access technology selection and resource allocation in heterogeneous wireless networks. *IEEE Transactions on Vehicular Technology*, 62(4):1819–1827, 2013. 18

[82] Mohamed Lahby, Leghris Cherkaoui, and Abdellah Adib. A comprehensive survey of rat selection algorithms for heterogeneous networks. *International Journal of Wireless Networks and Broadband Technologies*, 2(3):1–35, 2013. 18

[83] Lei Lei, Zhanyu Zhong, Kai Lin, and Xuemin Chen. Machine learning for vehicular networks: Recent advances and application examples. *IEEE Vehicular Technology Magazine*, 13(2):94–101, 2018. 18

[84] Dusit Niyato and Ekram Hossain. Dynamics of network selection in heterogeneous wireless networks: An evolutionary game approach. *IEEE Transactions on Vehicular Technology*, 58(4):2008–2017, 2009. 18

[85] Gaurav Naik, Biplav Choudhury, and Jung-Min Park. Ieee 802.11 bd & 5g nr v2x: Evolution of radio access technologies for v2x communications. *IEEE Access*, 7: 70169–70184, 2019. 18

[86] Mohamed Ismail, Weihua Zhuang, Erchin Serpedin, and Khalid Qaraqe. Energy-efficient resource allocation for heterogeneous wireless networks. *IEEE Wireless Communications*, 20(1):74–81, 2013. 18

[87] Maciej Dryjański, Łukasz Kułacz, and Adrian Kliks. Open ran—radio access network evolution, benefits and market trends. *Applied Sciences*, 11(1):408, 2021. 18

[88] Sepehr Niknam, Abhishek Roy, Harpreet S Dhillon, Sukhdeep Singh, Rakesh Banerji, Jeffrey H Reed, Navrati Saxena, and Seongjoon Yoon. Intelligent o-ran for beyond 5g and 6g wireless networks. *IEEE Access*, 10:33472–33488, 2022. 19

[89] Chungang Yang, Jiandong Li, Mohsen Guizani, Alagan Anpalagan, and Maged Elkashlan. Resource allocation in cloud radio access networks: A survey. *IEEE Communications Surveys & Tutorials*, 20(3):2428–2468, 2018. 20, 21

[90] Josep Xavier Salvat, Lanfranco Zanzi, Andres Garcia-Saavedra, Vincenzo Sciancalepore, and Xavier Costa-Perez. Overbooking network slices through yield-driven end-to-end orchestration. In *Proceedings of the 14th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '18, page 353–365. Association for Computing Machinery(ACM), 2018. doi: 10.1145/3281411.3281435. 21

[91] Mamta Agiwal, Abhishek Roy, and Navrati Saxena. Next Generation 5G Wireless Networks: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials*, 18(3):1617–1655, 2016. doi: 10.1109/COMST.2016.2532458. 23

[92] Jeffrey G. Andrews, Stefano Buzzi, Wan Choi, Stephen V. Hanly, Angel Lozano, Anthony C. K. Soong, and Jianzhong Charlie Zhang. What Will 5G Be? *IEEE Journal on Selected Areas in Communications*, 32(6):1065–1082, 2014. doi: 10.1109/JSAC.2014.2328098. 23

[93] Cisco Systems. Cisco visual networking index: global mobile data traffic forecast update, 2017–2022. *Cisco Visual Networking Index (VNI) Global Mobile Data Traffic Forecast Update*, 2019. 23

[94] Pablo Caballero, Albert Banchs, Gustavo de Veciana, Xavier Costa-Pérez, and Arturo Azcorra. Network Slicing for Guaranteed Rate Services: Admission Control and Resource Allocation Games. *IEEE Transactions on Wireless Communications*, 17(10):6419–6432, 2018. doi: 10.1109/TWC.2018.2859918. 23

[95] Muhammad Rehan Raza, Carlos Natalino, Peter Öhlen, Lena Wosinska, and Paolo Monti. A Slice Admission Policy based on Reinforcement Learning for a 5G Flexible RAN. In *2018 European Conference on Optical Communication (ECOC)*, pages 1–3. IEEE, 2018. 23, 24, 27, 34

[96] Borja Bordel, Ramón Alcarria, Diego Sánchez-de Rivera, and Álvaro Sánchez. An Inter-slice Management Solution for Future Virtualization-based 5G Systems. In *Advanced Information Networking and Applications: Proceedings of the 33rd International Conference on Advanced Information Networking and Applications (AINA-2019) 33*, pages 1059–1070. Springer, 2020. 23, 42

[97] Mourice O Ojijo and Olabisi E Falowo. A Survey on Slice Admission Control Strategies and Optimization Schemes in 5G Network. *IEEE Access*, 8:14977–14990, 2020. 23, 24, 42

[98] Ahmed El-Mekkawi, Xavier Hesselbach, and Jose Ramon Piney. A Novel Admission Control Scheme for Network Slicing Based on Squatting and Kicking Strategies. *Zaragoza (España)*, 22(24), 2019. 23, 24, 26, 30, 34

[99] Quang-Trung Luu, Sylvaine Kerboeuf, and Michel Kieffer. Admission Control and Resource Reservation for Prioritized Slice Requests With Guaranteed SLA under uncertainties. *IEEE Transactions on Network and Service Management*, 19(3):3136–3153, 2022. doi: 10.1109/TNSM.2022.3160352. 23

[100] Xueli An, Chan Zhou, Riccardo Trivisonno, Riccardo Guerzoni, Alexandros Kaloxylos, David Soldani, and Artur Hecker. On End to End Network Slicing for 5G Communication Systems. *Transactions on Emerging Telecommunications Technologies*, 28(4):e3058, 2017. 23

[101] Dario Bega, Marco Gramaglia, Albert Banchs, Vincenzo Sciancalepore, Konstantinos Samdanis, and Xavier Costa-Perez. Optimising 5G Infrastructure Markets: The Business of Network Slicing. In *IEEE INFOCOM 2017-IEEE conference on computer communications*, pages 1–9. IEEE, 2017. 24, 26, 34

[102] Bin Han, Vincenzo Sciancalepore, Di Feng, Xavier Costa-Perez, and Hans D Schotten. A Utility-driven Multi-queue Admission Control Solution for Network Slicing. In *IEEE INFOCOM 2019-IEEE conference on computer communications*, pages 55–63. IEEE, 2019. 24, 28, 34

[103] Kiana Noroozi, Mohammad Karimzadeh-Farshbafan, and Vahid Shah-Mansouri. Service Admission Control for 5G Mobile Networks with RAN and Core Slicing. In *2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2019. 24, 28, 34

[104] Bahador Bakhshi, Josep Mangues-Bafalluy, and Jorge Baranda. R-learning-based admission control for service federation in multi-domain 5g networks. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2021. 24, 26, 30, 40

[105] Nabhasmita Sen et al. Intelligent Admission and Placement of O-RAN Slices Using Deep Reinforcement Learning. In *2022 IEEE 8th International Conference on Network Softwarization (NetSoft)*, pages 307–311. IEEE, 2022. 24, 25, 26, 31, 33, 34, 40

[106] Amal Kammoun, Nabil Tabbane, Gladys Diaz, and Nadjib Achir. Admission Control Algorithm for Network Slicing Management in SDN-NFV Environment. In *2018 6th international conference on multimedia computing and systems (ICMCS)*, pages 1–6. IEEE, 2018. 24, 27

[107] Kammoun, Amal and Tabbane, Nabil and Diaz, Gladys and Dandoush, Abdulhalim and Achir, Nadjib. End-to-end efficient heuristic algorithm for 5g network slicing. In *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, pages 386–392. IEEE, 2018. 24, 28

[108] Amal Kammoun, Nabil Tabbane, Gladys Diaz, Nadjib Achir, and Abdulhalim Dandoush. Proactive Network Slices Management Algorithm Based on Fuzzy Logic System and Support Vector Regression Model. In *Advances on Broad-Band Wireless*

Computing, Communication and Applications: Proceedings of the 14th International Conference on Broad-Band Wireless Computing, Communication and Applications (BWCCA-2019) 14, pages 386–397. Springer, 2020. 24, 29, 34

[109] Sihem Bakri, Bouziane Brik, and Adlen Ksentini. On Using Reinforcement Learning for Network Slice Admission Control in 5G: Offline vs. Online. *International Journal of Communication Systems*, 34(7):e4757, 2021. 24, 29, 34

[110] Shivani Saxena and Krishna M Sivalingam. DRL-Based Slice Admission Using Overbooking in 5G Networks. *IEEE Open Journal of the Communications Society*, 4:29–45, 2022. 25, 26, 32, 33, 34, 40, 42

[111] Phudit Ampririt, Seiji Ohara, Ermioni Qafzezi, Makoto Ikeda, Keita Matsuo, and Leonard Barolli. An Integrated Fuzzy-based Admission Control System (IFACS) for 5G Wireless Networks: Its implementation and performance evaluation. *Internet of Things*, 13:100351, 2021. 25

[112] Phudit Ampririt, Seiji Ohara, Yi Liu, Makoto Ikeda, Hiroshi Maeda, and Leonard Barolli. A Fuzzy-based System for Admission Control in 5G Wireless Networks Considering Software-defined Network Approach. In *Advances in Internet, Data and Web Technologies: The 8th International Conference on Emerging Internet, Data and Web Technologies (EIDWT-2020)*, pages 73–81. Springer, 2020. 25

[113] Victor C. Ferreira, H. H. Esmat, Beatriz Lorenzo, Sandip Kundu, and França Felipe M. G. Reinforcement Learning based Multi-Attribute Slice Admission Control for Next-Generation Networks in a Dynamic Pricing Environment. In *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, pages 1–5, 2022. doi: 10.1109/VTC2022-Spring54318.2022.9860729. 26, 30, 40

[114] Miao Dai, Long Luo, Jing Ren, Hongfang Yu, and Gang Sun. PSACCF: Prioritized Online Slice Admission Control Considering Fairness in 5G/B5G Networks. *IEEE Transactions on Network Science and Engineering*, 9(6):4101–4114, 2022. 26, 30, 33, 40, 51, 52

[115] Weisheng Xie, Jiafeng Zhu, Changcheng Huang, Min Luo, and Wu Chou. Network Virtualization with Dynamic Resource Pooling and Trading Mechanism. In *2014 IEEE Global Communications Conference*, pages 1829–1835. IEEE, 2014. 26

[116] Jesutofunmi Ajayi, Antonio Di Maio, Torsten Braun, and Dimitrios Xenakis. An online multi-dimensional knapsack approach for slice admission control. In *2023 IEEE 20th Consumer Communications & Networking Conference (CCNC)*, pages 152–157, 2023. doi: 10.1109/CCNC51644.2023.10060460. 32

[117] Weiwei Jiang, Yafeng Zhan, Guanming Zeng, and Jianhua Lu. Probabilistic-forecasting-based Admission Control for Network Slicing in Software-defined Networks. *IEEE Internet of Things Journal*, 9(15):14030–14047, 2022. 34

[118] Muhammad Sulaiman, Arash Moayyedi, Mahdieh Ahmadi, Mohammad A Salahuddin, Raouf Boutaba, and Aladdin Saleh. Coordinated Slicing and Admission Control Using Multi-agent Deep Reinforcement Learning. *IEEE Transactions on Network and Service Management*, 2022. 34

[119] Md Ariful Haque and Vassilka Kirova. 5G Network Slice Admission Control Using Optimization and Reinforcement Learning. In *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 854–859. IEEE, 2022. 34

[120] Zhouxiang Wu, Genya Ishigaki, Riti Gour, and Jason P Jue. A Reinforcement Learning-based Admission Control Strategy for Elastic Network Slices. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pages 01–06. IEEE, 2021. 34

[121] William F. Villota-Jacome, Oscar Mauricio Caicedo Rendon, and Nelson L. S. da Fonseca. Admission Control for 5G Core Network Slicing Based on Deep Reinforcement Learning. *IEEE Systems Journal*, 16(3):4686–4697, 2022. doi: 10.1109/JSYST.2022.3172658. 34

[122] Ciro D'apice, Maria Pia D'arienzo, Alexander Dudin, and Rosanna Manzo. Admission control in priority queueing system with servers reservation and temporal blocking admission of low priority users. *IEEE Access*, 11:44425–44443, 2023. doi: 10.1109/ACCESS.2023.3273148. 51, 52

[123] Gordon Owusu Boateng, Guolin Sun, Daniel Ayepah Mensah, Daniel Mawunyo Doe, Ruijie Ou, and Guisong Liu. Consortium blockchain-based spectrum trading for network slicing in 5g ran: A multi-agent deep reinforcement learning approach. *IEEE Transactions on Mobile Computing*, 22(10):5801–5815, 2023. doi: 10.1109/TMC.2022.3190449. 51

[124] Guolin Sun, Daniel Ayepah-Mensah, Gordon Owusu Boateng, Noble Arden Kuadey, Mohamed Basher Omer, and Guisong Liu. Holistic roadmap of trends in radio access network slicing: A survey. *IEEE Communications Magazine*, 61(12):118–124, 2023. doi: 10.1109/MCOM.001.2300005. 59

[125] Petar Popovski, Čedomir Stefanović, Jimmy J. Nielsen, Elisabeth de Carvalho, Marko Angjelichinoski, Kasper F. Trillingsgaard, and Alexandru-Sabin Bana. Wireless access in ultra-reliable low-latency communication (urllc). *IEEE Transactions*

on Communications, 67(8):5783–5801, 2019. doi: 10.1109/TCOMM.2019.2914652. 59

[126] Wissal Attaoui, Essaid Sabir, Halima Elbiaze, and Mohsen Guizani. Vnf and cnf placement in 5g: Recent advances and future trends. *IEEE Transactions on Network and Service Management*, 20(4):4698–4733, 2023. doi: 10.1109/TNSM.2023.3264005. 59

[127] SK Telecom. Green mobile network: Energy saving efforts by sk telecom and ntt docomo. *White Paper, Feb*, 2023. 59, 69

[128] Attai Ibrahim Abubakar, Oluwakayode Onireti, Yusuf Sambo, Lei Zhang, G. K. Ragesh, and Muhammad Ali Imran. Energy efficiency of open radio access network: A survey. In *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, pages 1–7, 2023. doi: 10.1109/VTC2023-Spring57618.2023.10200477. 59

[129] R. Joda, P. E. Iturria-Rivera, and M. Erol-Kantarci. Deep reinforcement learning-based joint user association and cu-du placement in o-ran. *IEEE Transactions on Network and Service Management*, 2022. 60, 72

[130] E. Sarikaya and E. Onur. Placement of 5g ran slices in multi-tier o-ran 5g networks with flexible functional splits. In *2021 17th International Conference on Network and Service Management (CNSM)*, 2021. 60

[131] I. Tamim, A. Saci, M. Jammal, and A. Shami. Downtime-aware o-ran vnf deployment strategy for optimized self-healing in the o-cloud. In *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021. 61

[132] S. Mondal and M. Ruffini. Optical front/mid-haul with open access edge server deployment framework for sliced o-ran. *IEEE Transactions on Network and Service Management*, 19(3), 2022. 61, 88

[133] H. Hojeij, M. Sharara, S. Hoteit, and V. Vèque. Dynamic placement of o-cu and o-du functionalities in open-ran architecture. In *2023 20th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pages 330–338, 2023. doi: 10.1109/SECON58729.2023.10287529. 61, 71, 72, 73

[134] K. Ali and M. Jammal. Proactive vnf scaling and placement in 5g o-ran using ml. *IEEE Transactions on Network and Service Management*, 21(1):174–186, 2024. doi: 10.1109/TNSM.2023.3292986. 62

[135] X. Zhang, J. Zuo, Z. Huang, Z. Zhou, X. Chen, and C. Joe-Wong. Learning with side information: Elastic multi-resource control for the open ran. *IEEE Journal on Selected Areas in Communications*, 42(2):295–309, 2024. 62

[136] A. Ndao, X. Lagrange, N. Huin, G. Texier, and L. Nuaymi. Optimal placement of virtualized dus in o-ran architecture. In *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, pages 1–6, 2023. 63

[137] I. Tamim, S. Aleyadeh, and A. Shami. Intelligent o-ran traffic steering for urllc through deep reinforcement learning. *arXiv preprint arXiv:2303.01960*, 2023. 63

[138] H. Sfaxi, I. Lahyani, S. Yangui, and M. Torjmen. Latency-aware and proactive service placement for edge computing. *IEEE Transactions on Network and Service Management*, pages 1–12, 2024. 63

[139] R. Joda, S. Naseri, M. Hashemi, and C. Richards. Ue centric du placement with carrier aggregation in o-ran using deep q-network algorithm. In *2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1–6, 2023. 64, 79

[140] D. Mroziński, M. Klinkowski, and K. Walkowiak. Cost-aware du placement and flow routing in 5g packet xhaul networks. *IEEE Access*, 11:12710–12726, 2023. 64

[141] A. Ndao, X. Lagrange, N. Huin, G. Texier, and L. Nuaymi. Optimal du placement in an o-ran multi-band system. In *2023 6th International Conference on Advanced Communication Technologies and Networking (CommNet)*, pages 1–7, 2023. 64

[142] S. Mollahasani, T. Pamuklu, R. Wilson, and M. Erol-Kantarci. Energy-aware dynamic du selection and nf relocation in o-ran using actor-critic learning. *Sensors*, 22(13):5029, 2022. 64

[143] H. Li, P. Li, K. D. Assis, A. Aijaz, S. Shen, R. Nejabati, S. Yan, and D. Simeonidou. Netmind: Adaptive ran baseband function placement by gcn encoding and maze-solving drl. *arXiv preprint arXiv:2401.06722*, 2024. 65

[144] S. Mollahasani, M. Erol-Kantarci, and R. Wilson. Dynamic cu-du selection for resource allocation in o-ran using actor-critic learning. In *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, pages 1–6, 2021. 65

[145] H. Hojeij, G. I. Ricardo, M. Sharara, S. Hoteit, V. Vèque, and S. Secci. Flexible association and placement for open-ran. In *IEEE INFOCOM 2024 NG-OPERA: Next-generation Open and Programmable Radio Access Networks*, 2024. 65, 86, 91, 92

[146] Haiyuan Li, Amin Emami, Karcius Day R. Assis, Antonis Vafeas, Ruizhi Yang, Reza Nejabati, Shuangyi Yan, and Dimitra Simeonidou. Drl-based energy-efficient baseband function deployments for service-oriented open ran. *IEEE Transactions on*

Green Communications and Networking, 8(1):224–237, 2024. doi: 10.1109/TGCN.2023.3321195. 65, 88

[147] T. Pamuklu, S. Mollahasani, and M. Erol-Kantarci. Energy-efficient and delay-guaranteed joint resource allocation and du selection in o-ran. In *2021 IEEE 4th 5G World Forum (5GWF)*, pages 99–104, 2021. 66, 68

[148] Suvidha Mhatre, Ferran Adelantado, Kostas Ramantas, and Christos Verikoukis. Intelligent qos-aware slice resource allocation with user association parameterization for beyond 5g o-ran-based architecture using drl. *IEEE Transactions on Vehicular Technology*, 74(2):3096–3109, 2025. doi: 10.1109/TVT.2024.3483288. 68, 71, 72, 73, 80

[149] Yassir Al-Karawi, Hamed Al-Raweshidy, and Rajagopal Nilavalan. Power consumption evaluation of next generation open radio access network. In *2024 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–6, 2024. doi: 10.1109/ICCE59016.2024.10444418. 72, 81

[150] Ihtisham Khalid, Vasilis Maglogiannis, Dries Naudts, Adnan Shahid, and Ingrid Moerman. Optimizing hybrid v2x communication: An intelligent technology selection algorithm using 5g, cv2x pc5 and dsrc. *Future Internet*, 16(4):107, 2024. doi: 10.3390/fi16040107. 100, 101, 112

[151] M. Altahrawi, N. F. Abdullah, and R. Nordin. Service-oriented lstm multi-criteria rat selection scheme for vehicle-to-infrastructure communication. *IEEE Access*, 10: 110261–110284, 2022. doi: 10.1109/ACCESS.2022.3214852. 101, 112

[152] R. Jacob, N. Franchi, and G. Fettweis. Hybrid v2x communications: Multi-rat as enabler for connected autonomous driving. In *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1370–1376, Bologna, Italy, 2018. doi: 10.1109/PIMRC.2018.8580953. 101

[153] B. Y. Yacheur, T. Ahmed, and M. Mosbah. Drl-based rat selection in a hybrid vehicular communication network. In *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, pages 1–5, Florence, Italy, 2023. doi: 10.1109/VTC2023-Spring57618.2023.10199400. 102

[154] L. Bréhon-Grataloup, R. Kacimi, and A.-L. Beylot. Multi-rat-enabled edge computing for vehicle-to-everything architectures. *Ad Hoc Networks*, 154, 2024. 102

[155] G. Ghatak, A. De Domenico, and M. Coupechoux. Small cell deployment along roads: Coverage analysis and slice-aware rat selection. *IEEE Transactions on Communications*, 67(8):5875–5891, 2019. doi: 10.1109/TCOMM.2019.2916794. 102

[156] K. Waqas, N. Q. Nadia, C. Omer, A. Sadiq, Y. Heejung, and Z. U. H. Muhammad. Adaptive network selection for enhanced connectivity in vehicular-to-everything (v2x) communications. *SSRN*, 2024. doi: 10.2139/ssrn.5123284. 102

[157] M. Sepulcre and J. Gozalvez. Heterogeneous v2v communications in multi-link and multi-rat vehicular networks. *IEEE Transactions on Mobile Computing*, 20(1): 162–173, 2021. doi: 10.1109/TMC.2019.2939803. 103, 112

[158] Z. H. Mir, J. Toutouh, F. Filali, and E. Alba. Qos-aware radio access technology (rat) selection in hybrid vehicular networks. In *Communication Technologies for Vehicles*, volume 9066 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2015. 103

[159] J. Lianghai, A. Weinand, B. Han, and H. D. Schotten. Multi-rats support to improve v2x communication. In *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–6, Barcelona, Spain, 2018. doi: 10.1109/WCNC.2018.8377432. 103

[160] Nurudeen O. Salau, Sanaullah Manzoor, and Muhammad Z. Shakir. Machine learning approach of multi-rat selection for travelling users in 5g nsa networks. *IET Networks*, 14(1):e12124, 2025. doi: 10.1049/ntw2.12124. 103

[161] Xiaoman Shen, Jun Li, Lei Chen, Jiajia Chen, and Sailing He. Heterogeneous lte/dsrc approach to support real-time vehicular communications. In *2018 10th International Conference on Advanced Infocomm Technology (ICAIT)*, pages 122–127, 2018. doi: 10.1109/ICAIT.2018.8686612. 104, 111, 113