Author/s:
Mondal, Sourav

Title:
Strategic Deployment of Artificial Intelligence-Enhanced Cloudlets for Low-latency Human-to-Machine Applications

Date:
2020

Persistent Link:
https://hdl.handle.net/11343/241842

# Strategic Deployment of Artificial Intelligence-Enhanced Cloudlets for Low-latency Human-to-Machine Applications

by

**Sourav Mondal**

**ORCID: 0000-0002-5467-2545**

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the

Department of Electrical and Electronic Engineering
Melbourne School of Engineering
**THE UNIVERSITY OF MELBOURNE**

April 2020

# Declaration of Authorship

I, Sourav Mondal, declare that this thesis titled, "Strategic Deployment of Artificial Intelligence-Enhanced Cloudlets for Low-latency Human-to-Machine Applications" and the work presented in it are my own. I confirm that:

- the thesis comprises only my original work towards the Doctor of Philosophy except where indicated in the preface;
- due acknowledgement has been made in the text to all other material used; and
- the thesis is fewer than 100,000 words in length, exclusive of tables, maps, bibliographies and appendices as approved by the Research Higher Degrees Committee.

Signed: _____

Date: _____

# Abstract

The genesis of mobile cloud computing technology is one of the most significant technical advents of the last decade which can be seen as a marriage between cloud computing and mobile computing technologies. This technical paradigm brings mobile users, telecommunication network operators, and cloud service providers to a common playground, thus providing business opportunities for network operators and cloud service providers. The extension of this facility towards access networks by aggregation of edge-intelligence nodes like cloudlets is one more step forward. A cloudlet is a "data centre in a box" with enhanced mobility support to bring the cloud closer to mobile users and uses virtual machine abstraction for dynamic resource allocation to trusted mobile users, isolate untrusted mobile users, and support a wide variety of applications without being limited by their process structures, programming languages, or operating systems. To fulfil the ravenous demand for computational resources entangled with the crisp latency requirements of various computationally intensive and mission-critical applications related to augmented reality, autonomous transport, cognitive assistance, and Tactile Internet, installation of cloudlets near access seems to be a very promising solution because of its support for wide geographical network distribution, low latency, mobility and heterogeneity.

Finding the optimal cost of cloudlet deployment over urban, suburban, and rural deployment areas with an existing access network, essentially implies finding the optimal placement locations of the cloudlets over the entire deployment area and the optimal amount of computational and storage resources per cloudlet. Technically, this research question leads to an assignment problem, where we need to find the optimal interconnections between mobile devices and cloudlets. In this research, we propose a hybrid cost-optimal cloudlet placement framework over existing fibre-wireless access networks based on mixed-integer non-linear programming. We primarily focus on static cloudlet network planning and placement, i.e., identification of exact optimal cloudlet placement locations over urban, suburban and rural deployment scenarios to provide guidance on the installation cost and assess the workload distribution among different cloudlets and the percentage of incremental energy arising from the presence of cloudlets in the fibre-wireless access networks. Howbeit, we observed that mixed-integer programming based frameworks suffer from scalability issues with large networks and become completely useless when the network data is unavailable. Thus, to overcome this issue, we design analytical frameworks that

can provide a quick first-hand estimation of cloudlet deployment cost depending on mobile user density, network architecture, and QoS requirements. We verify that the results produced by this method can be considered as tight lower bounds to that produced by integer programming based frameworks for most practical scenarios. We further perform a parametric analysis to understand the dependence of cloudlet deployment cost on various network parameters.

However, depending on the mobility pattern and dynamically varying computational requirements of associated mobile devices, cloudlets at different parts of the network become either overloaded or under-loaded. Thus, we propose an economic and non-cooperative load balancing game for low-latency applications among neighbouring cloudlets, from same as well as different service providers. While addressing load balancing problems, most authors usually stress on minimising the end-to-end latency and do not consider the heterogeneity of neighbouring cloudlets. Nonetheless, in practice, if the job requests are processed within their requested QoS latency target, mobile users should be satisfied. Therefore, instead of formulating a conventional latency minimisation game, we propose a novel utility maximisation game to capture the multi-party economic interaction among heterogeneous neighbouring cloudlets. In this load balancing game, the participating cloudlets achieve their maximum utility when the end-to-end latency is equal to the QoS latency target. With this game formulation, each cloudlet is always interested in receiving some extra job requests and the associated incentives from their neighbouring cloudlets to push their utility towards the maximum point. To implement this game-theoretic load balancing framework, firstly, we propose a centralised mechanism where all the competing cloudlets send their predicted job request arrival rates to a neutral mediator. The mediator computes the Nash equilibrium load balancing strategies for the cloudlets and broadcasts to them before the actual job request arrival. This centralised mechanism also ensures that competing cloudlets are truthful while revealing private information e.g., total incoming job requests. Secondly, we propose a continuous-action reinforcement learning automata-based scheme, which allows each cloudlet to independently compute the Nash equilibrium in a completely distributed network setting. We critically study the convergence properties of the designed learning algorithm, scaffolding our understanding of the underlying load balancing game for faster convergence, and study the impacts of exploration and exploitation on learning accuracy.

After investigating the cloudlet placement and load balancing problems, we investigate the role of edge-intelligence servers like cloudlets in deploying low-latency

human-to-machine applications like teleoperation, immersive virtual/augmented reality, and industrial automotive control over long-distance access networks. Such applications are being realised through Tactile Internet that allows users to control remote things and involve the bi-directional transmission of video, audio, and haptic data. However, the end-to-end propagation latency presents a stubborn bottleneck, which can be alleviated by using various artificial intelligence-based application layer and network layer prediction algorithms, e.g., forecasting and preempting haptic feedback transmission. To gain proper insights, we study the experimental data on traffic characteristics of control signals and haptic feedback samples obtained through virtual reality-based human-to-machine teleoperation. Moreover, we propose the installation of edge-intelligence servers between master and slave devices to implement the preemption of haptic feedback from control signals. Harnessing virtual reality-based teleoperation experiments, we further propose a two-stage artificial intelligence-based module for forecasting haptic feedback samples. The first-stage unit is a supervised binary classifier that detects if haptic sample forecasting is necessary and the second-stage unit is a guided reinforcement learning unit that ensures haptic feedback samples are forecasted accurately when different types of material are present. Furthermore, by evaluating analytical expressions, we show the feasibility of deploying remote human-to-machine teleoperation over fibre backhaul by using our proposed artificial intelligence-based module, even under heavy traffic intensity.

# Preface

> "I do not think there is any thrill that can go through the human heart like that felt by the inventor as he sees some creation of the brain unfolding to success... such emotions make a man forget food, sleep, friends, love, everything." — Nikola Tesla

This thesis comprises only my original work (100%), which I conducted with the supervision of Prof. Elaine Wong (The University of Melbourne) and A/Prof. Goutam Das (Indian Institute of Technology Kharagpur) under Melbourne India Postgraduate Program (MIPP). This thesis has not been submitted for any other qualification. All the work towards the thesis was carried out after the enrolment in the degree. No third party editorial assistance was provided in the preparation of the thesis. Financial support for this undertaking was provided by the Australian Government and the University of Melbourne through the Melbourne Research Scholarship (MRS).

All the work in this thesis was conducted by the author of this thesis at the University of Melbourne, including problem formulation, mathematical analysis, implementation of the simulation environments, generation and analysis of the results and manuscript writing, which contributed about 80% to the work. Both the supervisors provided insightful technical comments and discussions which jointly contributed about 20% to the work. The publications arising from this thesis are listed in Chapter 1 Section 1.4. All of the journal and conference publications listed under Chapter 3 to Chapter 6, contain only the original work of the author of the thesis, including analytical models, computer simulations, and manuscript writing. Prof. Elaine Wong and A/Prof. Goutam Das provided helpful discussions and important feedback.

The 1$^{st}$ and 4$^{th}$ conference publications from Chapter 3 are based on the high-level research overview and written by Prof. Elaine Wong, where the author contributed about 70% to each publication. In the 1$^{st}$ conference paper from Chapter 6, the author contributed around 80%. Lihua Ruan helped in collecting experimental data and Prof. Elaine Wong helped with the manuscript preparation. Part of the research work for the 1$^{st}$ journal paper from Chapter 6 was carried out in Charles III University of Madrid in January 2020 and the author's total contributions for this paper is around 70%. Rest of the authors i.e., Lihua Ruan, Martin Maier, David Larrabeiti, Goutam Das, and Elaine Wong helped in discussing insightful research ideas and organising the manuscript.

# Acknowledgements

Undertaking this PhD program has been a truly life-enhancing experience for me and it would not have been possible to achieve without the support and guidance that I received from various people. First and foremost, I would like to express my profound and heartfelt gratitude to my doctoral advisory committee: Prof. Elaine Wong, A/Prof. Goutam Das, and Prof. William Shieh. I especially thank Prof. Wong for her invaluable guidance and relentless support throughout my doctoral research tenure. I always felt motivated working with her for her patience, boundless enthusiasm, dedication to excellence, and for providing various opportunities for my professional development. I truly feel blessed and privileged to have this opportunity to carry out my doctoral thesis work with the supervision of a passionate and dedicated researcher like her. I must express my deepest gratitude to my overseas co-supervisor A/Prof. Goutam Das for his consistent encouragements and patient guidance, which not only have motivated me to pursue higher studies, but also have helped me to overcome the research hurdles, and keep my spirit high during my toughest setbacks. In addition, I would also like to express my sincerest gratitude to my advisory chair, Prof. William Shieh, for his precious advice.

Alongside my advisory committee, I acknowledge The University of Melbourne for providing me with an opportunity to pursue the PhD program and financial assistance. In addition to Melbourne Research Scholarship (MRS), I also received 2017 Melbourne Abroad Traveling Scholarship (MATS), two one-time travel grants from Melbourne School of Engineering in 2017 and 2018, and 2019 John Collier Travel Scholarship. I am also thankful to Prof. Margreta Kuijper and Prof. Elaine Wong for providing me with casual tutoring and workshop demonstration jobs during semesters. During my short research visit to Charles III University of Madrid in January 2020, it was a great honour to collaborate with Prof. Martin Maier and Prof. David Larrabeiti. I was impressed by the charming personality and candid supervision style of Prof. Maier. I also enjoyed my time in Madrid very much as Prof. David introduced me to several delicacies of Spanish cuisine. I would also like to take this opportunity to thank all my teachers and professors since my childhood to this time. I am completely indebted to them for laying a solid foundation in my primary education, which eventually guided me to pursue the PhD program.

I would like to thank my friends and colleagues Partha De, Subhrasankha De, Jithin George, Samiru Gayan, Sampath Edirisinghege, Dibyendu Roy, Ishita Akhter,

Lihua Ruan, and several others for making my life at the University of Melbourne a truly memorable one. I thoroughly enjoyed most of the quality times as well as casual times I spent with you all, discussing technical, philosophical, and spiritual topics. I would also like to thank my lifetime friends Abhik Datta Banik, Priyadarshi Mukherjee, Sourav Bhattacharyya, Aritra Chatterjee, Subhrajit Roy, and Sanchari Baral, whose warm friendship kept me moving forward throughout the various moment of ups and downs in my life. Finally, I can never give enough thanks to my parents Mr. Sekharendu Mondal and Mrs. Mousumi Mondal Majumder, my grandmother, my cousins, and my other family members for their constant encouragement, unconditional love and support throughout my life.

*To my beloved parents, family members, and dearest friends*

*"Never regard your study as a duty, but as the enviable opportunity to learn the liberating beauty of the intellect for your own personal joy and for the profit of the community to which your later work will belong."*

— ***Albert Einstein (1879-1955)***

# Contents

# List of Figures

# List of Tables

# Abbreviations

| | |
|---|---|
| **AI** | **A**rtificial **I**ntelligence |
| **ANN** | **A**rtificial **N**eural **N**etwork |
| **BS** | **B**ase **S**tation |
| **CA** | **C**arrier **A**ggregation |
| **CDMA** | **C**ode **D**ivision **M**ultiple **A**ccess |
| **CO** | **C**entral **O**ffice |
| **CoMP** | **Co**ordinated **M**ulti **P**oint |
| **CPU** | **C**entral **P**rocessing **U**nit |
| **C-RAN** | **C**loud-**R**adio **A**ccess **N**etwork |
| **D2D** | **D**evice to(**2**) **D**evice |
| **DSSS** | **D**irect **S**equence **S**pread **S**pectrum |
| **FHSS** | **F**requency **H**opping **S**pread **S**pectrum |
| **eMBB** | **e**nhanced **M**obile **B**road**B**and |
| **FiWi** | **Fi**ber **Wi**reless |
| **H2M** | **H**uman to(**2**) **M**achine |
| **IoT** | **I**nternet **o**f **T**hings |
| **ITS** | **I**ntelligent **T**ransport **S**ystem |
| **KKT** | **K**arush **K**uhn **T**ucker |
| **LOS** | **L**ine **O**f **S**ight |
| **LTE** | **L**ong **T**erm **E**volution |
| **MCC** | **M**obile **C**loud **C**omputing |
| **MEC** | **M**ulti-access **E**dge **C**omputing |
| **MIMO** | **M**ultiple **I**nput **M**ultiple **O**utput |
| **MINLP** | **M**ixed **I**nteger **N**on-**L**inear **P**rogramming |

| | |
|---|---|
| **ML** | **M**achine **L**earning |
| **mMTC** | **m**assive **M**achine **T**ype **C**ommunications |
| **NE** | **N**ash **E**quilibrium |
| **NLOS** | **N**on-**L**ine **O**f **S**ight |
| **NR** | **N**ew **R**adio |
| **OFDM** | **O**rthogonal **F**requency **D**ivision **M**ultiplexing |
| **OLT** | **O**ptical **L**ine **T**erminal |
| **ONU** | **O**ptical **N**etwork **U**nit |
| **PON** | **P**assive **O**ptical **N**etwork |
| **QoE** | **Q**uality **o**f **E**xperience |
| **QoS** | **Q**uality **o**f **S**ervice |
| **R&F** | **R**adio **a**nd **F**iber |
| **RL** | **R**einforcement **L**earning |
| **RN** | **R**emote **N**ode |
| **RoF** | **R**adio **o**ver **F**iber |
| **RTT** | **R**ound **T**rip **T**ime |
| **TDM** | **T**ime **D**ivision **M**ultiplexed |
| **TDMA** | **T**ime **D**ivision **M**ultiple **A**ccess |
| **TI** | **T**actile **I**nternet |
| **uRLLC** | **u**ltra-**R**eliable **L**ow-**l**atency **C**ommunications |
| **UE** | **U**ser **E**quipment |
| **VI** | **V**ariational **I**nequality |
| **VM** | **V**irtual **M**achine |
| **WAP** | **W**ireless **A**cess **P**oint |
| **WCDMA** | **W**ideband **C**ode **D**ivision **M**ultiple **A**ccess |
| **WiFi** | **Wi**reless **Fi**delity |
| **WLAN** | **W**ireless **L**ocal **A**rea **N**etwork |
| **WMAN** | **W**ireless **M**etropolitan **A**rea **N**etwork |

# Chapter 1

# Introduction

"Begin at the beginning," the King said, very gravely,
"and go on till you come to the end; then stop."

---

Lewis Carroll, Alice in Wonderland

## 1.1 Cloudlet Computing Networks

The smart mobile devices of the present time e.g., smart-phones, smart-glasses, tablets, and Internet of Thing (IoT) devices, to name a few, have almost become an integral part of everyone's life and have merged into it almost indistinguishably. In his seminal paper [1], Weiser had shared his vision of ubiquitous computing in 21$^{st}$ century and looking around at the fact that we are able to execute all our communication and computational tasks while moving across various places, we can hardly have any doubt on that. In this wireless communication era, users all over the world expect to communicate "anytime, anywhere, with anyone". In smart cities and smart homes, almost every object around us is expected to have some computational capability and connectivity to the internet. Gartner predicts that, when IoT becomes completely commercialised along with the standardisation of 5G technologies, nearly 20 billion devices will connect to the Internet by 2020 [2]. In conjunction with this, so far mobile data traffic has increased by 63% till 2016 and by 2021, this is expected to increase by seven folds further [3].

The mobile computing technology provides distributed computation facilities on battery-powered, portable and mobile devices which are interconnected via mobile communication standards and protocols. Formally, mobile computing is defined as "a human-computer interaction by which a computer is expected to be transported during normal usage, which allows for the transmission of data, voice and video. Mobile computing involves mobile communication, mobile hardware, and mobile software". Communication issues include ad-hoc networks and infrastructure networks as well as communication properties, protocols, data formats and concrete technologies. Hardware includes mobile devices or device components. Mobile software deals with the characteristics and requirements of mobile applications [4]. In other words, this is just the convergence of wireless telephony and computational technologies. Hence, this provides the scope of executing miscellaneous useful applications on a single device to the users. A few well-known applications are the transmission of data, voice and video on the fly.

However, along with several fancy advantages, there exist some crucial limitations in mobile computing, e.g., resource constraints on battery, degradation of Quality of Service (QoS), limited bandwidth and connection latency. Some typical smart applications related to augmented reality, autonomous transport, cognitive assistance, and Tactile Internet applications like real-time patient monitoring that demand a network latency of 1-100 ms. However, mobile devices are designed to be lightweight and portable, which makes them poor in computational resources and memory and hence, very often the mobile devices become incapable of executing these applications over a long duration. Thus, to improve the performance of mobile devices, offloading the major computation components to remote cloud servers appeared to be a better solution [5].

The perception of cloud computing varies from person to person. Some think it as accessing software and storing data on the internet and use the associated services, whereas, some others think this as just a modernisation of the time-shared computational model deployed in the 1960s. Many authorities have defined cloud computing from their own perspective in recent past, e.g, the authors of [6] defined cloud computing as cloud computing is a model for enabling convenient, on-demand

Table 1.1: Comparison among fog, multi-access edge, and cloudlet servers

| | ***Fog computing*** | ***MEC*** | ***Cloudlet computing*** |
|---|---|---|---|
| **Device type** | Routers, switches, access points, gateways | Servers running in base stations | Data centre in a box |
| **Network location** | Varying between end devices and cloud | Radio network controller/ macro base station | Local/outdoor installation |
| **Software characteristics** | Fog abstraction layer based | Mobile orchestrator based | Cloudlet agent based |
| **Context awareness** | Medium | High | Low |
| **Proximity** | One or multiple hops | One hop | One hop |
| **Access network** | Bluetooth, WiFi, mobile networks | Mobile networks | WiFi |
| **Node-to-node communication** | Supported | Partial | Partial |

network access to a shared pool of configurable and reliable computing resources (e.g., networks, servers, storage, applications, services) that can be rapidly provisioned and released with minimal consumer management effort or service provider interaction.

Researchers started to explore the integration of cloud and mobile computing, also known as mobile-cloud computing (MCC) so that mobile devices can offload some of their computationally intensive and high memory consuming tasks to the remote cloud server. However, cloud servers located in the remote core network locations, which are usually very far from the edge and fail to meet the stringent latency requirements of low-latency applications. To overcome this hurdle, researchers from both industry and academia proposed edge computing solutions like multi-access edge computing (MEC), fog computing, and cloudlet computing [7]. A brief comparison of all these technical paradigms is summarised in Table 1.1. Nonetheless, it must be noted that the edge computing solutions cannot completely replace legacy cloud technology because clouds have several significant advantages, most notably network security and the scope for massive data storage and data processing power. When cloud and cloudlets coexist in the network, the combination can, however, improve end-user experience due to reduced latency.

The term "cloudlet" is commonly referred to as "data centre in a box" that "brings the cloud closer". Defined as a trusted, resource-rich computer or cluster of computers that are well-connected to the Internet and available for use by nearby mobile devices [8], cloudlets have several advantages and opportunities to improve network latency. Cloudlets can be installed over wireless as well as fibre-wireless

(FiWi) access networks in the closest vicinity of a group of mobile devices. This creates a three-tier network architecture, i.e., cloud-cloudlet-mobile edge devices, as shown in Fig 1.1. Thus, a cloudlet becomes a single-hop node from the mobile devices and can provide computational services much faster than a remote cloud. Cloudlets use virtual machines (VMs) to serve the job requests originating from mobile devices. Moreover, a cloudlet can work independently and without context-awareness, which makes it a very robust solution even in hostile environments [9].

## 1.2 Focus of the Thesis

Cloudlet servers are computers or clusters of computers installed in the proximity of mobile device users and are distributed across access networks. It is essential to place them optimally over access networks so that the maximum number of mobile devices in its proximity are connected. However, different wireless, optical, or FiWi access network technologies are associated with different network topologies. Hence, the placement of cloudlets at a strategic location such that the maximum number of mobile devices are connected poses an important research challenge. Therefore, this thesis explores efficient cloudlet placement frameworks over FiWi access networks subject to computational capacity, network connectivity, and latency constraints.

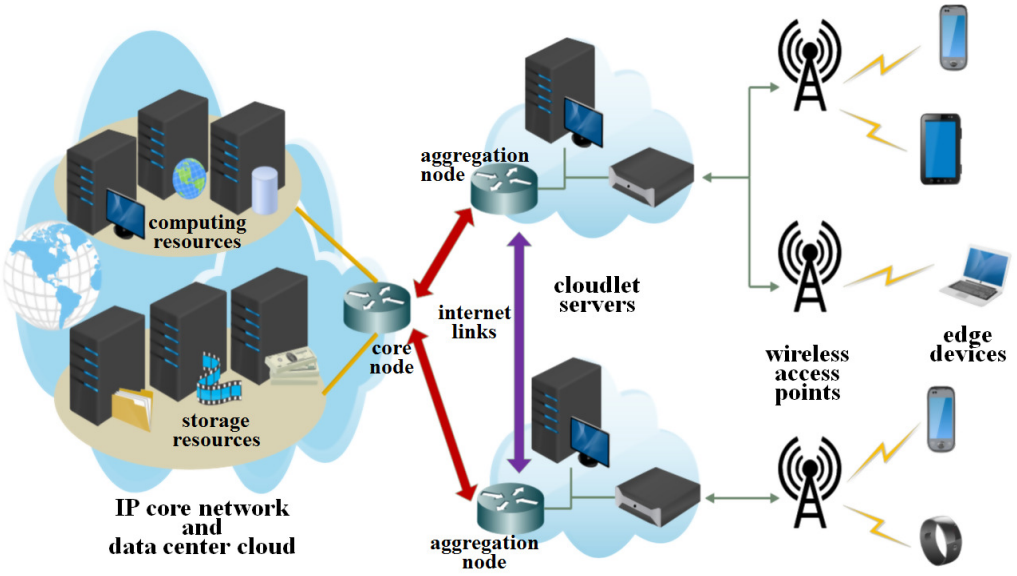



Figure 1.1: A simplified three-tier network architecture with cloud servers, cloudlet servers, and mobile edge devices.

The first objective of this thesis is to design cost-optimal cloudlet placement frameworks that can identify exact optimal cloudlet placement locations with any given instance of network data. Nonetheless, a critical issue arises when there is no existing network infrastructure or network data, i.e., the network topology between mobile devices and the cloudlets is unavailable. Thus, in addition to this, analytical frameworks are also designed in this thesis that can provide a quick first-hand estimation of cloudlet deployment cost depending on mobile user density, network architecture, and QoS requirements, even when the network data is unavailable.

As cloudlet computing systems are essentially distributed computing systems, hence, job request allocation and load balancing among neighbouring cloudlets problems appear to be the next important research challenges. Although job allocation frameworks allocate job requests to the most favourable cloudlets, due to the dynamic nature of the job request arrival process, cloudlets at different parts of the network become overloaded and under-loaded at different times. Therefore, the second objective of this thesis is to explore efficient load balancing methods among neighbouring cloudlets. Firstly, centralised frameworks are designed where a controller node decides the load balancing strategies for the neighbouring cloudlets. Secondly, distributed frameworks are designed where the neighbouring cloudlets find their load balancing strategies independently without exchanging extensive control messages among themselves.

It is interesting to note that the state-of-the-art access networks are still incapable of meeting the stringent low-latency requirements of various applications, especially with latest Tactile Internet applications like teleoperation, immersive virtual/augmented reality, and haptic communications where multi-modal bi-directional communication is involved. This latency bottleneck arises mainly from the stubborn propagation latency in a network. To solve such issues, the presence of distributed edge-intelligence nodes like cloudlets across access networks can be very much helpful in reducing the round-trip network latency. Usually, the reverse direction traffic follows the forward direction traffic and typically is bursty in nature as it depends

on the type of task is being performed. Thus, the cloudlets can decouple this bi-directional traffic by employing artificial intelligence that can predict various statistical properties of the haptic feedback traffic. Therefore, this thesis also investigates the performance of various machine learning and reinforcement learning algorithms with cloudlets that can help in reducing the round-trip network latency.

## 1.3 Outline and Contributions of the Thesis

This thesis consists of a total eight chapters including the current chapter. In the current Chapter 1 (Introduction), we discuss the primary motivation, objectives, outline, and the original contributions of the thesis. In Chapter 2, we introduce cloudlet computing as an edge computing paradigm for low-latency applications, review relevant research works, and discuss the recent progress in the field of cloudlet computing over wireless and FiWi access networks. We present the research work carried out for this thesis in detail from Chapter 3 to Chapter 6. Finally, we summarise the primary achievements of the thesis in Chapter 7 and identify some potential future research directions. A brief outline of the rest of the chapters and their originals contributions are presented in the following subsections.

### Chapter 2: Cloudlet Networks for Low-latency Applications

This chapter provides a discussion on recent research works carried out on various aspects of edge computing solutions like cloudlets over wireless and FiWi networks. At first, it provides a brief overview of both the state-of-the-art wireless and time-division multiplexed passive optical network (TDM-PON) based FiWi networks. Along with the wireless networks, the FiWi networks are also considered for cloudlet deployment due to their wide network coverage, low cost per bit, very high data transmission bandwidth, and immense potential for back/front-hauling of future generation wireless access networks. Followed by, we start to discuss the role of cloudlet computing in saving scarce computational, memory, and energy resources of mobile devices for computation-intensive and low-latency applications. Next, we review some of the recent cloudlet placement and job request allocation frameworks

over the state-of-the-art wireless and FiWi access networks. In particular, we critically discuss the performance of various frameworks that optimise either of the cost of installation, energy consumption, and end-to-end latency, subject to others as constraints. Next, we survey some recently proposed load balancing frameworks among neighbouring cloudlets and discuss the merits and demerits of different approaches taken in designing these frameworks. Finally, we review the recent research works that investigated various machine learning algorithms for reducing the round-trip network latency of applications with bi-directional communication.

### Chapter 3: Exact Cost-optimisation Frameworks for Cloudlet Placement over Fibre-Wireless Access Networks

In this chapter, we address the static network planning problem to obtain a first-hand estimation of the cloudlet deployment cost without considering any user mobility and complex system models. From our studies, we realised the necessity of a new network architecture for cloudlet placement over FiWi access networks that efficiently use the salient features of the underlying TDM-PON. Hence, we took the liberty of proposing an end-to-end hybrid cloudlet placement architecture over FiWi access networks that considers cloudlet placement in the field, at the central office (CO), and remote note (RN) locations. Moreover, we use mathematical optimisation techniques to find the exact cost-optimal cloudlet placement locations with a given instance of access network data. In particular, we design a mixed-integer non-linear programming (MINLP) based framework that can identify an optimal set of cloudlet placement locations with minimal installation cost subject to computational capacity, network connectivity, and latency constraints. We evaluate the performance of the proposed framework over urban, suburban, and rural population areas against very low end-to-end QoS latency requirements of 1 ms, 10 ms, and 100 ms. Furthermore, we make a comparative study between the hybrid cloudlet placement framework and the field cloudlet placement framework.

**Original contributions of Chapter 3**

- Proposed an end-to-end hybrid cloudlet placement architecture based on FiWi access networks with TDM-PON as back/fronthaul support.
- Formulated an MINLP problem to identify optimal cloudlet placement locations with hybrid cloudlet placement architecture over a given instance of FiWi access network data.
- Derived the MINLP formulation for the field cloudlet placement framework as a sub-problem of the MINLP formulation with the hybrid cloudlet placement architecture.
- Showed that the formulated MINLP frameworks with hybrid cloudlet placement architecture achieve a very low end-to-end latency e.g., 1 ms, 10 ms, and 100 ms.
- Linearised the objective function of the MINLP formulation by using some additional linear constraints to reduce the computational complexity of our design tool towards convergence to the optimal solution.
- Used the proposed MINLP framework against urban, suburban and rural scenarios to present a comparative study between field and hybrid cloudlet placement frameworks on deployment cost, workload distribution among various cloudlets, and the average number of processors per cloudlet.
- Demonstrated that the percentage of the incremental energy budget of the fibre-wireless access networks due to cloudlet installation in an urban, suburban and rural scenario is under 18%.

## Chapter 4: Analytical Cost-optimisation Frameworks for Cloudlet Placement over Fibre-Wireless Access Networks

Through our exercises in Chapter 3, we realised that the MINLP based framework is capable of providing us with the exact cloudlet placement locations, but we do

not gain any general insight about cost-optimal cloudlet deployment strategies, depending on the underlying network scenario. Moreover, this framework suffers from scalability issues with large data and becomes completely useless when the network data is unavailable. Thus, in this chapter, we propose a method to find lower-bounds of integer programming based cost optimisation frameworks with hybrid cloudlet placement architecture. We use the network homogeneity assumption to perform an average cloudlet installation cost analysis with a few network parameters e.g., TDM-PON split-ratio, mobile device user density, available bandwidth, and QoS latency requirements. We apply commonly used techniques to solve convex optimisation problems that resolve the scalability issues arising from integer programming based frameworks, thus providing a first-hand estimation of the cloudlet deployment cost against any deployment scenario. We verify that the results produced by this method can be considered as tight lower bounds to that produced by integer programming based frameworks for most practical scenarios. We further perform a parametric analysis to understand the dependence of cloudlet deployment cost on various network parameters.

**Original contributions of Chapter 4**

- Developed a technique to reduce the total network deployment into a smaller network unit to formulate a simple optimisation problem with a single TDM-PON based FiWi branch using the network homogeneity assumption.
- Formulated a convex optimisation problem to minimise cloudlet installation cost and showed that the primary latency constraint in the problem formulation is non-convex but quasi-convex.
- Solved the formulated optimisation problems by applying the KKT conditions on the Lagrangian functions and derive closed-form expressions for the cloudlet deployment cost under some cases.
- Used some numerical techniques for other cases but proved that the results still provide tight lower bounds on the optimal cost. In general, we derived lower

bounds of the overall cloudlet deployment cost corresponding to the actual network in a scalable manner.

- Validated our proposed method with extensive simulation and compared the performance with the MINLP based framework proposed in Chapter 3 over urban deployment area against 1 ms, 10 ms, and 100 ms target latency values with split-ratio $1 : N$, $N \in \{4, 8, 16\}$.

- Performed a detailed parametric analysis without any scalability issues and illustrated the impact of different network parameters e.g., split ratio of TDM-PON, population density and target QoS latency on the cloudlet deployment cost over FiWi networks.

### Chapter 5: Centralised and Decentralised Non-Cooperative Load-Balancing Games among Neighboring Cloudlets

In this chapter, we investigate the load balancing problem among neighbouring cloudlets. In a real deployment scenario, cloudlets from multiple service providers coexist, and this creates a heterogeneous cloudlet environment, where cloudlet service providers compete with each other over the same customer base. Thus, we formulate the load balancing problem as economic and non-cooperative games among multiple competitive cloudlets from same as well as different service providers under the supervision of a neutral mediator. With this game formulation, each competing cloudlet aims to maximise their utilities while achieving similar overall end-to-end latency. In addition, this game formulation ensures the truthful revelation of private information from the cloudlets through an incentive-compatible mechanism. Furthermore, we propose a continuous-action reinforcement learning automata-based algorithm, which allows each cloudlet to independently compute the Nash equilibrium (NE) in a completely distributed cloudlet network. We critically study the general convergence properties of the designed learning algorithm, scaffolding our understanding of the underlying load balancing game for faster convergence. Through extensive simulations, we study the impacts of exploration and exploitation on learning accuracy.

**Original contributions of Chapter 5**

- Formulated the load balancing problem among cloudlets from multiple service providers as a novel economic and non-cooperative game and ensured that each neighbouring cloudlet gets a scope to maximise its utility while meeting the QoS latency requirements of mobile device users.
- Showed that the formulated load balancing game can enforce the competing cloudlets to truthfully reveal their private information to a neutral mediator through an incentive-compatible mechanism.
- Demonstrated that both overloaded and under-loaded cloudlets achieve a utility greater than or equal to the utility without participating in the market competition by participating in the market competition and strategically offloading a fraction of their total incoming job requests to their neighbouring cloudlets according to the proposed NE strategy.
- Designed a distributed continuous-action reinforcement learning automata-based algorithm such that neighbouring cloudlets can independently compute the NE load balancing strategy, without exchanging any mutual control information. In turn, improved the convergence rate of reinforcement learning by scaffolding the properties of the underlying load balancing game.

## Chapter 6: Edge-Artificial Intelligence for Human-to-Machine Applications over Fibre-Wireless Access Networks

The recent research trends for achieving ultra-reliable and low-latency communication networks are largely driven by smart manufacturing and industrial Internet-of-Things applications. Such applications are being realised through Tactile Internet that allows users to control remote things and involve the bidirectional transmission of video, audio, and haptic data. The term "haptics" refers to both "kinesthetic perception", i.e., forces, torques, position, and velocity experienced by the muscles, joints, and tendons of the body and "tactile perception", i.e., surface texture and friction sensed by different types of mechanoreceptors in the skin [10]. However,

the end-to-end propagation latency presents a stubborn bottleneck, which can be alleviated by using various artificial intelligence-based application layer and network layer prediction algorithms, e.g., forecasting and preempting haptic feedback transmission. In this chapter, we study the experimental data on traffic characteristics of control signals and haptic feedback samples obtained through virtual reality-based human-to-machine teleoperation. Moreover, we propose the installation of edge-intelligence servers between master and slave devices to implement the preemption of haptic feedback from control signals. Harnessing virtual reality-based teleoperation experiments, we further propose a two-stage artificial intelligence (AI)-based module for forecasting haptic feedback samples. The first-stage unit is a supervised binary classifier that detects if haptic sample forecasting is necessary and the second-stage unit is a reinforcement learning (RL) unit that ensures haptic feedback samples are forecasted accurately when different types of material are present. Furthermore, by evaluating analytical expressions, we show the feasibility of deploying remote human-to-machine teleoperation over fibre backhaul by using our proposed AI-based module, even under heavy traffic intensity.

**Original contributions of Chapter 6**

- Firstly, we study various statistical characteristics of the control and haptic feedback data obtained from our virtual reality teleoperation experiments. We create histograms for various test scenarios from the packet arrival timestamps and successfully fit them with generalised Pareto distribution with optimal parameters.

- Secondly, we propose a two-stage AI-based haptic sample forecast unit, known as Event-driven HAptic feedback SAmple Forecast (EHASAF) unit. The first-stage uses ANN to implement a supervised learning algorithm on control signals from virtual reality gloves to detect if forecasting of haptic feedback samples is necessary. We show that our designed artificial neural network-based supervised learning algorithm can detect haptic feedback forecasting events from our experimental control signal data with $\sim 99\%$ accuracy.

- Thirdly, we propose a second-stage RL unit that uses a linear reward-inaction algorithm for run-time prediction of proper haptic feedback samples at every iteration. Through numerical evaluation, we perform a sensitivity analysis of the RL unit and investigate the impact of exploration and exploitation on the accuracy of the RL unit. Our results indicate $\sim 92\%$ accuracy for four different haptic materials with 90% mutual correlation in haptic feedback samples.

- Finally, we present the closed-loop latency for data transmission between master and slave devices over an optical fibre backhaul network against the overall network traffic intensity. In this context, we show that the deployment of remote human-to-machine teleoperation is infeasible for a 40 km master-slave closed-loop network with more than 65% overall traffic intensity. However, such deployment is possible by using our proposed EHASAF module with an intermediate human-to-machine server.

### Chapter 7: Conclusions and Future Directions

In this chapter, we make the concluding remarks with a summary of the research works reported. Furthermore, in this chapter, we discuss potential future directions of our research on cloudlet networks over FiWi access networks.

## 1.4 Publications

This section presents the journal and conference publications arising from the technical contributions described in the following chapters.

### Publications arising from Chapter 3 of Thesis

#### Journals

- **Sourav Mondal**, Goutam Das, and Elaine Wong, "Cost-optimal Cloudlet Placement Frameworks over Fibre-Wireless Access Networks for Low-latency Applications," *Journal of Network and Computer Applications*, vol. 138, pp. 27-38, 2019.

### Conferences

- Elaine Wong, **Sourav Mondal**, and Goutam Das, "Latency-aware Optimisation Framework for Cloudlet Placement," *in 2017 19th International Conference on Transparent Optical Networks (ICTON)*, Girona, July 2017.
- **Sourav Mondal**, Goutam Das, and Elaine Wong, "A Novel Cost optimization Framework for Multi-Cloudlet Environment over Optical Access Networks," *in 2017 IEEE Global Communications Conference (GLOBECOM)*, Singapore, December 2017.
- **Sourav Mondal**, Goutam Das, and Elaine Wong, "CCOMPASSION: A Hybrid Cloudlet Placement Framework over Passive Optical Access Network," *in IEEE Conference on Computer Communications (INFOCOM)*, Honolulu, HI, USA, April 2018.
- **Sourav Mondal**, Goutam Das, and Elaine Wong, "Support of Low Latency Applications through Hybrid Cost-Optimised Cloudlet Placement," *in 2018 20th International Conference on Transparent Optical Networks (ICTON)*, Bucharest, July 2018.

## Publications arising from Chapter 4 of Thesis

### Journals

- **Sourav Mondal**, Goutam Das, and Elaine Wong, "An Analytical Cost-Optimal Cloudlet Placement Framework over Fibre-Wireless Networks with Quasi-Convex Latency Constraint." *Electronics*, vol. 8, no. 4, 404, 2019.
- **Sourav Mondal**, Goutam Das, and Elaine Wong, "Efficient Cost-optimization Frameworks for Hybrid Cloudlet Placement over Fibre-Wireless Networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 11, no. 8, pp. 437-451, August 2019.

## Publications arising from Chapter 5 of Thesis

### Journals

- **Sourav Mondal**, Goutam Das, and Elaine Wong, "Computation Offloading in Optical Access Cloudlet Networks: A Game-Theoretic Approach," *IEEE Communications Letters*, vol. 22, no. 8, pp. 1564-1567, August 2018.

- **Sourav Mondal**, Goutam Das, and Elaine Wong, "A Game-Theoretic Approach for Non-cooperative Load Balancing among Competing Cloudlets," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 226-241, 2020.

- **Sourav Mondal**, Goutam Das, and Elaine Wong, "Centralized and Decentralized Non-Cooperative Load-Balancing Games among Competing Cloudlets," *IEEE/ACM Transactions on Networking*, April 2020. (under review)

## Publications arising from Chapter 6 of Thesis

### Journals

- **Sourav Mondal**, Lihua Ruan, Martin Maier, David Larrabeiti, Goutam Das, and Elaine Wong, "Enabling Remote Human-to-Machine Applications with AI-Enhanced Servers over Access Networks," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 889-899, 2020.

### Conferences

- **Sourav Mondal**, Lihua Ruan, and Elaine Wong, "Remote Human-to-Machine Distance Emulation through AI-Enhanced Servers for Tactile Internet Applications," *in The Optical Networking and Communication Conference & Exhibition (OFC)*, March 2020.

- Elaine Wong, **Sourav Mondal**, and Lihua Ruan, "Alleviating the Master-Slave Distance Limitation in H2M Communications through Remote Environment Emulation," *in International Conference on Optical Network Design and Modeling (ONDM)*, May 2020.

## Other related Publications

### Conferences

- Lihua Ruan, **Sourav Mondal**, and Elaine Wong, "Machine Learning based Bandwidth Prediction in Tactile Heterogeneous Access Networks," *in IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Honolulu, HI, USA, April 2018, pp. 1-2.

- Lihua Ruan, **Sourav Mondal**, and Elaine Wong, "Low-Latency Federated Reinforcement Learning-Based Resource Allocation in Converged Access Networks," *in The Optical Networking and Communication Conference & Exhibition (OFC)*, March 2020.

# Chapter 2

# Literature Review

"Research is to see what everybody else has seen, and to think what nobody else has thought."

Albert Szent-Gyorgyi

## 2.1 Introduction

Over the past few years, the focus of Internet technologies has started to shift from storage and distribution of large-scale data to ultra-low system latency with support for context-awareness and mobility. This shift is motivated by the massive growth of application-based service demands of smart mobile devices e.g., augmented reality, on-line gaming, real-time environmental monitoring and Tactile Internet (TI) applications. The TI finds enormous potential applications in the fields of the automation industry, autonomous driving, robotics and healthcare, just to name a few [11]. These applications require end-to-end latency of the order of 1-100 ms or less, extremely reliable telecommunication infrastructure for prompt exchange of control messages and perseverance of security and privacy of the data [12].

However, it is observed that to execute such applications, lot of computational, storage and battery resources are required, and mobile devices suffer from resource poverty due to limited capacity for such resources. With mobile cloud computing, mobile devices can offload their computational jobs to remote cloud servers to compensate for computational resources as well as extend battery life, known as "cyber

foraging" [13]. It is defined as a mechanism to augment the computational, storage, and energy capabilities of mobile devices through the opportunistic discovery of servers in the environment. However, there is a trade-off between the amount of required computation and transmission latency. If the computation is large and transmission time is reasonable, then the best decision for the mobile devices is to offload, whereas if the computation amount is less and transmission time is more, then it is better not to offload. For intermediate conditions, an optimal decision needs to be taken depending on bandwidth and other network constraints [5]. In the presence of IoT and other mobile devices, quick allocation of bandwidth resources and prompt to-and-fro response time between edge devices and cloud server will become highly essential. However, most often the round-trip-time (RTT) from cloud servers becomes a bottleneck due to large geographic separation between the edge devices in the access and cloud servers situated beyond core networks, which leads to degradation of QoS and Quality of Experience (QoE). To overcome this hurdle, a new edge computing (EC) layer, sandwiched between access layer and cloud computing layer is introduced and examples of a few recent such implementations are the fog computing, the multi-access edge computing and the cloudlet computing [14].

Mobile devices can access the cloudlet computing nodes through wireless access technologies like long term evolution-advanced (LTE-A), wireless fidelity, millimetre wave (mmWave), 5G new radio (NR), to name a few. Besides, popular optical access technologies like TDM-PON and wavelength-division multiplexed passive optical network (WDM-PON) and converged FiWi technologies like radio over fibre (RoF) and radio and fibre (R&F), cloud radio access network (C-RAN) are also useful due to their wide deployment coverage, low cost per bit, very high data transmission bandwidth, efficient network virtualisation and network scalability [15]. The primary topics reviewed in this chapter, that are relevant to this thesis, are summarised in Fig. 2.1. We firstly review some of the latest optical, wireless, and fibre-wireless access technology standards. We then discuss the importance of cloudlets towards the feasibility of low-latency applications. We also identify some of the primary research problems with cloudlets and discuss some of the recent works done to address these

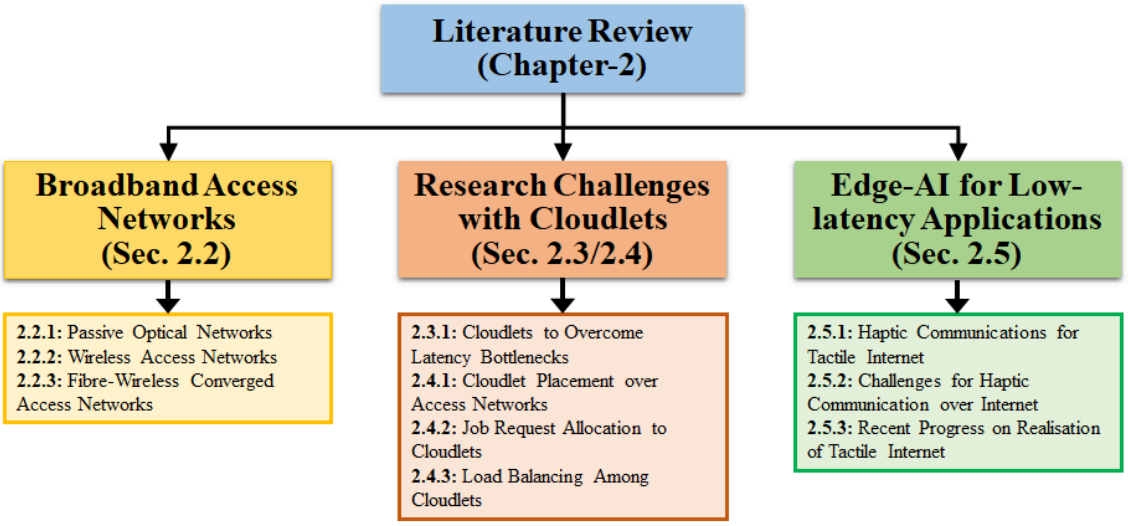



Figure 2.1: An overview of the relevant topics reviewed in this chapter.

problems. Moreover, we discuss the characteristics and networking challenges of recently evolved ultra-reliable and low-latency immersive TI applications.

The rest of this chapter is organised as follows. In Section 2.2, we discuss state-of-the-art broadband access technologies and their key features. In Section 2.3, the role of cloudlets in the practical deployment of low-latency applications. In Section 2.4, we briefly discuss some of the recent works done to address various research challenges related to cloudlet deployment. In Section 2.5, we discuss the characteristics of immersive TI applications, networking challenges and recent progress to deploy such applications over the Internet.

## 2.2 State-of-the-Art Broadband Access Networks

The last mile user access networks can be broadly categorised into wired and wireless access networks. In this section, we discuss the state-of-the-art wired and wireless access technologies and their primary features. The digital subscriber line (DSL) [16] and cable modem [17] are some of the very first generation wired broadband access network technologies. In many countries, asymmetric DSL (ADSL) [18] and very-high-speed DSL (VDSL) [19] are still being used to provide broadband access in residential areas. Nonetheless, after the introduction of optical fibre communication in the access network, the popularity of PON started to grow as it can provide a

very high data rate and a much longer reach. In addition to this, wireless broadband access technologies also came across a long way in the past few decades in terms of their bandwidth efficiency. Furthermore, wireless technologies are able to support the mobility of end-users and provide broadband access to remote areas where wired access technologies are infeasible.

### 2.2.1 Passive Optical Access Networks

Any state-of-the-art PON architecture primarily consists of an optical line terminal (OLT) at the service provider's CO and several optical network units (ONUs) near end users [20]. Any active element is not used in the optical path between the CO and its corresponding OLTs. A typical architecture of a TDM-PON is shown in Fig. 2.2. An OLT aggregates all optical signals from different service providers subscribed by the ONUs into a single multiplexed wavelength. It acts as an interface between PON and the backbone network. On the other hand, ONUs connect end-user devices to the TDM-PON and provides optical to electrical signal conversion. A passive splitter situated at the remote node (RN) splits the feeder fibre emanating from OLT into several distribution fibre paths that terminate at an ONU placed near customer premises. A standard splitter usually has a split ratio of 1:4, 1:8, 1:16, 1:32, 1:64, and so on. The number of ONUs connected to the OLT is dependent on this ratio. The split-ratio of passive splitters indicates the bandwidth available to each ONU and a higher split-ratio indicates lower bandwidth per ONU because the same channel bandwidth is shared amongst more number of ONUs. Usually, downlink signals follow a broadcast protocol, and encryption techniques are used to prevent eavesdropping. Uplink signals from multiple ONUs are combined using TDM access protocol such that each ONU transmits uplink data in its timeslot and any data collision is avoided. Through dynamic bandwidth allocation (DBA) protocols, the throughput of TDM-PONs can be further increased several times by exploiting statistical multiplexing [21].

Based on the location of the optical end terminals, the fibre-based passive access networks are termed as fibre-to-the-x (FTTx) architectures, e.g., FTTC, FTTB, and

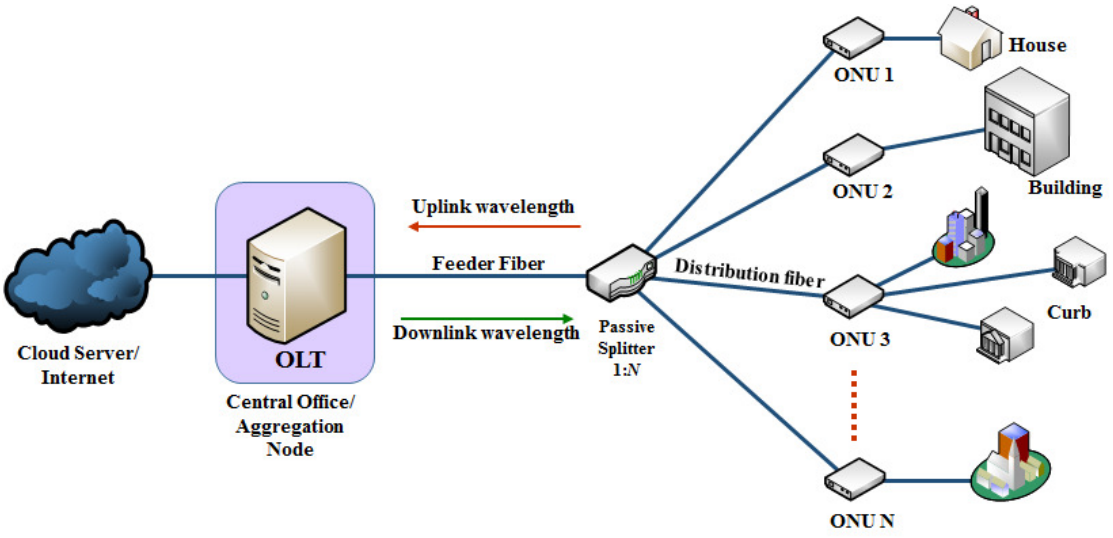



Figure 2.2: The basic network architecture of a TDM-PON.

FTTH, if the fibre is terminated at a curb, at a building, or house premises, respectively. Thus far, many variations of PON such as TDM-PON, wavelength division multiplexing (WDM) PON, orthogonal frequency division multiplexing (OFDM) PON, sub-carrier multiplexing (SCM), and code division multiplexing (CDM) PON are introduced [22]. Although TDM-PON and WDM-PON are equally suitable for practical deployment [23], TDM-PON gained became more popular to service providers due to its cost-effectiveness and ability to support higher bandwidth [24]. Therefore, we primarily consider TDM-PON for our work in this thesis. For uplink and downlink data transmissions in a TDM-PON, two distinct wavelengths are used. The currently deployed TDM-PON standards include ATM-PON (APON), Broadband PON (BPON), Ethernet PON (EPON), Gigabit PON (GPON), 10G-EPON, and Next-generation PON (NG-PON) that provide different data rates, e.g., asymmetric 10G-PON (XG-PON1) has 10 Gbps downlink and 2.5 Gbps uplink, but symmetric 10G-PON (XG-PON2) has both 10 Gbps downlink and uplink. The network architectures of APON/BPON, GPON, and NG-PON were standardised by the Full Service Access Network (FSAN), an affiliation of network operators and telecom vendors [25]. The EPON and 10G-EPON are standardised by the Institute of Electrical and Electronics Engineers (IEEE) 802 study group [26].

### 2.2.2 Wireless Access Networks

Till date, various wireless access technologies are also developed by IEEE and the Third Generation Partnership Project (3GPP). A typical wireless access network architecture is shown in Fig. 2.3. Wireless device users connect to the access network through the wireless base stations (BSs), which are connected to the wireless core network through a backhaul or fronthaul network. The wireless core network can provide connectivity with other networks through gateways. It also consists of management entities that are responsible for managing wireless users connected to the network. Next, we briefly discuss the key features of some recent state-of-the-art wireless broadband access network technologies.

#### Long Term Evolution (LTE) and LTE-Advanced (LTE-A)

LTE was first proposed by NTT-Docomo Japan in 2004 and it is an all-IP based network [27]. The core network of LTE is known as the evolved packet core (EPC), which is designed to replace the GPRS Core Network, consists of packet gateways and a mobility management entity. The wireless devices to access the network is known as user equipment (UE) and the BSs are known as evolved-nodeBs (eNBs) [28]. The eNBs allocate radio resources for the UEs and grant them access to the core network. The eNBs and the EPC are also connected through an interface known as S1 interface. Moreover, the neighbouring eNBs are connected via a new interface

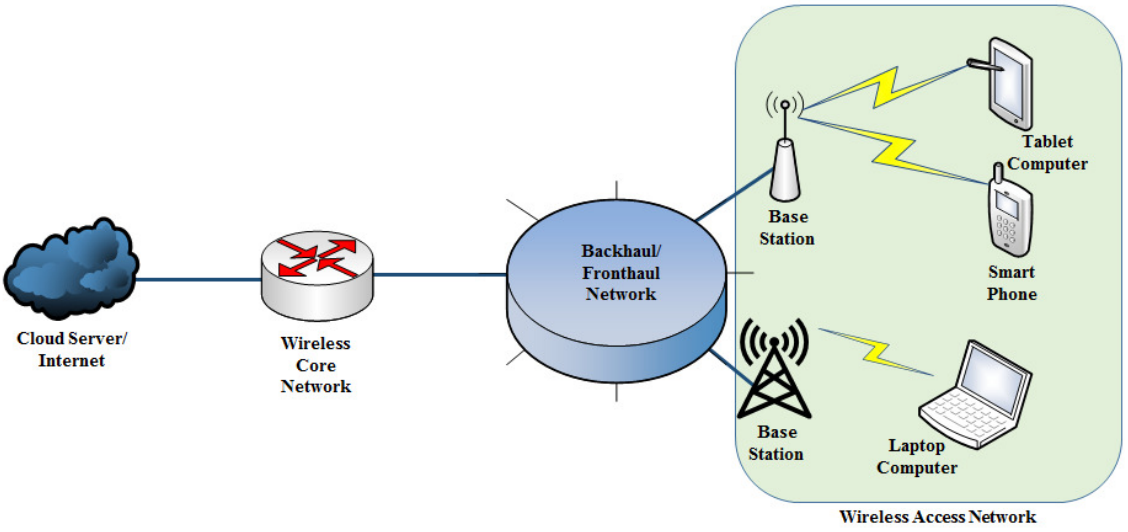



Figure 2.3: The basic network architecture of a wireless access network.

known as X2. The OFDM multiple access technology is used in the downlink and the single-carrier frequency division multiple access (SC-FDMA) technology is used in the uplink of the LTE network [29]. The LTE networks are expected to provide a downlink peak rate of 300 Mbps, an uplink peak rate of 75 Mbps, and QoS provisions that permit a transfer latency of less than 5 ms. In addition to this, LTE can manage fast-moving mobile devices with seamless handovers for both voice and data, supports scalable carrier bandwidth within 1.4 MHz to 20 MHz, and supports both frequency division duplexing and time-division duplexing [30].

Nonetheless, LTE could not successfully meet all the requirements of 4G standard such as a peak data rate of 1 Gbps and hence, was called 3.9G (beyond 3G but pre-4G). Therefore, an evolved standard known as LTE-A is proposed that supports coordinated multi-point (CoMP) data transmission and reception, $2 \times 2$ MIMO, scalable bandwidth from 20 MHz to 100 MHz, carrier aggregation (CA) of the contiguous and non-contiguous spectrum, hybrid OFDMA and SC-FDMA in the uplink, cognitive radio, to name a few. The CA functionality helps to make better use of existing multi-antenna techniques (MIMO) and support for relay nodes [31]. This advanced standard is expected to provide a downlink peak rate of 1 Gbps, an uplink peak rate of 500 Mbps, and at least 3 times greater spectrum efficiency than LTE. With the CA functionality, many service providers that do not have sufficient contiguous spectrum to support the required bandwidths for the very high data rates can utilise multiple channels from either in the same bands or different areas of the spectrum. Another functionality is LTE relaying, that enables signals to be forwarded by remote stations from the main base station for coverage improvement. The CoMP functionality introduces several techniques for dynamic coordination of data transmission and reception between a UE and a variety of different base stations to increase the overall utilisation of the network. Furthermore, LTE device-to-device (D2D) functionality is also introduced, particularly for the enabling fast swift direct communication for emergency services [32].

### Wireless Fidelity (WiFi)

WiFi is a very popular wireless local area network (WLAN) technology from the IEEE 802.11 family of standards. Initially, WiFi was introduced to facilitate wireless access only in private environments, e.g., in a house or an office. However, now it is also being used in many public places to provide broadband access [33]. Note that these WiFi nodes are needed to be supported by wired or wireless backhaul/fronthaul networks. It is interesting to note that despite a shorter reach of the WiFi network compared to LTE and LTE-A networks, it is identified as a very cost-efficient solution to provide high bandwidth support to end-users [34]. Usually, an indoor WiFi access point (or hotspot) has a range of about 20 metres, whereas some modern access points that are used for outdoors, are capable of covering up to 150-metre. The classical 802.11 protocol was designed to support a maximum data rate of 2 Mbps for 2.4 GHz WiFi with frequency-hopping spread spectrum (FHSS) and direct sequence spread spectrum (DSSS) modulation techniques. Followed by 802.11a and 802.11b standards are published that operate on 5 GHz and 2.4 GHz bands, respectively. Note that 802.11a protocol is OFDM based with 54 Mbps maximum data rate and 802.11b provides 11 Mbps maximum data rate. Finally, in 802.11g protocol is published that has similar functionality as 802.11a but operates at 2.4 GHz band [35]. Nonetheless, along with several advantages, one major disadvantage of WiFi is that it is potentially more vulnerable to cyber-attacks than wired networks. The reason behind this is that any malicious user within range of a WiFi can attempt to access the network with a wireless network interface controller [36].

### Millimeter Wave (mmWave)

Currently, the 4G network spectrum is spanned from 600 MHz to 3 GHz, but studies show that this spectrum is insufficient to meet the 5G QoS requirements. Therefore, millimetre wave (mmWave) bands ranging from 30 to 300 GHz, started to gain significant research interest recently because of extremely wide available bandwidths [37]. However, this entire band is not equally useful, because severe signal attenuation is observed in the 60 GHz, 120 GHz, 180 GHz bands. Only a few special bands

like 35 GHz, 94 GHz, 140 GHz, and 220 GHz are most commonly chosen as they experience lesser attenuation [38]. The mmWave technology facilitates large antenna arrays to be packed in small physical dimension due to their short wavelengths and narrow beams. Recently IEEE designed 802.11ad protocol at the 60 GHz carrier frequency band. It can support a maximum data rate of 4.6-7 Gbps, depending on the data modulation format used [39]. This protocol can stream high definition audio and video data but within a much shorter range than WiFi. Furthermore, mmWave channels of 28 GHz, 38 GHz, 60 GHz, and 73 GHz with line-of-sight (LOS) and non-line-of-sight (NLOS) transmissions are designed for both indoor and outdoor environments [40].

### 5G New Radio (NR)

Some of the latest usage scenarios defined for 5G wireless systems are Enhanced Mobile Broadband (eMBB), Ultra-Reliable Low Latency Communications (uRLLC), and Massive Machine Type Communications (mMTC) [41]. The eMBB in 5G can be considered as an evolution from 4G LTE standard, with faster connections, higher throughput, and more capacity. The uRLLC is proposed to support mission-critical applications that require seamless, reliable, and robust data exchange over the network. The mMTC is designed to connect a large number of low power, low-cost devices to the Internet for high scalability and increased battery lifetime over a wide area. Recently, 3GPP has standardised several latency-reducing and reliability-enhancing features especially to address uRLLC services through 5G New Radio (NR) radio interface [42]. Similar to 3GPP LTE standard, OFDM waveform is used in NR system also, but NR provides more flexible numerology as different sub-carrier spacings are used for signal generation, thus leading to OFDM symbols of different lengths. In LTE, the OFDM sub-carrier spacing is 15 kHz whereas a sub-carrier spacing of 120 kHz is used in NR. Thus, a 14-symbol transmission slot is very easily reduced from 1 ms to 125 $\mu$s. Moreover, mini-slots are introduced for uRLLC traffic to use even shorter time slots and preemption of other ongoing traffic is also allowed for uRLLC traffic to reduce queueing latency at the transmitter. It is also observed

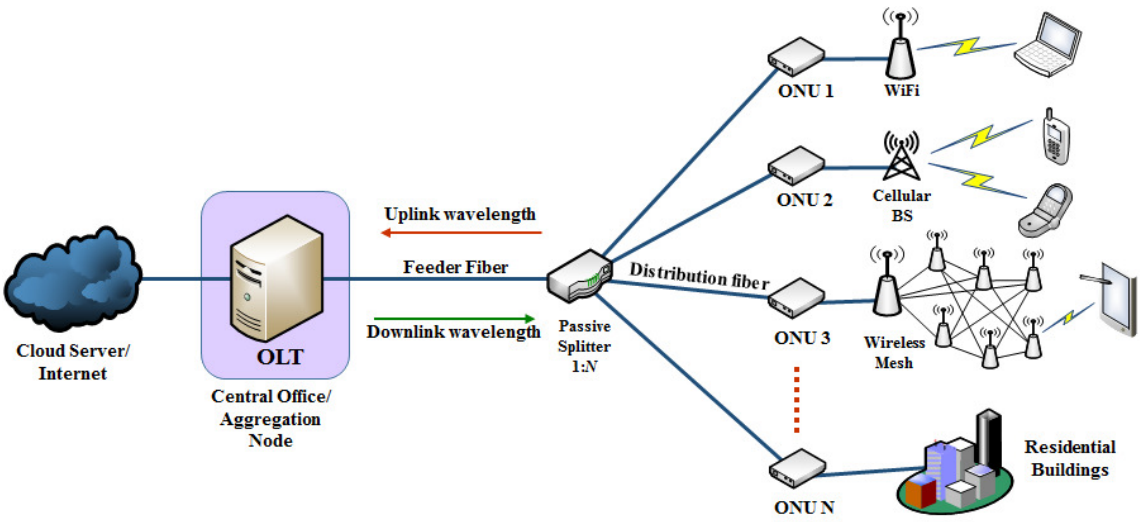



Figure 2.4: The basic network architecture of a fibre-wireless access network.

that depending on the user configuration, the lowest achievable one-way transmission latencies that can be guaranteed with high reliability over the NR radio access network range from sub-millisecond level to a few milliseconds [11]. Different NR configurations also lead to different spectral efficiencies and coverage levels of the access network [43]. The NR usually includes lower frequencies (FR1), below 6 GHz, as well as higher frequencies (FR2), above 24 GHz.

### 2.2.3 Fibre-Wireless Converged Access Networks

FiWi networks combine the capacity of optical fibre networks with the ubiquity and mobility of wireless networks to provide a diverse platform for emerging future network applications [44]. A typical fibre-wireless access network architecture is shown in Fig. 2.4. In RoF networks, optical fibre is used as an analog transmission medium between a CO and wireless access points (WAPs) associated with ONUs, and a central BS controls the access to both optical and wireless channels. The primary advantages of using fibre optical links over all-electrical signal transmission are lower transmission losses and lower noise and electromagnetic interference. On the other hand, in R&F networks, access to optical and wireless channels are controlled separately at their interface. We provide a brief review on the recent developments of RoF and R&F technology standards as follows.

### Radio over fibre (RoF) Networks

In RoF networks, mainly two types of data transmission systems are used, e.g., radio frequency (RF)-over-fibre and intermediate frequency (IF)-over-fibre, depending on the range of frequency of the radio signals. In RF-over-fibre systems, the high RF signals are directly generated in the optical domain for distribution to BSs, where optical-to-electrical conversion is performed. Therefore, frequency up-down conversions at wireless BSs are avoided and simple, cost-effective implementation is possible [45]. However, in IF-over-fibre systems, IF radio signals are used for modulating optical signals for transmission through optical links and converted to electrical signals at the BS for radio transmission. Modulation of the optical signals are performed by techniques like four-wave mixing in a highly nonlinear dispersion-shifted fibre (HNL-DSF), cross-phase modulation in a nonlinear optical loop mirror in conjunction with a straight pass in HNL-DSF, or all-optical up-conversion employing cross-absorption modulation in an electro-absorption modulator [46]. Nonetheless, external intensity and phase modulation schemes are the most practical solutions due to low cost, simplicity, and long-distance transmission performance.

### Radio and fibre (R&F) Networks

In R&F based FiWi networks, cascaded TDM IEEE 802.3ah EPON, IEEE 802.3av 10G-EPON or WDM-PON are used for backhaul and IEEE 802.11 a/b/g/n/s WLAN mesh frontend is used [47]. In the optical network segment, directly modulated external cavity lasers, multi-section distributed feedback (DFB)/ distributed Bragg reflector (DBR) lasers, and tunable vertical-cavity surface-emitting laser (VCSEL) is used for data transmission units. Tunable receivers are realised by using a tunable optical filter and a broadband photo-diode. Moreover, colourless ONUs based on reflective semiconductor optical amplifiers (RSOAs), which remotely modulate centralised optical signals, burst-mode laser drivers and burst-mode receivers are also used with high sensitivity, wide dynamic range, and fast time response to arriving data bursts.

#### Cloud-Radio Access Network (C-RAN)

The C-RAN was first introduced by China Mobile Research Institute in April 2010, which is a centralised, cloud computing-based wireless network architecture that supports 2G, 3G, 4G and all future wireless communication standards [48]. C-RAN has an evolved network architecture that takes advantages from several recently developed wireless, optical, and IT communication systems, e.g., mmWave, coarse or dense wavelength-division multiplexing (CWDM/DWDM) technology, to name a few. It also integrates the latest data centre technologies to facilitate a low cost, high reliability, low latency and high bandwidth interconnected network with the base-band unit (BBU) pool. In C-RAN, several remote radio heads (RRHs) are connected to a centralised BBU pool and the maximum allowed distance is 20 km in fibre link for 4G (LTE/LTE-A), and (40 km 80 km) for 3G wideband code division multiple access or time division synchronous code division multiple access (WCDMA/TD-SCDMA) and 2G global system for mobile communication or code division multiple access (GSM/CDMA) standards. Neighbouring BBUs can communicate with each other with a very high data rate ($\geq$ 10 Gbps) and low latency ($\sim$ 10 $\mu$s) communication link [49]. This is a very important advantage of C-RAN as it can implement LTE-Advanced features such as CoMP with very low latency among multiple RRHs. It is also interesting to note that C-RAN utilises open platforms and real-time virtualisation with the aid of cloud/edge computing to achieve dynamic shared resource allocation for supporting a multi-vendor environment [50].

## 2.3 Cloudlets for Low-latency Applications

A cloudlet is expected to provide all the basic functionalities provided by a cloud server. The cloud computing paradigm can be summarised as a layered "everything as a service" (XaaS) architecture, consisting of the following layers: (a) Software as a Service (SaaS): The end-users can run all applications directly on a cloud server by using this layer, (b) Platform as a Service (PaaS): The PaaS layer provides an application or development platform on which end-users can create SaaS applications/services, (c) Infrastructure as a Service (IaaS): The underlying resource on

which PaaS/SaaS rely on, i.e. storage, computing and networking, are provided by this lowest level. The five essential characteristics of the cloud computing are viz., on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service; the service models are SaaS, PaaS and IaaS; and the deployment models are private cloud, community cloud, public cloud and hybrid cloud [6].

Recently, Akamai launched five cloudlet solutions which are located at the network edge and are capable of executing different function modules [51]. The edge redirector cloudlet provided services enable redirecting uniform resource locator (URL) requests so that these requests need not access the origin data centre every time to retrieve the new uniform resource locator URL, which saves the RTT to a great extent. The visitor prioritisation cloudlet enables the cloudlets to maintain a visitor prioritisation. All the visitors are routed to the application server directly when the visitor traffic to the application server is normal. The image converter cloudlet enables the service provider to append application programming interface (API) commands to image URLs. The images can be dynamically manipulated on the cloudlet through the API commands. The Forward Rewrite Cloudlet helps the service providers to improve the user experience by showing the desired URL in the visitors address bars while delivering the web content from the original URL. With IP/geo access cloudlet, access control service is provided to cloudlet for easy-to-manage white-list and black-list according to the IP address or geographic location. In the next subsections, we briefly discuss the system requirements for low-latency applications and the role of cloudlets in their deployment.

### 2.3.1 Cloudlets to Overcome Latency Bottlenecks

Long before the proposal of cloudlets by a team of researchers led by M. Satyanarayanan, researchers were studying the impact of distance on end-user experience latency-sensitive applications. The authors of [52] made such studies with transient display applications. Both remote cloud and single-hop cloudlets were used to offload computational tasks in their experiment. A typical cloudlet can be imagined as a commercial personal computer or workstation on which a customised virtual machine (VM) is running. Through experimental results, the authors of [53] confirmed

through quantitative evidence that execution of the offloaded task by a nearby computer is always more latency efficient over a distant server for five example applications, e.g., face recognition (FACE), speech recognition (SPEECH), object and pose identification (OBJECT), mobile augmented reality (AUGREAL), physical simulation and rendering (FLUID). Amazon EC2 service provided data centres located in Virginia, Oregon, Ireland and Singapore were used. A cloudlet equivalent machine "1-WiFi" was installed at their university campus in Pittsburgh, PA and purposefully a machine, relatively slower than Amazon data centres were installed. Still, they observed a better latency-efficient performance from 1-WiFi in most of the cases, e.g., an augmented reality application when executed on a nearby cloudlet, experiences a response time up to 100 ms while the same application experiences 800 ms when running on a distant Amazon EC2 server. Such experiments motivated researchers in concretising the idea of edge computing servers like cloudlets that can facilitate low-latency applications over the Internet by overcoming the latency bottleneck.

In addition, the performance of cloudlets can be improved in several ways by studying and improving different metrics. The authors of [54] examined the cloudlet access probability, task success rate and task execution speed to evaluate their impact on the performances of mobile users. The authors of [55] focused on improving the overall performance of cloudlet systems by the accelerating the "dynamic VM synthesis" process on cloudlets. They proposed a novel just-in-time method that provisions a cloudlet with a non-trivial VM image in 10 seconds. The basic approach for dynamic VM synthesis is explained as follows:

- The VM image used for offloading is referred to as a "launch VM" and it is created by installing the relevant software into a "base VM".
- The compressed binary difference between the base VM image and the launch VM image is known as a "VM overlay". This idea of the binary difference between VM images is used to reduce storage and network transfer costs.

- During execution, dynamic VM synthesis reverses the process of overlay creation. Typically a mobile device offloads the VM overlay to a cloudlet that already contains the corresponding base VM.
- The cloudlet decompresses the overlay and by applying it to the base VM, derives the launch VM to create a VM instance from it.
- After this, the mobile device can start periodic offloading operations on this instance. When the session ends, the instance is also destroyed, but the cloudlet can preserve the launch VM for future use.

It is interesting to note that the cloudlet and mobile device can have different hardware architectures and no restrictions are imposed on a guest operating system of the base VM, i.e., they can work with both Linux and Windows. The authors of [56] presented a new concept "offload shaping" that reduces the offloaded workload for the mobile devices. This technique helps to reduce the resource demand of mobile devices very much significantly, without losing any application-level fidelity. The authors of [57] further demonstrated that by adopting efficient VM provisioning algorithms viz., optimised VM synthesis, application virtualisation, cached VM, cloudlet push and on-demand VM provisioning, the computation latency can be reduced to a great extent for tactile applications.

## 2.4 Research Challenges with Cloudlets

Note that cloudlet computing systems are essentially distributed computing systems, and in any distributed computational networked system, placement of computational nodes over a network, job request allocation from end-user devices to computational nodes, and load balancing among neighbouring computational nodes are fundamental research problems. Primarily, centralised and decentralised control models are used in the existing literature to address these problems [58]. A common objective function and a series of constraints are formulated in centralised optimisation method based models to determine the optimal solution. These models can provide quick and efficient solutions to the problem, but they are hard to implement over a large

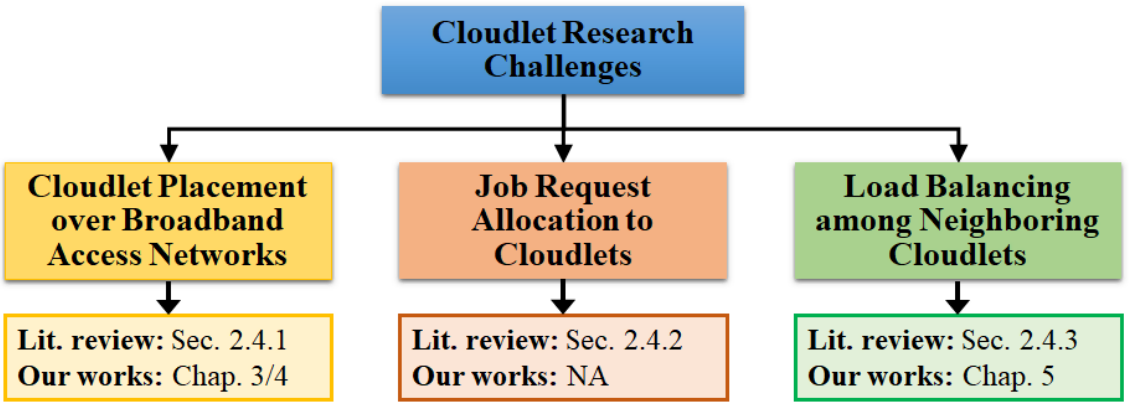



Figure 2.5: Primary research problems with cloudlet computing networks.

geographically spanned or heterogeneous network. On the contrary, in decentralised control models, all distributed nodes exchange their local control information among themselves and determine the optimal solution without any central controller node. Although for large networks such models are more robust, they cause inefficient sharing of control messages and computational burden on the network. In Fig. 2.5, we summarise the research problems related to cloudlet computing networks and in the following subsections, we briefly review some recent works done on networking problems with cloudlets.

### 2.4.1 Cloudlet Placement over Access Networks

The authors of [59–61] proposed computation offloading frameworks for energy saving in edge devices, assuming that cloudlets are already deployed over access networks. However, usually, the user distribution under a typical wireless network deployment is very complicated. Also, in crowded areas cloudlets can be accessed by a large number of mobile users, whereas in sparsely populated areas only a few users will intend to access cloudlets. Thus, optimal placement of cloudlets over wireless access networks and placement of an optimal number of VMs in cloudlets appeared to be essential research problems to several researchers, because these aspects improve the cloudlet resource utilisation significantly [62]. The authors of [63, 64] proposed an edge-cloud network design framework that determines where to install cloudlet facilities among available sites. In [65–67], the authors studied the cloudlet placement problem over a large-scale wireless metropolitan area network (WMAN) with

many wireless access points and formulated an integer linear programming (ILP). The authors also proposed heuristic algorithms with guaranteed approximation ratio to solve the ILP. The authors of [68] proposed an optimisation framework, known as the Cost Aware cloudlet PlAcement in moBiLe Edge computing strategy (CAPABLE), to minimise the cloudlet deployment cost against the end-to-end latency. In [69], the authors proposed a profit maximisation framework, PRofIt Maximisation Avatar pLacement (PRIMAL), that minimises the cost of VM avatar migration subject to the migration gain, i.e., reduction of the end-to-end latency. In addition, the authors of [70, 71] identified that optical access networks can be useful in the next few decades to support edge computing technologies. In [72], the authors studied the performance of centralised and decentralised bandwidth allocation algorithms in a long-reach optical access network to investigate the feasibility of computation offloading to edge-computing servers and develop an analytical framework to validate against simulated results.

### 2.4.2 Job Request Allocation to Cloudlets

Note that cloudlets are placed over an access network, depending on the average or maximum mobile user density. However, due to the mobility of mobile devices, cloudlets at different parts of the network become overloaded and under-loaded at different times. Thus, assigning job requests to the most suitable cloudlets is the next important research challenge. In addition to the heuristic for cloudlet placement ILP, the authors of [65, 66] devised online algorithms to dynamically allocate job requests from mobile devices to different cloudlets. The authors of [63, 64] also proposed heuristics to assign sets of access points to cloudlets that supports VM orchestration as well as satisfies service-level agreements. The authors [68] designed a job allocation scheme for minimising the end-to-end latency of user requests by assigning the job requests from each region to suitable cloudlets. The authors of [73] proposed a linear programming solution for computation offloading by considering the QoS requirements of mobile users while maximising the revenue of service providers. In [74, 75], the authors developed non-cooperative game-theoretic models for computation offloading among mobile devices to mobile cloudlet servers, proved

the existence of the Nash equilibrium (NE) of the formulated games, and provided polynomial-time algorithms for computing NE solutions. Moreover, the authors of [76] proposed another computation offloading non-cooperative game among multiple mobile users and computed the NE of the game by showing it as an exact potential game. They further proposed a reinforcement learning-based algorithm to compute the NE solution. The authors of [77] proposed a price-based distributed computation offloading framework, formulating a Stackelberg game to model the interaction among edge cloud and mobile users. In this game formulation, the edge cloud demands a price to maximise its revenue subject to its computational resources.

### 2.4.3 Load Balancing Among Cloudlets

Load balancing among edge computing nodes such as cloudlets is a major research issue and some researchers recently proposed load balancing models based on optimisation and game-theoretical methods. The authors of [78] compared the latency performance of three load balancing systems through closed-form expressions, i.e., no sharing, random sharing, and less loaded sharing with different degrees of cooperation between neighbouring cloudlets. A centralised problem of latency minimisation is formulated by the authors of [79] and proposed a network-flow based heuristic algorithm for solving it. Recently, there is a growing interest in applying cooperative and non-cooperative game-theoretical models to various network-related issues, as game theory offers many effective tools for evaluating and researching the relationship between distributed agents in conflict and cooperation [80]. Furthermore, reinforcement learning algorithms also seem to be a valuable approach to solving load balance problems but can present different complexity and convergence issues in real-time [81]. The authors of [82] proposed a cooperative load balancing scheme where underloaded cloudlets coordinate with their neighbouring overloaded cloudlets to reduce their blocking probability and processing latency. The authors of [83] proposed a distributed non-cooperative load balancing game in small cell networks among the neighbouring cloudlets, and compared its findings with a centralised load balancing system that leverage the Lyapunov-drift technique to maximise the long-term system performance. By identifying the estimated latency as the dis-utility function of

every cloudlet, the authors of [84] devised a non-cooperative load balancing game where cloudlets try to minimise its disutility and proposed an iterative proximal algorithm to compute the NE solution. In this framework, none of the cloudlets is allowed to offload until their incoming job requests reach a certain threshold.

## 2.5 Edge-Intelligence for Ultra-reliable and Low-latency Communications

The recent advent of the Tactile Internet (TI) [11], with an intention to provide ultra-low latency and ultra-high reliability communications, has shifted the traditional content-oriented communication to control-oriented communication. In general, the low-latency applications demand much more stringent end-to-end latency and more network bandwidth than conventional audio and video applications [85]. For some of the typical low-latency applications, an end-to-end latency of 1 ms with a packet loss rate of $\leq 10^{-2}$ is required [11]. In factory automation applications, where real-time control of machines are involved, a highly reliable communication link with end-to-end latency between 0.25 ms to 10 ms with a packet loss rate of $10^{-9}$ is required [86]. In intelligent transportation systems (ITS), different services like autonomous driving, road safety, and traffic efficiency services are involved with different latency requirements. Overall, an end-to-end latency of 10 ms to 100 ms is required with a packet loss rate of $10^{-3}$ to $10^{-5}$ [87]. In future, for handling several dangerous areas in construction and maintenance sectors, remote-controlled robots and telepresence

Table 2.1: Typical end-to-end latency and data rate requirements of some recent low-latency applications

| ***Application*** | ***Latency*** | ***Data rate*** |
|---|---|---|
| Factory automation | 0.25-10 ms | 1 Mbps |
| Intelligent transportation systems | 10-100 ms | 10-700 Mbps |
| Robotics and Telepresence | 1 ms | 100 Mbps |
| Virtual/Augmented Reality | 1 ms | 1 Gbps |
| Health care | 1 ms | 100 Mbps |
| Online gaming | 1 ms | 1 Gbps |
| Education | 5-10 ms | 1 Gbps |

applications will be highly useful with a real-time synchronous visual-haptic feedback mechanism. However, to facilitate such applications, a high reliability/availability communication system is required that guarantees a system response time less than a few milliseconds including all network latencies [88]. Similarly, for virtual reality (VR) and augmented reality (AR) applications, where remote manipulation of objects in virtual or users field of view environments are performed, an end-to-end latency of 1 ms is required [89]. Recently, in the health care sector, applications like telediagnosis, telesurgery, and telerehabilitation started gaining popularity that requires an RTT latency of 1-10 ms and a highly-reliable data transmission link [11]. For immersive online gaming experience also, an end-to-end latency of 1 ms is required and latency of more than 30-50 ms usually results in cyber-sickness to users [85]. These low-latency applications are also very much useful in the education sector where remote training and interactive teaching is possible through multi-modal human-machine interfaces that perceive visual, auditory, and haptic information. A summary of end-to-end latency and data rate requirements of the above mentioned low-latency applications is outlined in Table 2.1.

### 2.5.1 Haptic Communications for Tactile Internet

In addition to delivering human intelligence, the human-to-machine (H2M) applications pave the path for the remote distribution of human expertise digitally, thus inaugurating "Internet of skills" [90]. For the realisation of such latency-sensitive and mission-critical H2M applications, TI appears to be highly relevant which involves a tight integration of control mechanisms with communication systems [91]. According to the H2M paradigm, human multi-sensory information must be shared for contact and connectivity with the remote world. A haptic communication system provides this platform for the H2M applications, by exchanging kinesthetic and tactile information, and the possibility of transmitting distant physical experiences globally. The human haptic feedback system actively processes both kinesthetic and tactile sensations. These two haptic sub-modalities are interpreted through different sensing pathways [92]. The feedback to the haptic codecs may be generated in one mode or the other, or both, depending on the situation for Tactile Internet use or

the device scenario. Note that these two modalities are also regarded separately concerning haptic technologies because separate principles of sensing and actuation apply. Nevertheless, for a human user, all forms of information are combined into a joint touch impression. In the following, we discuss the importance of kinesthetic and tactile information exchange in a few H2M applications.

(i) **Kinesthetic feedback in bi-lateral teleoperation:** Bilateral teleoperation with kinesthetic feedback is a classical example that involves the exchange of kinesthetic information, e.g., telemaintenance and telesurgery. In these applications, a human user is usually connected to a kinesthetic input/output device. A remote teleoperator is a robotic arm at a distant or dangerous environment with force sensors, a video camera, a microphone and some operating tools. It is observed that instead of considering the kinesthetic sub-modality only, better user experience can be obtained through the combination of kinesthetic and tactile feedback [93]. Furthermore, object surface properties can be remotely perceived better by a combination of low-frequency kinesthetic force feedback, high-frequency tactile signals, and thermal feedback [94].

(ii) **Tactile feedback in e-commerce:** Presenting object surface properties on touch screens helps innovative online-shopping implementations, which is called T-Commerce. Nonetheless, a high-fidelity T-Commerce scenario involves extra efforts in the collection, delivery, and display of product data. To provide users with high quality and detailed remote touch experience, there should be a full description of the entity that contains all applicable kinesthetic and tactile properties. Ideally, users should not be able to distinguish between feeling the actual object physically and through the online interface offered [95].

(iii) **Tactile feedback in telepresence:** Telepresence technologies of today (e.g., video conferencing or video chat) share high quality audio and video between two or more users. Although these devices satisfy, at least partly, the goal of immersing a user into a distant space and creating a certain degree of presence, they lack the opportunity to share contact sensations during interactions.

Therefore, if the consumers are fitted with tactile actuators, tactile feedback sensations like vibro-tactile stimuli can also be given [95].

(iv) **Kinesthetic and Tactile feedback enhanced virtual reality:** During the initial stages of deployment, VR systems only displayed users' visual and auditory information. However, the most natural reaction of users in a VR is an attempt to grasp and interact with objects. In recent years a significant number of wearable haptic technologies have introduced to facilitate such features [96]. Although, such systems to incorporate kinesthetic and tactile feedback typically lack the compatibility with a lightweight device.

#### 2.5.1.1 Characteristics of Human Tactile Perception

The authors of [97] and [98] described six different types of exploration patters that humans typically use to identify unknown objects. After the encounter with objects, its enclosure and contour reveal spatial information regarding the shape of the object and its coarse contour characteristics. Moreover, to measure the weight of objects, humans lift items. A static contact is used by the bare finger to define the thermal conductance. Pressing on the content exposes details about its rigidity. Finally, random sliding motions require the perception of the fine roughness and the friction properties of the surface of the material, often referred to as haptic texture.

The authors in [99] have defined five main tactile dimensions, based on the assessment of the adjectives used to define tactile information in previous studies. During sliding movements, friction between a bare finger and a surface causes the person to apply a specific lateral force [100]. Components of the friction models are usually the surface-specific force to break the adhesion with it, or the resistance needed to slip the bare finger [101]. Hardness perception results from specific exploration patterns such as tapping on an object surface, pinching an object, or pressing on the surface [102]. The thermal receptors in the human skin sense temperature conductivity [98]. The balance of ambient temperature and a material's thermal conductivity defines whether is experienced as warm or cold to the direct touch [95]. The thermal receptor response range lies between 5-45 $^{o}$C. The duplex nature of roughness [95], consisting of microscopic and macroscopic roughness, has

been described and confirmed in works like [98] and is based on the presence of different mechanoreceptors in the human skin. Microscopic roughness, also known as fine roughness, results from high-frequency vibrations during active surface-tool or surface-finger sliding motions. Vibrations between 40 and 400 Hz are perceived by the Pacinian corpuscles, also known as FA2 receptors [95].

#### 2.5.1.2 Collection and Presentation of Tactile Information

The first step toward the realisation of TI applications is to collect appropriate tactile information. With each of the five main tactile dimensions, we list potential sensing and actuation rules in the following.

(i) **Friction** involves normal and tangential contact forces (Coulomb friction model) [103] and is typically measured by using 3 degree-of-freedom (DoF) force sensors, or by the motor current needed to drag a linear stage [101], or as a combination of force-sensitive resistors [104].

(ii) **Hardness** is commonly defined as spring stiffness following Hooke's law [105] and can be expressed using force sensor values divided by the indentation depth measurements. Besides, it was found that acceleration signals help to reflect hardness as high-frequency components of tappings [106]. The authors of [107] have verified that these high-frequency tapping transients reported by the accelerometer can accurately reflect the dynamic portion of the material hardness and can be used for haptic display using vibrotactile actuators as well.

(iii) **Warmth** can be measured using thermistors [101] or non-destructive thermal camera-based infrared images [108, 109]. Note that thermal sensing of any system requires pre-heating of the object's surface, e.g. using a laser, to determine the temporal dissipation of thermal energy [108]. If a fluid is part of the sensing system, the thermal conductivity can be determined directly without the surface heating beforehand [101].

(iv) The research in [104, 110, 111] on **macroscopic roughness** used a laser scanner or an infrared reflective sensor to detect objects' surface texture during

non-contact scans. Height profiles are then produced to simulate the coarsely structural detail. Most recently, authors of [112] proposed stereoscopy-based approaches to the assessment of material surface structures.

(v) To capture **microscopic roughness**, conventional haptic rendering algorithms and tools can not produce high-fidelity reproductions of finger-surface interactions [113]. The motor drive circuitry as well as the device's inertia and stability, hinder their ability to accurately replicate high-frequency vibrations. As a result, the simulated surface projection also does not have haptic feedback, and instead feels flat and slick. Although collecting gradually shifting data, such as temperature and pressure, is often simple, accurate monitoring of high-frequency vibrations is a more difficult challenge as such signals are highly dependent on scanning force and scan speed [114].

It is possible to use multiple tactile interface systems to convey tactile information to human users and three major emerging developments are focused on vibro-tactile, ultrasonic and electrostatic actuation. Vibro-tactile actuators, available in different design modes such as voice coil actuators, eccentric mass motors, piezo-ceramic actuators or [115, 116] tactile pattern displays. Voice coil actuators are most commonly used to replicate high-frequency mechanical sounds within a range of 50 Hz to 1 kHz. During the surface activity, different tips of 3D printed steel tools can be used to capture vibrotactile data. The recorded signals are processed for further use, e.g. for implementation of compression systems, and then presented on the attached vibrotactile actuator for subjective assessment of the reported data trail, or objective test measurements using another acceleration sensor [95].

### 2.5.2 Challenges for Haptic Communication over Internet

In TI applications like teleoperation, telemanipulation, and telesurgery with haptic communication systems, visual and audio feedback are also required to be provided to the master subsystem from the slave subsystem. Thus, a very high packet rate is needed for making these applications feasible over the Internet. Therefore, Packet loss and variable latency present major challenges related to the management and synchronisation of the data streams as summarised below:

### Demand for large network bandwidth

Haptic sensor readings from kinesthetic devices are normally sampled, packed and transmitted at a rate of 1 kHz or higher to maintain system reliability and transparency [117]. In order to demonstrate the balance between the sampling rate, the overall stiffening rate is seen and the damping of a system, the authors of [118–120] performed stability analysis. A working teleoperation device with lower levels of sampling may somehow continue to work and an operator can somehow do a job. However, the average stiffness shown is lower than that of the higher sampled rate, but it retains system stability, so that the device may need more damping to stabilise hard contact. Kinesthetic information processing for teleoperation systems involves the transfer from a master to a slave of a thousand or more haptic data packets per second. This high volume of packets would result in vast quantities of network capacity being used in tandem with the transmission of audio and video data, resulting in inefficient data communication. Haptic data reduction or a reduction in packet rate is therefore required in teleoperation systems.

### Guarantee of System Stability

Teleoperation systems are particularly vulnerable to data loss and latency, meaning that the master and the slave devices must send and receive a packet every millisecond in order to guarantee the reliability of a haptic communication system [117]. In that context, the system latency and the quantity of data loss a computer can tolerate are essential questions. The authors of [121] showed that almost 90% data reduction is possible, whereas the stability of a bilateral teleoperation network can be disturbed with only a little latency. Teleoperation systems may display stability issues, which could lead to a degradation of teleoperation efficiency and mission performance, even with a small delay in communication or packet loss rate. This problem becomes unavoidable as the Internet becomes implemented across mobile networks. A key objective of telemanipulation systems is, therefore, to ensure system reliability and enhance QoS and QoE performance [122]. The task performance quality provides the quality of task and is typically quantified by a simple calculation of the completion time of the task. Certain efficiency metrics include the

number of squared forces, maximum forces, error/failure rate of the mission, the trajectory of a haptic system, movement range and speed [123]. At the other hand, the network infrastructure itself must be capable of ensuring sufficient resources and communication efficiency to best QoE and decrease the dependency at altering haptic information, if it is enhanced to the degree that all needs are met.

To address these issues, researchers identify three main solution spaces that can improve haptic communication systems, e.g., (i) the communication networks, that covers all network dimensions and mobile/wireless communication allowing the TI, (ii) data processing solutions for reducing data transmission using perceptual thresholds or methods of prediction to account for the latency induced by long-distance communication, and (iii) stability control solutions for reducing the effect of extra latency and maintaining the reliability of the control loop. Currently, research and development are being done to improve all the above-mentioned solution spaces of haptic communication.

Nonetheless, in this thesis, we keep our research focus only on communication networks to reduce the closed-loop latency between master and slave devices. Enhancement of the distance between the master and slave devices is a very important research challenge and provides a very stubborn bottleneck due to the propagation latency of data. Decoupling of haptic feedback from the control signal and forecasting of the haptic feedback samples from an intermediate server showing a great promise. However, the generation of proper contents of the haptic feedback samples need a thorough understanding of the type of application and haptic materials are involved. In the previous subsections, we have discussed some of the typical characteristics of the haptic materials, based on which we shall attempt to propose various haptic feedback sample, forecasting models.

### 2.5.3 Recent Progress on Realisation of Tactile Internet

Recently, some technical advances are made towards the realisation of low-latency and TI applications with the latest Internet technologies. Key technologies that facilitate this are software-defined networking (SDN), network function virtualisation (NFV), and edge computing, to name a few.

- The SDN is expected to be an important feature of 5G networks that decouples the control and data planes and provides locally centralised control to network elements. In a centralised control network, the management of traffic within a network becomes easier and mobility can be managed more efficiently [124] and with less latency while taking advantage of abstraction [125]. Also, software-based nature and programmability allow QoS to be delivered in accordance with granular and flow-based policies [126].

- With NFV, virtualisation and softwarisation of network functions are possible that reduce the dependency on hardware, which in turn, increases the scalability and reliability of the network. The NVF also simplifies sharing of resources among various network functions and transferring of network functions across the network to maximise the latency performance of a service [127].

- The authors of [128] showed that the decoupling of uplink control signals and downlink haptic feedback connections will provide extra reliability in heterogeneous networks. Edge-intelligence nodes like cloudlets can play an important role to deploy such schemes where haptic communications can be offloaded from the remote teleoperator [129]. By using the aforementioned NFV and cloudlets, "intelligence" can be efficiently distributed across network edges.

- The new radio standard supports services with diverse latency requirements. LTE standards currently offer 10 ms to 20 ms RTT between air interfaces only. Nevertheless, 5G standards require an end-to-end latency of less than 1 ms for the user plane [130]. Moreover, the implementation of massive MIMO would ensure that the bit error rate for reliable low latency communication is kept at minimum [131].

TI is undoubtedly an ever-evolving area which mainly combines robotics, telecommunications and data processing. According to [132], Tactile Internet systems need modern and updated technology. The next steps of the research work will help to improve the user experience and the teleoperation systems' effectiveness. Various control and communication techniques were developed to overcome the difficulties of haptic communication for time-delayed TI, as discussed in the previous sections.

Nonetheless, the facets of control and communication have been researched largely separately and through abstracting or neglecting essential properties of the underlying network. Therefore, a more holistic approach is needed to incorporate TI systems using practical networking facilities, whether wired or wireless IP networks. To achieve a reliable, consistent and effective system architecture, the implementation of real-world packet-switched networks involves a mutual analysis of the control and communication aspects. The state-of-the-art systems often vary in their robustness against various QoS network parameters and artefacts that are implemented into the device [133]. In general, there is no common understanding of the optimal architecture for certain QoS parameters, nor universally applicable effects on the necessary QoS parameters to achieve a certain level of TI system performance.

# Chapter 3

# Exact Cost-optimisation Frameworks for Cloudlet Placement over Fibre-Wireless Access Networks

> "Most practical questions can be reduced to problems of largest and smallest magnitudes ... and it is only by solving these problems that we can satisfy the requirements of practice which always seeks the best, the most convenient."
>
> Pafnuty Lvovich Chebyshev

## 3.1 Introduction

Cloudlet is defined as a trusted, resource-rich computer or cluster of computers that is well-connected to the Internet and available for use by nearby mobile devices [8]. A cloudlet can work independently as well, which makes it a very robust solution even in hostile environments [9]. Mobile devices can offload their computational tasks to meet several objectives. The most common objective is to reduce execution latency for energy-saving. When multiple objectives arise, one of them can be chosen as the objective and the rest as constraints. Computation offloading frameworks

like MAUI, CloneCloud, and COMET help offloading devices to make optimal decisions on "whether" and "when" to offload such that overall system performance is maximised based on the real-time network conditions, such as latency, bandwidth, and packet loss [134].

The authors of [53] showed that the system performance in terms of latency for the requested jobs from a single-hop cloudlet server is always better than a distant cloud server for low-latency applications. Thus, the optimal placement of cloudlets over wireless access networks seemed to be an important research problem to enhance system performance. Along with the wireless access networks, researchers started to focus on optical and FiWi access networks for edge-computing solutions deployment due to their wide deployment coverage, low cost per bit, very high data transmission bandwidth, efficient network virtualisation and network scalability [15]. The TDM-PON has also been considered as a front/back-haul support for wireless access network, i.e., access technologies like Radio-over-fibre (RoF), WiFi, LTE-A and 5G can be integrated with ONUs for a higher user coverage.

In this chapter, we address the "static network planning" problem as it provides a first-hand estimation for the cloudlet deployment cost without considering any user mobility and complex system models. We observe that existing literature mainly focuses on the traditional placement problem that is concerned mostly about "where" to place the cloudlets over wireless access networks. However, answering "how much" computational resources to be placed per cloudlet is also important in any cost optimisation framework as this prevents resources to be under or over-utilised. Moreover, the optimised design and planning of the hybrid placement of cloudlets based on TDM-PON supported FiWi architecture is yet to be addressed, i.e., the cloudlets are allowed to be placed in the field, RN and CO locations. The hybrid cloudlet placement architecture provides improved flexibility in where to place cloudlets over the field-based architectures, because depending on the deployment scenario and end-user QoS and QoE requirements, cloudlets can be placed in the field, RN or CO.

These gaps in existing literature motivate us to propose a hybrid cloudlet placement architecture and capital expenditure minimisation framework over existing

10G-PON infrastructure, subject to the cloudlet computational capacity and desired QoS latency constraints. Our proposed optimisation framework identifies optimal cloudlet placement locations as well as the average computational resources in each cloudlet. We use mixed-integer non-linear programming (MINLP) as a mathematical design tool for this framework. Through our framework evaluation, we show that the maximum percentage of incremental cost due to cloudlets with field architecture is around 12% for a 5 km $\times$ 5 km area, whereas percentages at 10% or lower are achievable with hybrid architectures. We also show that higher computational resources are required in the field or RN with stringent QoS latency requirements, e.g., 1 ms or 10 ms, but with a less stringent QoS requirement like 100 ms, more resources can be placed in CO cloudlets. Moreover, we show that the percentage of the incremental energy budget of the network with both field and hybrid architectures lie below 18%.

The rest of this chapter is organised as follows. In Section 3.2, we briefly review some of the key concepts of MINLP. In Section 3.3, the FiWi over TDM-PON based field and hybrid cloudlet placement network architectures are described in detail. In Section 3.4, the system model and the general MINLP formulation are presented. In Section 3.5, the simulation results are presented and their insights are discussed. Finally, in Section 3.6 the key achievements and observations from this entire work are summarised.

## 3.2 Background on Mixed-Integer Non-linear Programming

A large set of real-world optimisation problems are formulated as integer programming problems and MINLP is a special subset of this class of problems. In such problems, some of the variables are restricted to take only integer values and the feasible region is defined by a set of non-linear functions. Integer variables are very much useful in the modeling of problems that deal with logical relationships, fixed charges, indivisible objects, disjunctive constraints, or piece-wise linear functions. Moreover,

to accurately reflect physical properties, covariance, and large-scale economies, non-linear functions appear to be appropriate. Due to these characteristics, MINLPs belong to a large class of challenging optimisation problems, because they mix the complexity of optimising over integer variables while managing nonlinear functions. If we limit our model to include only linear functions, then MINLP reduces to a mixed-integer linear program (MILP), an NP-Hard problem [135]. Again, if we limit our model to have no integer variable but allow for general nonlinear functions in the objective or constraints, then MINLP reduces to a nonlinear program (NLP), which is also considered to be NP-Hard [136]. Nonetheless, the combination of integrality and non-linearity sometimes may lead to undecidable examples of MINLP as well [137]. A general MINLP problem can be formulated as follows:

$$\begin{aligned} &\min f(x,y) \\ \text{subject to } &g_i(x,y) \leq 0, \quad i = 1,\ldots,q, \\ &h_i(x,y) = 0, \quad i = 1,\ldots,l, \\ &x \in X \subseteq \mathbb{R}^n, y \in Y \subseteq \mathbb{Z}^m, \end{aligned}$$

where $f : X \times Y \to \mathbb{R}$, $g_i : X \times Y \to \mathbb{R}$ $(i = 1,\ldots,q)$, $h_i : X \times Y \to \mathbb{R}$ $(i = 1,\ldots,l)$ and $\mathbb{Z}^m$ denotes the set of integer vectors in $\mathbb{R}^m$. Moreover, we assume that $X$ is a nonempty convex set in $\mathbb{R}^n$ and $Y$ is a finite integer set in $\mathbb{Z}^m$ [138]. Due to the integrality of some of the variables, the feasible region of a general MINLP problem is not connected. This issue can be fixed by relaxing the integrality of the integer variables such that the MINLP becomes a continuous problem over a convex feasible region. It is one of the basic strategies adopted among various solution methods for solving MINLP problems. Another basic idea behind the solution methods for MINLP is to isolate the non-linearity from the mixed-integer model so that the main problem can be reduced to sub-problems that are comparatively easier to solve using well-known solution methods. If the integrality restriction on $y$ is relaxed, then a non-linear programming sub-problem can be derived that contains only continuous variables $(x, y)$ and can provide a lower-bound to the MINLP. If a suitable value for the integer variable $y$ is fixed, then an NLP sub-problem of continuous variable $x$ can

be derived that provides an upper bound to MINLP. Furthermore, a MILP or convex program for proving a lower-bound MINLP can be formulated by constructing linear or convex underestimation of $f$ or $g_i$ at certain known points. All these solution techniques are adopted in some of the recently proposed algorithms like branch-and-bound (BB) method, generalised Benders decomposition (GBD) method, and outer approximation (OA) method [138]. In this chapter, we use a spatial branch-and-bound algorithm to solve our proposed MINLP formulation, because they are efficient in handling non-convex terms by replacing them with their convex envelopes and forming a convex MINLP.

## 3.3 Cloudlet Placement Architectures

In this section, we first describe the proposed end-to-end physical network infrastructure of hybrid cloudlet support network with TDM-PON based FiWi access networks, as illustrated in Fig. 3.1. With a tree-and-branch network topology of TDM-PON, the ONUs are connected to optical passive splitters with split-ratio $1:N$, where $N \in \{4, 8, 16\}$, depending on QoS requirements [20]. Furthermore, we discuss how the field cloudlet placement architecture can be derived as a subset of the hybrid architecture.

### 3.3.1 Hybrid Cloudlet Placement Architecture

In a hybrid cloudlet placement scheme, for the sake of proximity with the end-users, we choose cloudlets to be installed either in the field, RN or CO locations. We plan and design the network such that a majority of the job requests from end-users are processed at cloudlets to achieve the low-latency QoS and QoE requirements. Under overload conditions, cloudlets can offload excess job requests to the remote cloud server. A detailed network architecture for hybrid cloudlet placement is presented in Fig. 3.1.

**Field Cloudlets:** The field cloudlets are connected with ONUs with point-to-point fibre links (brown links). Ideally, cloudlet should be connected to the Internet as well and hence, additional point-to-point fibre links between each cloudlet and

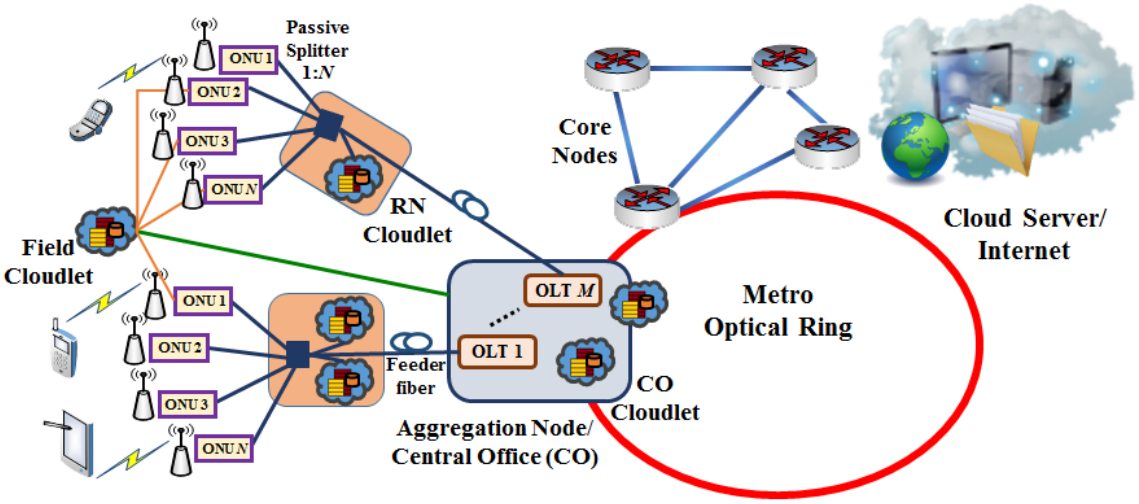



Figure 3.1: The hybrid cloudlet placement framework showing cloudlet placement locations in the field, RN and CO with a single FiWi branch.

the nearest COs (green links) are also installed. Due to the point-to-point fibre links from ONUs to cloudlets, ONUs can take advantage of the dedicated bandwidth.

**RN Cloudlets:** In the hybrid architecture, cloudlets are allowed to be installed at RN locations with passive splitters, as shown in Fig. 3.1. In this scheme, the ONUs can re-use the existing TDM-PON distribution fibres for communication with the cloudlets using new wavelengths. Either a single wavelength can be time-shared by multiple ONUs or a different wavelength can be used by each ONU, depending on the budget of network vendor.

**CO Cloudlets:** The idea of installing a cloudlet at the CO is advantageous over field and RN cloudlets in terms of new infrastructure setup, because the existing feeder and distribution fibres can be reused for cloudlets. However, the transmission latency from the ONUs to CO cloudlets is higher compared to cloudlets in the field and RN cloudlets due to increased geographic separation. Moreover, ONUs are expected to use around 70% and 50% of the default downlink and uplink channel bandwidths respectively, for regular data communication [139] and the rest of the bandwidth can be used for communications with cloudlets placed at the CO.

Note that whenever an ONU is connected to a field or RN cloudlet, a new set of transceivers will be required to be installed and should be accounted for in the incremental expenditure of cloudlet deployment. The reason behind this is that the default uplink and downlink channels connect ONUs directly to the CO nodes

only. Moreover, in a general cloudlet deployment scenario, any ONU can connect to multiple cloudlets, but in this work, we consider that each ONU is connected to one and only one cloudlet, as we assume that cloudlets are being deployed by a single service provider and can install multiple processors to handle all incoming job requests. This implies that if an ONU is connected to an RN cloudlet, it cannot be connected to either a field or a CO cloudlet.

### 3.3.2 Field Cloudlet Placement Architecture

Based on the field cloudlet placement architecture, initially explored by the authors of [140], we propose a cost-optimisation framework for the field cloudlet architecture over existing TDM-PON based FiWi networks. In the field cloudlet placement architecture, cloudlets are installed only in the field locations and ONUs are connected with them by using point-to-point fibres. Moreover, instead of placing the cloudlets at field locations, we can install cloudlets in some of the ONUs and create new fibre connections with other ONUs that share the cloudlets. If the ONUs were already connected by PON protection fibres, then this additional cost of new fibre installation can be avoided. Interestingly, if the options of placing cloudlets in RN and CO locations are not allowed, then the hybrid architecture can be easily reduced to the field cloudlet placement architecture. Note that in the field cloudlet placement architecture, the split ratio of the passive splitters does not play any role as new point-to-point fibre connections are created among ONUs and cloudlets.

## 3.4 System Model and Problem Formulation

In this section, we define the cloudlet network planning problem and present the system model with MINLP formulation.

### 3.4.1 Static Cloudlet Network Planning Problem

The static cloudlet placement framework considers the network status as static in time, i.e., it neither takes user mobility nor VM mobility into account and identifies optimal cloudlet locations over an existing access network infrastructure. This is

essential to a cost-optimal network design framework for cloudlet placement and assignment of ONUs to cloudlets [63].

### 3.4.2 Network optimisation Parameters

We assume that each cloudlet contains a finite set of processors, e.g., $m \in K = \mathbb{Z}_+$ processors with an average service rate of $\mu$ for each processor. We model the cloudlet system by assuming that the service process of each job request is independent of each other and are exponentially distributed with a service rate of $\mu$. We also assume that all job requests are homogeneously parallel processed by all the processors present in the cloudlet. The job requests the arrival process to any cloudlet from the connected ONUs are assumed to be independent and Poisson distributed with an arrival rate of $\lambda$. Based on these considerations, we model the cloudlets as $M/M/1$ queueing systems similar to [141]. This model is the most appropriate when we are just looking for a first-hand impression of a system's performance as it provides the upper bound on processing latency of a cloudlet when the aggregated processing rate of all the processors are considered. In general, a cloudlet service provider usually executes the static planning framework only at the initial stage of cloudlet deployment. Hence, by adopting the $M/M/1$ queueing model, we ensure that all the incoming job requests are processed within $D_Q$. The number of processors $m$ may vary from cloudlet to cloudlet, depending on the number of ONUs they are connected to. If a cloudlet location (either in the field or RN or CO) is not chosen, then the total number of processors should also be zero for that particular location.

The total incoming job request rate to a cloudlet is calculated by adding all the job requests arriving from all ONUs connected to that cloudlet. The operational cloudlets may choose either to process the total incoming job request all by itself or offload some fraction to the remote cloud while meeting the latency constraints. Offloading incoming jobs to the remote cloud is a cost minimising the opportunity for cloudlets, but meeting the latency requirements becomes challenging due to very high end-to-end latency from a remote cloud server. When a cloudlet chooses to process the total incoming job request all by itself, the average processing time for any cloudlet in the field, RN or CO location $z \in \{a, b, c\}$ is expressed by [142]:

Table 3.1: Hybrid Cloudlet Network optimisation Parameters

| ***Symbol*** | ***Definition*** |
|---|---|
| $A$ | Set of possible field cloudlet locations |
| $B$ | Set of possible RN cloudlet locations |
| $C$ | Set of possible CO cloudlet locations |
| $D$ | Set of locations of existing ONUs in the TDM-PON network |
| $K$ | Set of number of processors in a cloudlet location |
| $\alpha$ | Cost of installing a single processor cloudlet at field, RN or CO |
| $\xi_a$ | New infrastructure installation cost at field location $a \in A$ |
| $\xi_b$ | New infrastructure installation cost at RN location $b \in B$ |
| $\xi_c$ | New infrastructure installation cost at CO location $c \in C$ |
| $\eta$ | Cost of installing new optical fibre per kilometer |
| $L_{max}$ | Maximum allowed fibre length between field cloudlet $a \in A$ and ONU $d \in D$ (km) |
| $L_{zd}$ | Actual optical fibre length between a cloudlet at $z \in \{a, b, c\}$ and ONU $d \in D$ (km) |
| $L_{ac}$ | Actual optical fibre length between a field cloudlet at $a \in A$ and the nearest CO $c \in C$ (km) |
| $x_{bd}^{adj}$ | Element of adjacency matrix denoting connectivity between ONUs $d \in D$ and RNs $b \in B$ |
| $x_{cd}^{adj}$ | Element of adjacency matrix denoting connectivity between ONUs $d \in D$ and COs $c \in C$ |
| $BW_{dz}$ | Bandwidth of the optical fibre link between cloudlet at $z \in \{a, b, c\}$ and ONU $d \in D$ in the uplink direction |
| $BW_{zd}$ | Bandwidth of the optical fibre link between cloudlet at $z \in \{a, b, c\}$ and ONU $d \in D$ in the downlink direction |
| $D_{zd}$ | Total end-to-end propagation latency between cloudlet at $z \in a, b, c$ and ONU at $d \in D$ |
| $D_{QoS}$ | Maximum allowed latency to maintain desired QoS |
| $\mu$ | Average service rate of each processor in a cloudlet (jobs/s) |
| $\lambda_z$ | Total job arrival rate at cloudlet at $z \in \{a, b, c\}$, from all $d \in D$ONUs connected to it (jobs/s) |
| $\lambda_d$ | Job arrival rate from a single ONU at $d \in D$ (jobs/s) |
| $\mu_z$ | Total service rate of a cloudlet at $z \in \{a, b, c\}$ containing $m \in K$ processors (jobs/s) |
| $\Lambda$ | The round-trip-time to offload a job request to remote cloud server |
| $\sigma_{ul}$ | Average number of bits an ONU $d \in D$ sends to cloudlet $z \in \{a, b, c\}$ for processing |
| $\sigma_{dl}$ | Average number of bits an ONU $d \in D$ receives from cloudlet $z \in \{a, b, c\}$ after processing |
| $n_\lambda$ | Number of wavelengths shared by ONUs to communicate to the corresponding RN cloudlet |
| $\beta_{dl}$ | Background load in the downlink of the considered TDM PON |
| $\beta_{ul}$ | Background load in the uplink of the considered TDM PON |
| $\delta_{zd}$ | Polling cycle latency between cloudlet at $z \in \{a, b, c\}$ and ONU $d \in D$ |
| $v_\lambda$ | Velocity of light within optical fibre ($2 \times 10^8$ m/s) |

$$\tau_z^{cloudlet} = \frac{1}{(\mu_z - \lambda_z)}, \forall z \in \{a, b, c\}. \tag{3.1}$$

The core cloud servers are usually over-provisioned and possess huge computational and storage resources. Hence can be assumed as an M/M/$\infty$ queuing system [143]. The total latency for a job request, when offloaded to cloud, can be expressed as follows [143]:

$$\tau^{cloud} = \Lambda + \frac{1}{\mu}. \tag{3.2}$$

A summary of network optimisation parameters with their definitions are presented in Table 3.1 for convenience.

### 3.4.3 The Decision Variables

To formulate the objective function and constraints of our MINLP problem, we choose a set of binary and fractional decision variables as follows. The variables $x_a$, $x_b$ and $x_c$ are binary variables, indicating the decision whether to install a cloudlet at a field site $a \in A$, at an RN site $b \in B$, and at a CO site $c \in C$, respectively.

$$x_a = \begin{cases} 1; & \text{if a cloudlet is installed} \\ & \text{in the field location } a \in A \\ 0; & \text{otherwise} \end{cases}$$

$$x_b = \begin{cases} 1; & \text{if a cloudlet is installed} \\ & \text{at RN location } b \in B \\ 0; & \text{otherwise} \end{cases}$$

$$x_c = \begin{cases} 1; & \text{if a cloudlet is installed} \\ & \text{at CO location } c \in C \\ 0; & \text{otherwise} \end{cases}$$

The variables $n_{am}$, $n_{bm}$ and $n_{cm}$ are binary variables to indicate whether $m$ number of processors are chosen to be placed in cloudlet location $a \in A$, $b \in B$ and

$c \in C$, respectively.

$$n_{am} = \begin{cases} 1; & \text{if the cloudlet in } a \in A \\ & \text{contains } m \text{ processors} \\ 0; & \text{otherwise} \end{cases}$$

$$n_{bm} = \begin{cases} 1; & \text{if the cloudlet in } b \in B \\ & \text{contains } m \text{ processors} \\ 0; & \text{otherwise} \end{cases}$$

$$n_{cm} = \begin{cases} 1; & \text{if the cloudlet in } c \in C \\ & \text{contains } m \text{ processors} \\ 0; & \text{otherwise} \end{cases}$$

We define the following binary variables as the product of some particular combinations of the aforementioned binary variables, i.e., $\overline{x_a} = x_a n_{am}$, $\overline{x_b} = x_b n_{bm}$, and $\overline{x_c} = x_c n_{cm}$.

The variables $x_{ad}$, $x_{bd}$ and $x_{cd}$ are binary variables to indicate whether the connectivity between an ONU at $d \in D$ and a cloudlet at $a \in A$, $b \in B$ and $c \in C$ respectively, is created or not.

$$x_{ad} = \begin{cases} 1; & \text{if ONU } d \in D \text{ is connected} \\ & \text{to field cloudlet } a \in A \\ 0; & \text{otherwise} \end{cases}$$

$$x_{bd} = \begin{cases} 1; & \text{if ONU } d \in D \text{ is connected} \\ & \text{to RN cloudlet } b \in B \\ 0; & \text{otherwise} \end{cases}$$

$$x_{cd} = \begin{cases} 1; & \text{if ONU } d \in D \text{ is connected} \\ & \text{to CO cloudlet } c \in C \\ 0; & \text{otherwise} \end{cases}$$

The variables $\varphi_a$, $\varphi_b$, and $\varphi_c \in [0, 1]$ indicate the fraction of the total incoming job requests a cloudlet at $a \in A$, $b \in B$ and $c \in C$ chooses to process by itself and offloads the rest of the workloads, i.e., $(1 - \varphi_a)$, $(1 - \varphi_b)$, and $(1 - \varphi_c)$ to remote cloud server.

### 3.4.4 Objective Function and Constraints

We formulate the objective function to minimise the overall cloudlet installation expenditures as follows:

$$\begin{aligned} \min \Bigg\{ & \sum_{a \in A} \sum_{m \in K} (m\alpha)\overline{x_a} + \sum_{b \in B} \sum_{m \in K} (m\alpha)\overline{x_b} + \sum_{c \in C} \sum_{m \in K} (m\alpha)\overline{x_c} \\ & + \sum_{a \in A} \sum_{d \in D} \eta L_{ad} x_{ad} + \sum_{a \in A} (\xi_a + \eta L_{ac})\, x_a + \sum_{b \in B} \xi_b x_b + \sum_{c \in C} \xi_c x_c \Bigg\}. \end{aligned} \tag{3.3}$$

The above expression includes cost components arising from multiple sectors. The first, second and third terms denote the total cost of processors installed in the field, the RN and the CO cloudlet locations, respectively. The fourth term denotes the total cost of installing new fibre connections among ONUs and field cloudlets. Finally, the fifth, sixth and seventh terms denote the additional infrastructure setup costs in the field, the RN and the CO cloudlet locations, respectively. The fifth term also contain the cost of additional point-to-point fibre links between each field cloudlet and nearest CO. Note that in this problem formulation, we consider only the cloudlet installation cost and this is a one-time expense. The operational expenditures of cloudlet deployment are not included in this formulation because it involves an implicit time component and should be minimised separately through dynamic resource allocation problems.

Actually the first three terms of the objective function (3.3) are quadratic in nature due to $\overline{x_a} = x_a n_{am}$, $\overline{x_b} = x_b n_{bm}$ and $\overline{x_c} = x_c n_{cm}$, but we linearise them using

three linear inequalities for each of them, as shown in (3.4)-(3.6) [144].

$$\overline{x_a} \leq n_{am}, \overline{x_a} \leq x_a, \overline{x_a} \geq n_{am} + x_a - 1, \forall a \in A, m \in K, \tag{3.4}$$

$$\overline{x_b} \leq n_{bm}, \overline{x_b} \leq x_b, \overline{x_b} \geq n_{bm} + x_b - 1, \forall b \in B, m \in K, \tag{3.5}$$

$$\overline{x_c} \leq n_{cm}, \overline{x_c} \leq x_c, \overline{x_c} \geq n_{cm} + x_c - 1, \forall c \in C, m \in K. \tag{3.6}$$

The set of constraints (3.7)-(3.9) are the capacity constraints and are used to ensure that only one value of $m$ (the number of processors in a particular cloudlet) is chosen. If a particular cloudlet is not chosen to be activated at either of field, RN or CO location, then no processors are placed at that location.

$$x_a \leq \sum_{m \in K} n_{am} \text{ and } \sum_{m \in K} n_{am} \leq 1, \forall a \in A, m \in K, \tag{3.7}$$

$$x_b \leq \sum_{m \in K} n_{bm} \text{ and } \sum_{m \in K} n_{bm} \leq 1, \forall b \in B, m \in K, \tag{3.8}$$

$$x_c \leq \sum_{m \in K} n_{cm} \text{ and } \sum_{m \in K} n_{cm} \leq 1, \forall c \in C, m \in K. \tag{3.9}$$

Constraints described next are connectivity constraints. Constraint (3.10) denotes that any ONU can be chosen to be connected to a field cloudlet, only if their Euclidean separation is $\leq L_{max}$. Constraint (3.11) denotes that a field cloudlet will be activated if there exists at least one connected ONU.

$$x_{ad} \leq \max\left[0, \left(\frac{L_{ad} - L_{max}}{|L_{ad} - L_{max}|}\right)\right], \forall a \in A, d \in D, \tag{3.10}$$

$$x_a \geq x_{ad}, \forall a \in A, d \in D. \tag{3.11}$$

In a TDM-PON, any arbitrary ONU cannot be connected to any RN cloudlet, rather only those ONUs can be connected which already has an existing fibre connection to a particular RN and the same condition applies for connectivity between ONU and CO as well. These conditions are enforced by constraints (3.12)-(3.13). Furthermore, constraints (3.14)-(3.15) imply that a cloudlet in the RN and in the

CO can be activated if there is at least one ONU connected to it, respectively.

$$x_{bd} \leq x_{bd}^{adj}, \forall b \in B, d \in D, \tag{3.12}$$

$$x_{cd} \leq x_{cd}^{adj}, \forall c \in C, d \in D, \tag{3.13}$$

$$x_b \geq x_{bd}, \forall b \in B, d \in D, \tag{3.14}$$

$$x_c \geq x_{cd}, \forall c \in C, d \in D. \tag{3.15}$$

The constraint in (3.16) ensures that every ONU is connected to one and only one cloudlet,

$$\sum_{a \in A} x_{ad} + \sum_{b \in B} x_{bd} + \sum_{c \in C} x_{cd} = 1, \forall d \in D. \tag{3.16}$$

Finally, the most important constraint, i.e., the QoS latency constraint that provides the upper limit of total allowed latency for each ONU, is described in (3.17),

$$\begin{aligned} & x_{ad}\left[\varphi_a\left\{\frac{1}{\mu_a - \lambda_a\varphi_a} + D_{ad} + \frac{\sigma_{ul}}{BW_{da}} + \frac{\sigma_{dl}}{BW_{ad}}\right\}\right. \\ & \qquad \left. +(1-\varphi_a)\left(\Lambda + \frac{1}{\mu}\right)\right] \\ & +x_{bd}\left[\varphi_b\left\{\frac{1}{\mu_b - \lambda_b\varphi_b} + D_{bd} + \frac{\sigma_{ul}\sum_{d\in D} x_{bd}}{n_\lambda BW_{db}} + \frac{\sigma_{dl}\sum_{d\in D} x_{bd}}{n_\lambda BW_{bd}}\right\}\right. \\ & \qquad \left. +(1-\varphi_b)\left(\Lambda + \frac{1}{\mu}\right)\right] \\ & +x_{cd}\left[\varphi_c\left\{\frac{1}{\mu_c - \lambda_c\varphi_c} + D_{cd} + \frac{\sigma_{ul}\sum_{d\in D} x_{cd}}{BW_{dc} - \beta_{ul}} + \frac{\sigma_{dl}\sum_{d\in D} x_{cd}}{BW_{cd} - \beta_{dl}}\right\}\right. \\ & \left. +(1-\varphi_c)\left(\Lambda + \frac{1}{\mu}\right)\right] \leq D_{QoS}, \forall a \in A, b \in B, c \in C, d \in D. \end{aligned} \tag{3.17}$$

Note that, the total job arrival rate to any cloudlet, depending on connectivity among ONUs and cloudlets, and the total service rate of any cloudlet is computed as follows:

$$\lambda_z = \sum_{d \in D} x_{zd}\lambda_d, \text{ and } \mu_z = \sum_{m \in K} mn_{zm}\mu, \forall z \in \{a, b, c\}. \tag{3.18}$$

Total end-to-end propagation latency between cloudlet at $z \in \{a, b, c\}$ and ONU at $d \in D$ is defined as $D_{zd} = \left(2\frac{L_{zd}}{v_\lambda} + \frac{\delta_{zd}}{2}\right), \forall z \in \{a, b, c\}$. We assume that the job request packets from the ONUs and the job response packets from cloudlets

**Algorithm 3.1** Spatial Branch-and-Bound Algorithm of COUENNE

**Input:** The MINLP problem with objective (3.3) and constraints (3.4)-(3.17). Denote this problem by $\mathcal{P}$.
**Output:** The optimal value $z^*$ of the MINLP problem $\mathcal{P}$.
**Define:** Set $L$ of subproblems; let $L \leftarrow \{\mathcal{P}\}$
**Define:** $z_u$ as an upper-bound for $\mathcal{P}$; let $z_u \leftarrow +\infty$
**while** $L \neq \emptyset$ **do**
    Choose $\mathcal{P}_k \in L$;
    $L \leftarrow \{\mathcal{P}_k\}$;
    Apply bounds tightening to $\mathcal{P}_k$;
    **if** bounds tightening did not prove $\mathcal{P}_k$ infeasible **then**
        Generate a linear relaxation $\mathcal{LP}_k$ of $\mathcal{P}_k$;
        **repeat**
            Solve $\mathcal{LP}_k$;
            Let $x_k^*$ be an optimum and $z_k^*$ its objective value;
            Refine linearisation $\mathcal{LP}_k$;
        **until** $x_k^*$ is feasible for $\mathcal{P}_k$ or $z_k^*$ does not improve sufficiently;
    **if** $x_k^*$ is feasible for $\mathcal{P}_k$ **then**
        Let $z_u \leftarrow \min\{z_u, z_k^*\}$;
    (Optional) find a local optimum $z_k'$ of $\mathcal{P}_k$;
    $z_u \leftarrow \min\{z_u, z_k'\}$;
    **if** $z_k^* \leq z_u - \epsilon$ **then**
        Choose a variable $x_i$;
        Choose a branching point $x_{bi}$;
        Create sub-problems:
        $\mathcal{P}_{k-}$ with $x_i \leq x_{bi}$;
        $\mathcal{P}_{k+}$ with $x_i \geq x_{bi}$;
        $L \leftarrow L \cup \{\mathcal{P}_{k-}, \mathcal{P}_{k+}\}$;
Output $z^* := z_u$;

are interleaved with background traffic to be cleared within one polling cycle of the considered TDM-PON.

The constraint (3.17) consists of three terms of the weighted sum of total latencies when the incoming jobs are either executed in the field or RN or CO cloudlets. However, due to (3.16) each ONU can connect to only one cloudlet. Each of these three weighted latency terms consists of four components. The first component denotes the processing latency, the second component denotes the propagation latency, the third component denotes the transmission latency to offload total bits for the job by ONU to cloudlet and the fourth component denotes the transmission latency to receive total bits post-processing by ONU from cloudlet.

## 3.5 Framework Evaluation

We implemented the MINLP model described in Section 3.4 with the A Modeling Language for Mathematical Programming (AMPL) platform and used the open-source solver COUENNE package for MINLPs to solve the problem [145]. Any MINLP problem is NP-Hard in general and in our optimisation problem formulation, the underlying decision problem is equivalent to a multidimensional assignment problem as we try to assign each ONU to a cloudlet installed either in the field, at RN, and at CO cloudlets, such that the desired QoS latency is met at the minimum cost. This problem is an NP-complete problem [146] and may take several hours to days to compute the optimal solution [147]. Note that static planning frameworks are evaluated in the pre-deployment stage and hence this computation time can be tolerated. COUENNE aims to find the global optima of non-convex MINLPs by implementing linearisation, bound reduction, and branching methods within a branch-and-bound framework. Thus, the underlying spatial branch-and-bound algorithm is summarised in Algorithm 3.1. A detailed complexity analysis of the spatial branch-and-bound algorithm used in COUENNE is done by the authors of [148].

### 3.5.1 Random Dataset Generation

To test the designed framework, we use stochastically-generated hypothetical 5 km $\times$ 5 km standard city/towns to represent urban, suburban and rural deployment scenarios. Following the guidelines provided in [149] we consider the population densities of 4000, 2500 and 1500 per square kilometre (km$^2$) for urban, suburban and rural areas, respectively. Here we assume that each ONU can serve around 1000 users by wireless connection and hence we divide population density by 1000 to calculate the average number of ONUs/km$^2$, e.g., an average 4 ONUs/km$^2$ in the urban area. We use the Poisson-point process to distribute the total population across the considered area to identify the ONU locations. We then use a $k$-means clustering algorithm to identify potential field cloudlet placement locations [150]. The value of $k$ needs to be sufficiently large to maintain the feasibility of the optimal solution. In this work, we arbitrarily choose $k = 20$. The ONUs are supported by the standard 10G-PON infrastructure with 1:$N$, $N \in \{4, 8, 16\}$ split ratio and 10 Gbps

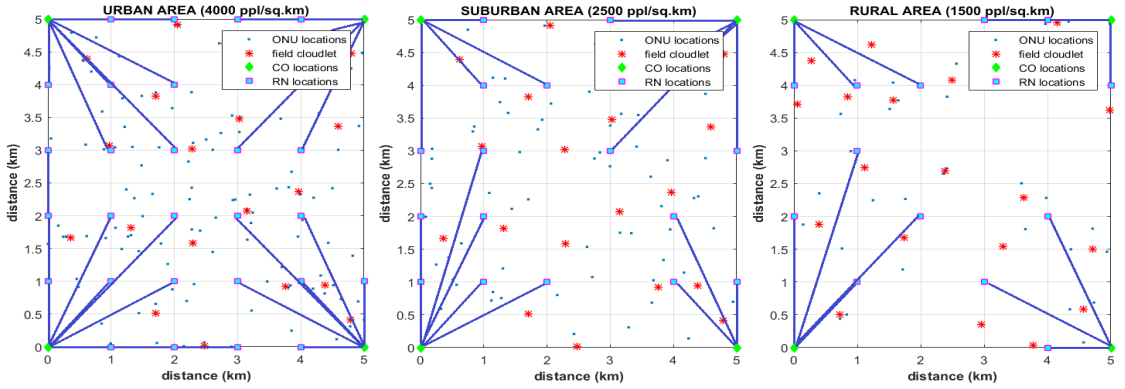



Figure 3.2: Randomly generated urban, suburban and rural areas for cloudlet deployment with existing TDM-PON network infrastructure with 1:4 split ratio. The blue links from CO to RN locations represent the feeder fibres. Distribution fibres from RN to each ONU are not shown.

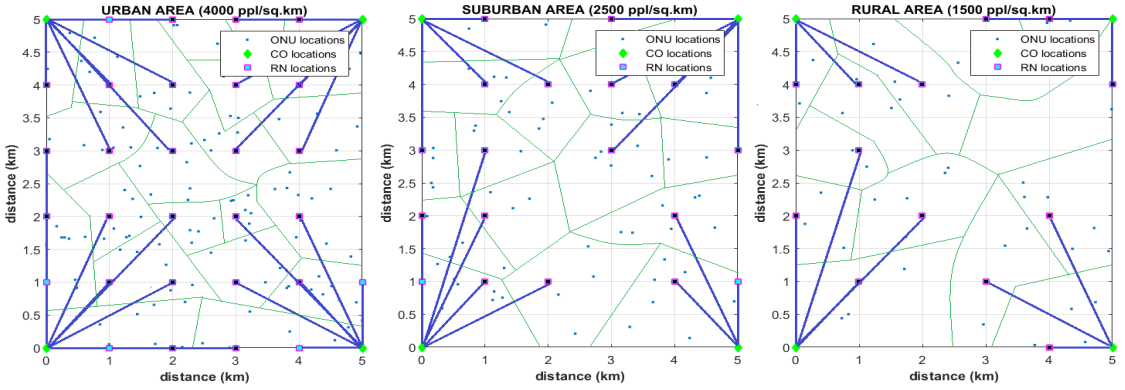



Figure 3.3: Optimal hybrid cloudlet placement locations over the considered urban, suburban and rural areas for 1 ms latency constraint. In this solution, cloudlets are installed at darkened face RNs and all COs.

datarate in both downlink and uplink directions. The normalised costs of installing new point-to-point fibre per kilometre ($\eta$) are 50, 35 and 20 for urban, suburban and rural, respectively [139]. The datarate of these new point-to-point fibre links are considered to be $BW_{ad} = BW_{da} = 1$ Gbps and the maximum allowed distance among field cloudlets and ONUs is $L_{max} = 4$ km, such that feasibility of optimal solution is maintained. We consider that $n_\lambda = 1$ wavelength (where a maximum 10 Gbps datarate is shared among ONUs) to communicate with the cloudlet at their corresponding RN location and hence $BW_{bd} = BW_{db} = 10$ Gbps. As ONUs are allowed to use the default uplink and downlink channels to communicate with cloudlets at their corresponding CO, therefore $BW_{cd} = BW_{dc} = 10$ Gbps.

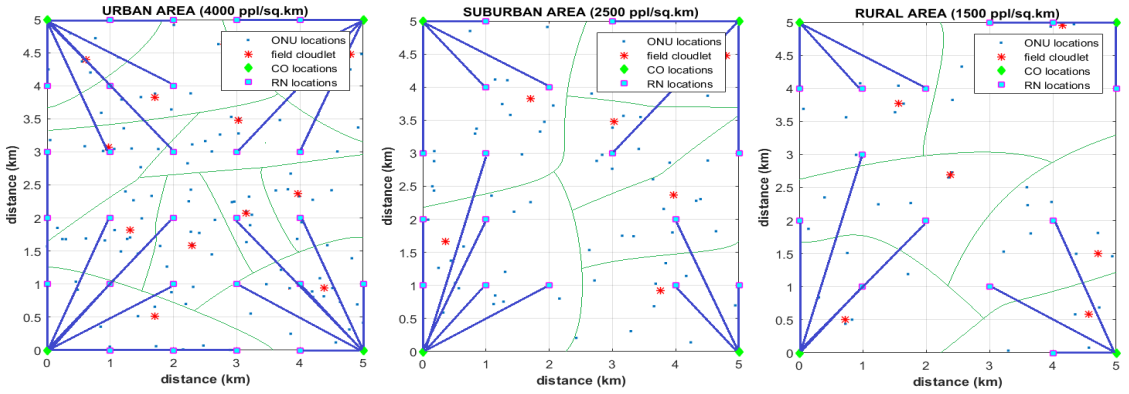



Figure 3.4: Optimal field cloudlet placement locations over the considered urban, suburban and rural areas for 1 ms latency constraint. In this solution, cloudlets are installed only in the field.

### 3.5.2 Values of Network Parameters

We consider the approximate downlink and uplink background loads as $\beta_{dl} = 7$ Gbps and $\beta_{ul} = 5$ Gbps, respectively[139]. The normalised cost of installing one processor is considered as $\alpha = 1$ [151]. We also consider the normalised costs of installing new cloudlet infrastructure in the field locations is $\xi_a = 8$ and at RN locations is $\xi_b = 6$, and at CO locations, i.e., $\xi_c = 4$ [151]. These parameters implicitly include the costs of the associated cooling and maintenance units. Due to our assumption that each ONU can serve a maximum of 1000 users the maximum job arrival rate from each ONU $d \in D$ is $\lambda_d = 1000$ jobs/s. The average job request and response sizes of some commonly used low-latency applications considered are $\sigma_{ul} = 100$ KB and $\sigma_{dl} = 100$ B [53]. Here, we assume that each VM is serving job requests from only one user, but this can be very easily extended to one VM serving job requests from multiple users [63]. We consider the service rate of each processor in cloud or cloudlet is $\mu = 2500$ jobs/s [151]. We consider $\delta_{ad} = 0$ as point-to-point fibre connections are used between ONUs and field cloudlets, and $\delta_{bd} = \delta_{cd} = 0.5$ msec [152]. The round-trip-time to offload a job to a cloud server ($\Lambda$) is relatively large due to large geographic separation, jitter buffer, and queueing latencies of the intermediate routers and hence is considered to be 500 ms [153].

### 3.5.3 Results and Discussions

In this subsection, we present a comparative study between the field and hybrid cloudlet placement architectures over 5 km × 5 km urban, suburban and rural deployment scenarios for $D_{QoS}$ values of 1 ms, 10 ms and 100 ms. Fig. 3.2 shows an instance of a randomly generated dataset considered for our framework evaluation. The small blue dots indicate the ONU locations that are generated using Poisson-point process. The light-green corner points indicate the CO locations and cyan squares denote the RN locations. The red asterisks denote the possible field cloudlet placement locations, which are identified by applying $k$-means clustering algorithm on the ONU locations. The thick blue lines indicate the feeder fibre of the TDM-PONs, connecting the RNs and their respective COs. The distribution fibres are not shown to avoid unnecessary overcrowding. With this dataset shown in Fig. 3.2, the hybrid cloudlet placement framework yields the final cloudlet deployment locations and their corresponding coverage areas for $D_{QoS} = 1$ ms as shown in Fig. 3.3 and the same with the field cloudlet placement framework as shown in Fig. 3.4. The green borders indicate the coverage areas of each cloudlet, i.e., the ONUs served by each cloudlet.

Fig. 3.5 illustrates the percentage of incremental cost over existing TDM-PON with field and hybrid cloudlet placement architectures in urban, suburban and rural scenarios against $D_{QoS} = 1$ ms, 10 ms, and 100 ms. Fig. 3.5a shows the percentage of incremental cost over the existing TDM-PONs for field cloudlet architecture, whereas Figs. 3.5b, 3.5c, and 3.5d show the same for hybrid cloudlet architecture with $1 : N$, $N \in \{4, 8, 16\}$ split ratios, respectively. The total incremental cost is calculated by the addition of the cost of processors, the cost of newly installed fibre, the cost of newly installed additional optical transceivers and the cost of new infrastructure setup [139]. The results indicate that with the hybrid cloudlet placement architecture, the incremental cost of cloudlet is higher for split-ratio 1:4 than that with split-ratio 1:8 and 1:16, as more number of cloudlets are required to be installed. Moreover, the incremental cost of installing cloudlets is highest in the urban scenario and least in the rural scenario with all split ratios. This is primarily because more cloudlets are required in the urban and suburban scenarios compared

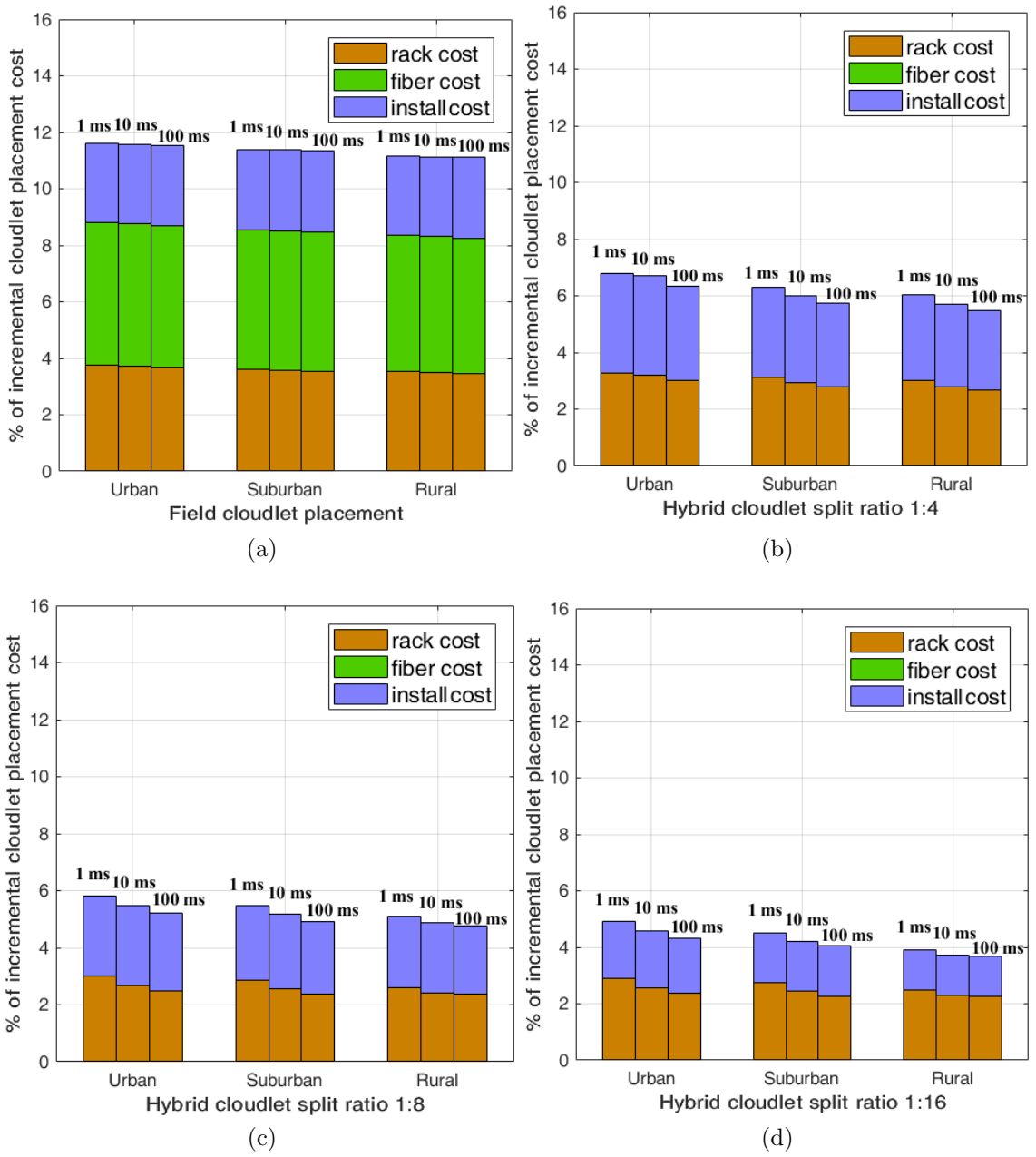



Figure 3.5: Comparison of workload distribution between field and hybrid cloudlet placement architectures in urban, suburban and rural scenarios against $D_{QoS}$ = 1 ms, 10 ms, and 100 ms.

to the rural scenario to meet latency requirements. Note that the incremental cost of cloudlet installation is relatively higher with field cloudlet architecture than that of the hybrid cloudlet architecture, primarily due to the cost of new fibre installation, which is unavoidable with field cloudlet architecture.

Although we considered the scope of offloading some job requests to the remote cloud server, we observed from the optimal solution that to meet the target QoS

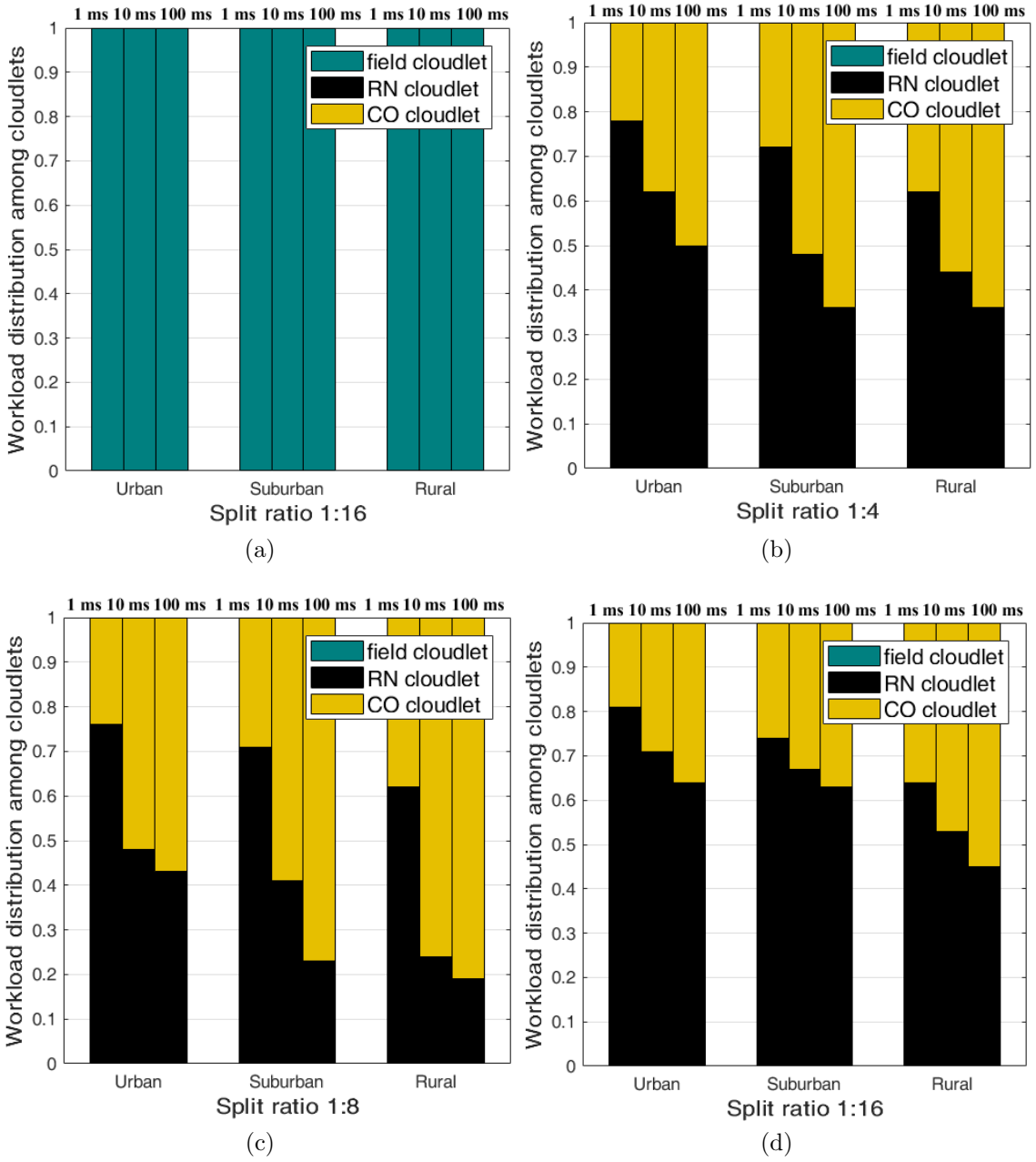



Figure 3.6: Comparison of workload distribution between field and hybrid cloudlet placement architectures in urban, suburban and rural scenarios against $D_{QoS}$ = 1 ms, 10 ms, and 100 ms.

latency in the range 1-100 ms, no job requests were offloaded to the remote cloud server. Rather, 100% of the job requests were serviced by field, RN and CO cloudlets. In Fig. 3.6 we show the distribution of workload amongst the cloudlets installed in the field, RN or CO for urban, suburban and rural scenarios with 1:$N$, $N \in \{4, 8, 16\}$ split ratios, for the all three $D_{QoS}$ values of 1 ms, 10 ms and 100 ms. With the field

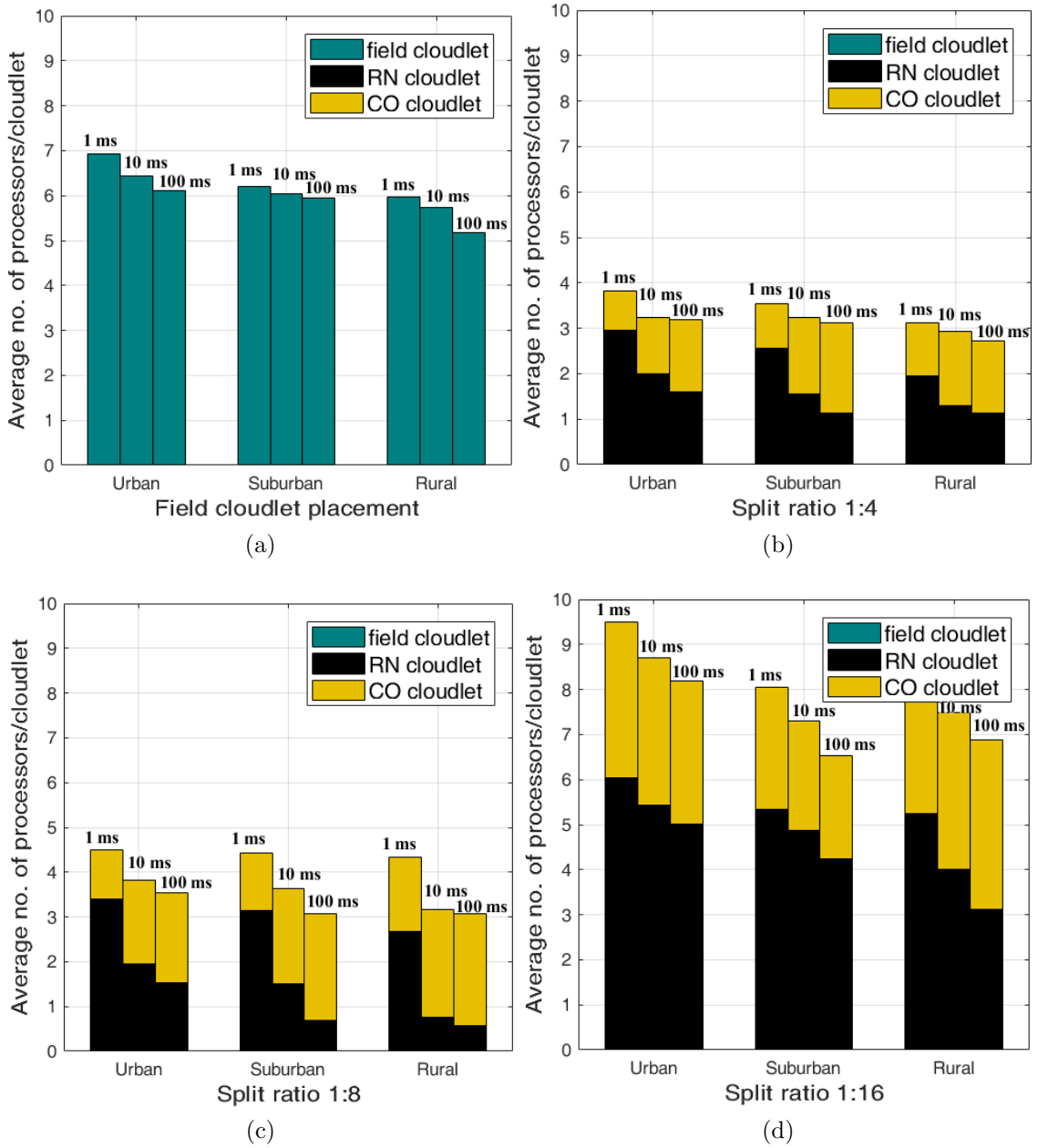



Figure 3.7: Comparison of average number of racks/cloudlets between field and hybrid cloudlet placement architectures in urban, suburban and rural scenarios against $D_{QoS}$ = 1 ms, 10 ms, and 100 ms.

cloudlet architecture, the entire load is handled by the cloudlets installed in the field, as shown in Fig. 3.6a. The hybrid cloudlet placement framework assigns the majority of the job requests to the RN cloudlets, as depicted by Figs. 3.6b to 3.6d for all deployment scenarios against a stringent $D_{QoS}$ = 1 ms requirement. However, against relatively relaxed $D_{QoS}$ = 10 ms and 100 ms, a higher percentage of workload is assigned to the CO cloudlets. Therefore, when the QoS latency requirement is

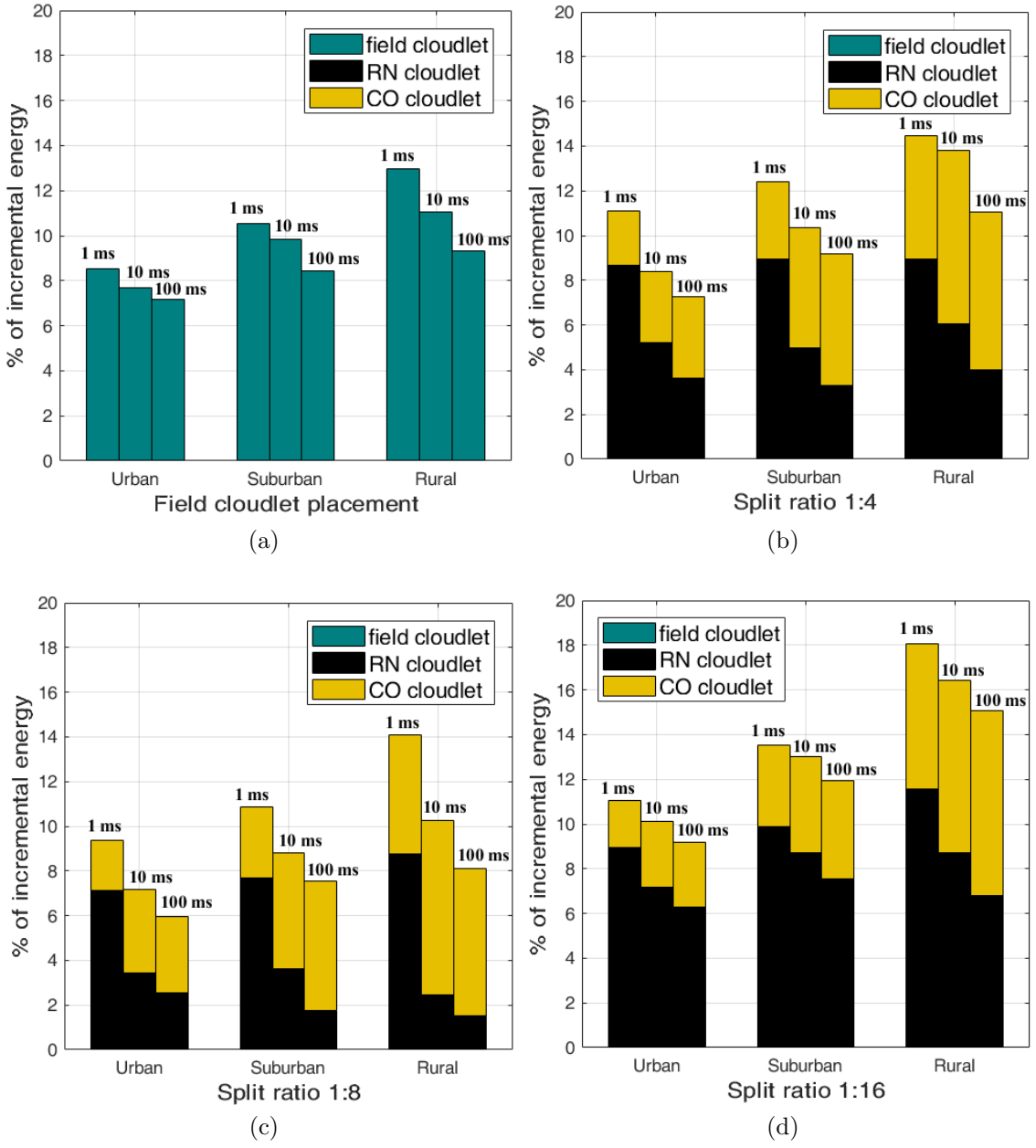



Figure 3.8: Comparison of incremental energy consumption over existing TDM-PON between field and hybrid cloudlet placement architectures in urban, suburban and rural scenarios against $D_{QoS}$ = 1 ms, 10 ms, and 100 ms.

stringent, installation of field or RN cloudlets are preferred despite higher additional installation expenses, whereas installation of CO cloudlets are preferred against a less stringent QoS latency requirement.

Fig. 3.7 shows the average amount of computational resources, i.e., the average number of processors required per cloudlet, against the urban, suburban and rural

scenarios. Fig. 3.7a presents the average number of processors required per cloudlet with the field cloudlet framework, which tries to install a lower number of cloudlet sites and place an optimal number of processors in them. Figs. 3.7b, 3.7c and 3.7d show the same with the hybrid cloudlet framework with $1 : N$, $N \in \{4, 8, 16\}$ split ratios against $D_{QoS}$ values of 1 ms, 10 ms and 100 ms. It is evident from these figures that for $D_{QoS} = 1$ ms, a higher number of processors/cloudlet are required, whereas when $D_{QoS} = 10$ ms or 100 ms, a relatively lower number of processors/cloudlet can handle the same amount of job requests.

In Fig. 3.8, we present the percentage of increase in energy consumption for the urban, suburban and rural scenarios with field cloudlet architecture (Fig. 3.8a) and hybrid cloudlet architectures with 1:$N$, $N \in \{4, 8, 16\}$ split ratios against $D_{QoS}$ values of 1 ms, 10 ms and 100 ms (Figs. 3.8b, 3.8c and 3.8d). The total energy budget of the deployed 10G-PON is already known, which includes the energy consumption of COs, ONUs, wireless transceivers integrated with ONUs, and all-optical transceivers [139]. The energy budget of the optimally installed cloudlets is computed from the technical specifications of Lenovo ThinkStation P900 [151]. Note that, these results provide an upper bound of the incremental energy consumption as we assume that all devices are always functional. However, for all considered 5 km $\times$ 5 km urban, suburban and rural scenarios with all $1 : N$, $N \in \{4, 8, 16\}$ split ratios, the percentage of incremental energy budget arising from cloudlet infrastructure installation is less than 18%.

## 3.6 Conclusions

In this chapter, we have proposed a FiWi over TDM-PON based hybrid cloudlet placement framework, where cloudlets can be optimally placed in the field, RN or CO locations. In this framework, an MINLP model was developed to evaluate the optimal cost of cloudlet placement. We have evaluated both field and hybrid frameworks with 1:$N$, $N \in \{4, 8, 16\}$ split ratios over hypothetical urban, suburban and rural deployment scenarios against latency constraints of 1 ms, 10 ms and 100 ms and have made a comparative study among them. We have shown that the installation of more RN and CO cloudlets provides improved cost-optimal solution

than the installation of field cloudlets alone. We have shown that the field cloudlet framework can be derived as a subset of the hybrid cloudlet placement framework. In general, the results help us to understand that "cost-optimal cloudlet placement locations" and "computational resources per cloudlet" are scenario dependent, i.e., user density, network architecture, TDM-PON split-ratio and QoS requirements primarily dictate the optimal cloudlet placement. We have also shown that the percentage of the incremental energy budget of the access networks due to active cloudlet installation is reasonably small. These fundamental insights provide a first account of guiding network providers in making suitable cloudlet placement choices for future cloudlet based network designing and planning.

# Chapter 4

# Analytical Cost-optimisation Frameworks for Cloudlet Placement over Fibre-Wireless Access Networks

"Our life is frittered away by detail ... Simplify, simplify. Instead of three meals a day, if it be necessary eat but one; instead of a hundred dishes, five; and reduce other things in proportion".

Henry David Thoreau, Walden

## 4.1 Introduction

In Chapter 3 we focused on cost-optimal cloudlet placement problem and proposed frameworks that provide a first-hand cost estimation without considering any user mobility at the initial network deployment stage. To the best of our knowledge, for the very first time, we proposed a cost-optimisation framework for hybrid cloudlet placement, where cloudlets are allowed to be installed in the field, RN and at CO locations subject to the capacity and latency constraints. We gathered several fundamental insights on optimal cloudlet placement through numerical evaluation of our proposed framework. When QoS latency requirement is very stringent and user

density is high, cloudlets need to be placed in the field and RN locations (close to mobile users), whereas when the QoS latency requirement is relaxed and user density is low or moderate, cloudlets can be placed at CO or even metro/core network nodes (relatively far from mobile users). Nonetheless, with this numerical evaluation, we realised that this framework is capable of providing us with the exact cloudlet placement locations, but it does not provide any general insight about cost-optimal cloudlet deployment strategies, depending on the underlying network scenario. Moreover, the solution algorithm does not scale very well with large data sets.

In addition, while reviewing some of the recent works on cloudlet placement in Chapter 2, we observed that the authors mainly proposed integer/mixed-integer linear/non-linear programming based frameworks. In such problem formulations, some of the decision variables are restricted to be only integers and some variables are free to take any numerical value. In practice, most of the times these problems display NP-hardness and hence, it is not known how quickly and accurately to find a solution for these problems, although solutions can be checked within a polynomial time when given. Thus, the authors designed heuristic algorithms for their proposed framework evaluation. Although some heuristic algorithms can be used to solve NP-hard optimisation problems very fast converging and scalable, optimality, precision, and accuracy are sacrificed. Some commonly adopted heuristic algorithms are swarm intelligence, tabu search, simulated annealing, genetic algorithms, and artificial neural networks, to name a few. Therefore, the results shown by the authors may often have a serious scalability issue and a very strong bias on the dataset used for their framework evaluation. Another critical issue arises when there is no existing network infrastructure and some service provider intends to install cloudlet servers while deploying green-field fibres. A greenfield implementation includes building up and setting up a network in the case of telecommunication networks, where there was no network previously. On the other hand, an update or extension to the existing network and using some obsolete components is a brownfield implementation. In such a scenario, due to the lack of knowledge on the underlying access network, most of the existing frameworks fail to provide any estimation of cloudlet deployment cost

or insight on optimal cloudlet deployment strategy.

This gap in the existing literature forbids us from estimating the total cost for the cloudlet deployment in a simple way considering only the key network parameters such as user density, TDM-PON split ratio, uplink-downlink bandwidth, to name a few. This intriguing research challenge motivates us to design a method that can provide a tight lower bound on the optimal deployment cost for the cost-optimisation frameworks on a hybrid cloudlet placement architecture. Here we make a fundamental assumption that the network is "homogeneous", which means that each TDM-PON has same split ratio and each ONU serves the same number of users. This assumption appears to be very much useful for any analysis in the average sense. Furthermore, we assume that the distances between all ONUs and the field, RN, and CO cloudlets are equal to their respective average distances. These assumptions aid us to perform an average cost analysis of a network. Our assumptions also help us to reduce the analysis of the entire volume of a network to a fundamental network unit, i.e., just a single TDM-PON based FiWi branch, and to derive a lower bound of the actual cloudlet deployment cost depending on a few system variables, e.g., where to place cloudlets and how much computational resources per cloudlet. These variables play a major role in determining the total cost of hybrid cloudlet deployment.

The rest of this chapter is organised as follows. In Section 4.2, we briefly review some of the key concepts of convex optimisation theory. In Section 4.3, the static cloudlet placement problem is formulated as a two-stage convex optimisation problem that provides lower bounds on optimal solutions from MINLP frameworks. In Section 4.4, the proposed method to find lower bounds on optimal solutions from MINLP frameworks is validated. In Section 4.5, a parametric analysis is presented and the importance of primary network parameters on cloudlet deployment cost are discussed. Finally, in Section 4.6 concluding remarks are made, summarising the primary findings from this entire work.

## 4.2 Background on Convex optimisation Theory

In this section, at first, we review some basic terminologies related to optimisation theory and followed by review the necessary and sufficient conditions of optimality. For detailed discussions on these topics, please refer to [154, 155]. Constrained optimisation problems are generally represented by a constraint set $X$ and a cost/objective function $f$ that maps elements of $X$ into real numbers. The set $X$ consists of the available decision variables $x$ and the scalar cost function $f(x)$ measures the undesirability of choosing a decision $x$. When we want to find an optimal solution, we want to find a $x^* \in X$ such that,

$$f(x^*) \leq f(x), \qquad \forall x \in X.$$

### Vector Sequence, Set Topology, and Convex Sets

We denote a vector sequence as $\{x_k\} \subset \mathbb{R}$ and proceed to define the following:

- A sequence is "bounded" when for some scalar $C$, we have, $||x_k|| \leq C, \forall k$.
- A sequence "converges to a vector" $\tilde{x} \in \mathbb{R}^n$, when $\lim_{k\to\infty} ||x_k - \tilde{x}|| = 0$.
- A vector $y$ is an "accumulation point of the sequence" when there exists a sub-sequence $\{x_{k_i}\}$ such that $\{x_{k_i}\} \to y$ as $i \to \infty$.
- **Bolzano Theorem:** A bounded sequence $\{x_k\} \subset \mathbb{R}$ has at least one limit point.

We consider that $X$ is a subset of $\mathbb{R}^n$ and proceed to define the following:

- A vector $x$ is an "accumulation point" or a "limit point" of the set $X$ if there exists a sequence $\{x_k\} \subset X$ such that $x_k \to x$.
- The set X is a "closed set" if it contains all its accumulation points.
- The set X is a "open set" if its complement set $X^c = \{x \in \mathbb{R}^n | x \notin X\}$ is a closed set.
- The set X is a "bounded set" if for some arbitrary $C$, $||x|| \leq C, \forall x \in X$.

- The set X is a "compact set" if it is both closed and bounded.

Furthermore, we define some commonly occurring examples of special and convex sets. We consider $X$ as:

- **subspace:** if $\alpha x + \beta y \in X$ for all $x, y \in X$ and any arbitrary $\alpha, \beta \in \mathbb{R}$,
- **affine:** if it is a translated subspace, i.e., $X = x + S$ for an $x \in X$ and a subspace $S$,
- **cone:** if $x \in X$ for every $\lambda \geq 0$ and every $x \in X$,
- **convex:** if $\alpha x + (1 - \alpha)y \in X$ for any $x, y \in X$ and $\alpha \in [0, 1]$.

## Mappings, Special Mappings and Functions

We consider $X \subseteq \mathbb{R}^n$ as before and denote $F : X \to \mathbb{R}^m$ as a "mapping" $F$ from the set $X$ to $\mathbb{R}^m$. Now, we can define the following:

- $F$ is referred to as a "function" when $F : X \to R$ $(m = 1)$.
- $X$ is referred to as the "domain" of the mapping $F$.
- The "image of $X$ under mapping $F$" is denoted by the following set: $F(X) = \{y \in \mathbb{R}^m | y = F(x)\}$ for some $x \in X$.
- The "inverse image of $Y \subseteq \mathbb{R}^m$ under mapping $F$" is denoted by the following set: $F^{-1}(Y) = \{x \in X | F(x) \in Y\}$.

Note that a mapping $F : X \to \mathbb{R}^m$ with $X \subseteq \mathbb{R}^n$ is considered as:

- **affine:** if $F(x) = Ax + b$, for all $x \in X$, a matrix $A$ and a vector $b \in \mathbb{R}^m$,
- **linear:** if $F(x) = Ax$, for all $x \in X$ and a matrix $A$.

Moreover, a function $f : \mathbb{R}^n \to \mathbb{R}$ is considered as:

- **quadratic:** if $f(x) = x^T Q x + a^T x + b$, for all $x$, a matrix $Q$, and vectors $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$,
- **affine:** if $f(x) = a^T x + b$, for all $x$, vectors $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$,
- **linear:** $f(x) = a^T x$ for a vector $a \in \mathbb{R}^n$.

## Convex/Quasi-convex Functions and Optimality Conditions

A function $f$ is a "convex function" if $dom(f)$ is a convex set and the following condition holds:

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y),$$

for all $x, y \in dom(f)$ and $\alpha \in [0,1]$. The function $f$ is a "strictly convex function" for all $x, y \in dom(f)$ and $\alpha \in (0,1)$. Moreover, $f$ is a "concave function" is $-f$ is a convex function and a "strictly concave function" is $-f$ is a strictly convex function. In addition, the function $f$ is called "quasi-convex function" if $f(\alpha x + (1-\alpha)y) \leq \max\{f(x), f(y)\}$ and "quasi-concave function" if $f(\alpha x + (1-\alpha)y) \geq \min\{f(x), f(y)\}$, for all $x, y \in dom(f)$ and $\alpha \in [0,1]$. The function $f$ is called "strictly quasi-convex function" if $f(\alpha x + (1-\alpha)y) < \max\{f(x), f(y)\}$, and "strictly quasi-concave function" if $f(\alpha x + (1-\alpha)y) > \min\{f(x), f(y)\}$, for all $x, y \in dom(f)$ and $\alpha \in (0,1)$.

We assume that $f$ is a "twice differentiable" function over its domain $dom(f)$. Therefore, the "gradient of $f$" at each point $x \in dom(f)$ is defined as,

$$\nabla f(x) = \left( \frac{\partial f(x)}{\partial x_1}, \frac{\partial f(x)}{\partial x_2}, \ldots, \frac{\partial f(x)}{\partial x_n} \right).$$

The "Hessian of $f$" is a symmetric $n \times n$ matrix whose elements are the second-order partial derivatives of $f$ at each $x \in dom(f)$,

$$\left[\nabla^2 f(x)\right]_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}, \forall i, j = 1, \ldots, n.$$

The "$r$th order bordered Hessian of $f$" for $r = 1, \ldots, n$ at each $x \in dom(f)$ is defined as,

$$\mathcal{H}_B = \begin{bmatrix} 0 & \frac{\partial f(x)}{\partial x_1} & \cdots & \frac{\partial f(x)}{\partial x_r} \\ \frac{\partial f(x)}{\partial x_1} & \frac{\partial^2 f(x)}{\partial x_1^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_r} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f(x)}{\partial x_r} & \frac{\partial^2 f(x)}{\partial x_r \partial x_1} & \cdots & \frac{\partial^2 f(x)}{\partial x_r^2} \end{bmatrix}.$$

**First-Order Condition for Convexity**

The function $f$ is a convex function if and only if $dom(f)$ is a convex set and

$$f(x) + \nabla f(x)^T(z - x) \leq f(z), \forall x, z \in dom(f).$$

**Second-Order Condition for Convexity**

The function $f$ is a convex function if and only if $dom(f)$ is a convex set and

$$\nabla^2 f(x) \succeq 0, \forall x \in dom(f).$$

Moreover, $f$ is a strictly convex function if $dom(f)$ is a convex set and

$$\nabla^2 f(x) \succ 0, \forall x \in dom(f).$$

**Conditions for Quasi-convexity and Quasi-concavity**

We denote the determinant of $r$-th order bordered Hessian matrix by $D_r$.

- If $f$ is a "quasi-concave function" then $D_1(x) \leq 0, D_2(x) \geq 0, \ldots, D_r(x) \leq 0$, i.e., $D_n(x) \leq 0$ when $n$ is odd and $D_n(x) \geq 0$ when n is even, for all $x \in dom(f)$.
- If $f$ is a "quasi-convex function" then $D_1(x) \leq 0, D_2(x) \leq 0, \ldots, D_r(x) \leq 0$, $r = 1, \ldots, n$, for all $x \in dom(f)$.
- If $D_1(x) < 0, D_2(x) > 0, \ldots, D_r(x) < 0$, i.e., $D_n(x) < 0$ when $n$ is odd and $D_n(x) > 0$ when n is even, for all $x \in dom(f)$, then $f$ is a "strictly quasi-concave function".
- If $D_1(x) < 0, D_2(x) < 0, \ldots, D_r(x) < 0$, $r = 1, \ldots, n$, for all $x \in dom(f)$, then $f$ is a "strictly quasi-convex function".

## Necessary Conditions for Constrained optimisation

We consider a continuous constrained optimisation problem as follows:

$$\begin{aligned} \mathcal{P}: \qquad & \min f(x) \\ & \text{subject to } g_i(x) \leq 0, \quad i = 1, \ldots, p, \\ & \qquad\qquad h_i(x) = 0, \quad i = 1, \ldots, m, \end{aligned}$$

where $f$, $g_i$, and $h_i$ are continuously differentiable functions over $\mathbb{R}^n$. The set $X = \{x \in \mathbb{R}^n | g_i(x) \leq 0, h_i(x) = 0\}$ is defined as the "feasible set" for the above constrained optimisation problem. For a given "feasible vector" $x \in X$, an "active set of (inequality) constraints" is denoted by $\mathcal{A}(x) = \{i | g_i(x) = 0\}$. If $i \notin \mathcal{A}(x)$, we consider the $i$-th constraint to be "inactive" at $x$. Moreover, we consider a vector $x$ to be "regular" if the gradients $\nabla h_1(x), \ldots, \nabla h_m(x)$ and $\nabla g_i(x)$ for $i \in \mathcal{A}(x)$ are linearly independent [156].

**Theorem 4.1.** ***(Weierstrass Theorem)***
*Let $X$ be a nonempty compact subset of a finite dimensional Euclidean space and let $f : X \to \mathbb{R}$ be a continuous function. Then there exists an optimal solution to the optimisation problem $\mathcal{P}$.*

The Lagrangian function $\mathcal{L}(x, \lambda, \mu)$ associated with the convex optimisation problem $\mathcal{P}$ is defined by,

$$\mathcal{L}(x, \lambda, \mu) = f(x) + \sum_{i=1}^{m} \lambda_i h_i(x) + \sum_{i=1}^{p} \mu_i g_i(x),$$

where $\lambda_i \in \mathbb{R}$ for all $i = 1, \ldots, m$, and $\mu_i \in \mathbb{R}_+$ for all $i = 1, \ldots, p$. Now, we summarise the first-order and second order Karush-Kuhn-Tucker (KKT) necessary conditions below [156].

**Theorem 4.2.** ***(First-Order KKT Necessary Conditions)***
*Let $x^*$ be a local minimum of the equality/inequality constrained problem $\mathcal{P}$. Also, assume that $x^*$ is regular. Then, there exist unique multipliers $\lambda^*$ and $\mu^*$ such that,*

- $\nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = 0$, *[$\mathcal{L}$ is the Lagrangian function]*

- $\mu^* \geq 0$ *for all* $i = 1, \ldots, p$,
- $\mu^* = 0$ *for all* $i \notin \mathcal{A}(x)$. *[complementary slackness conditions]*

**Theorem 4.3.** ***(Second-Order KKT Necessary Conditions)***
*Let* $x^*$ *be a local minimum of the equality/inequality constrained problem* $\mathcal{P}$. *Also, assume that* $x^*$ *is regular and that* $f$, $h_i$, $g_i$ *are twice continuously differentiable. Then, there exist unique multipliers* $\lambda^*$ *and* $\mu^*$ *such that,*

- $\nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = 0$,
- $\mu^* \geq 0$ *for all* $i = 1, \ldots, p$,
- $\mu^* = 0$ *for all* $i \notin \mathcal{A}(x)$,
- *for any vector* $y$ *such that* $\nabla h_i(x^*)^T y = 0$ *for all* $i$ *and* $\nabla g_i(x^*)^T y = 0$ *for all* $i \in \mathcal{A}(x)$, *the following relation holds:*

$$y^T \nabla^2_{xx} \mathcal{L}(x^*, \lambda^*, \mu^*) y \geq 0.$$

## 4.3 Optimal Cloudlet Placement Problem

In this section, we present the system model and optimisation problem formulations. As we focus mainly on low-latency applications (1-100 ms), we can ignore the possibility of offloading job requests from mobile users to remote cloud servers. In contrast to the cloud servers, the cloudlets only contain a finite number of processors with parallel processing enabled. From the Google cluster-usage traces, it can be shown that job request arrival and their service times follow exponential distributions, and hence can be considered as Poisson processes [157]. Therefore, we model the cloudlets as $M/M/1$ queueing systems [143] because it is simple and useful if we just want to achieve a first-cut estimation of a system's performance. With an average service rate $\mu_c$ and average job request arrival rate $\lambda_c$, the average processing latency of any cloudlet is given by, $\tau_{cloudlet} = \frac{1}{(\mu_c - \lambda_c)}$ [142].

In the static cloudlet placement problem, we consider that the network status is static in time, i.e., we ignore both the user mobility and VM mobility and identify

Table 4.1: Network optimisation Parameters

| ***Symbol*** | ***Definition*** |
|---|---|
| $\alpha$ | Cost of installing a single processor in a cloudlet |
| $\varphi$ | Average number of jobs a single processor can simultaneously process |
| $\xi_z$ | New infrastructure installation cost at location $z \in \{f, r, o\}$ |
| $\eta$ | Cost of installing new optical fibre per kilometer |
| $L_z$ | Average length between cloudlet at $z \in \{f, r, o\}$ and ONUs |
| $\delta_z$ | Polling cycle latency between cloudlet at $z \in \{f, r, o\}$ and ONUs |
| $BW_z$ | Bandwidth of the optical fibre link between cloudlets at $z \in \{f, r, o\}$ and ONUs for both uplink and downlink |
| $n_\lambda$ | Number of wavelengths shared by ONUs to communicate to their corresponding RN cloudlet |
| $D_z$ | Transmission latency between cloudlet at $z \in \{f, r, o\}$ and ONUs |
| $D_{QoS}$ | Maximum value of allowed QoS latency requirement |
| $p$ | Number of end-users served by each ONU |
| $N$ | Split-ratio of the passive splitter of the TDM-PON branches |
| $\sigma_{ul}$ | Average number of bits an ONU sends to cloudlet at $z \in \{f, r, o\}$ for processing |
| $\sigma_{dl}$ | Average number of bits an ONU receives from cloudlet at $z \in \{f, r, o\}$ after processing |
| $\beta_{ul}$ | Background load in the uplink of the considered TDM PON |
| $\beta_{dl}$ | Background load in the downlink of the considered TDM PON |

optimal cloudlet locations over an existing access network infrastructure. This is justified for a cost-optimal cloudlet placement framework and assignment of ONUs to cloudlets [64]. We consider just one TDM-PON based FiWi branch and cloudlets can be installed in the field, RN and CO locations, as shown in Fig. 3.1. In Table 4.1, we briefly define the required network parameters. As this problem is formulated at the network planning stage without considering the user and VM mobility, it is best to design the system for the maximum workload, i.e., all the users supported by all the ONUs send job requests to the cloudlets. Thus, the total job request arrival rate is $\lambda_{max} = pN$ (jobs/s). The total transmission latency between a cloudlet and ONU is computed as the sum of to-and-fro data transmission latency and average polling cycle latency, i.e., $D_z = \left(\frac{2L_z}{v_c} + \frac{\delta_z}{2}\right), \forall z \in \{f, r, o\}$, where $v_c$ denotes the velocity of light. The job request packets from ONUs and job response packets from cloudlets are interleaved with background traffic to be cleared within one polling cycle of the considered TDM-PON.

As we assume that the network is "homogeneous" hence, each TDM-PON has

the same split ratio, each ONU serves the same number of users, and the distances between all ONUs and the field, RN, and CO cloudlets are equal to their respective average distances. We define the decision variables as follows: $x_f :=$ the fraction of total incoming workload assigned to the field cloudlets, $x_r :=$ the fraction of total incoming workload assigned to the RN cloudlets, $x_o :=$ the fraction of total incoming workload assigned to the CO cloudlets, $b_f :=$ (binary variable) takes the value of 1 if a field cloudlet is installed, and $\mu :=$ total service rate of all the processors installed in the field, at the RN and the CO (jobs/s). For this problem setup, all the variables are non-negative i.e., $\geq 0$. The objective function to minimise the overall cloudlet installation expenditure is given below:

$$\min_{\mu, x_f, x_r, x_o, b_f} \left( \frac{\alpha}{\varphi}\mu + \eta L_f N b_f + \xi_f x_f + \xi_r x_r + \xi_o x_o \right) \tag{4.1}$$

In the above expression, the first component denotes the cost of processors, the second component denotes the cost of new point-to-point fibre installation cost, and the third, fourth and fifth components denote the cloudlet installation costs in the field, at RN, and CO locations, respectively and are scaled according to the workload assigned. However, the cost of new fibre installation for supporting field cloudlet installation needs to be associated with a binary variable $b_f$, as it cannot be associate with the fractional variable $x_f$. In (4.1), the binary variable $b_f$ follows the boundary constraint $x_f \leq b_f \leq (x_f + 0.999)$, which implies that cost of new fibre installation is included in the objective function only when some workload is assigned to the field cloudlets. This makes the problem a mixed-integer programming problem and hence, cannot be solved by using common convex optimisation techniques. To overcome this issue, we reformulate this problem into a two-stage convex optimisation problem. In our first-stage problem formulation, we do not consider the scope of field cloudlet installation and use the variables $x_r$, $x_o$, and $\mu$ only. In other words, we consider the presence of RN and CO cloudlets, only. In the second-stage problem formulation, we consider that the field cloudlets are installed along with RN and CO cloudlets. Therefore, we include the cost of new fibre installation directly in the objective function.

### 4.3.1 First-Stage optimisation Problem Formulation

The modified objective function to minimise the overall cloudlet installation expenditures is:

$$\min_{\mu, x_r, x_o} \left( \frac{\alpha}{\varphi}\mu + \xi_r x_r + \xi_o x_o \right) \tag{4.2}$$

The following constraint ensures that all the ONUs are served either by RN or CO cloudlet. Moreover, this constraint helps to reduce one decision variable as well.

$$x_r + x_o = 1 \quad \text{or,} \quad x_o = 1 - x_r \tag{4.3}$$

The non-negativity constraints on the decision variables are given as follows:

$$(\mu - pN) \geq 0 \quad \text{and} \quad 0 \leq x_r \leq 1 \tag{4.4}$$

The latency constraint that ensures that the overall system latency is not greater than $D_{QoS}$ is given below:

$$\begin{aligned} & x_r \left\{ \frac{1}{\mu x_r - pN x_r} + D_r + \frac{N\sigma_{ul}}{n_\lambda BW_r} + \frac{N\sigma_{dl}}{n_\lambda BW_r} \right\} \\ & + x_o \left\{ \frac{1}{\mu x_o - pN x_o} + D_o + \frac{N\sigma_{ul}}{BW_o - \beta_{ul}} + \frac{N\sigma_{dl}}{BW_o - \beta_{dl}} \right\} \leq D_{QoS} \end{aligned} \tag{4.5}$$

In (4.5), each factor multiplied with $x_r$ and $x_o$ consists of four terms, i.e., processing latency of the cloudlet, the transmission latency, the latency to offload total bits for the job request from ONU to cloudlet, and the latency to receive total bits post-processing by ONU from cloudlet, respectively.

Now, using the constraint (4.3), moving denominator terms to numerator and rearranging all terms, we rewrite (4.5) as below:

$$2 + \{(A - B)x_r + (B - D_{QoS})\}(\mu - pN) \leq 0 \tag{4.6}$$

where,

$$A = D_r + \frac{N\sigma_{ul}}{n_\lambda BW_r} + \frac{N\sigma_{dl}}{n_\lambda BW_r}$$
$$B = D_o + \frac{N\sigma_{ul}}{BW_o - \beta_{ul}} + \frac{N\sigma_{dl}}{BW_o - \beta_{dl}}$$

**Proposition 4.4.** *The function* $f(x_r, \mu) = 2 + \{(A-B)x_r + (B - D_{QoS})\}(\mu - pN)$ *is non-convex but quasi-convex in nature.*

*Proof.* The first-order partial derivatives of $f(x_r, \mu)$ with respect to $x_r$ and $\mu$ are given below:

$$\frac{\partial f}{\partial x_r} = (A-B)(\mu - pN)$$
$$\frac{\partial f}{\partial \mu} = \{(A-B)x_r + (B - D_{QoS})\}$$

To prove the first part of the statement, let us evaluate the Hessian matrix [158] for the function $f(x_r, \mu)$ as follows:

$$\mathcal{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_r^2} & \frac{\partial^2 f}{\partial x_r \partial \mu} \\ \frac{\partial^2 f}{\partial \mu \partial x_r} & \frac{\partial^2 f}{\partial \mu^2} \end{bmatrix} = \begin{bmatrix} 0 & (A-B) \\ (A-B) & 0 \end{bmatrix} \tag{4.7}$$
$$\det(\mathcal{H}) = -(B-A)^2$$

Clearly, $\mathcal{H}$ is not positive semi-definite as $\det(\mathcal{H}) < 0$ and hence, $f(x_r, \mu)$ is a non-convex function [158]. Now, to prove the second part of the statement, we evaluate the bordered Hessian matrix [158] for the function $f(x_r, \mu)$ as follows:

$$\mathcal{H}_B = \begin{bmatrix} 0 & \frac{\partial f}{\partial x_r} & \frac{\partial f}{\partial \mu} \\ \frac{\partial f}{\partial x_r} & \frac{\partial^2 f}{\partial x_r^2} & \frac{\partial^2 f}{\partial x_r \partial \mu} \\ \frac{\partial f}{\partial \mu} & \frac{\partial^2 f}{\partial \mu \partial x_r} & \frac{\partial^2 f}{\partial \mu^2} \end{bmatrix} \tag{4.8}$$

$$\det(\mathcal{H}_B) = -2(B-A)^2(\mu - pN)\{(B-A)x_r - (B - D_{QoS})\}$$

It can be shown that when constraint (4.6) holds, then $\det(\mathcal{H}_B) \leq 0$. Therefore, as the determinants of both the first-order and second-order bordered Hessian matrices are negative, we can consider $f(x_r, \mu)$ as a quasi-convex function [158]. □

We observe that the objective function (4.2) and the constraint (4.4) are linear. Moreover, the Proposition 4.4 proves that the constraint (4.6) is quasi-convex. Therefore, in the optimisation problem with objective function (4.2) and constraints (4.4) and (4.6), if at least one of the Lagrange multipliers that satisfy KKT conditions exist, then any local optimum of this problem is also the global optimum. Hence, the first-order KKT (necessary) conditions are also sufficient conditions for optimality [158]. Now, we write the Lagrangian function for the objective function (4.2) and constraints (4.4) and (4.6) as follows:

$$\mathcal{L}(x_r, \mu; \boldsymbol{\lambda}) = \frac{\alpha}{\varphi}\mu + \xi_r x_r + \xi_o(1 - x_r) - \lambda_1 x_r - \lambda_2(1 - x_r) \\ - \lambda_3(\mu - pN) + \lambda_4[2 + \{(A - B)x_r + (B - D_{QoS})\}(\mu - pN)] \tag{4.9}$$

We obtain the first-order KKT conditions for optimality from the Lagrangian function (4.9) as follows:

$$\frac{\partial \mathcal{L}}{\partial \mu} = \frac{\alpha}{\varphi} - \lambda_3 + \lambda_4\{(A - B)x_r + (B - D_{QoS})\} = 0 \tag{4.10}$$

$$\frac{\partial \mathcal{L}}{\partial x_r} = \xi_r - \xi_o - \lambda_1 + \lambda_2 + \lambda_4(A - B)(\mu - pN) = 0 \tag{4.11}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_1} = -x_r \quad [\because x_r \neq 0, \therefore \lambda_1 = 0] \tag{4.12}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_2} = -(1 - x_r) \quad [\because (1 - x_r) \neq 0, \therefore \lambda_2 = 0] \tag{4.13}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_3} = -(\mu - pN) \quad [\because (\mu - pN) \neq 0, \therefore \lambda_3 = 0] \tag{4.14}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_4} = 2 + \{(A - B)x_r + (B - D_{QoS})\}(\mu - pN) = 0 \tag{4.15}$$

As (4.12)-(4.14) indicate that $\lambda_1 = 0$, $\lambda_2 = 0$, and $\lambda_3 = 0$ respectively, hence by eliminating $\lambda_4$ from (4.10)-(4.11), we obtain the expression for $(\mu - pN)$, as shown below:

$$\mu - pN = \frac{\varphi(\xi_r - \xi_o)\{(A - B)x_r + (B - D_{QoS})\}}{\alpha(A - B)} \tag{4.16}$$

Again, rearranging the different terms in (4.15), we obtain another expression for $(\mu - pN)$ as below:

$$\mu - pN = \frac{-2}{(A-B)x_r + B - D_{QoS}} \tag{4.17}$$

Eliminating $(\mu - pN)$ from (4.16) and (4.17), we obtain the closed form expression for $x_r$ as shown below:

$$x_r = \frac{1}{(B-A)}\left[(B - D_{QoS}) \mp \sqrt{\frac{2\alpha(B-A)}{\varphi(\xi_r - \xi_o)}}\right] \tag{4.18}$$

Putting the above expression for $x_r$ in (4.18) and (4.11), we obtain the closed form expressions for $\mu$ and $\lambda_4$, respectively as shown below:

$$\mu = \left[pN \mp 2\sqrt{\frac{\varphi(\xi_r - \xi_o)}{2\alpha(B-A)}}\right] \tag{4.19}$$

$$\lambda_4 = \mp\frac{\alpha}{\varphi}\sqrt{\frac{\varphi(\xi_r - \xi_o)}{2\alpha(B-A)}} \tag{4.20}$$

Now, we evaluate the bordered Hessian matrix for the Lagrangian function (4.9) as follows:

$$\mathcal{H}'_B = \begin{bmatrix} 0 & \frac{\partial^2\mathcal{L}}{\partial x_r \partial \lambda_4} & \frac{\partial^2\mathcal{L}}{\partial \mu \partial \lambda_4} \\ \frac{\partial^2\mathcal{L}}{\partial x_r \partial \lambda_4} & \frac{\partial^2\mathcal{L}}{\partial x_r^2} & \frac{\partial^2\mathcal{L}}{\partial x_r \partial \mu} \\ \frac{\partial^2\mathcal{L}}{\partial \mu \partial \lambda_4} & \frac{\partial^2\mathcal{L}}{\partial \mu \partial x_r} & \frac{\partial^2\mathcal{L}}{\partial \mu^2} \end{bmatrix} \tag{4.21}$$

where,

$$\begin{aligned} \frac{\partial^2\mathcal{L}}{\partial x_r^2} &= \frac{\partial^2\mathcal{L}}{\partial \mu^2} = 0 \\ \frac{\partial^2\mathcal{L}}{\partial x_r \partial \mu} &= \frac{\partial^2\mathcal{L}}{\partial \mu \partial x_r} = \lambda_4(A-B) \\ \frac{\partial^2\mathcal{L}}{\partial x_r \partial \lambda_4} &= (A-B)(\mu - pN) \\ \frac{\partial^2\mathcal{L}}{\partial \mu \partial \lambda_4} &= \{(A-B)x_r + (B - D_{QoS})\} \end{aligned}$$

$$\det(\mathcal{H}'_B) = -2\lambda_4(B-A)^2(\mu - pN)\{(B-A)x_r - (B - D_{QoS})\} \tag{4.22}$$

Clearly, with the conditions $(B-A)^2 > 0$, $\xi_r > \xi_o$, and for the following set of values, $\det(\mathcal{H}'_B) < 0$:

$$\begin{aligned}
x_r^* &= \frac{1}{(B-A)}\left[(B - D_{QoS}) + \sqrt{\frac{2\alpha(B-A)}{\varphi(\xi_r - \xi_o)}}\right] \\
x_o^* &= 1 - x_r^* \\
\mu^* &= \left[pN + 2\sqrt{\frac{\varphi(\xi_r - \xi_o)}{2\alpha(B-A)}}\right] \\
\lambda_4^* &= \frac{\alpha}{\varphi}\sqrt{\frac{\varphi(\xi_r - \xi_o)}{2\alpha(B-A)}} \\
\det(\mathcal{H}'_B) &= -\frac{4\alpha}{\varphi}(B-A)^2\sqrt{\frac{\varphi(\xi_r - \xi_o)}{2\alpha(B-A)}} < 0
\end{aligned} \tag{4.23}$$

Therefore, the minima of the optimisation problem with objective function (4.2) and constraints (4.4) and (4.6) exists at $(x_r^*, \mu^*, \lambda_4^*)$ and the corresponding minimal cost value is given as follows:

$$\begin{aligned}
\mathbb{C}_1 = &\frac{\alpha}{\varphi}\left[pN + 2\sqrt{\frac{\varphi(\xi_r - \xi_o)}{2\alpha(B-A)}}\right] \\
&+ \frac{(\xi_r - \xi_o)}{(B-A)}\left[(B - D_{QoS}) + \sqrt{\frac{2\alpha(B-A)}{\varphi(\xi_r - \xi_o)}}\right] + \xi_o
\end{aligned} \tag{4.24}$$

Instead of choosing RN and CO cloudlets, we can select field and CO cloudlets and solving the problem in a similar method as above, we obtain the minimal cost value as follows:

$$\begin{aligned}
\tilde{\mathbb{C}}_1 = &\frac{\alpha}{\varphi}\left[pN + 2\sqrt{\frac{\varphi(\xi_f - \xi_o)}{2\alpha(B-C)}}\right] + \eta L_f N \\
&+ \frac{(\xi_f - \xi_o)}{(B-C)}\left[(B - D_{QoS}) + \sqrt{\frac{2\alpha(B-C)}{\varphi(\xi_f - \xi_o)}}\right] + \xi_o
\end{aligned} \tag{4.25}$$

where,

$$C = D_f + \frac{\sigma_{ul}}{BW_f} + \frac{\sigma_{dl}}{BW_f}$$

However, in most practical scenarios installation of RN and CO cloudlets are preferred more than the field and CO cloudlets as installation of field cloudlets includes additional installation cost of point-to-point fibres and new transceiver pairs. Therefore, in general $\mathbb{C}_1 < \tilde{\mathbb{C}}_1$ holds and hence, $\mathbb{C}_1$ is chosen over $\tilde{\mathbb{C}}_1$. Similar conclusion is also valid for the field and RN cloudlet combination. Note that, in the above solution we considered that $\lambda_1 = 0$, $\lambda_2 = 0$, and $\lambda_3 = 0$ from the complementary-slackness conditions (4.12)-(4.14). Nonetheless, there may be instances when the cost optimisation framework decides not to install any RN cloudlet. In such a case, we need to enforce $x_r = 0$ such that $\lambda_1 \neq 0$, $\lambda_2 = 0$ and $x_o = 1 - x_r = 1$. However, as the framework tries to keep the processing time finite and below $D_{QoS}$, (4.14) is valid always. Therefore, this situation arises when the following condition gets satisfied:

$$B - \sqrt{\frac{2\alpha(B-A)}{\varphi(\xi_r - \xi_o)}} \leq D_{QoS} \tag{4.26}$$

In this situation, the above solution $(x_r^*, \mu^*, \lambda_4^*)$ is no longer valid and we need to reduce the problem with just one variable $\mu$. The modified objective function is given below:

$$\min_{\mu} \left( \frac{\alpha}{\varphi}\mu + \xi_o \right) \tag{4.27}$$

Also, the modified constraints are given as follows:

$$(\mu - pN) \geq 0 \tag{4.28}$$

$$\left\{ \frac{1}{\mu - pN} + D_o + \frac{N\sigma_{ul}}{BW_o - \beta_{ul}} + \frac{N\sigma_{dl}}{BW_o - \beta_{dl}} \right\} \leq D_{QoS} \tag{4.29}$$

The constraint (4.29) is a linear constraint and we can rewrite the above constraint as follows:

$$1 - (D_{QoS} - B)(\mu - pN) \leq 0 \tag{4.30}$$

The Lagrange function for this reduced form of problem is given below:

$$\begin{aligned}\mathcal{L}'(\mu; \boldsymbol{\nu}) =& \frac{\alpha}{\varphi}\mu + \xi_o - \nu_1(\mu - pN) \\ &+ \nu_2[1 - (D_{QoS} - B)(\mu - pN)]\end{aligned} \tag{4.31}$$

Therefore, the first-order KKT conditions are given as follows:

$$\frac{\partial \mathcal{L}'}{\partial \mu} = \frac{\alpha}{\varphi} - \nu_1 - \nu_2(D_{QoS} - B) = 0 \tag{4.32}$$

$$\frac{\partial \mathcal{L}'}{\partial \nu_1} = -(\mu - pN) \quad [\because (\mu - pN) \neq 0, \therefore \nu_1 = 0] \tag{4.33}$$

$$\frac{\partial \mathcal{L}'}{\partial \nu_2} = 1 - (D_{QoS} - B)(\mu - pN) = 0 \tag{4.34}$$

By using the conditions (4.32)-(4.34), we obtain the optimal solution and to avoid ambiguity, we denote the optimal solution by $(\bar{\mu}^*, \nu_2^*)$ as follows:

$$\bar{\mu}^* = pN + \frac{1}{D_{QoS} - B} \tag{4.35}$$

$$\nu_2^* = \frac{\alpha}{\varphi(D_{QoS} - B)} \tag{4.36}$$

With these latest values of the variables, the cost function (4.27) yields the corresponding optimal cost value as follows:

$$\mathbb{C}_2 = \frac{\alpha}{\varphi}\left[pN + \frac{1}{D_{QoS} - B}\right] + \xi_o \tag{4.37}$$

If we want to install only RN cloudlets, then we need to enforce $x_r = 1$ such that $\lambda_1 = 0$, $\lambda_2 \neq 0$ and $x_o = 1 - x_r = 0$. Formulating a similar optimisation problem as

above, we find the optimal value of $\mu$ (denoted by $\bar{\bar{\mu}}^*$) and optimal cost as follows:

$$\bar{\bar{\mu}}^* = pN + \frac{1}{D_{QoS} - A} \tag{4.38}$$

$$\mathbb{C}_3 = \frac{\alpha}{\varphi}\left[pN + \frac{1}{D_{QoS} - A}\right] + \xi_r \tag{4.39}$$

Nonetheless, when installation RN and CO cloudlets are no more sufficient to meet the QoS requirements, we need to install additional field cloudlets and proceed to the second stage optimisation problem formulation. This situation arises when the following condition holds:

$$A - \sqrt{\frac{2\alpha(B-A)}{\varphi(\xi_r - \xi_o)}} \geq D_{QoS} \tag{4.40}$$

### 4.3.2 Second-Stage optimisation Problem Formulation

In the second-stage of optimisation problem formulation, we assume that all of the field, RN, and CO cloudlets are installed and hence we use the variables $x_f$, $x_r$, $x_o$, and $\mu$. We assume that all $N$ ONUs are connected to the field cloudlet and hence $N$ new point-to-point fibres are installed, each with a dedicated bandwidth $BW_f$. Thus the corresponding objective function to minimise the overall CAPEX is given below:

$$\min_{\mu, x_f, x_r, x_o} \left(\frac{\alpha}{\varphi}\mu + \eta L_f N + \xi_f x_f + \xi_r x_r + \xi_o x_o\right) \tag{4.41}$$

Note that, in the above cost function, the binary variable $b_f$ is no longer included. To ensures that all cloudlets are served either by field, RN or CO cloudlet we write the following constraint and one decision variable as follows:

$$x_f + x_r + x_o = 1 \quad \text{or,} \quad x_o = 1 - x_f - x_r \tag{4.42}$$

As the variables can take only non-negative values, we write the following constraints on the parameters:

$$(\mu - pN) \geq 0 \quad \text{and} \quad 0 \leq x_f \leq 1 \quad \text{and} \quad 0 \leq x_r \leq 1 \tag{4.43}$$

The new latency constraint ensuring that the overall system latency is not exceeding $D_{QoS}$ is given below:

$$\begin{aligned} & x_f \left\{ \frac{1}{\mu x_f - pNx_f} + D_f + \frac{N\sigma_{ul}}{NBW_f} + \frac{N\sigma_{dl}}{NBW_f} \right\} \\ & + x_r \left\{ \frac{1}{\mu x_r - pNx_r} + D_r + \frac{N\sigma_{ul}}{n_\lambda BW_r} + \frac{N\sigma_{dl}}{n_\lambda BW_r} \right\} \\ & + x_o \left\{ \frac{1}{\mu x_o - pNx_o} + D_o + \frac{N\sigma_{ul}}{BW_o - \beta_{ul}} + \frac{N\sigma_{dl}}{BW_o - \beta_{dl}} \right\} \leq D_{QoS} \end{aligned} \tag{4.44}$$

The factors multiplied with each of $x_f$, $x_r$, and $x_o$ contain terms similar to that in (4.5). Thus, by using (4.44) and adjusting the terms from denominator to numerator, we obtain the following:

$$3 + \{(C-B)x_f + (A-B)x_r + (B - D_{QoS})\}(\mu - pN) \leq 0 \tag{4.45}$$

**Proposition 4.5.** *The function* $g(x_f, x_r, \mu) = 3 + \{(C-B)x_f + (A-B)x_r + (B - D_{QoS})\}(\mu - pN)$ *is neither convex nor quasi-convex/concave in nature.*

*Proof.* The first-order partial derivatives of $g(x_f, x_r, \mu)$ with respect to $x_f$, $x_r$, and $\mu$ are given below:

$$\begin{aligned} \frac{\partial g}{\partial x_f} &= (C-B)(\mu - pN) \\ \frac{\partial g}{\partial x_r} &= (A-B)(\mu - pN) \\ \frac{\partial g}{\partial \mu} &= \{(C-B)x_f + (A-B)x_r + (B - D_{QoS})\} \end{aligned}$$

The Hessian matrix for the function $g(x_f, x_r, \mu)$ is evaluated as follows:

$$\mathcal{H}'' = \begin{bmatrix} \frac{\partial^2 g}{\partial \mu^2} & \frac{\partial^2 g}{\partial \mu \partial x_f} & \frac{\partial^2 g}{\partial \mu \partial x_r} \\ \frac{\partial^2 g}{\partial x_f \partial \mu} & \frac{\partial^2 g}{\partial x_f^2} & \frac{\partial^2 g}{\partial x_f \partial x_r} \\ \frac{\partial^2 g}{\partial x_r \partial x_f} & \frac{\partial^2 g}{\partial x_r \partial \mu} & \frac{\partial^2 g}{\partial x_r^2} \end{bmatrix} = \begin{bmatrix} 0 & (C-B) & (A-B) \\ (C-B) & 0 & 0 \\ (A-B) & 0 & 0 \end{bmatrix} \tag{4.46}$$

Clearly, $\det(\mathcal{H}'') = 0$. Therefore, $\mathcal{H}''$ is not positive semi-definite and hence $g(x_f, x_r, \mu)$ is neither a convex nor a concave function [158]. This completes the first part of the proof. Now to prove the second part of the statement, we evaluate the

bordered Hessian matrix as follows:

$$\mathcal{H}''_B = \begin{bmatrix} 0 & \frac{\partial g}{\partial \mu} & \frac{\partial g}{\partial x_f} & \frac{\partial g}{\partial x_r} \\ \frac{\partial g}{\partial \mu} & \frac{\partial^2 g}{\partial \mu^2} & \frac{\partial^2 g}{\partial \mu \partial x_f} & \frac{\partial^2 g}{\partial \mu \partial x_r} \\ \frac{\partial g}{\partial x_f} & \frac{\partial^2 g}{\partial x_f \partial \mu} & \frac{\partial^2 g}{\partial x_f^2} & \frac{\partial^2 g}{\partial x_f \partial x_r} \\ \frac{\partial g}{\partial x_r} & \frac{\partial^2 g}{\partial x_r \partial x_f} & \frac{\partial^2 g}{\partial x_r \partial \mu} & \frac{\partial^2 g}{\partial x_r^2} \end{bmatrix} \tag{4.47}$$

Putting in the values of all the elements in the above matrix (4.47), we find that $\det(\mathcal{H}''_B) = 0$. Hence, the function $g(x_f, x_r, \mu)$ cannot be considered as a quasi-convex/concave function [158]. This completes the proof. □

Due to Proposition 4.5, we cannot apply KKT conditions on the Lagrangian function for objective function (4.41) and constraints (4.43) and (4.45). If we apply the KKT conditions forcefully on this optimisation problem, then it will lead to a saddle point and we will not be able to find any optima [158]. This intractability compels us to reformulate our second-stage optimisation problem similar to the first-stage problem with some numerical techniques to find the optimal solution. We define a new variable, $x_{ro}$ := the fraction of total incoming workload assigned to the RN and CO cloudlet together and the weighted average of the cloudlet installation cost is $\xi_{ro} = \xi_r \epsilon + \xi_o(1 - \epsilon)$, where $\epsilon \in [0, 1]$, but its value is unknown at this stage. Thus, the reformulated objective function is given below:

$$\min_{\mu, x_f, x_{ro}} \left( \frac{\alpha}{\varphi} \mu + \eta L_f N + \xi_f x_f + \xi_{ro} x_{ro} \right) \tag{4.48}$$

Moreover, the reformulated constraints are given as follows:

$$x_f + x_{ro} = 1 \quad \text{or,} \quad x_{ro} = 1 - x_f \tag{4.49}$$

$$(\mu - pN) \geq 0 \quad \text{and} \quad 0 \leq x_f \leq 1 \tag{4.50}$$

$$x_f \left\{ \frac{1}{\mu x_f - pNx_f} + D_f + \frac{N\sigma_{ul}}{NBW_f} + \frac{N\sigma_{dl}}{NBW_f} \right\} + x_{ro}\epsilon \left\{ \frac{1}{\mu x_{ro}\epsilon - pNx_{ro}\epsilon} + D_r + \frac{N\sigma_{ul}}{n_\lambda BW_r} + \frac{N\sigma_{dl}}{n_\lambda BW_r} \right\} + x_{ro}(1-\epsilon) \left\{ \frac{1}{\mu x_{ro}(1-\epsilon) - pNx_{ro}(1-\epsilon)} + D_o + \frac{N\sigma_{ul}}{BW_o - \beta_{ul}} + \frac{N\sigma_{dl}}{BW_o - \beta_{dl}} \right\} \leq D_{QoS} \tag{4.51}$$

Using (4.49) and readjusting terms from denominator to numerator, we can rewrite the constraint (4.51) as follows:

$$3 + \{(C-Q)x_f + (Q - D_{QoS})\}(\mu - pN) \leq 0 \tag{4.52}$$

where,

$$Q = \epsilon \left( D_r + \frac{N\sigma_{ul}}{n_\lambda BW_r} + \frac{N\sigma_{dl}}{n_\lambda BW_r} \right) + (1-\epsilon) \left( D_o + \frac{N\sigma_{ul}}{BW_o - \beta_{ul}} + \frac{N\sigma_{dl}}{BW_o - \beta_{dl}} \right)$$

The function $h(x_f, \mu) = 3 + \{(C-Q)x_f + (Q - D_{QoS})\}(\mu - pN)$ is similar to $f(x_r, \mu)$ and hence is non-convex but quasi-convex in nature. Therefore, we can apply KKT conditions on the Lagrangian function of this newly formulated second-stage problem to find optimal solution, and write the Lagrangian function as follows:

$$\mathcal{L}''(\mu, x_f; \boldsymbol{\gamma}) = \frac{\alpha}{\varphi}\mu + \eta L_f N + \xi_f x_f + \xi_{ro}(1 - x_f) - \gamma_1 x_f - \gamma_2(1 - x_f) - \gamma_3(\mu - pN) + \gamma_4[3 + \{(C-Q)x_f + (Q - D_{QoS})\}(\mu - pN)] \tag{4.53}$$

Thus, we derive the first-order KKT conditions from the Lagrangian function (4.53) as shown below:

$$\frac{\partial \mathcal{L}''}{\partial \mu} = \frac{\alpha}{\varphi} - \gamma_3 + \gamma_4\{(C-Q)x_f + (Q - D_{QoS})\} = 0 \tag{4.54}$$

$$\frac{\partial \mathcal{L}''}{\partial x_f} = \xi_f - \xi_{ro} - \gamma_1 + \gamma_2 - \gamma_4(C-Q)(\mu - pN) = 0 \tag{4.55}$$

$$\frac{\partial \mathcal{L}''}{\partial \gamma_1} = -x_f \quad [\because x_f \neq 0, \therefore \gamma_1 = 0] \tag{4.56}$$

$$\frac{\partial \mathcal{L}''}{\partial \gamma_2} = -(1 - x_f) \quad [\because (1 - x_f) \neq 0, \therefore \gamma_2 = 0] \tag{4.57}$$

$$\frac{\partial \mathcal{L}''}{\partial \gamma_3} = -(\mu - pN) \quad [\because (\mu - pN) \neq 0, \therefore \gamma_3 = 0] \tag{4.58}$$

$$\frac{\partial \mathcal{L}''}{\partial \gamma_4} = 3 + \{(C - Q)x_f + (Q - D_{QoS})\}(\mu - pN) = 0 \tag{4.59}$$

From the conditions (4.56)-(4.58), we find that $\gamma_1 = 0$, $\gamma_2 = 0$, and $\gamma_3 = 0$. Thus we evaluate $(\mu - pN)$ by eliminating $\gamma_4$ from (4.54)-(4.55) as follows:

$$\mu - pN = \frac{(\xi_f - \xi_{ro})\varphi\{(C - Q)x_f + (Q - D_{QoS})\}}{\alpha(C - Q)} \tag{4.60}$$

From (4.55) we obtain another expression for $(\mu - pN)$ as below:

$$\mu - pN = \frac{-3}{(C - Q)x_f + (Q - D_{QoS})} \tag{4.61}$$

Comparing (4.60) and (4.61), we eliminate $(\mu - pN)$ to obtain the closed form expression for $x_f$ as shown below:

$$x_f = \frac{1}{(Q - C)} \left[ (Q - D_{QoS}) \mp \sqrt{\frac{3\alpha(Q - C)}{\varphi(\xi_f - \xi_{ro})}} \right] \tag{4.62}$$

By using the above expression for $x_f$ in (4.62), we obtain the closed form expressions for $\mu$ and $\gamma_4$, respectively as shown below:

$$\mu = \left[ pN \mp 3\sqrt{\frac{\varphi(\xi_f - \xi_{ro})}{3\alpha(Q - C)}} \right] \tag{4.63}$$

$$\gamma_4 = \mp \frac{\alpha}{\varphi} \sqrt{\frac{\varphi(\xi_f - \xi_{ro})}{3\alpha(Q - C)}} \tag{4.64}$$

By observing the similarity of the reformulated second-stage problem with the first-stage problem, we can consider that the minima of this problem exist at the following points, with conditions $(Q - C)^2 > 0$ and $\xi_f > \xi_{ro}$. To avoid ambiguity, we denote the optimal number of processors after second-stage problem by $\hat{\mu}^*$, the optimal fraction of total workload handled by RN cloudlet by $\bar{x_r^*}$, and the optimal

fraction of total workload handled by CO cloudlet by $\bar{x_o^*}$.

$$x_f^* = \frac{1}{(Q-C)}\left[(Q - D_{QoS}) + \sqrt{\frac{3\alpha(Q-C)}{\varphi(\xi_f - \xi_{ro})}}\right] \tag{4.65}$$

$$\bar{x_r^*} = (1-x_f^*)\epsilon^*, \bar{x_o^*} = (1-x_f^*)(1-\epsilon^*) \tag{4.66}$$

$$\hat{\mu}^* = \left[pN + 3\sqrt{\frac{\varphi(\xi_f - \xi_{ro})}{3\alpha(Q-C)}}\right] \tag{4.67}$$

$$\gamma_4^* = \frac{\alpha}{\varphi}\sqrt{\frac{\varphi(\xi_f - \xi_{ro})}{3\alpha(Q-C)}} \tag{4.68}$$

However, we cannot use the above expressions yet as $\epsilon$ is unknown to us, but we can use our first-stage optimisation problem to solve this issue. We observe from (4.59) that the latency constraint is active at the boundary and hence equality holds. Moreover, field cloudlets handle $pNx_f^*$ amount of workload and only $pN(1-x_f^*)$ amount of workload is handled by RN and CO cloudlets. Therefore, we can use the modified parameters $N' = N(1-x_f^*)$ and $D'_{QoS} = D_{QoS} - x_f^*\left\{1/(\hat{\mu}^* x_f^* - pNx_f^*) + C\right\}$ on our first-stage optimisation problem formulation. This allows us to compute the modified network parameters $A' = A(1-x_f^*)$ and $B' = B(1-x_f^*)$. As $\epsilon$ is equivalent to $x_r$ in the first-stage problem, we can write an expression for $\epsilon^*$ by using (4.18) as follows:

$$\epsilon^* = \frac{1}{(B'-A')}\left[(B' - D'_{QoS}) + \sqrt{\frac{2\alpha(B'-A')}{\varphi(\xi_r - \xi_o)}}\right] \tag{4.69}$$

Now, we can numerically solve the system of non-linear equations consisting of (4.65) and (4.69) to find the final values of $x_f^*$ and $\epsilon^*$. Followed by we can compute the expressions in (4.66)-(4.68). The following theorem shows that the optimal solution obtained from the reformulated problem is equal to the optimal solution of the actual problem.

**Theorem 4.6.** *The optimal solution of the actual second-stage optimisation problem with objective function (4.41) and constraints (4.42)-(4.44) is identical to that of the reformulated second-stage optimisation problem with objective function (4.48) and constraints (4.49)-(4.51) and (4.69).*

*Proof.* We consider $\breve{\mu}^*, \breve{x}_f^*, \breve{x}_r^*$, and $\breve{x}_o^*$ as the optimal values from the actual second-stage problem and can safely assume that $\breve{x}_f^*, \breve{x}_r^*, \breve{x}_o^* \neq 0$. Although we showed that this problem cannot be solved analytically, but by simple inspection we observe that the objective function (4.41) is linear. Hence the optimal solution must exist at the boundary of the solution space. Equality can not hold for (4.43), hence it must hold for (4.44) at the optima. The optimal cost is $\left(\frac{\alpha}{\varphi}\breve{\mu}^* + \eta L_f N + \xi_f \breve{x}_f^* + \xi_r \breve{x}_r^* + \xi_o \breve{x}_o^*\right)$ and the latency constraint is as follows:

$$\breve{x}_r^* \left\{ \frac{1}{\breve{\mu}^* \breve{x}_r^* - pN\breve{x}_r^*} + A \right\} + \breve{x}_o^* \left\{ \frac{1}{\breve{\mu}^* \breve{x}_o^* - pN\breve{x}_o^*} + B \right\} + \breve{x}_f^* \left\{ \frac{1}{\breve{\mu}^* \breve{x}_f^* - pN\breve{x}_f^*} + C \right\} = D_{QoS} \tag{4.70}$$

In the reformulated second-stage optimisation problem, the optimal values are $\hat{\mu}^*, x_f^*, x_{ro}^*, \bar{x}_r^*, \bar{x}_o^*$ and $\epsilon^*$. Therefore, the optimal value of the objective function is

$$\begin{aligned} &\frac{\alpha}{\varphi}\hat{\mu}^* + \eta L_f N + \xi_f x_f^* + \xi_{ro} x_{ro}^* \\ &\quad = \frac{\alpha}{\varphi}\hat{\mu}^* + \eta L_f N + \xi_f x_f^* + \xi_r \bar{x}_r^* + \xi_o \bar{x}_o^* \end{aligned} \tag{4.71}$$

As the objective values of both the actual and reformulated problems are similar, therefore, the optimal values $\hat{\mu}^*, x_f^*, \bar{x}_r^* = \epsilon^* x_{ro}^*$, and $\bar{x}_o^* = (1-\epsilon^*)x_{ro}^*$ from the reformulated problem are also similar to the optimal values $\breve{\mu}^*, \breve{x}_f^*, \breve{x}_r^*$, and $\breve{x}_o^*$ from the actual problem.

Now, from the KKT condition (4.59), we see that equality holds for the latency constraint in this problem and hence, from (4.51),

$$\bar{x}_r^* \left\{ \frac{1}{\hat{\mu}^* \bar{x}_r^* - pN\bar{x}_r^*} + A \right\} + \bar{x}_o^* \left\{ \frac{1}{\hat{\mu}^* \bar{x}_o^* - pN\bar{x}_o^*} + B \right\} + x_f^* \left\{ \frac{1}{\hat{\mu}^* x_f^* - pN x_f^*} + C \right\} = D_{QoS} \tag{4.72}$$

As we use our first-stage problem with modified parameters $N', A', B'$ and $D'_{QoS}$ to find $\epsilon^*$ and $x_f^*$, therefore with some algebraic manipulation on (4.5) and by using $\bar{x}_r^* = \epsilon^*(1 - x_f^*)$, and $\bar{x}_o^* = (1-\epsilon^*)(1-x_f^*)$, we again find (4.72).

Clearly, (4.70) and (4.72) appear to be similar constraints. Therefore, both the actual and reformulated optimisation problems have similar objective functions and solution spaces. Moreover, we can show that the first order derivatives, $\frac{\mathrm{d}x_f}{\mathrm{d}\epsilon} \geq 0$ from (4.65) and $\frac{\mathrm{d}\epsilon}{\mathrm{d}x_f} \geq 0$ from (4.69), when the conditions $0 \leq \epsilon \leq 1$, $0 \leq x_f \leq 1$, $(Q - C) \geq 0$, and $(B - A) \geq 0$ hold. Therefore, both $\epsilon$ and $x_f$ are monotonically increasing with respect to each other and hence, the solution of non-linear system of equations consisting of (4.65) and (4.69) is unique. Also, this solution is optimal when computed by any numerical technique. Thus, the optimal solution of the actual problem is equal to the optimal solution of the reformulated problem. □

Therefore, we evaluate the minimal cost value of the optimisation problem with objective function (4.48) and constraints (4.50) and (4.52) at $(x_f^*, \hat{\mu}^*, \gamma_4^*)$ as shown below:

$$\begin{aligned}\mathbb{C}_4 = \frac{\alpha}{\varphi}\left[pN + 3\sqrt{\frac{\varphi(\xi_f - \xi_{ro})}{3\alpha(Q-C)}}\right] + \eta L_f N \\ + \frac{(\xi_f - \xi_{ro})}{(Q-C)}\left[(Q - D_{QoS}) + \sqrt{\frac{3\alpha(Q-C)}{\varphi(\xi_f - \xi_{ro})}}\right] + \xi_{ro}\end{aligned} \tag{4.73}$$

Again, similar to the first-stage problem, there may arise situations when we have to enforce $x_f = 0$ and correspondingly $\gamma_1 \neq 0$ in (4.56). Hence, $\bar{x_r}^* = x_r^*$ and $\bar{x_o}^* = x_o^*$. Intuitively, in this situation, the second-stage problem reduces to the first-stage problem and this situation arises under the following condition:

$$Q - \sqrt{\frac{3\alpha(Q-C)}{\varphi(\xi_f - \xi_{ro})}} \leq D_{QoS} \tag{4.74}$$

However, in the situations when $x_f = 1$ and $x_r = 0$, $x_o = 0$ are needed to be enforced, then $\gamma_2 \neq 0$ in (4.57). This is similar to the field cloudlet placement model discussed in Chapter 3. In this situation, the following condition is satisfied:

$$C - \sqrt{\frac{3\alpha(Q-C)}{\varphi(\xi_f - \xi_{ro})}} \geq D_{QoS} \tag{4.75}$$

Clearly, we can formulate a reduced optimisation problem similar to the case of $x_r = 0$, $x_o = 1$ in the first-stage problem and we find the optimal value of $\mu$ (denoted

by $\tilde{\mu}^*$) and optimal cost as follows:

$$\tilde{\mu}^* = pN + \frac{1}{D_{QoS} - C} \tag{4.76}$$

$$\mathbb{C}_5 = \frac{\alpha}{\varphi}\left[pN + \frac{1}{D_{QoS} - C}\right] + \xi_f \tag{4.77}$$

Note that, if we observe $x_r^* = 0$ and $x_o^* = 1$ after solving the first-stage problem, then solving the second-stage problem becomes unnecessary, because it seems less likely that the cost minimisation framework decides to install more expensive field cloudlets without installing the RN cloudlets. In general, while solving both the first-stage and second-stage problems, we can figure out the most appropriate formula for cloudlet deployment cost amongst $\mathbb{C}_1$, $\mathbb{C}_2$, $\mathbb{C}_3$, $\mathbb{C}_4$ and $\mathbb{C}_5$ and accordingly can decide whether to install cloudlets in the field, at RN or CO. Table 4.2 provides a summary of the cost formulas used against different network conditions:

## 4.4 Framework Validation

In this section, we validate the lower bounds on static cloudlet deployment cost derived in Section 4.3. To achieve this goal, we proceed in two steps. Firstly, we stochastically generate multiple instances of network data over a circular deployment area, i.e., ONU locations, RN locations, and CO location. We evaluate the MINLP based framework designed in Chapter 3 on these data and obtain an empirical average of the results. Secondly, we derive a theoretical model for the circular

Table 4.2: Summary of Cost Formula Usage

| ***Formula*** | ***Cloudlet Locations*** | ***Network Conditions*** |
|---|---|---|
| $\mathbb{C}_1$ | RN, CO | $Q - \sqrt{\frac{3\alpha(Q-C)}{\varphi(\xi_f - \xi_{ro})}} \leq D_{QoS}$ |
| $\tilde{\mathbb{C}}_1$ | CO, Field ($\mathbb{C}_1 \geq \tilde{\mathbb{C}}_1$) | $C - \sqrt{\frac{3\alpha(Q-C)}{\varphi(\xi_f - \xi_{ro})}} \leq D_{QoS}$ |
| $\mathbb{C}_2$ | CO | $B - \sqrt{\frac{2\alpha(B-A)}{\varphi(\xi_r - \xi_o)}} \leq D_{QoS}$ |
| $\mathbb{C}_3$ | RN | $B - \sqrt{\frac{2\alpha(B-A)}{\varphi(\xi_r - \xi_o)}} \geq D_{QoS}$ |
| $\mathbb{C}_4$ | RN, CO, Field | $A - \sqrt{\frac{2\alpha(B-A)}{\varphi(\xi_r - \xi_o)}} \geq D_{QoS}$ |
| $\mathbb{C}_5$ | Field | $C - \sqrt{\frac{3\alpha(Q-C)}{\varphi(\xi_f - \xi_{ro})}} \geq D_{QoS}$ |

network deployment area such that the circular area can be reduced to the smallest circular wedge under the network homogeneity assumption. We apply the suitable expressions derived in Section 4.3 on this circular wedge and scale up the results to obtain similar results as obtained from the first step.

### 4.4.1 Random Dataset Generation and Evaluation with Hybrid MINLP Framework

We use the Poisson-point process [150] to generate random ONU locations over a circular area of diameter 4 km, as shown in Fig. 4.1(a). We consider that each ONU has associated wireless access points to serve around 1000 users and hence we divide population density by 1000 to calculate the average number of ONUs/km$^2$, e.g., in a typical Australian urban area, the standard population density is 4000 persons/km$^2$ [149], thus on an average of 4 ONUs/km$^2$. To identify potential field cloudlet placement locations, we use $k$-means clustering algorithm [150] with $k = 12$, such that the feasibility of the MINLP framework designed in Chapter 3 is maintained. We consider that the ONUs are supported by 10G-PONs and hence enjoy a total of 10 Gbps datarate in both uplink and downlink directions. The split-ratio is $1 : N$, $N \in \{4, 8, 16, \dots\}$, but for this framework validation purpose, we consider $N \in \{4, 8, 16\}$. Any field cloudlet can be connected to ONUs belonging to different PON branches, but the RN and CO cloudlets can be connected to the ONUs that belong to that particular PON branch. We consider that the CO for all the PON branches is located at the centre of the circular area.

We implement the MINLP model designed in Chapter 3 with the A Modeling Language for Mathematical Programming (AMPL) platform and the open-source solver COUENNE package for MINLPs [145] to evaluate the model on the chosen dataset. We generate random instances of network deployment data with split-ratio $1 : N$, $N \in \{4, 8, 16\}$, i.e., ONU locations, RN locations, and potential field cloudlet locations, over the circular area with diameter 4 km and find the average cost of cloudlet deployment over this area against $D_{QoS}$ latency values 1 ms, 10 ms, and 100 ms. Fig. 4.1(b) shows the optimal cloudlet placement locations in the field, RN

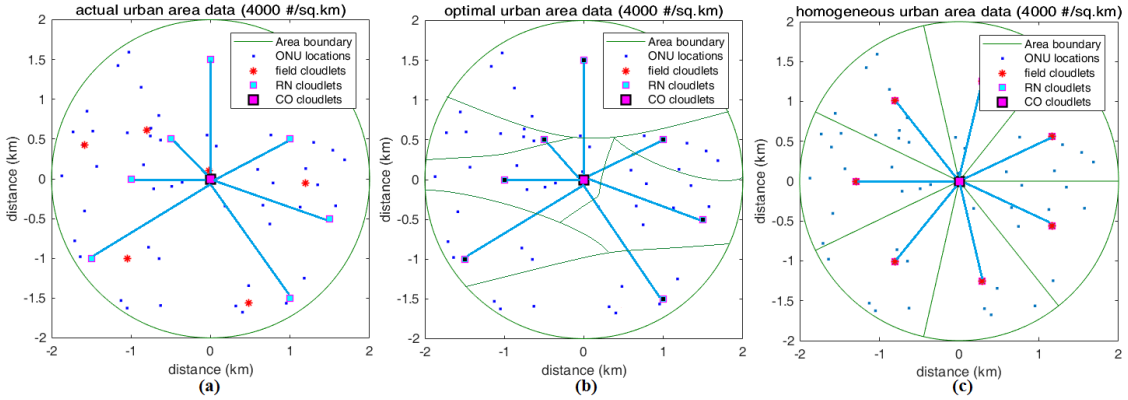



Figure 4.1: (a) An instance of the stochastically generated ONU locations (blue dots), RN locations (cyan-faced squares), CO locations (pink-faced squares) and potential field cloudlet locations (red asterisks) with split-ratio 1:8 of an urban population area with density 4000 people/km$^2$; (b) optimally chosen field, RN and CO cloudlet placements and their respective coverage areas (green borders) using MINLP framework; (c) equivalent field, RN and CO cloudlet placement locations under homogeneous network assumption (Only the feeder fibres are shown in all figures).

and CO along with the corresponding coverage areas for split-ratio 1:8 and $D_{QoS}$ = 1 ms.

### 4.4.2 Reduction of Circular Area to Circular Wedge for Deriving Lower Bounds

For the average analysis, we assume that the network deployment is homogeneous. Therefore, with $\Gamma$ ONUs/km$^2$ there are a total $\lceil \Gamma \pi R^2 \rceil$ number of ONUs in a circular area with radius $R$ and hence, $\Omega = \lceil \Gamma \pi R^2 / N \rceil$ TDM-PONs of split-ratio $N$ can serve all the ONUs. Thus, we can consider the entire circular area as a combination of $\Omega$ identical circular wedges inscribing a central angle, say, $2\theta = 2\pi/\Omega$. Fig. 4.1(c) shows the example case where for $\Gamma = 4$ ONUs/km$^2$, $R = 2$ km, and $N = 8$, there are $\Omega = 7$ identical circular wedges with $2\theta = 2\pi/7$.

Due to our initial assumption that the ONUs are homogeneously distributed over the entire area of the circular wedge, we can further consider that the RN is located at the centroid. Note that, ideally the field cloudlet location should also be the same as RN, i.e., the centroid of the circular wedge. The coordinates of the centroid of a circular wedge with radius $R$ and spanned from $-\theta$ to $\theta$ are $(\bar{x}, \bar{y}) = \left(\frac{2R}{3\theta} \sin\theta, 0\right)$

[159]. Thus the distance between CO and RN is, $L_{cr} = L_f = (\frac{2R}{3\theta}\sin\theta)$ km. We consider $(x, y)$ to be the coordinates of any random point within the circular wedge. Therefore, we compute the average distance of $(x, y)$ from the centroid $(\bar{x}, \bar{y})$ as follows:

$$\begin{aligned} L_{ro} &= \frac{1}{\theta R^2}\int_{-\theta}^{\theta}\int_{0}^{R} |x-\bar{x}|\, r\, dr\, d\theta \\ &= \frac{1}{\theta R^2}\left[\int_{-\theta}^{\theta}\int_{0}^{\bar{x}} (\bar{x}-x)\, r\, dr\, d\theta + \int_{-\theta}^{\theta}\int_{\bar{x}}^{R} (x-\bar{x})\, r\, dr\, d\theta\right] \end{aligned} \tag{4.78}$$

Evaluating the double integrals in (4.78) by putting $x = r\cos\theta$ and $\bar{x} = 2R\sin\theta/3\theta$, we obtain the final expression as follows:

$$L_{ro} = \frac{16R}{27\theta^3}\sin^3\theta\left(1 - \frac{2}{3\theta}\sin\theta\right) \tag{4.79}$$

According to the tree-and-branch topology of TDM-PON architecture [20], the total length between an ONU and CO is the sum of the distances from CO to RN and RN to ONU. Therefore, in the present context, the average distance between an ONU and CO ($L_{co}$) is the sum of the distance from the origin to the centroid ($L_{cr}$) and the average distance of any random point from the centroid ($L_{ro}$) within the circular wedge, i.e., $L_{co} = L_{cr} + L_{ro}$.

With this geometrical setup, we can use the lower bound of cloudlet deployment cost expressions derived in Section 4.3 to calculate the empirical average cloudlet deployment cost with just one TDM-PON based FiWi branch over the smallest circular wedge area, we then scale up the results by multiplying with factor $\Omega$ to calculate the total cloudlet deployment cost over the entire circular area.

### 4.4.3 Comparison of Cloudlet Deployment Costs and Lower Bounds

While evaluating either of the expressions for cost values $\mathbb{C}_1$, $\mathbb{C}_2$, $\mathbb{C}_3$, $\mathbb{C}_4$ and $\mathbb{C}_5$, as required, we need to validate that our two-stage optimisation problem formulation

produces an equivalent result to that of the primary optimisation problem formulation with objective function (4.1) and constraints (4.42)-(4.44). Hence, we evaluate both these problem formulations over a single TDM-PON based FiWi branch and scale up the results for the entire dataset. In addition to this, we need to show that the scaled up results are close enough to the results obtained by using the MINLP framework proposed in Chapter 3 on the complete dataset and hence, we compare these cost values with the empirical average of cost values obtained by evaluating the MINLP framework on multiple instances of randomly generated complete datasets. The values of all the network parameters are considered such that consistency is maintained with the MINLP framework designed in Chapter 3, e.g., the normalised cost of installing a processor is $\alpha = 1$ [151], the normalised costs of installing a cloudlet in the field is $\xi_f = 8$, at RN is $\xi_r = 6$ and at CO is $\xi_o = 5$ [151], the normalised cost of installing new point-to-point fibre per kilometer is $\eta = 20$ [139], data rate of each new point-to-point fibre links is $BW_f = 1$ Gbps, $n_\lambda = 1$ wavelength(s) of datarate $BW_r = 10$ Gbps is shared among all $N$ ONUs to communicate with RN cloudlet, the datarate of default communication link between ONU and CO is $BW_o = 10$ Gbps, the approximate downlink and uplink background loads are $\beta_{dl} = 7$ Gbps and $\beta_{ul} = 5$ Gbps, respectively [139], each ONU serves $p = 1000$ users, each cloudlet processor supports $\varphi = 2500$ jobs [63], average number of bits an ONU sends to a cloudlet as job requests in uplink $\sigma_{ul} = 1$ MB and average number of bits an ONU receives from a cloudlet as response of the job requests in downlink $\sigma_{dl} = 50$ KB [53]. We consider $\delta_f = 0$ as point-to-point fibre connections are used between ONUs and field cloudlets, and $\delta_r = \delta_o = 0.5$ ms [152].

Fig. 4.2 presents a comparison of the normalised cloudlet deployment cost/100 users over a circular area of diameter 4 km dense urban deployment scenario (population density = 4000 persons/km$^2$). We consider split-ratio $1 : N$, $N \in \{4, 8, 16\}$ and $D_{QoS}$ values of 1 ms, 10 ms, and 100 ms. Note that, we indicate the optimal solutions of the MINLP framework proposed in Chapter 3, the optimal solution of the primary optimisation problem formulation with objective function (4.1) and constraints (4.42)-(4.44), and the optimal solution of the two-stage problem formulation by the labels "MINLP-complete", "Primary problem", and "Two-stage",

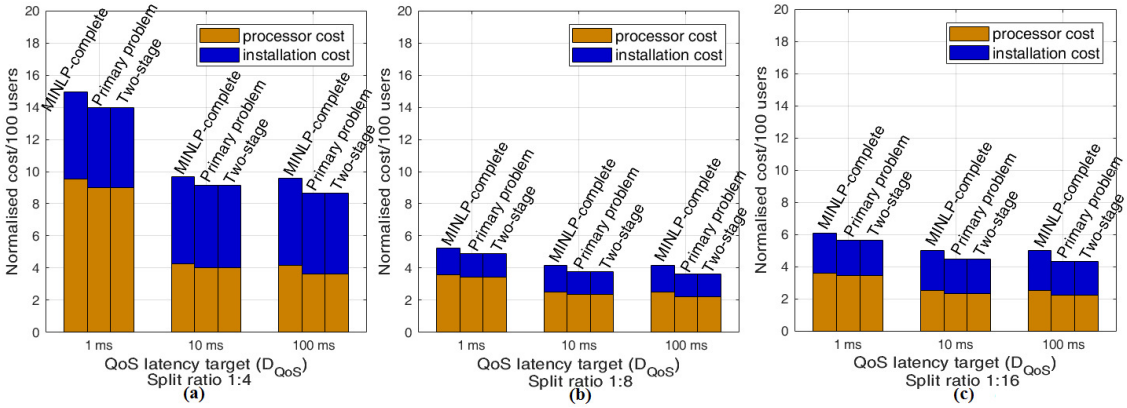



Figure 4.2: Comparison of normalised cloudlet deployment cost/100 users over dense urban deployment scenario obtained from the complete MINLP framework, primary optimisation problem with a single TDM-PON FiWi branch, and the corresponding two-stage problem formulation against $D_{QoS}$ values of 1 ms, 10 ms, and 100 ms with (a) split-ratio 1:4, (b) split-ratio 1:8, and (c) split-ratio 1:16.

respectively. The optimal cost values obtained by scaling up the results from the primary optimisation problem and appropriate cost formula derived from the two-stage optimisation problem formulation are equal. This justifies all our assumptions and validates the two-stage optimisation problem formulation. Moreover, both of these cost values are in agreement with the empirical average cost value obtained by evaluating the MINLP framework on the entire network in all the test scenarios. Thus, our derived cloudlet deployment cost expressions provide a tight lower bound to the cost value obtained using the MINLP framework. Hence, we can use the expressions derived in Section 4.3 to draw various insightful conclusions on cloudlet deployment strategies using TDM-PON based FiWi access network.

## 4.5 Results and Discussions

In this section, we show the variation of cloudlet deployment cost over 10G-PON based FiWi network depending on various network parameters, i.e., a parametric analysis. Fig. 4.3 shows the workload distribution among the field, RN and CO cloudlets against TDM-PON split-ratio for a circular area with diameter = 4 km, population density = 4000 people/km$^2$, and $D_{QoS}$ = 1 ms. With smaller split-ratio values e.g., 1:4 and 1:8, CO cloudlet alone is capable to handle the total workload,

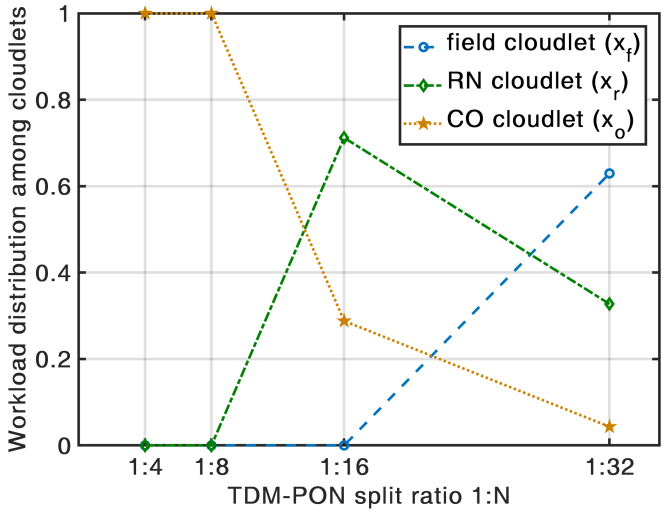



Figure 4.3: Workload distribution among field, RN and CO cloudlets against TDM-PON split ratio for a circular area with diameter = 4 km, population density = 4000 people/km$^2$, and $D_{QoS} = 1$ ms.

hence $x_o = 1$, $x_r = 0$ and $x_f = 0$. This follows from the network condition (4.26). However, for higher split-ratio values, the installation of RN and field cloudlets becomes essential. With split-ratio 1:16, the total workload is shared between RN and CO cloudlets following from the network condition (4.74), and with split-ratio 1:32, the total workload is shared amongst the field, RN and CO cloudlets following from the network condition (4.40). With a TDM-PON split-ratio $1 : N$ where $N \in \{4, 8, 16\}$, the first-stage optimisation problem is sufficient, but for a split-ratio of 1:32 or higher, we need to solve the second-stage problem as well. In this case, the curve of $x_o$ is monotonically decreasing, $x_f$ is monotonically increasing, and $x_r$ is increasing first and then decreases again. Note that, it is easy to show from the analytical expressions that this trend of variation of values of $x_f$, $x_r$, and $x_o$ against split-ratio is generic with given values of all other network parameters. Initially $x_o = 1$, $x_r = 0$ and $x_f = 0$; as $N$ increases $x_o$ decreases monotonically and $x_r$ increases monotonically with $x_f = 0$. For a further higher value of $N$, $x_f$ starts to monotonically increase, $x_r$ starts to monotonically decrease, and $x_o$ decreases monotonically as earlier. As we change the values of different network parameters, these critical points of $N$ also change, but the trend of variation of $x_f$, $x_r$, and $x_o$ remains the same. It is interesting to note that in all the cases $x_f + x_r + x_o = 1$ should hold as per our optimisation problem design constraint.

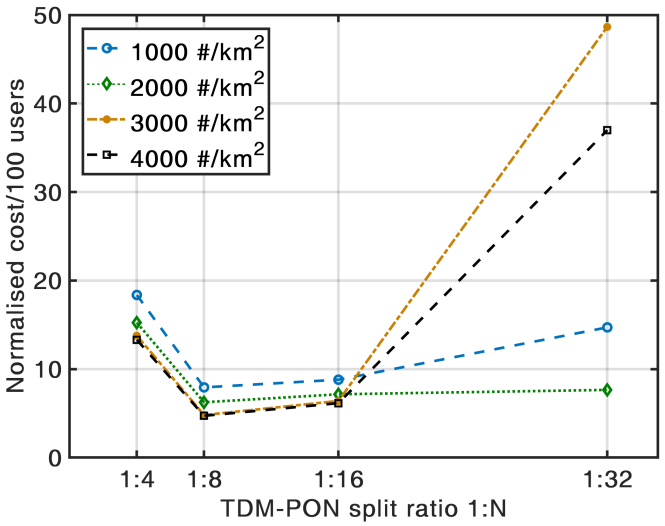



Figure 4.4: Variation of normalised cloudlet deployment cost/100 users against TDM-PON split-ratio $1 : N$ for a circular area with diameter = 4 km and $D_{QoS} = 1$ ms. Population density is varied from 1000 to 4000 people/km$^2$.

Next, Fig. 4.4 shows the variation of normalised cloudlet deployment cost/100 users against TDM-PON split-ratio $1 : N$ where $N \in \{4, 8, 16, 32\}$ with $D_{QoS} = 1$ ms. We plot multiple such graphs while varying the population density from 1000 to 4000 people/km$^2$. From these plots, we primarily observe that the normalised cost decreases as we increase the TDM-PON split-ratio from 1:4 to 1:8, because a lower number of TDM-PON branches are present in the network and only CO cloudlets are capable of handling the total workload. However, the normalised cost slightly increases for the split-ratio of 1:16 because RN cloudlets are required to be installed in this scenario. With split-ratio 1:32 we observe a sudden jump in the normalised cost. This is due to the installation of field cloudlets which incurs an additional cost of new point-to-point fibre installation. Moreover, for $N \in \{4, 8, 16\}$, normalised cost is higher for lower population density, e.g., with split-ratio 1:4, normalised cost for 1000 people/km$^2$ is higher than that of 4000 people/km$^2$. However, the situation is reversed for higher split-ratio like 1:32. In general, from these results, we can deduce that for a stringent QoS requirement of $D_{QoS} = 1$ ms, cloudlet installation becomes highly expensive with a TDM-PON split-ratio 1:32 or higher.

In Fig. 4.5, we show the variation of normalised cloudlet deployment cost/100 users against population density of the considered area while varying the TDM-PON

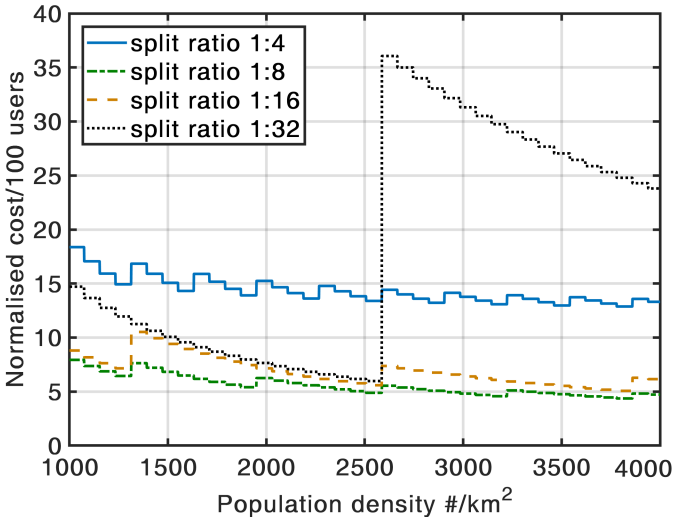



Figure 4.5: Variation of normalised cloudlet deployment cost/100 users against population density of the cloudlet deployment circular area with diameter = 4 km and $D_{QoS} = 1$ ms. The value of $N$ in TDM-PON split-ratio $1 : N$ is varied from 4 to 32.

split-ratio $1 : N$ where $N \in \{4, 8, 16, 32\}$ with $D_{QoS} = 1$ ms. As population density increases, the overall cloudlet installation cost also increases, but the normalised cost/100 users slowly decrease for a particular split-ratio. For split-ratio $1 : N$ where $N \in \{4, 8, 16\}$, the normalised cost is minimum with split-ratio 1:8 and maximum with split-ratio 1:4. However, for the split-ratio 1:32, the normalised cost becomes abruptly higher due to the installation of expensive field cloudlets.

Fig. 4.6 shows the variation of normalised cloudlet deployment cost/100 users with population density 4000 people/km$^2$ against TDM-PON split-ratio $1 : N$ where $N \in \{4, 8, 16, 32\}$ while varying $D_{QoS}$. The plots show similar patterns as in Fig. 4.4, i.e., the normalised cost decreases from split-ratio 1:4 to 1:8, then it slightly increases for the split-ratio 1:16, and then abruptly increases for the split-ratio 1:32. However, the normalised costs are slightly higher for a stringent QoS requirement of $D_{QoS} = 1$ ms and gradually become lower as the QoS requirement is made lenient to $D_{QoS} = 100$ ms. This primarily happens due to the reason that we require lower computational resources, i.e., processors per cloudlet for a lenient $D_{QoS} = 100$ ms than that of a stringent $D_{QoS} = 1$ ms.

In Fig. 4.7, we show the variation of normalised cloudlet deployment cost/100 users with population density 4000 people/km$^2$ and $D_{QoS} = 1$ ms against TDM-PON

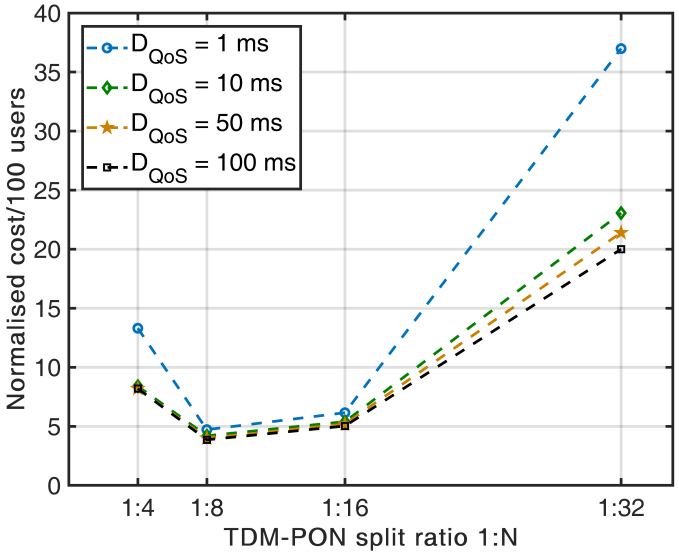



Figure 4.6: Variation of normalised cloudlet deployment cost/100 users against TDM-PON split-ratio $1 : N$ for a circular area with diameter = 4 km and population density = 4000 people/km$^2$. The QoS latency requirement $D_{QoS}$ is varied from 1 to 100 ms.

split-ratio $1 : N$ where $N \in \{4, 8, 16, 32\}$ while varying the data rate of TDM-PON. We considered 10G-PON in all the previous results which have 10 Gbps data rate in both uplink and downlink and observed that the cost of cloudlet deployment first decreases with split-ratio 1:8 from 1:4, then gradually increases with split-ratio 1:16 and 1:32. However, as we use higher data rate PONs over the same user database,

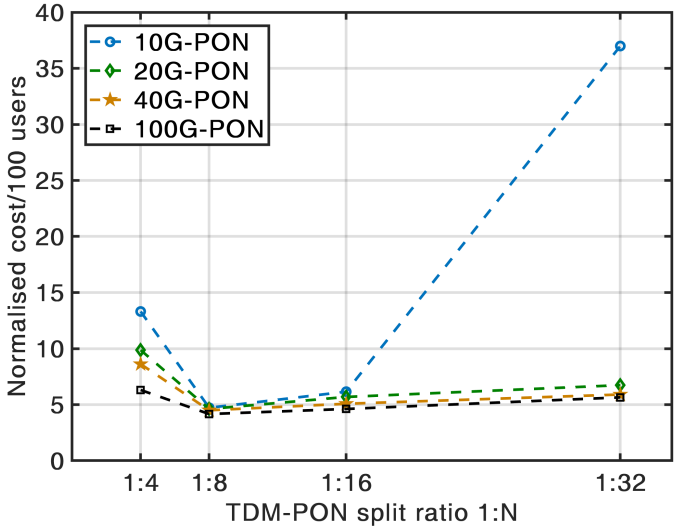



Figure 4.7: Variation of normalised cloudlet deployment cost/100 users against TDM-PON split-ratio $1 : N$ for a circular area with diameter = 4 km and population density = 4000 people/km$^2$. The uplink and downlink datarate of TDM-PON is varied from 10 Gbps to 100 Gbps.

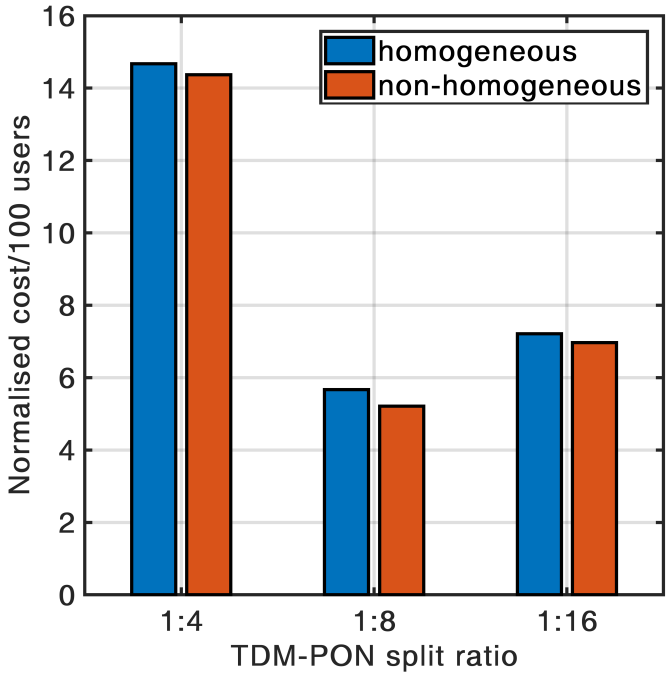



Figure 4.8: Comparison of normalised cloudlet deployment cost/100 users with homogeneous and non-homogeneous network over a circular area with diameter = 4 km, population density = 4000 people/km$^2$, and $D_{QoS} = 1$ ms.

e.g., 20G-PON, 40G-PON, and 100G-PON that have 20 Gbps, 40 Gbps, and 100 Gbps data rates, respectively, the cloudlet deployment cost decreases. The cloudlet deployment cost first decreases as usual with split-ratio 1:8 from 1:4, but it does not increase much with split-ratio 1:16 and 1:32. This happens because with higher data rate PONs, each ONU gets more bandwidth over the default uplink and downlink channels and hence, CO cloudlets get to execute a major share of the total workload.

Finally, in Fig. 4.8, we show a comparison of normalised cloudlet deployment cost/100 users against $D_{QoS} = 1$ ms with homogeneous and non-homogeneous network deployment. We choose the same circular area with a diameter of 4 km and population density 4000 people/km$^2$. To simulate non-homogeneity, we choose a bunch of TDM-PONs and randomly vary their split-ratios such that their mean is equal to the corresponding homogeneous split-ratio $1 : N$ where $N \in \{4, 8, 16\}$. For example, for a homogeneous split-ratio of 1:8, we uniformly choose split-ratios of the TDM-PONs from the interval $[6, 10]$ such that their mean is 8, and so on. We observe from our simulated results that the performance of the non-homogeneous framework is not worse than 5% of that with the homogeneous framework.

## 4.6 Conclusions

In this chapter, we have proposed a novel method to find lower bound of integer programming based cost optimisation frameworks for TDM-PON based hybrid cloudlet placement architecture, where cloudlets are allowed to be optimally placed in the field, at RN or CO locations. This technique very easily resolves the scalability issues in our previously designed MINLP based frameworks. To solve the cloudlet placement problem, we have used KKT conditions and numerical techniques for system of nonlinear equations, and no other heuristic algorithms. We have used MINLP frameworks on multiple instances of similar datasets e.g., a circular area of diameter 4 km with population density 4000 people/km$^2$ and have compared their empirical average with the corresponding lower bound derived. We have observed that the lower bounds obtained by using our proposed method lie within 10% of the solutions obtained by using the MINLP framework. We have also performed a parametric analysis and observed the dependency of cloudlet deployment cost on network parameters e.g., user density, network architecture, TDM-PON split-ratio and QoS requirements. Cloudlet deployment cost decreases for increasing split-ratio from 1:4 to 1:8 as only CO cloudlets are sufficient to handle the total workload, but starts to increase again for higher split-ratios like 1:16 and 1:32 as relatively expensive RN and field cloudlets are required to be installed. Moreover, the cloudlet deployment cost is higher for a stringent QoS requirement of 1 ms, but becomes relatively lower for lenient QoS requirements of 10 ms or 100 ms, because more processors are required for $D_{QoS} = 1$ ms than that of $D_{QoS} = 10$ ms or 100 ms. Therefore, in the network planning stage, this method to find a lower bound of integer programming based cost optimisation frameworks appears to be a very much user-friendly tool to the cloudlet service providers for developing fundamental insights on the cloudlet deployment and making an approximate estimate for the total cost.

# Chapter 5

# Centralised and Decentralised Non-Cooperative Load-Balancing Games among Neighbouring Cloudlets

> "When you have balance in your life, work becomes an entirely different experience. There is a passion that moves you to a whole new level of fulfillment and gratitude, and that's when you can do your best... for yourself and for others."
>
> Cara Delevingne

## 5.1 Introduction

Edge computing servers like cloudlets are essentially a computer or cluster of computers installed in the proximity of mobile device users and distributed across access networks [8]. Thus, the authors of [63, 66, 140, 143, 160–162] proposed efficient cloudlet placement frameworks over wireless and fibre-wireless access networks. As cloudlet computing systems are essentially distributed computing systems, the authors of [74–77, 163–166] addresses the optimal job request allocation

problem from mobile devices to cloudlets while meeting computation and communication constraints. Although job allocation frameworks allocate job requests to the most favourable cloudlets, due to the dynamic nature of job request arrival process, cloudlets at different parts of a large network become overloaded and under-loaded at different times. Thus, the authors of [78, 79, 82–84] designed efficient load balancing frameworks among neighbouring cloudlets.

In this chapter, we mainly focus on the load balancing problem among neighbouring cloudlets. We critically observe that most of the existing works on load balancing among cloudlets stress on the minimisation of end-to-end latency of the cloudlets. However, if the job requests are processed within their requested quality-of-service (QoS) latency target, the users are satisfied in practice. Therefore, minimising latency beyond the QoS latency target is not always a desired objective for the cloudlets. Nonetheless, failing to meet the QoS latency target should incur a significant penalty on the cloudlets, especially for low-latency applications. This realisation motivated us to design a novel game-theoretic objective function that yields the maximum utility for cloudlets when the end-to-end latency is equal to the QoS latency target. In turn, this objective function makes each cloudlet interested in receiving some extra job requests from their neighbouring cloudlets and gain some economic benefit, whenever the respective cloudlet is meeting the desired QoS latency target.

For load balancing among neighbouring cloudlets from the same service provider, network optimisation-based frameworks proposed in [78, 79, 82–84] perform very efficiently. Nonetheless, in a real heterogeneous deployment scenario, usually multiple cloud service providers install cloudlets over the same customer base and a game-theoretic framework is required as different service providers are non-cooperative in general. Thus, to capture this multi-party economic interaction among heterogeneous neighbouring cloudlets, our problem formulation acts as an optimisation problem among cloudlets from the same service provider and acts like a non-cooperative game among cloudlets from different service providers. Moreover, as the existing load balancing frameworks make the load balancing decisions after the actual job request arrival to cloudlets, the overhead time of the load balancing algorithms makes these

frameworks highly unfit, especially for low-latency applications. To deal with such scenarios, we make the cloudlets predict the job request arrival rates and make the load balancing decisions beforehand so that immediate processing after actual job request arrival is possible.

To compute the Nash equilibrium (NE) load balancing strategies of the cloudlets, firstly we propose a centralised framework where all the competing cloudlets send their predicted job request arrival rates to a neutral mediator. In practice, a common network infrastructure provider can act as a mediator among the competing cloudlets who can be considered as the *tenants* of that network. The mediator computes the NE load balancing strategies for the cloudlets and broadcasts to them before the actual job request arrival. It is important to note that competing cloudlets are not always necessarily truthful while revealing private information e.g., total incoming job requests, and may adopt strategies to gain some additional economic benefit from the market. Thus, we propose a scheme where the neutral mediator is present in the system to impose efficient incentive mechanisms such that the revelation of truthful information is ensured [167].

Secondly, we propose a distributed framework to compute NE load balancing strategies among competing cloudlets, which is independent of the truthfulness of cloudlets. Although, distributed frameworks are more robust than centralised frameworks, all the cloudlets need to exchange extensive control information among themselves [58]. If we design an iterative algorithm, then all cloudlets need to exchange several control messages at every iteration within every timeslot with its neighbours until convergence to Nash equilibrium is achieved. This may not pose any network problem for a regular load balancing framework, but a network bottleneck may appear when the timeslots are very small (a few milliseconds). This issue can be resolved by using various artificial intelligence-based schemes to learn network conditions and make load balancing decisions. However, due to the randomness of job request arrival process, designing complex supervised learning models like artificial neural networks based on historical data to make load balancing decisions will prove to be highly inefficient, especially in a dynamic network scenario where there is a low correlation between the trained data and real-time data. Therefore, we propose a

reinforcement learning automata-based algorithm such that quick convergence is ensured. This empowers the cloudlets to make load balancing decisions independently, without exchanging any control information among themselves [168].

The rest of this chapter is organised as follows. Section 5.2 provides a brief background review on non-cooperative game theory and reinforcement learning automata theory. In Section 5.3, the details of the system model are presented. In Section 5.4, a non-cooperative game-theoretic problem among competing cloudlets for computation offloading is formulated. In Section 5.5, a dominant strategy incentive compatible mechanism is designed. In Section 5.6, a distributed continuous-action reinforcement learning automata-based algorithm is proposed. In Section 5.7, the proposed load balancing framework is evaluated and compared with a few other existing frameworks. Finally, in Section 5.8, our primary observations and achievements by using the game-theoretic framework are summarised.

## 5.2 Background on Non-cooperative Game Theory and Reinforcement Learning Automata

### Non-cooperative Game Theory

The notion of strategic (or normal) form proves to be one of the most common manifestations when describing a static or dynamic non-cooperative game [169]. In general, strategic (or normal) non-cooperative "pure strategy game" game has three components: the set of players, their strategies, and the payoffs or utilities. Formally, we define a strategic game as follows:

**Definition 5.1.** *A non-cooperative game in strategic (or normal) form can be represented by a touple* $\mathcal{G} = \langle \mathcal{N}, (A_i)_{i\in\mathcal{N}}, (U_i(a_i, \boldsymbol{a}_{-i}))_{i\in\mathcal{N}} \rangle$*, where:*

- $\mathcal{N} = \{1, 2, \ldots, N\}$ *represents a finite set of players;*
- $A_i$ *represents the set of available actions of player* $i \in \mathcal{N}$*;*
- $U_i : \boldsymbol{A} \rightarrow \mathbb{R}$ *represents the utility (payoff) function of player* $i$*, with* $\boldsymbol{A} = A_1 \times A_2 \times \cdots \times A_N$ *(complete set of joint actions of all the players).*

Note that utility of each player $i$ depends on the action $a_i \in A_i$ of the agent $i$ as well as on the joint action $\boldsymbol{a}_{-i} = (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_N)$ of all other agents, i.e., $U_i(\boldsymbol{a}) = U_i(a_i, \boldsymbol{a}_{-i}) = U_i(a_1, \ldots, a_i, \ldots, a_N)$. Nonetheless, if players choose actions according to some independent probability distributions or "mix" their strategies, then such a game is called a "mixed strategy game". We use the notation $\tilde{\mathcal{G}} = \langle \mathcal{N}, (\Sigma_i(A_i))_{i\in\mathcal{N}}, (\tilde{U}_i)_{i\in\mathcal{N}} \rangle$ as a mixed strategy extension of the pure strategy game $\mathcal{G} = \langle \mathcal{N}, (A_i)_{i\in\mathcal{N}}, (U_i)_{i\in\mathcal{N}} \rangle$, where $\Sigma_i(A_i)$ is the set of probability distributions defined on the set $A_i$ and $\tilde{U}_i$ is the mathematical expectation of $U_i$ with the joint distribution $\boldsymbol{\sigma}$ from the set $\Sigma(\boldsymbol{A}) = \Sigma_1(A_1) \times \Sigma_2(A_2) \times \cdots \times \Sigma_N(A_N)$. Therefore, for a discrete action pure strategy game $\mathcal{G}$, the utility function of the $i^{\text{th}}$ player in the corresponding mixed strategy game $\tilde{\mathcal{G}}$ is,

$$\tilde{U}_i(a_i, \boldsymbol{a}_{-i})) = \tilde{U}_i(\sigma_1, \sigma_2, \ldots, \sigma_N) = \sum_{\boldsymbol{a}\in A} U_i(\boldsymbol{a})\sigma_1(a_1)\ldots\sigma_N(a_N),$$

where, $\sigma_j \in \Sigma_j(A_j)$ is a mixed strategy of the $j^{\text{th}}$ player. Similarly, if we consider a continuous action game in pure strategies, then the utility functions $\tilde{U}_i$ are expressed by the Lebesgue integrals as follows,

$$\tilde{U}_i(\boldsymbol{\sigma}) = \int_A U_i(\boldsymbol{x}) d\sigma(\boldsymbol{x}),$$

where $\sigma(\boldsymbol{x})$ is the players joint distribution of the mixed strategies [169]. If players in a pure or mixed strategy game are selfish and make independent choices, the game will have a stable outcome where every player chooses their best response against the strategies of all other players.

**Definition 5.2. (Best response)** *In a pure strategy game $\mathcal{G} = \langle \mathcal{N}, (A_i)_{i\in\mathcal{N}}, (U_i)_{i\in\mathcal{N}} \rangle$ or a mixed strategy game $\mathcal{G} = \langle \mathcal{N}, (A_i)_{i\in\mathcal{N}}, (U_i)_{i\in\mathcal{N}} \rangle$, an action $a_i^* \in A_i$ or mixed strategy $\sigma_i^* \in \Sigma_i(A_i)$ is a best response for the player $i \in \mathcal{N}$ against the joint action of other players $\boldsymbol{a}_{-i}$ or joint mixed strategy of other players $\boldsymbol{\sigma}_{-i}$, if*

$$U_i(a_i^*, \boldsymbol{a}_{-i}) = \max_{a_i\in A_i} U_i(a_i, \boldsymbol{a}_{-i}), \quad \tilde{U}_i(\sigma_i^*, \boldsymbol{\sigma}_{-i}) = \max_{\sigma_i\in\Sigma_i} \tilde{U}(\sigma_i, \boldsymbol{\sigma}_{-i}).$$

**Definition 5.3. (Nash equilibrium)** *A joint action $\boldsymbol{a}^* \in \boldsymbol{A}$ or joint mixed strategy*

*$\boldsymbol{\sigma}^* \in \sigma(\boldsymbol{A})$ is a pure strategy or a mixed strategy Nash equilibrium, if the action $\boldsymbol{a}_i^*$ or the mixed strategy $\boldsymbol{\sigma}_i^*$ is a best response to the joint action of other players $\boldsymbol{a}_{-i}^*$ or joint mixed strategy of other players $\boldsymbol{\sigma}_{-i}^*$ for each player $i \in \mathcal{N}$, i.e.,*

$$U_i(a_i^*, \boldsymbol{a}_{-i}^*) \geq U_i(a_i, \boldsymbol{a}_{-i}^*), \forall a_i \in A_i, \quad \tilde{U}_i(\sigma_i^*, \boldsymbol{\sigma}_{-i}^*) \geq \tilde{U}_i(\sigma_i, \boldsymbol{\sigma}_{-i}^*), \forall \sigma_i \in \Sigma_i(A_i).$$

From the concept of best-response function, we can consider a Nash equilibrium as the intersection of the best response functions of all the players and derive an alternative definition of a pure-strategy Nash equilibrium as follows [170]:

**Definition 5.4.** *A pure strategy profile $\boldsymbol{a}^* \in \boldsymbol{A}$ or a mixed strategy profile $\boldsymbol{\sigma}^* \in \Sigma(\boldsymbol{A})$ is a Nash equilibrium of a non-cooperative game if and only if every players strategy is a best response ($b_i(a_{-i})$ or $b_i(\sigma_{-i})$) against the strategies of other players, i.e., $a_i^* \in b_i(\boldsymbol{a}_{-i}^*)$ or $\sigma_i^* \in b_i(\boldsymbol{\sigma}_{-i}^*)$, for each player $i$.*

In practice, it is a very challenging task to prove the existence of pure-strategy Nash equilibrium and find this equilibrium in a generic non-cooperative game. For this purpose, there exist some theorems to investigate whether a Nash equilibrium in pure strategies exists in a given game, especially for continuous-kernel games. Some of the most commonly used theorems to prove the existence of Nash equilibrium are as follows [170]:

**Theorem 5.5.** ***(Debreu, Glicksberg, Fan theorem)*** *If in a non-cooperative game in strategic form $\mathcal{G} = \langle \mathcal{N}, (A_i)_{i\in\mathcal{N}}, (U_i)_{i\in\mathcal{N}} \rangle$, if $\forall i \in \mathcal{N}$, every strategy set $A_i$ is compact and convex, $U_i(a_i, \boldsymbol{a}_{-i})$ is a continuous function in the profile of strategies $\boldsymbol{a} \in \boldsymbol{A}$ and quasi-concave in $a_i$, then the game has at least one pure-strategy Nash equilibrium.*

**Theorem 5.6.** *In a strategic game $\mathcal{G} = \langle \mathcal{N}, (A_i)_{i\in\mathcal{N}}, (U_i)_{i\in\mathcal{N}} \rangle$, if each $A_i$ are non-empty compact metric spaces and each $U_i(a_i, \boldsymbol{a}_{-i}), \forall i \in \mathcal{N}$ are continuous functions, then the game has a mixed-strategy Nash equilibrium.*

## Reinforcement Learning Automaton

A learning automaton is defined as *an adaptive decision-making unit, which learns the optimal action against a random environment through repeated interactions with*

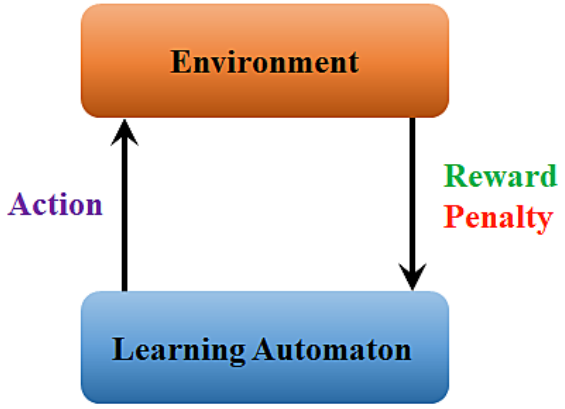



Figure 5.1: Working principle of a reinforcement learning automaton.

*the environment*. The interaction with the environment and the evolution of the action probabilities in all automated models proceed in discreet time steps. At each time instant, $n, n = 0, 1, 2, \ldots$, the automaton randomly chooses an action, $\alpha(n)$, based on its current probability distribution over the action space. This action is an input into the environment that responds to reactions or reinforcements, $\beta(n)$. Again, this "environmental reaction" or "reinforcement", i.e., a "reward" or a "penalty" is the input to the automaton. The automaton updates its distribution of the probability of action and the process of a random choice of action, eliciting reinforcement and updating of probabilities of action, repeats. In Fig. 5.1, a general block diagram of an automaton that works in a random setting is shown [171].

### Finite Action Learning Automaton

We can describe a finite action learning automaton by the quadruple $(\mathcal{A}, B, \mathcal{T}, \boldsymbol{p}^{(n)})$, where $\mathcal{A} = \{\alpha_1, \alpha_2, \ldots, \alpha_N\}$, is the finite set of actions, $B$ is the set of all possible inputs or reinforcements to the automaton, $\mathcal{T}$ is the learning algorithm for updating action probabilities and $\boldsymbol{p}^{(n)}$ is the action probability vector at instant $n$ given by $\boldsymbol{p}^{(n)} = [p_1^{(n)}, p_2^{(n)}, \ldots, p_N^{(n)}]^T$. We have $p_i^{(n)} \geq 0, \forall i, n$ and $\sum_{i=1}^{N} p_i^{(n)} = 1, \forall n, n = 0, 1, 2, \ldots$. Here $p_i^{(n)}$ is the probability with which action $\alpha_i$ is chosen at time instant $n$, i.e.,

$$p_i^{(n)} = \text{Prob}[\alpha^{(n)} = \alpha_i | \boldsymbol{p}^{(n)}], i = 1, 2, \ldots, N.$$

The objective of a learning automaton is to find the optimal action although

it does not know the random reinforcement distribution in response to various actions. The automaton gradually updates the action probability distribution based on the reinforcement received through a learning algorithm. A variety of learning algorithms can be used to achieve this objective. The "linear reward-inaction" algorithm is one of the earliest finite action learning automaton algorithms [171], where the action probabilities are updated as described below.

Let $\alpha^{(n)} = \alpha_i$. Then the action probability vector $\boldsymbol{p}^{(n)}$ is updated as:

$$p_i^{(n+1)} = p_i^{(n)} + \lambda\beta^{(n)}(1 - p_i^{(n)}),$$
$$p_j^{(n+1)} = p_j^{(n)} - \lambda\beta^{(n)}p_j^{(n)}, \forall j \neq i,$$

where $\lambda$ is the learning rate (or step-size) parameter that satisfies $0 < \lambda < 1$.

#### Continuous Action Learning Automaton

In this case, we consider the action space $\mathcal{A}$ to be continuous and define $\boldsymbol{f}$ as a non-parametric probability density function over $\mathcal{A}$. The learning automaton initially starts with the uniform distribution over the whole action space $\mathcal{A}$ and after exploring action $\alpha^{(n)} \in \mathcal{A}$ at time-step $n$, the probability distribution is updated as:

$$\boldsymbol{f}^{(n+1)}(\alpha) = \begin{cases} \gamma^{(n)} \left( \boldsymbol{f}^{(n)}(\alpha) + \beta^{(n)}(\alpha)\nu \exp\left\{ -\frac{1}{2}\left(\frac{\alpha-\alpha^{(n)}}{\omega}\right)^2 \right\} \right), & \forall \alpha \in \mathcal{A}, \\ 0, & \forall \alpha \notin \mathcal{A}, \end{cases}$$

where, $\nu$ and $\omega$ are referred to as learning and spreading rate parameters, respectively [172]. The first parameter $\nu$ indicates exploration of new observations, and the second parameter $\omega$ determines the spread of the information from a given observation to the neighbouring actions. Finally, $\gamma$ can be considered as a normalisation factor so that $\int_{-\infty}^{+\infty} \boldsymbol{f}^{(n+1)} dz = 1$.

## 5.3 System Model for Load Balancing Game

In this section, we discuss the considered system model and the primary assumptions made. We consider a general heterogeneous deployment scenario for cloudlets, where

the neighbouring cloudlets may belong to same as well as different service providers. We denote the total number of competing cloudlets in the network by $N$ and the set of cloudlets is denoted by $\mathcal{C} = \{1, 2, \ldots, N\}$ where $N \geq 2$. The set of competing cloudlets $\mathcal{C}$ in a network is "common knowledge" [173].

**(a) Job Request Service Process:** We assume that the number of processors in each of the cloudlets is finite. Furthermore, we assume that each unit processor present in the network has similar job processing capabilities. If we assume that the $i^{\text{th}}$ cloudlet has a single processor then the average service rate is $\mu_i$ (jobs/s) depending on the incoming job requests. Therefore, $\mu_i$ indicates a parametric description of the job requests arrived at the $i^{\text{th}}$ cloudlet. We further assume that incoming job requests from mobile devices are maximally parallelised by cloudlets. By using Google cluster-usage traces, the authors of [157] showed that job request arrival and their service times follow exponential distributions, and hence can be considered as Poisson processes. Therefore, we model the cloudlets as $M/M/1$ queuing systems while considering the aggregated processing rate of all processors [83]. If the $i^{\text{th}}$ cloudlet has $n_i$ number of processors with complete parallel processing enabled, then the required service rate for supporting the total CPU cycles of all the job requests received within a time-slot rate can be determined as $\mu_{ii} = n_i\mu_i$. Moreover, when $i^{\text{th}}$ cloudlet offloads some job requests to its neighbouring $j^{\text{th}}$ cloudlet, then the corresponding service rate for the respective job requests is defined as $\mu_{ij} = n_j\mu_i$.

**(b) Job Request Arrival Process:** We denote the average job request arrival rate from all the corresponding mobile devices to a cloudlet $i \in \mathcal{C}$ by $\lambda_i$. We presume that the network has enough bandwidth and by counting the number of incoming packets over each time-slot, the job request arrival rate $\lambda_i$ can be calculated. Each cloudlet prioritises the processing of the incoming job requests internally or offloads to a neighbouring cloudlet through some internal scheduling algorithm (beyond the scope of this paper). As the average job request arrival varies from time instance to time instance, we assume that each $\lambda_i$ is independently and uniformly distributed over the support $\Lambda_i = [0, \lambda_i^{max}], \forall i \in \mathcal{C}$. Therefore, the computation job request profile or true type of all the competing cloudlets is represented as $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_N) \in \boldsymbol{\Lambda} = (\Lambda_1 \times \Lambda_2 \times \cdots \times \Lambda_N)$. In general, the job

request arrival process to cloudlets is a non-stationary process, but it possesses some pseudo-stationary characteristics such that the mean job request arrival rate varies gradually. This facilitates the cloudlets to predict the incoming job request arrival rate by employing efficient traffic prediction algorithms like auto-regressive and moving average (ARMA) algorithm [174]. Each cloudlet also estimates the transmission latency of the incoming job requests from the mobile devices and the intermediate transmission latencies with its neighbouring cloudlets [175]. Nonetheless, the design of load-predictive algorithms is beyond the scope of this paper.

**(c) QoS Latency Requirements of Job Requests:** The individual job requests from mobile devices demand a certain number of CPU cycles to process the jobs within a predefined QoS latency target $D_Q$ [76]. However, in this paper, rather than individual job requests, we are considering a batch of incoming job requests to cloudlets. Thus, we denote the computational and latency requirements of all the incoming job requests to $i^{\text{th}}$ cloudlet by the consolidated tuple $(\mu_i, \lambda_i, D_Q)$. We assume that all job requests belong to a similar type of low-latency applications and hence, the value of $D_Q$ is same for all the neighbouring cloudlets. This implies that the duration of each timeslot is also equal to $D_Q$ and if any highly overloaded cloudlet cannot process some of its total incoming job requests within $D_Q$, it needs to drop those job request and pay a penalty for that. We consider a more generalised model with multiple classes of job requests with heterogeneous $D_Q$ values as our future work, where we shall use suitable *processor slicing* models. Note that, $M/M/1$ queue provides the upper bound of processing latency of a cloudlet when the aggregated processing rate of all the processors are considered, i.e., the worst-case processing latency of the cloudlets is considered. This ensures that when the average latency of each cloudlet meets the QoS latency target $D_Q$, then all the incoming job requests are processed within $D_Q$.

**(d) User Mobility Model:** We assume that the mobile users cannot move beyond the coverage area of a cloudlet within 1-10 ms, thus consider the quasi-static mobility model for mobile users. This means that mobile users can be considered almost stationary to the corresponding cloudlets during computation offloading period, but may move on later [76]. The cloudlets either start processing the job

requests received from corresponding mobile devices or strategically offload a fraction of them to their neighbouring cloudlets to satisfy the QoS latency target $D_Q$. If any cloudlet becomes highly overloaded to process all the incoming job requests over a certain timeslot and satisfy $D_Q$, we consider that it needs to pay a penalty as well as clear the pending job requests before starting to process the jobs requests for the next timeslot.

## 5.4 Economic and Non-cooperative Load Balancing Game among Cloudlets

In this section, we formulate the load balancing problem among $N \geq 2$ neighbouring cloudlets from same as well as different service providers, under the supervision of a neutral mediator or government body, as a continuous-kernel non-cooperative game. In a practical deployment scenario, overloaded cloudlets intend to offload a fraction of its job requests to its under-loaded neighbouring cloudlets and the under-loaded cloudlets intend to receive some additional job requests from its overloaded neighbouring cloudlets. The complete job request offloading strategy space of all cloudlets is defined as a matrix $\Phi = (\Phi_1^T, \Phi_2^T, \ldots, \Phi_N^T)^T \subset \mathbb{R}^{N \times N}$, where $\boldsymbol{\varphi}_i = (\varphi_{i1}, \varphi_{i2}, \ldots, \varphi_{iN}) \in \Phi_i \subset \mathbb{R}^N$, $\varphi_{ij} \in \Phi_{ij} = [0, 1] \subset \mathbb{R}$, and $\sum_{j=1}^{N} \varphi_{ij} = 1, \forall i \in \mathcal{C}$. Each $\varphi_{ij}$ denotes the fraction of job requests $i^{\text{th}}$ cloudlet offloads to its $j^{\text{th}}$ neighbouring cloudlet. In a stable market scenario, all the service providers tend to install cloudlets with similar processing capabilities (i.e., $n_i = n_j, \forall i, j$) to maintain the feasibility of the non-cooperative load balancing competition over the same customer base and the corresponding service rate request of arrived jobs at each cloudlet tends to become equal as they arrive from the almost similar locality and customer base (i.e., $\mu_i = \mu_j, \forall i, j$). Thus, we consider that the processors of all the neighbouring cloudlets have similar service rates for the incoming job requests from their associated mobile devices (please refer to Appendix A for analysis with different service rates), and the overall processing and queuing latency of the job requests at $i^{\text{th}}$

cloudlet can be derived as follows:

$$\mathbb{T}_i(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) = \frac{1}{\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \varphi_{ji}\lambda_j}. \tag{5.1}$$

### 5.4.1 Economic and Non-cooperative Game Formulation

In this work, we consider the most commonly used pricing schemes e.g., pay-as-you-go policy, where users pay a fixed price per job request without any long-term commitments [176]. Note that, each cloudlet receives a linearly proportional price per workload ($\Omega_1$) for the total amount of incoming job requests from all the connected mobile devices. Each cloudlet pays a linearly proportional price per workload ($\Omega_2$) for offloading job requests to a neighbouring cloudlet from a different service provider and also, receives a linearly proportional price for executing its neighbour's offloaded jobs. We define a parameter $\gamma_{ij}$ to distinguish the price for offloading a job request to neighbouring cloudlets as follows:

$$\gamma_{ij} = \begin{cases} 1; & \text{if a cloudlet offloads job requests to a cloudlet} \\ & \text{from a different service provider} \\ 0; & \text{if a cloudlet offloads job requests to a cloudlet} \\ & \text{that shares the same service provider} \end{cases}$$

This implies that every $i^{\text{th}}$ cloudlet needs to pay a price to $j^{\text{th}}$ cloudlet for offloading any job requests only when it belongs to a different service provider, i.e., $\gamma_{ij} = 1$. Nonetheless, the cost parameter $\Omega_2$ can also be considered as strategy of cloudlets that leads to a cooperative game-theoretic model, which is part of our future work. In addition to these, each cloudlet pays a penalty price with a proportionality cost factor ($\Omega_3$) for exceeding the QoS target latency $D_Q$. In this work, we consider a linear penalty price similar to the linear latency cost designed in [83]. Therefore, all the competing cloudlets with a utility function $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}), \forall i \in \mathcal{C}$, where $\boldsymbol{\varphi}_{-i} = (\boldsymbol{\varphi}_1, \ldots, \boldsymbol{\varphi}_{i-1}, \boldsymbol{\varphi}_{i+1}, \ldots, \boldsymbol{\varphi}_N)$, in the load balancing game intend to solve the maximisation problem as follows:

$$\max_{\boldsymbol{\varphi}_i \in \Phi_i} \mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \sum_{j=1, j\neq i}^{N} \gamma_{ji}\varphi_{ji} \frac{\lambda_j}{\mu_{ii}} - \Omega_2 \sum_{j=1, j\neq i}^{N} \gamma_{ij}\varphi_{ij} \frac{\lambda_i}{\mu_{jj}}$$
$$- \Omega_3 \left[ \left( 1 - \sum_{j=1, j\neq i}^{N} \varphi_{ij} \right) \frac{\lambda_i}{\mu_{ii}} \max \left\{ 0, \left( t_{ui} + \frac{1}{\mu_{ii} - (1 - \sum_{j\neq i} \varphi_{ij})\lambda_i - \sum_{j\neq i} \varphi_{ji}\lambda_j} - D_Q \right) \right\} \right.$$
$$\left. + \sum_{j=1, j\neq i}^{N} \varphi_{ji} \frac{\lambda_j}{\mu_{ii}} \max \left\{ 0, \left( t_{uj} + \frac{1}{\mu_{ii} - (1 - \sum_{j\neq i} \varphi_{ij})\lambda_i - \sum_{j\neq i} \varphi_{ji}\lambda_j} + t_{ji} - D_Q \right) \right\} \right] \tag{5.2}$$

$$\text{subject to} \quad 0 \leq \varphi_{ij} \leq 1, \sum_{j=1}^{N} \varphi_{ij} = 1. \tag{5.3}$$

The first term in (5.2) denotes the total payment received by the cloudlet from mobile users and is linearly proportional to the average workload. The second term denotes the payment $i^{\text{th}}$ cloudlet receives from $j^{\text{th}}$ cloudlet to execute its offloaded job requests and the third term denotes the payment $i^{\text{th}}$ cloudlet makes to $j^{\text{th}}$ cloudlet for offloading job requests. Note that these terms are essential to distinguish payments for offloading job requests among heterogeneous cloudlets from same as well as different service providers. Finally, the fourth and fifth terms denote the penalty $i^{\text{th}}$ cloudlet pays for overall latency (sum of transmission, processing, and queueing latencies) if it exceeds $D_Q$ against the total incoming job requests, otherwise no penalty is applied. Note that we consider the utility function only under the condition of stable operation, i.e., $\left[\mu_{ii} - \left(1 - \sum_{j\neq i} \varphi_{ij}\right) \lambda_i - \sum_{j\neq i} \varphi_{ji}\lambda_j\right] \geq 0, \forall i, j \neq i \in \mathcal{C}$. We denote the average round-trip data transmission latency among mobile devices and the corresponding $i^{\text{th}}$ cloudlet by $t_{ui}$. Also, we denote the inter-cloudlet round-trip data transmission latency by $t_{ij}, \forall i, j \neq i \in \mathcal{C}$.

Note that the utility of each cloudlet in this load balancing game is an affine function when the total latency is within $D_Q$, otherwise, it becomes a non-linear function whose maxima is achieved when the total latency is equal to $D_Q$. Hence, the cloudlets are always interested in gaining some economic benefits by receiving some extra job requests from neighbouring cloudlets as long as the QoS latency requirement $D_Q$ is met. The maximum utility is achieved at the point where

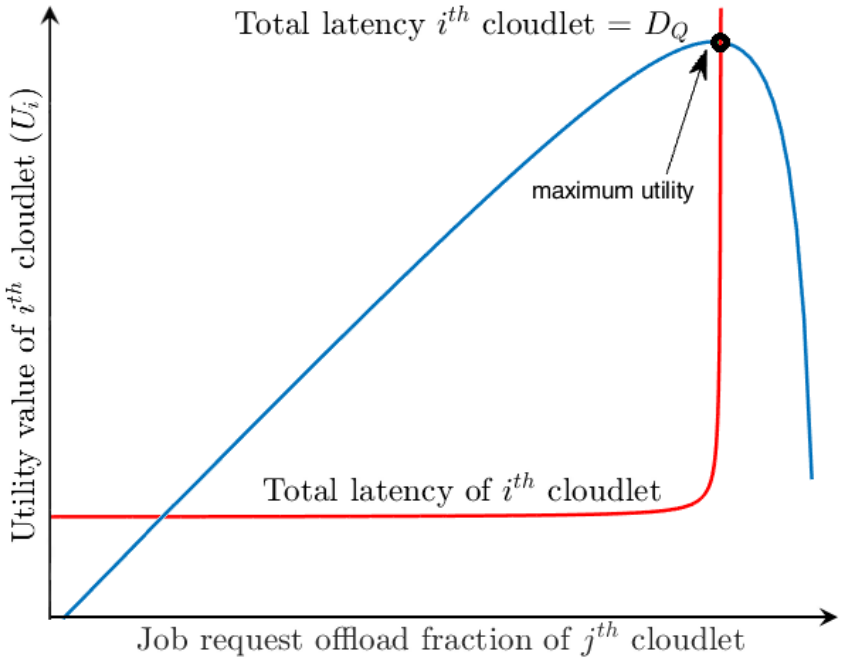



Figure 5.2: A sample utility function of $i^{\text{th}}$ cloudlet against job request offload fraction of neighbouring $j^{\text{th}}$ cloudlet.

the total end-to-end latency is equal to $D_Q$ and the utility starts to decrease beyond this point. Moreover, the individual rationality of each competing cloudlet is maintained, because under all network conditions a default utility, $\mathcal{U}_i^0 = \Omega_1 \frac{\lambda_i}{\mu_{ii}} - \Omega_3 \frac{\lambda_i}{\mu_{ii}} \max\left\{0, \left(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i} - D_Q\right)\right\}, \forall i \in \mathcal{C}$ is ensured.

It is interesting to note that in this load balancing game, each competing cloudlet is interested in *maximizing their individual utilities rather than strictly minimizing the average end-to-end latency* as most of the existing works. Hence, the cloudlets are always interested in receiving some job requests from neighbouring cloudlets as long as the QoS latency requirement $D_Q$ is met and some extra incentive is gained. Fig. 5.2 shows that with a sufficient amount of job requests and a set of properly chosen parameters $\Omega_1$, $\Omega_2$, and $\Omega_3$, the utility function $i^{\text{th}}$ cloudlet monotonically increases as more job requests are offloaded by the neighbouring $j^{\text{th}}$ cloudlet until the total end-to-end latency is equal to the target QoS latency value $D_Q$. The maximum utility is achieved at the point where the total end-to-end latency is equal to $D_Q$ and the utility starts to decrease beyond this point. Therefore, the fraction of incoming job requests offloaded by an overloaded cloudlet is controlled by the overloaded cloudlet itself as well as its under-loaded neighbouring cloudlets.

Furthermore, note that due to the utility function (5.2) and constraint (5.3), which does not provide an explicit latency bound on the participating cloudlets, even highly over-loaded cloudlets can participate in the game and can offload some of the job requests to the relatively under-loaded neighbouring cloudlets. This makes their utility higher than the utility by not participating in the game. Nonetheless, under such conditions the game formulation in [83] that has explicit delay bound on participating cloudlets becomes infeasible and hence, a valid NE solution can not be obtained. We prefer to investigate the NE of the game $\Gamma$, because none of the competing cloudlets find it beneficial to deviate unilaterally from the NE computational offload strategy $\boldsymbol{\varphi}^* = (\boldsymbol{\varphi}_1^{*T}, \boldsymbol{\varphi}_2^{*T}, \ldots, \boldsymbol{\varphi}_N^{*T})$.

**Lemma 5.7.** *The utility functions $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}), \forall i \in \mathcal{C}$ are quasi-concave functions of $\boldsymbol{\varphi}_i$.*

*Proof.* Please refer to Appendix B. □

**Theorem 5.8.** *For the game $\Gamma = \langle \mathcal{C}, (\Phi_i)_{i \in C}, (\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}))_{i \in C} \rangle$, there exists at least one pure strategy NE.*

*Proof.* In the game $\Gamma$, the strategy spaces of all the competing cloudlets $\Phi_i$ are compact and convex in nature. The utility functions $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}), \forall i \in \mathcal{C}$ are continuous functions of $(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i})$ with the condition of stable operation, $[\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \varphi_{ji}\lambda_j] \geq 0, \forall i, j \neq i \in \mathcal{C}$. Moreover, we showed in Lemma 5.7 that $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}), \forall i \in \mathcal{C}$ are quasi-concave functions of $\boldsymbol{\varphi}_i$. These are the sufficient conditions to ensure the existence of a pure strategy NE for the non-cooperative load balancing game $\Gamma$. □

### 5.4.2 Computation of the Pure-Strategy Nash Equilibrium

As the utility functions $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}), \forall i \in \mathcal{C}$ are non-differentiable in nature, we cannot derive the "best response functions" of the cloudlets by directly differentiating the utility functions. Therefore, we use the necessary conditions, i.e., the first-order KKT conditions to compute the pure-strategy NE [177]. To derive analytical closed form expressions for NE of the load balancing game, at first we consider only two

cloudlets and followed by we extend this analysis to a multi-cloudlet game. The utility function of each $i^{\text{th}}$ cloudlet is defined as follows:

$$\begin{aligned}\mathcal{U}_i^2(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) &= \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \gamma_{ji} \varphi_{ji} \frac{\lambda_j}{\mu_{ii}} - \Omega_2 \gamma_{ij} \varphi_{ij} \frac{\lambda_i}{\mu_{jj}} \\ &- \Omega_3 \left[ (1-\varphi_{ij}) \frac{\lambda_i}{\mu_{ii}} \max\left\{0, \left( t_{ui} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q \right) \right\} \right. \\ &\left. + \varphi_{ji} \frac{\lambda_j}{\mu_{ii}} \max\left\{0, \left( t_{uj} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} + t_{ji} - D_Q \right) \right\} \right]. \end{aligned} \tag{5.4}$$

Note that this utility definition (5.4) takes two different forms depending on $i^{\text{th}}$ cloudlet meets or exceeds the QoS latency target $D_Q$. Therefore, the objective of an under-loaded cloudlet is interpreted as follows:

$$\boldsymbol{\mathcal{P}^u}: \quad \max_{\varphi_i \in \Phi_i} \mathcal{U}_i^2(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \gamma_{ji} \varphi_{ji} \frac{\lambda_j}{\mu_{ii}} - \Omega_2 \gamma_{ij} \varphi_{ij} \frac{\lambda_i}{\mu_{jj}}$$

$$\text{subject to} \quad 0 \leq \varphi_{ij} \leq 1,$$
$$\left( t_{uj} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} + t_{ji} - D_Q \right) \leq 0.$$

Similarly, the objective of an overloaded cloudlet is interpreted as follows:

$$\begin{aligned}\boldsymbol{\mathcal{P}^o}: \quad \max_{\varphi_i \in \Phi_i} \mathcal{U}_i^2(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) &= \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \gamma_{ji} \varphi_{ji} \frac{\lambda_j}{\mu_{ii}} - \Omega_2 \gamma_{ij} \varphi_{ij} \frac{\lambda_i}{\mu_{jj}} \\ -\Omega_3 &\left[ (1-\varphi_{ij}) \frac{\lambda_i}{\mu_{ii}} \left( t_{ui} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q \right) \right. \\ &\left. + \varphi_{ji} \frac{\lambda_j}{\mu_{ii}} \left( t_{uj} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} + t_{ji} - D_Q \right) \right] \end{aligned}$$

$$\text{subject to} \quad 0 \leq \varphi_{ij} \leq 1,$$
$$\left( t_{ui} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q \right) \geq 0.$$

Therefore, to show the uniqueness of NE of the game $\Gamma$, we need to derive the NE under different network conditions, e.g., all cloudlets are under-loaded, all cloudlets are overloaded, or some cloudlets are under-loaded and some cloudlets are

overloaded.

**Case-1:** $\left[\left(t_{ui}+\frac{1}{\mu_{ii}-\lambda_i}\right)<D_Q,\left(t_{uj}+\frac{1}{\mu_{jj}-\lambda_j}\right)<D_Q\right]$

In this case, both the cloudlets are under-loaded and their Lagrangian function can be written from $\boldsymbol{\mathcal{P}^u}$ as follows:

$$\mathcal{L}_i^u = \Omega_1\frac{\lambda_i}{\mu_{ii}} + \Omega_2\gamma_{ji}\varphi_{ji}\frac{\lambda_j}{\mu_{ii}} - \Omega_2\gamma_{ij}\varphi_{ij}\frac{\lambda_i}{\mu_{jj}} + \alpha_i\varphi_{ij} + \beta_i(1-\varphi_{ij}) - \xi_i\left(t_{uj}+\frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j}+t_{ji}-D_Q\right), \tag{5.5}$$

where $\alpha_i, \beta_i, \xi_i \geq 0$ are Lagrange multipliers corresponding to the respective constraints. Thus, the necessary first-order KKT conditions (FOC) $\forall i, j \neq i \in \mathcal{C}$ are derived as follows:

$$\frac{\partial\mathcal{L}_i^u}{\partial\varphi_{ij}} = -\Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}} + \alpha_i - \beta_i + \xi_i\left[\frac{\lambda_i}{(\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j)^2}\right] = 0. \tag{5.6}$$

In addition to this, the complementary slackness conditions (CSC) $\forall i, j \neq i \in \mathcal{C}$ are written as follows:

$$\alpha_i\varphi_{ij} = 0, \tag{5.7}$$

$$\beta_i(1-\varphi_{ij}) = 0, \tag{5.8}$$

$$\xi_i\left(t_{uj}+\frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j}+t_{ji}-D_Q\right) = 0. \tag{5.9}$$

In this case, both the cloudlets have sufficient computational resources to meet the QoS latency target and hence, $(t_{uj}+\frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j}+t_{ji}-D_Q)<0$. Therefore, from (5.9) we get $\xi_i^* = \xi_j^* = 0$ and it is easy to show that the unique NE solution is $\varphi_{ij}^* = \varphi_{ji}^* = 0$. Intuitively, this implies that none of the cloudlets offload any job requests to each other and from FOC (5.6) and CSC (5.7)-(5.9), we get the corresponding values of Lagrange multipliers $\alpha_i^* = \Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}}$, $\beta_i^* = 0$, $\alpha_j^* = \Omega_2\gamma_{ji}\frac{\lambda_j}{\mu_{ii}}$, and $\beta_j^* = 0$. A more detailed analysis is shown in Appendix C.

**Case-2:** $\left[\left(t_{ui}+\frac{1}{\mu_{ii}-\lambda_i}\right)\geq D_Q,\left(t_{uj}+\frac{1}{\mu_{jj}-\lambda_j}\right)\geq D_Q\right]$

In this case, both the cloudlets are overloaded and hence, the corresponding Lagrangian function from $\boldsymbol{\mathcal{P}^o}$:

$$\begin{aligned}\mathcal{L}_i^o &= \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \gamma_{ji}\varphi_{ji}\frac{\lambda_j}{\mu_{ii}} - \Omega_2 \gamma_{ij}\varphi_{ij}\frac{\lambda_i}{\mu_{jj}} \\ &- \Omega_3 \left[(1-\varphi_{ij})\frac{\lambda_i}{\mu_{ii}}\left(t_{ui} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q\right)\right. \\ &\left. + \varphi_{ji}\frac{\lambda_j}{\mu_{ii}}\left(t_{uj} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} + t_{ji} - D_Q\right)\right] \\ &+ \alpha_i\varphi_{ij} + \beta_i(1-\varphi_{ij}) \\ &+ \xi_i\left(t_{ui} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q\right). \end{aligned} \tag{5.10}$$

Again, the necessary FOC and CSC $\forall i, j \neq i \in \mathcal{C}$ are derived as follows:

$$\begin{aligned}\frac{\partial \mathcal{L}_i^o}{\partial \varphi_{ij}} &= -\Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}} + \Omega_3\frac{\lambda_i}{\mu_{ii}}\left[t_{ui} + \frac{\mu_{ii}}{(\mu_{ii}-(1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j)^2} - D_Q\right] \\ &+ \alpha_i - \beta_i - \xi_i\left[\frac{\lambda_i}{(\mu_{ii}-(1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j)^2}\right] = 0, \end{aligned} \tag{5.11}$$

$$\alpha_i\varphi_{ij} = 0, \tag{5.12}$$

$$\beta_i(1-\varphi_{ij}) = 0, \tag{5.13}$$

$$\xi_i\left(t_{ui} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q\right) = 0. \tag{5.14}$$

As both the cloudlets are overloaded, therefore, in this case, $(t_{ui} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j} - D_Q) > 0$ and (5.14) yields $\xi_i^* = \xi_j^* = 0$. In this case also, it is obvious that the unique NE strategy for both the cloudlets is not to offload any job requests to each other, i.e., $\varphi_{ij}^* = \varphi_{ji}^* = 0$. Along with this, from FOC (5.11) and CSC (5.12)-(5.14), we get the corresponding values of Lagrange multipliers $\alpha_i^* = \Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}} - \Omega_3\frac{\lambda_i}{\mu_{ii}}\left[t_{ui} + \frac{\mu_{ii}}{(\mu_{ii}-\lambda_i)^2}\right]$, $\beta_i^* = 0$, $\alpha_j^* = \Omega_2\gamma_{ji}\frac{\lambda_j}{\mu_{ii}} - \Omega_3\frac{\lambda_j}{\mu_{jj}}\left[t_{uj} + \frac{\mu_{jj}}{(\mu_{jj}-\lambda_j)^2}\right]$, and $\beta_j^* = 0$.

**Case-3:** $\left[\left(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i}\right) \geq D_Q, \left(t_{uj} + \frac{1}{\mu_{jj}-\lambda_j}\right) < D_Q\right]$

In this case, we consider that $i^{\text{th}}$ cloudlet is overloaded but $j^{\text{th}}$ cloudlet is underloaded. Thus, $i^{\text{th}}$ cloudlet needs to offload some job requests to $j^{\text{th}}$ cloudlet to meet the QoS target latency $D_Q$, as long as $j^{\text{th}}$ cloudlet does not exceed $D_Q$. This implies

that $0 < \varphi_{ij}^* < 1$ and $\varphi_{ji}^* = 0$. Moreover, we get $(t_{ui} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j} - D_Q) = 0$, $\xi_i^* > 0$ and $(t_{uj} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j} + t_{ji} - D_Q) \leq 0$, $\xi_j^* = 0$. Considering the FOC (5.11) and CSC (5.12)-(5.14) for $i^{\text{th}}$ cloudlet and FOC (5.6) and CSC (5.7)-(5.9) for $j^{\text{th}}$ cloudlet, we get the following unique NE solution:

$$\varphi_{ij}^* = 1 - \frac{1}{\lambda_i}\left[\mu_{ii} - \frac{1}{D_Q - t_{ui}}\right] \leq \frac{1}{\lambda_i}\left[\mu_{jj} - \lambda_j - \frac{1}{D_Q - t_{ui} - t_{ij}}\right]. \tag{5.15}$$

Along with this, we get the Lagrange multiplier values $\alpha_i^* = 0$, $\beta_i^* = 0$, $\alpha_j^* = 0$, and $\beta_j^* = 0$. It is interesting to note from (5.15) that for the overloaded $i^{\text{th}}$ cloudlet, the computation offloading decision is not entirely controlled by itself. As long as the under-loaded $j^{\text{th}}$ cloudlet can process the entire extra load from $i^{\text{th}}$ cloudlet, $\varphi_{ij}^* = 1 - \frac{1}{\lambda_i}\left[\mu_{ii} - \frac{1}{D_Q - t_{ui}}\right]$ is acceptable. However, when $j^{\text{th}}$ cloudlet cannot process the entire extra load, then the offload fraction is not allowed to exceed the upper-bound, and hence, $\varphi_{ij}^* = \frac{1}{\lambda_i}\left[\mu_{jj} - \lambda_j - \frac{1}{D_Q - t_{ui} - t_{ij}}\right]$. Therefore, from the above analysis we showed that we can compute a unique NE for our proposed load balancing game among competing cloudlets.

Now, we extend the NE solution for a general $N \geq 2$ cloudlet load balancing scenario. From the above analysis, it is evident that when all the cloudlets are under-loaded or overloaded, then the NE is not to offload any job requests to each other, i.e., $\boldsymbol{\varphi}_i^* = \mathbf{0}$. However, when some cloudlets are under-loaded and some are overloaded, we need to propose a general solution method. As in Case-3 we observed that the NE load balancing strategies are not entirely controlled by the overloaded cloudlets but also the under-loaded cloudlets, hence we introduce a new set of decision variables for each $i^{\text{th}}$ cloudlet denoting the fraction of job requests received from all $j^{\text{th}}$ cloudlets i.e., $\boldsymbol{\psi}_i = (\psi_{1i}, \psi_{2i}, \ldots, \psi_{Ni})$ with the equality constraints, $\varphi_{ji} = \psi_{ji}, \forall i, j \neq i \in \mathcal{C}$. We can mark the $N_u$ under-loaded cloudlets and $N_o$ overloaded cloudlets by observing $\lambda_i$ such that $N_u + N_o = N$. Moreover, the latency constraints and FOC

for under-loaded cloudlets respectively are written as follows:

$$\left(t_{uj} + \frac{1}{\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \psi_{ji}\lambda_j} + t_{ji} - D_Q\right) \leq 0, \forall i \in \mathcal{C}, \tag{5.16}$$

$$\nabla_{\varphi_i} \mathcal{U}_i^N + \nabla_{\varphi_i} \left[\boldsymbol{\alpha}_i^T \boldsymbol{\varphi}_i + \boldsymbol{\beta}_i^T (\mathbf{1} - \boldsymbol{\varphi}_i)\right.$$
$$\left. -\boldsymbol{\xi}_i^T \left(t_{uj} + \frac{1}{\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \psi_{ji}\lambda_j} + t_{ji} - D_Q\right)\right] = 0. \tag{5.17}$$

Similarly, the latency constraints and FOC for the overloaded cloudlets respectively are written as follows:

$$\left(t_{ui} + \frac{1}{\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \psi_{ji}\lambda_j} - D_Q\right) \geq 0, \forall i \in \mathcal{C}, \tag{5.18}$$

$$\nabla_{\varphi_i} \mathcal{U}_i^N + \nabla_{\varphi_i} \left[\boldsymbol{\alpha}_i^T \boldsymbol{\varphi}_i + \boldsymbol{\beta}_i^T (\mathbf{1} - \boldsymbol{\varphi}_i)\right.$$
$$\left. +\boldsymbol{\xi}_i^T \left(t_{ui} + \frac{1}{\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \psi_{ji}\lambda_j} - D_Q\right)\right] = 0. \tag{5.19}$$

**Theorem 5.9.** *The constrained optimisation problem* $\boldsymbol{\mathcal{P}}$ *is equivalent to the game* $\Gamma = \langle \mathcal{C}, (\Phi_i)_{i \in C}, (\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}))_{i \in C} \rangle$.

$$\boldsymbol{\mathcal{P}}: \quad \text{Minimise} \sum_{i=1}^{N} \left[\boldsymbol{\alpha}_i^T \boldsymbol{\varphi}_i + \boldsymbol{\beta}_i^T (1 - \boldsymbol{\varphi}_i)\right]$$
$$- \sum_{i=1}^{N_u} \boldsymbol{\xi}_i^T \left(t_{uj} + \frac{1}{\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \psi_{ji}\lambda_j} + t_{ji} - D_Q\right)$$
$$+ \sum_{i=1}^{N_o} \boldsymbol{\xi}_i^T \left(t_{ui} + \frac{1}{\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \psi_{ji}\lambda_j} - D_Q\right)$$

$$\text{subject to} \quad \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \boldsymbol{\xi}_i \succeq \mathbf{0}, \forall i \in \mathcal{C},$$
$$0 \leq \varphi_{ij} \leq 1, \sum_{j=1}^{N} \varphi_{ij} = 1, \forall i, j \neq i \in \mathcal{C},$$
$$\varphi_{ji} = \psi_{ji}, \forall i, j \neq i \in \mathcal{C},$$
$$\text{constraints (5.16)-(5.19)}, \forall i \in \mathcal{C}.$$

*Proof.* We observe that the objective of the problem $\boldsymbol{\mathcal{P}}$ is non-negative with any feasible solution. If we can find a vector $\boldsymbol{\varphi}^* \in \Phi$ as an NE point of the game $\Gamma$, then there exist a set of non-negative values of the variables $\boldsymbol{\alpha}_i^* \in \mathbb{R}^{N-1}$, $\boldsymbol{\beta}_i^* \in \mathbb{R}^{N-1}$, and $\boldsymbol{\xi}_i^* \in \mathbb{R}^{N-1}$ such that all the necessary FOC and CSC conditions of $\Gamma$ are satisfied. Clearly, with these values of the variables, the objective function of $\boldsymbol{\mathcal{P}}$ becomes 0 and hence, the non-negative optimal value is reached. On the other hand, due to the concavity of the constraint functions, the KKT conditions are sufficient for $\boldsymbol{\varphi}_i^*$ to be the maxima of utilities of each cloudlet $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i})$. □

Nonetheless, it is noteworthy that the problem $\boldsymbol{\mathcal{P}}$ is not a convex optimisation problem because of the equality constraints (5.17) and (5.19). The feasible set of $\boldsymbol{\mathcal{P}}$ is created by the intersection of all the inequality and equality constraints, which becomes non-convex in general. Hence, we can reformulate the problem as a convex optimisation problem by adding the squares of left-hand sides of (5.17) and (5.19) to the objective [178]. With this modification, the objective is a convex function as it is a sum of a few convex functions and the feasible set is also convex as it is created by the intersection of convex inequality constraints only. This problem now can be solved by using a "gradient-projection algorithm" or any commercially available solver package. An algorithm to solve the game $\Gamma$ is summarised in Algorithm 5.1.

## 5.5 Centralised Load Balancing Framework

Private information of a cloudlet is defined as the information that is known only to that cloudlet and no other entity in the network [179]. The competing cloudlets are non-cooperative and rational utility maximisers, whereas a neutral mediator, on the other hand does not have any utility associated with the incoming job requests. The mediator acts like a supervisor that ensures fair participation of all the competing cloudlets in the market competition and installs a computational facility in the proximity of the competing cloudlets to compute the NE for all the cloudlets over that period by using a centralised algorithm and inform the cloudlets, as shown in Fig. 5.3. In a centralised framework, the mediator firstly addresses the "preference elicitation problem" to elicit truthful information from competing cloudlets by

**Algorithm 5.1** Projection Algorithm with Constant Step Size

1: **Input:** Network parameters $\mu_{ii}$, $\lambda_i$, $t_{ui}$, $t_{ij}, \forall i, j \neq i \in \mathcal{C}$.
2: **Output:** The pure-strategy NE of the non-cooperative load balancing game.
3: **Initialisation:** Choose any Lagrange multipliers $\boldsymbol{\alpha}^{(0)}$, $\boldsymbol{\beta}^{(0)}$, $\boldsymbol{\xi}^{(0)} \geq 0$, step size $\omega > 0$, and tolerance limit $\epsilon > 0$. Set the index $t = 0$.
4: **if** all cloudlets are under-loaded, i.e., $(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i}) < D_Q$, or all cloudlets are overloaded, i.e., $(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i}) \geq D_Q$ **then** choose not to offload, i.e., set $\boldsymbol{\varphi}^* = \boldsymbol{I}_N$: STOP;
5: **else** identify the FOC of the under-loaded cloudlets as (5.16)-(5.17) and the FOC of the overloaded cloudlets as (5.18)-(5.19). Use the corresponding CSC to formulate the objective function of $\mathcal{P}$.
6: **if** $\boldsymbol{\varphi}^{(t)}(\boldsymbol{\chi}^{(t)})$, $\boldsymbol{\alpha}^{(t)}(\boldsymbol{\chi}^{(t)})$, $\boldsymbol{\beta}^{(t)}(\boldsymbol{\chi}^{(t)})$, $\boldsymbol{\xi}^{(t)}(\boldsymbol{\chi}^{(t)})$ satisfies the desirable tolerance limit $\epsilon$ **then** STOP;
7: With given $\boldsymbol{\chi}^{(t)}$, compute $\boldsymbol{\varphi}^{(t)}(\boldsymbol{\chi}^{(t)})$, $\boldsymbol{\alpha}^{(t)}(\boldsymbol{\chi}^{(t)})$, $\boldsymbol{\beta}^{(t)}(\boldsymbol{\chi}^{(t)})$, $\boldsymbol{\xi}^{(t)}(\boldsymbol{\chi}^{(t)})$ as the solution of convexified optimisation problem $\mathcal{P}$;
8: Update the Lagrange multipliers $\boldsymbol{\chi}$ corresponding to the constraints of $\mathcal{P}$ by gradient projection as follows:

$$\boldsymbol{\chi}^{(t+1)} = [\boldsymbol{\chi}^{(t)} + \omega\Theta(\boldsymbol{\chi}^{(t)})]^+,$$

where $\Theta$ is the feasible set of $\mathcal{P}$.
9: Set $t \leftarrow t + 1$; go to Step 6.

imposing efficient mechanisms on the cloudlets who periodically reveal their private information about incoming job requests. We represent the "revealed type profile" of the competing cloudlets as $\hat{\boldsymbol{\lambda}} = (\hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_N) \in \boldsymbol{\Lambda}$. However, we assume that cloudlets usually truthfully reveal other network parameters like $\mu_{ii}$, $t_{ui}$, and $t_{ij}$, because $\mu_{ii}$ is dependent on the type of job requests and usually do not change frequently, and $t_{ui}$, $t_{ij}$ can be cross-verified as they are reported by multiple cloudlets. This mechanism provides us rules/guidelines on how the cloudlets and the mediator should communicate with each other. Secondly, the mediator addresses the "preference aggregation problem" to compute the NE load balancing strategies for each cloudlet based on the communicated information, to induce the desired strategic behavior from the cloudlets [179]. The fundamental stages of the overall control design are shown in Fig. 5.4 and are summarised below:

(a) Each cloudlet periodically executes a load-predictive learning algorithm at every $n^{\text{th}}$ time-slot by using the information from $(n-1)^{\text{th}}$ time-slot and the data

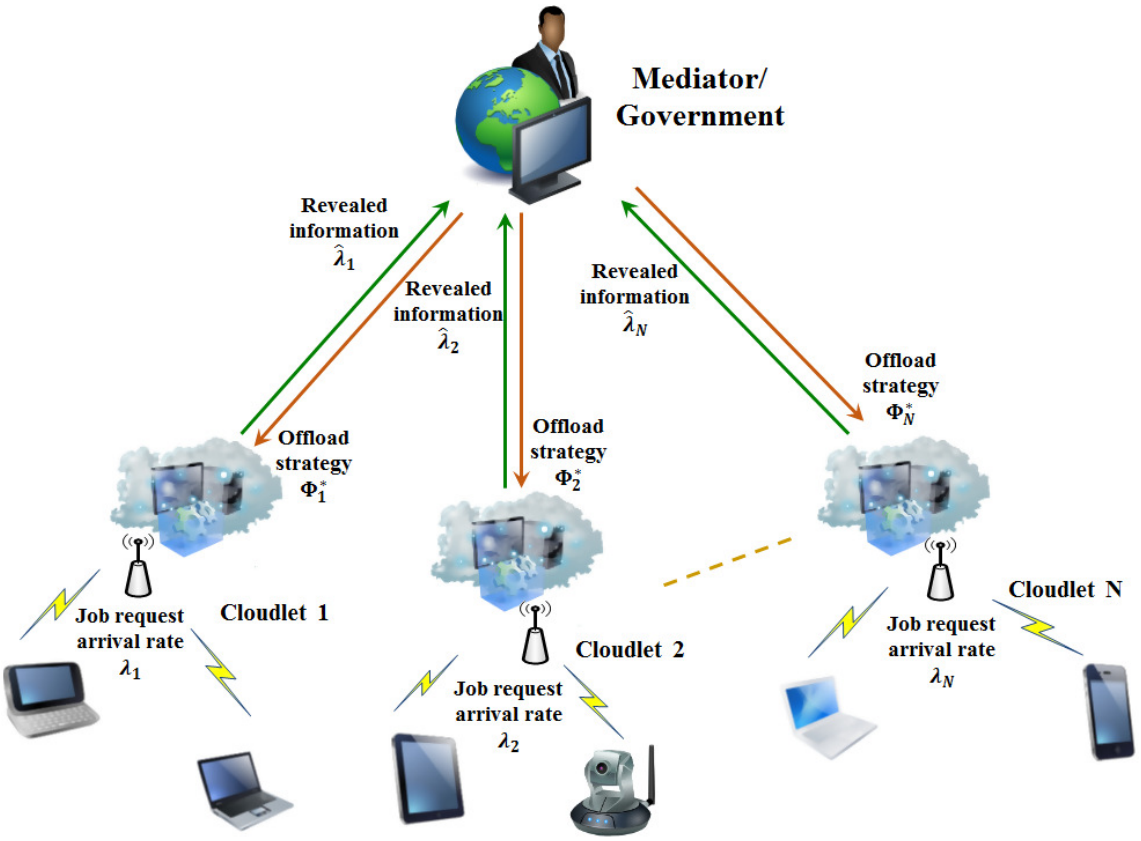



Figure 5.3: A simplified schematic diagram showing the interaction among $N$ competing cloudlets supervised by a neutral mediator.

available regarding the job arrival history to predict the incoming job request arrival rate of the $(n+1)^{\text{th}}$ time-slot.

(b) Along with job request arrival rate, each cloudlet also estimates the transmission latency of the incoming job requests from the mobile devices by using the given stochastic parameters of the interface between mobile devices and cloudlets, as well as, estimates the intermediate transmission latencies with its neighbouring cloudlets.

(c) After learning all this information, each cloudlet communicates the estimated incoming job request arrival rate and the transmission latencies to the computational facility installed by the mediator.

(d) The mediator computes the NE computation offloading strategy by employing a centralised algorithm and broadcasts to all the competing cloudlets before the $(n+1)^{\text{th}}$ time-slot.

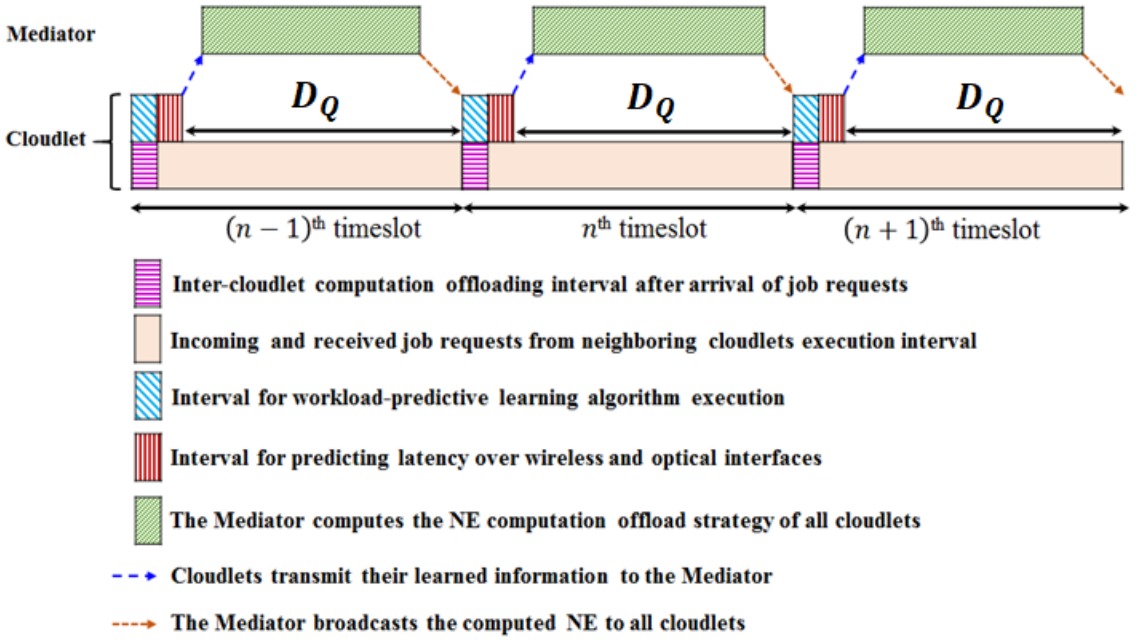



Figure 5.4: A timing diagram illustrating the overall control design for interactions among $N$ competing cloudlets, supervised by a neutral mediator.

(e) Accordingly, the cloudlets offload some fraction of their total incoming job requests to their neighbouring cloudlets when the $(n+1)^{\text{th}}$ time-slot arrives.

### 5.5.1 Incentive Compatible Mechanism Design

In this section, we show that our proposed load balancing game is incentive compatible and ensures the truthful revelation of private information from the competing cloudlets. To take advantage of the analytical closed-form expressions derived in Section 5.4, at first we consider the load balancing game between two cloudlets and show that truthful revelation of private information to the mediator is a weakly-dominated strategy for $i^{\text{th}}$ cloudlet irrespective of the information shared by $j^{\text{th}}$ cloudlet. Recall that the true types of the cloudlets are $(\lambda_i, \lambda_j)$ and their revealed types to the mediator are $(\hat{\lambda}_i, \hat{\lambda}_j)$. With the revealed type information, the mediator solves the load balancing game and broadcasts the NE load balancing strategies $\hat{\varphi}^*_{ij}$ and $\hat{\varphi}^*_{ji}$ to the competing cloudlets. The utility of each cloudlet based on its true type is $\mathcal{U}_i^2$ and the utility computed by each cloudlet based on their revealed type is denoted by $\hat{\mathcal{U}}_i^2$. The incentive mechanism design is truthful if $(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) \leq 0$ always holds true.

**Case-1: [Both cloudlets are under-loaded]**

In this case, $(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i}) < D_Q, (t_{uj} + \frac{1}{\mu_{jj}-\hat{\lambda}_j}) < D_Q$. Now, if the revealed type $\hat{\lambda}_i < \lambda_i$ or, $\hat{\lambda}_i \geq \lambda_i$ such that $(t_{ui} + \frac{1}{\mu_{ii}-\hat{\lambda}_i}) < D_Q$, then the NE does not change from the true NE, i.e., $\hat{\varphi}^*_{ij} = 0$ and $\hat{\varphi}^*_{ji} = 0$. Hence, the utility of $i^{\text{th}}$ cloudlet does not improve, because $(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} - \Omega_1 \frac{\lambda_i}{\mu_{ii}} = 0$. Intuitively, a truly under-loaded cloudlet cannot improve its utility by revealing a job request arrival rate such that it is still under-loaded. Note that the payment received from mobile users in $\hat{\mathcal{U}}_i^2$ always remains same as in $\mathcal{U}_i^2$.

However, if $\hat{\lambda}_i \geq \lambda_i$ such that $(t_{ui} + \frac{1}{\mu_{ii}-\hat{\lambda}_i}) \geq D_Q$, then the $i^{\text{th}}$ cloudlet needs to offload $\hat{\varphi}^*_{ij}\hat{\lambda}_i$ job requests to the $j^{\text{th}}$ cloudlet. The NE solution with the revealed information is $\hat{\varphi}^*_{ij} = 1 - \frac{1}{\hat{\lambda}_i}\left[\mu_{ii} - \frac{1}{D_Q - t_{ui}}\right] \leq \frac{1}{\hat{\lambda}_i}\left[\mu_{jj} - \hat{\lambda}_j - \frac{1}{D_Q - t_{ui} - t_{ij}}\right]$ and $\hat{\varphi}^*_{ji} = 0$. Clearly, the utility of the $i^{\text{th}}$ cloudlet decreases because $(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} - \Omega_2 \gamma_{ij} \hat{\varphi}^*_{ij} \frac{\hat{\lambda}_i}{\mu_{jj}} - \Omega_1 \frac{\lambda_i}{\mu_{ii}} = -\Omega_2 \gamma_{ij} \hat{\varphi}^*_{ij} \frac{\hat{\lambda}_i}{\mu_{jj}} < 0$. In other words, the utility of a truly under-loaded cloudlet decreases if it reveals itself as overloaded when the neighbouring cloudlet is under-loaded.

**Case-2: [One overloaded and one under-loaded cloudlet]**

Firstly we consider $(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i}) \geq D_Q, (t_{uj} + \frac{1}{\mu_{jj}-\hat{\lambda}_j}) < D_Q$, i.e., the $i^{\text{th}}$ cloudlet is overloaded and the $j^{\text{th}}$ cloudlet is under-loaded. If $\hat{\lambda}_i \geq \lambda_i$, then $i^{\text{th}}$ cloudlet offloads more or same amount of job requests, depending on $\hat{\lambda}_j$, i.e., $\hat{\varphi}^*_{ij}\hat{\lambda}_i \geq \varphi^*_{ij}\lambda_i$ and $\hat{\varphi}^*_{ij}, \varphi^*_{ij}$ can be computed from (5.15) with $\hat{\varphi}^*_{ji} = 0$. Thus, the utility of the $i^{\text{th}}$ cloudlet decreases because $(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} - \Omega_2 \gamma_{ij} \hat{\varphi}^*_{ij} \frac{\hat{\lambda}_i}{\mu_{jj}} - \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \gamma_{ij} \varphi^*_{ij} \frac{\lambda_i}{\mu_{jj}} = -\Omega_2 \frac{\gamma_{ij}}{\mu_{jj}} (\hat{\varphi}^*_{ij}\hat{\lambda}_i - \varphi^*_{ij}\lambda_i) < 0$. Therefore, an overloaded cloudlet revealing itself as more overloaded, needs to offload more than actually needed to meet the QoS target latency, and hence the utility decreases.

On the other hand, if $\hat{\lambda}_i < \lambda_i$ but $(t_{ui} + \frac{1}{\mu_{ii}-\hat{\lambda}_i}) \geq D_Q$, then $i^{\text{th}}$ cloudlet offloads less than that required to meet the QoS target latency, i.e., $\hat{\varphi}^*_{ij}\hat{\lambda}_i \leq \varphi^*_{ij}\lambda_i$ and $\hat{\varphi}^*_{ij}, \varphi^*_{ij}$ can be computed from (5.15) with $\hat{\varphi}^*_{ji} = 0$. This implies that $i^{\text{th}}$ cloudlet needs to pay lesser incentives than before but the penalty for latency decreases its utility

value as follows:

$$(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = -\Omega_2 \frac{\gamma_{ij}}{\mu_{jj}} (\hat{\varphi}_{ij}^* \hat{\lambda}_i - \varphi_{ij}^* \lambda_i) - \Omega_3 \left( \frac{\lambda_i - \hat{\varphi}_{ij}^* \hat{\lambda}_i}{\mu_{ii}} \right) \left[ t_{ui} + \frac{1}{\mu_{ii} - (\lambda_i - \hat{\varphi}_{ij}^* \hat{\lambda}_i)} - D_Q \right]. \tag{5.20}$$

We shall soon derive a necessary condition between $\Omega_2$ and $\Omega_3$ that ensured that $(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) \leq 0$ for expressions like (5.20). Again, if $\hat{\lambda}_i < \lambda_i$ such that $(t_{ui} + \frac{1}{\mu_{ii} - \hat{\lambda}_i}) < D_Q$, then $i^{\text{th}}$ cloudlet is revealed as under-loaded and hence it is not allowed to offload any job requests. Hence, the NE solution with the revealed information is $\hat{\varphi}_{ij}^* = 0$, $\hat{\varphi}_{ji}^* = 0$ and the utility of $i^{\text{th}}$ cloudlet decreases due to the latency penalty as follows:

$$(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_2 \gamma_{ij} \varphi_{ij}^* \frac{\lambda_i}{\mu_{jj}} - \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[ t_{ui} + \frac{1}{\mu_{ii} - \lambda_i} - D_Q \right]. \tag{5.21}$$

Therefore, in this case, an overloaded cloudlet cannot achieve a better utility by revealing false information. Now, we consider $(t_{ui} + \frac{1}{\mu_{ii} - \lambda_i}) < D_Q, (t_{uj} + \frac{1}{\mu_{jj} - \hat{\lambda}_j}) \geq D_Q$, i.e., the $i^{\text{th}}$ cloudlet is under-loaded and the $j^{\text{th}}$ cloudlet is overloaded. In this case, if $\hat{\lambda}_i < \lambda_i$ or, $\hat{\lambda}_i \geq \lambda_i$ such that $(t_{ui} + \frac{1}{\mu_{ii} - \hat{\lambda}_i}) < D_Q$ and can process the entire job request offload requests $\hat{\varphi}_{ji}^* \hat{\lambda}_j$ from the $j^{\text{th}}$ cloudlet, then the NE solution is $\hat{\varphi}_{ij}^* = 0$ and $\hat{\varphi}_{ji}^* = 1 - \frac{1}{\hat{\lambda}_j} \left[ \mu_{jj} - \frac{1}{D_Q - t_{uj}} \right] \leq \frac{1}{\hat{\lambda}_j} \left[ \mu_{ii} - \hat{\lambda}_i - \frac{1}{D_Q - t_{uj} - t_{ji}} \right]$. Thus, the utility of the $i^{\text{th}}$ cloudlet remains unchanged because the same amount of job requests are offloaded irrespective of the falsely revealed information and hence, $(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \gamma_{ji} \hat{\varphi}_{ji}^* \frac{\hat{\lambda}_j}{\mu_{ii}} - \Omega_1 \frac{\lambda_i}{\mu_{ii}} - \Omega_2 \gamma_{ji} \hat{\varphi}_{ji}^* \frac{\hat{\lambda}_j}{\mu_{ii}} = 0$.

Again, if $\hat{\lambda}_i \geq \lambda_i$ such that $(t_{ui} + \frac{1}{\mu_{ii} - \hat{\lambda}_i}) < D_Q$ but cannot process the entire job request offload requests $\hat{\varphi}_{ji}^* \hat{\lambda}_j$ from the $j^{\text{th}}$ cloudlet, then lesser amount of job requests are offloaded, i.e., $\hat{\varphi}_{ji}^* \hat{\lambda}_j \leq \varphi_{ji}^* \hat{\lambda}_j$ and $\hat{\varphi}_{ji}^*, \varphi_{ji}^*$ can be computed from (5.15) with $\hat{\varphi}_{ij}^* = 0$. This implies that the utility of the $i^{\text{th}}$ cloudlet decreases because $(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \gamma_{ji} \hat{\varphi}_{ji}^* \frac{\hat{\lambda}_j}{\mu_{ii}} - \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \gamma_{ji} \varphi_{ji}^* \frac{\hat{\lambda}_j}{\mu_{ii}} = -\Omega_2 \frac{\gamma_{ji}}{\mu_{ii}} (\varphi_{ji}^* \hat{\lambda}_j - \hat{\varphi}_{ji}^* \hat{\lambda}_j) < 0$. Therefore, in this case, an under-loaded cannot achieve a better utility by revealing a higher amount of job requests than the true value with an overloaded neighbouring cloudlet.

Finally, if $\hat{\lambda}_i < \lambda_i$ but actually $i^{\text{th}}$ cloudlet cannot process the entire extra load from $j^{\text{th}}$ cloudlet, then $\hat{\varphi}^*_{ji}\hat{\lambda}_j \geq \varphi^*_{ji}\hat{\lambda}_j$ and $\hat{\varphi}^*_{ji}, \varphi^*_{ji}$ can be computed from (5.15) with $\hat{\varphi}^*_{ij} = 0$. However, the $i^{\text{th}}$ cloudlet fails to meet the QoS latency target and the latency penalty decreases the utility in spite of getting some extra incentives as follows:

$$\begin{aligned}(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_2\gamma_{ji}\hat{\varphi}^*_{ji}\frac{\hat{\lambda}_j}{\mu_{ii}} - \Omega_2\gamma_{ji}\varphi^*_{ji}\frac{\hat{\lambda}_j}{\mu_{ii}} \\ - \Omega_3\frac{\lambda_i}{\mu_{ii}}\left[t_{ui} + \frac{1}{\mu_{ii} - \lambda_i - \hat{\varphi}^*_{ji}\hat{\lambda}_j} - D_Q\right] \\ - \Omega_3\frac{\hat{\varphi}^*_{ji}\hat{\lambda}_j}{\mu_{ii}}\left[t_{uj} + \frac{1}{\mu_{ii} - \lambda_i - \hat{\varphi}^*_{ji}\hat{\lambda}_j} + t_{ji} - D_Q\right].\end{aligned} \tag{5.22}$$

Therefore, we also verified that in this case, any under-loaded cloudlet also cannot achieve a better utility by revealing lower amount of job requests than the true value when the neighbouring cloudlet is overloaded.

**Case-3: [Both cloudlets are overloaded]**

In this case, $(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i}) \geq D_Q, (t_{uj} + \frac{1}{\mu_{jj}-\hat{\lambda}_j}) \geq D_Q$. Now, $\hat{\lambda}_i \geq \lambda_i$ or, $\hat{\lambda}_i < \lambda_i$ such that $(t_{ui} + \frac{1}{\mu_{ii}-\hat{\lambda}_i}) \geq D_Q$, then the NE is same as the true NE, i.e., $\hat{\varphi}^*_{ij} = 0$ and $\hat{\varphi}^*_{ji} = 0$. Hence, the utility of $i^{\text{th}}$ cloudlet does not improve, because $(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_1\frac{\lambda_i}{\mu_{ii}} - \Omega_3\frac{\lambda_i}{\mu_{ii}}\left[t_{ui} + \frac{1}{\mu_{ii}-\lambda_i} - D_Q\right] - \Omega_1\frac{\lambda_i}{\mu_{ii}} + \Omega_3\frac{\lambda_i}{\mu_{ii}}\left[t_{ui} + \frac{1}{\mu_{ii}-\lambda_i} - D_Q\right] = 0$. Therefore, a truly overloaded cloudlet cannot improve its utility by revealing itself as more overloaded.

Again, if $\hat{\lambda}_i < \lambda_i$ such that $(t_{ui} + \frac{1}{\mu_{ii}-\hat{\lambda}_i}) < D_Q$, then the $i^{\text{th}}$ cloudlet is considered as under-loaded and $j^{\text{th}}$ cloudlet offloads some job requests to it. The NE based on the revealed information is $\hat{\varphi}^*_{ij} = 0$ and $\hat{\varphi}^*_{ji} = 1 - \frac{1}{\hat{\lambda}_j}\left[\mu_{jj} - \frac{1}{D_Q - t_{uj}}\right] \leq$

$\frac{1}{\hat{\lambda}_j}\left[\mu_{ii} - \hat{\lambda}_i - \frac{1}{D_Q - t_{uj} - t_{ji}}\right]$. This may help the $i^{\text{th}}$ cloudlet to earn some extra incentives but the latency penalty decreases the utility as follows:

$$\begin{aligned}(\hat{\mathcal{U}}_i^2 - \mathcal{U}_i^2) = \Omega_2 \gamma_{ji} \hat{\varphi}_{ji}^* \frac{\hat{\lambda}_j}{\mu_{ii}} + \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[t_{ui} + \frac{1}{\mu_{ii} - \lambda_i} - D_Q\right] \\ - \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[t_{ui} + \frac{1}{\mu_{ii} - \lambda_i - \hat{\varphi}_{ji}^* \hat{\lambda}_j} - D_Q\right] \\ - \Omega_3 \frac{\hat{\varphi}_{ji}^* \hat{\lambda}_j}{\mu_{ii}} \left[t_{uj} + \frac{1}{\mu_{ii} - \lambda_i - \hat{\varphi}_{ji}^* \hat{\lambda}_j} + t_{ji} - D_Q\right]. \end{aligned} \tag{5.23}$$

Clearly, the above analysis clearly shows that we ensured that any cloudlet cannot achieve a better utility by revealing a higher or lower job request arrival rates than the actual job request. Therefore, truthful revelation of all the cloudlets is a weakly-dominated strategy, irrespective of the information shared by the neighbouring cloudlets. Note that the load balancing game among $N \geq 2$ cloudlets is a direct linear extension from the game between two cloudlets. Thus, all the properties of the NE solution remain consistent and the incentive mechanism design is also applicable.

**Proposition 5.10.** *For the successful implementation of the proposed incentive mechanism with our non-cooperative load balancing game among competing cloudlets,* $\Omega_2 \geq \Omega_3 \left[\max\{t_{ui}\} + \frac{1}{\max\{\mu_{ii}\}} + \max\{t_{ij}\} - D_Q\right], \forall i, j \neq i \in \mathcal{C}$ *is a necessary condition.*

*Proof.* From the NE solution of the proposed non-cooperative load balancing game, we know that any overloaded cloudlet offloads only a small fraction of its total job request arrival rate, i.e., $\varphi_{ij}^* \approx 0$, $\varphi_{ij}^* \lambda_i \ll \lambda_i$. Now, for the proposed incentive mechanism to work perfectly, we need to find a common relation among the primary networks parameters such that $(\hat{\mathcal{U}}_i^N - \mathcal{U}_i^N) \leq 0$ holds for all the expressions (5.20)-(5.23). By inspection, we can conclude that $\Omega_2 \geq \Omega_3 \left[\max\{t_{ui}\} + \frac{1}{\max\{\mu_{ii}\}} + \max\{t_{ij}\} - D_Q\right]$, $\forall i, j \neq i \in \mathcal{C}$ is one such relation and hence, it is a necessary condition for the successful implementation of the incentive mechanism. □

## 5.6 Decentralised Load Balancing Framework

In a distributed cloudlet network, the incoming job request arrival rate $\lambda_i$ to each cloudlet $i \in \mathcal{C}$ is known only to that cloudlet and no other entity in the network. We still assume that the competing cloudlets are non-cooperative and rational utility maximisers and hence, they do not share this "private information". Therefore, a reinforcement learning automata-based algorithm helps the cloudlets to independently make load balancing decisions only from their private information and to aid this decision-making process, we formulate an economic and non-cooperative game with a unique NE at both overloaded and under-loaded network conditions. Moreover, some particular characteristics of the underlying game formulation help to reduce the search space of the reinforcement learning algorithm and improve the convergence rate of the algorithm to a great extent. The fundamental stages of the overall control design are summarised below:

(a) Each cloudlet periodically executes a load-predictive learning algorithm at every $n^{\text{th}}$ time-slot by the data available regarding the job arrival history to predict the incoming job request arrival rate of the $(n+1)^{\text{th}}$ time-slot.

(b) Along with job request arrival rate, each cloudlet also estimates the transmission latency of the incoming job requests from the mobile devices by using the given stochastic parameters of the interface between mobile devices and cloudlets, as well as, estimates the intermediate transmission latencies with its neighbouring cloudlets.

(c) After learning all this information, each cloudlet shares a random amount of its job requests to the neighbouring cloudlets, depending on the latest probability distribution over its strategy space and observes the utility and rewards received. Based on the reward values received at $n^{\text{th}}$ and $(n-1)^{\text{th}}$ time-slots, each cloudlet updates the probability distribution over its strategy space for $(n+1)^{\text{th}}$ time-slot.

### 5.6.1 Distributed Reinforcement Learning Algorithm

In this subsection, we design a continuous-action reinforcement learning automata-based algorithm for learning the NE of the continuous-kernel non-cooperative load

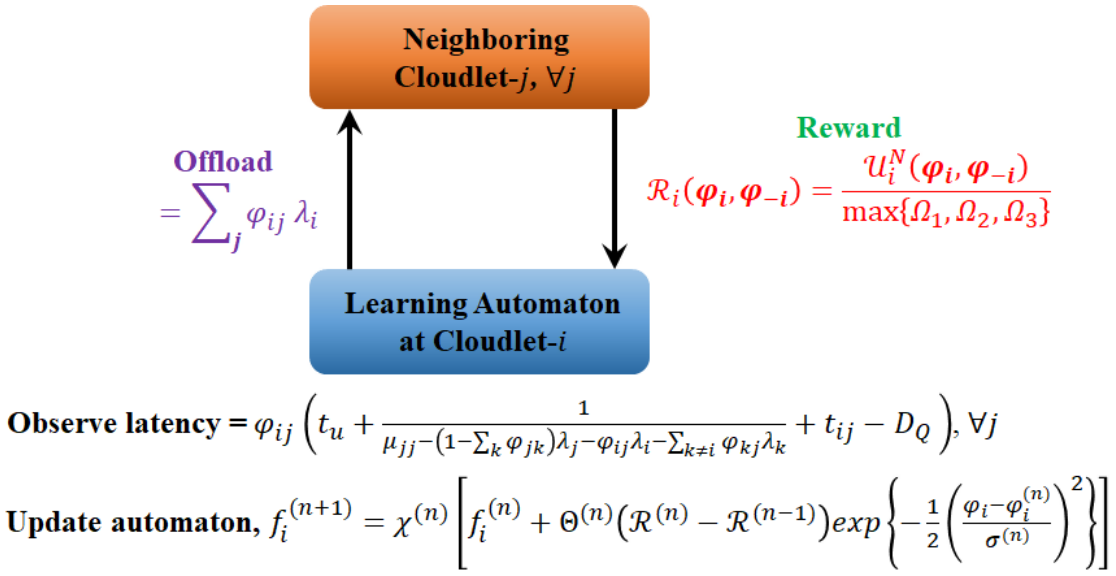



Figure 5.5: A closed loop action-reward diagram of the proposed load balancing reinforcement learning automaton.

balancing game formulated in Section 5.4. At first, we define the "mixed-strategy" of each $i^{\text{th}}$ cloudlet as continuous probability density function (PDF) $\boldsymbol{f}_i(\boldsymbol{\varphi}_i)$ over its pure-strategy space $\Phi_i$. Therefore, the probability of randomly choosing an action within a close neighbourhood of $\varphi_{ij}$ by $i^{\text{th}}$ cloudlet can be determined from the corresponding PDF $f_{ij}(\varphi_{ij})$. The complete mixed-strategy of all the cloudlets is defined as $\mathcal{F} := \boldsymbol{f}_1 \times ... \times \boldsymbol{f}_N$ over the complete pure-strategy space $\Phi$. When each $i^{\text{th}}$ cloudlet chooses an action $\varphi_{ij}$, i.e., offloads $\varphi_{ij}\lambda_i$ job requests to neighbouring $j^{\text{th}}$ cloudlets, then the environment responds with a random reward $\mathcal{R}_i(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) \in [0, 1]$, which is defined as:

$$\mathcal{R}_i(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) = \frac{\mathcal{U}_i^N(\boldsymbol{\lambda}, \boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i})}{\max\{\Omega_1, \Omega_2, \Omega_3\}}. \tag{5.24}$$

In the load balancing game, with a continuous-action reinforcement learning automata-based algorithm, each $i^{\text{th}}$ cloudlet starts with uniform probability distributions as their mixed-strategies over their individual pure-strategy action spaces and keeps on updating the PDFs based on the received rewards in the following time-slots to ultimately find their pure-strategy NE, as shown in Fig. 5.5 [172]. After exploring an action $\boldsymbol{\varphi}_i^{(n)} \in \Phi_i$ during $n^{\text{th}}$ time-slot, the PDFs are updated for

$(n+1)^{\text{th}}$ time-slot by the following update rule:

$$\boldsymbol{f}_i^{(n+1)}(\boldsymbol{\varphi}_i) = \chi^{(n)} \left[ \boldsymbol{f}_i^{(n)}(\boldsymbol{\varphi}_i) + \Theta^{(n)} \left( \mathcal{R}_i^{(n)} - \mathcal{R}_i^{(n-1)} \right) \exp \left\{ -\frac{1}{2} \left( \frac{\boldsymbol{\varphi}_i - \boldsymbol{\varphi}_i^{(n)}}{\sigma^{(n)}} \right)^2 \right\} \right], \tag{5.25}$$

where, $\Theta$ is the "learning rate parameter", $\sigma$ is the "spreading rate parameter" and $\chi$ is a "normalisation factor" so that $\int_{-\infty}^{+\infty} f^{(n+1)} dz = 1$, for any $z$. Note that, our proposed reinforcement learning automata algorithm (5.25) is essentially a "gradient bandit" algorithm, based on the idea of "stochastic gradient ascent" algorithms [180]. Moreover, the term $(\mathcal{R}_i^{(n)} - \mathcal{R}_i^{(n-1)})$ used in this model makes this algorithm highly robust in tracking a non-stationary job request arrival process. The PDFs are continuously updated by the cloudlets based on their private information and rewards received at every time-slot to learn the pure-strategy NE of the non-cooperative load balancing game [181].

**Theorem 5.11.** *The continuous-action reinforcement learning automata-based algorithm with update rule (5.25) converges to the pure-strategy Nash equilibrium of the non-cooperative load balancing game.*

*Proof.* We considered that the strategy space $\Phi_i$ is a compact set in the load balancing game $\Gamma$ and the stochastic reward values are normalised, i.e., $\mathcal{R}_i : \mathbb{R} \rightarrow [0, 1], \forall i \in \mathcal{C}$. The authors of [172] showed that with such conditions, a continuous action reinforcement learning automata-based PDF update rule like (5.25) can be guaranteed to converge to a local optimal NE if the following necessary restrictions are imposed:

(i) We choose a sufficiently small value of $\Theta$ such that each $i^{\text{th}}$ cloudlet can match their expected strategy through iterations.

(ii) We choose a value of $\sigma$ such that the equilibrium point of $\boldsymbol{f}_i(\boldsymbol{\varphi}_i)$ has an upper bound $\frac{1}{\sigma\sqrt{2\pi}}$.

Thus, if the above restrictions are satisfied, then the continuous action reinforcement learning automata-based algorithm will converge to at least one of the existing

**Algorithm 5.2** Distributed Reinforcement Learning Automata-based Algorithm for learning NE of the Load Balancing Game

1: **Initialisation:** Set the iteration index $n = 0$ and $f_{ij}^{(n)}(\varphi_{ij}) = \text{uniform}(\Phi_{ij}), \forall i, j \in \mathcal{C}$.
2: **Output:** The pure-strategy NE of the non-cooperative load balancing game.
3: **if** $i^{\text{th}}$ cloudlet is under-loaded, i.e., $(t_{ui} + \frac{1}{\mu_i - \lambda_i}) < D_Q$ **then** choose not to offload, i.e., $\boldsymbol{\varphi}_i^{(n)} = \mathbf{0}$;
4: **else** $i^{\text{th}}$ cloudlet randomly choose an action $\boldsymbol{\varphi}_i^{(n)}$ based on its latest mixed-strategy $\boldsymbol{f}_i^{(n)}(\boldsymbol{\varphi}_i)$.
5: **if** $j^{\text{th}}$ cloudlet is overloaded or partially processes the jobs received from all $i^{\text{th}}$ cloudlets **then** indicate all $i^{\text{th}}$ cloudlets that only $\hat{\psi}_{ij}\lambda_i$ jobs are processed $(0 \leq \hat{\psi}_{ij} < \varphi_{ij})$, where $\hat{\psi}_{ij}\lambda_i$ are chosen according to the ratio of $\varphi_{ij}\lambda_i$.
6: At the end of $n^{\text{th}}$ time-slot, each $i^{\text{th}}$ cloudlet receives a reward $\mathcal{R}_i^{(n)}$ from the environment.
7: Each $i^{\text{th}}$ cloudlet update their mixed-strategy $\boldsymbol{f}_i^{(n+1)}(\boldsymbol{\varphi}_i)$ by the reinforcement learning automata for $\boldsymbol{\varphi}_i^{(n)} \in \Phi_i$:

$$\boldsymbol{f}_i^{(n+1)}(\boldsymbol{\varphi}_i) = \chi^{(n)} \left[ \boldsymbol{f}_i^{(n)}(\boldsymbol{\varphi}_i) + \Theta^{(n)} \left( \mathcal{R}_i^{(n)} - \mathcal{R}_i^{(n-1)} \right) \exp \left\{ -\frac{1}{2} \left( \frac{\boldsymbol{\varphi}_i - \boldsymbol{\varphi}_i^{(n)}}{\sigma^{(n)}} \right)^2 \right\} \right]$$

8: Set $n \leftarrow n + 1$; go to Step 3.

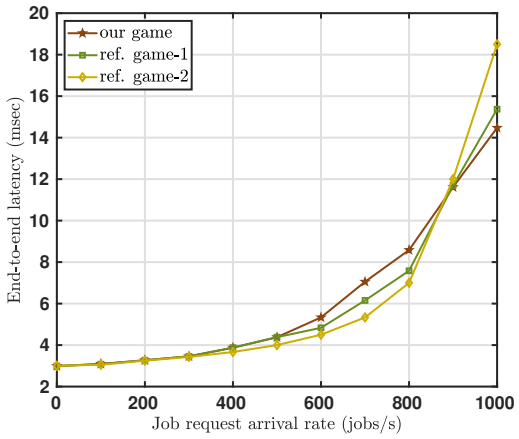
NE of the underlying continuous-kernel game. However, we computed a unique pure-strategy NE under both overloaded and under-loaded network conditions in Section 5.4. Therefore, our proposed algorithm with update rule (5.25) will always converge to the pure-strategy Nash equilibrium of the underlying non-cooperative load balancing game. □

Although, it is ensured that our proposed algorithm converges to the NE of the load balancing game for all competing cloudlets in both overloaded and under-loaded conditions, but we can *speed up the convergence rate of the algorithm* several times more by scaffolding our understanding about the underlying load balancing game. We observe that whenever some cloudlet is in under-loaded condition, i.e., $(t_{ui} + \frac{1}{\mu_{ii} - \lambda_i}) < D_Q$, its NE strategy is not to offload any job requests to its neighbouring cloudlets. Thus, all cloudlets can update their PDFs accordingly without exploring many job request offloading strategies as long as the under-load condition persists. In addition to this, during overload condition, i.e., $(t_{ui} + \frac{1}{\mu_{ii} - \lambda_i}) \geq D_Q$, each cloudlet will offload only a portion of the received job requests and try to shift the peak of

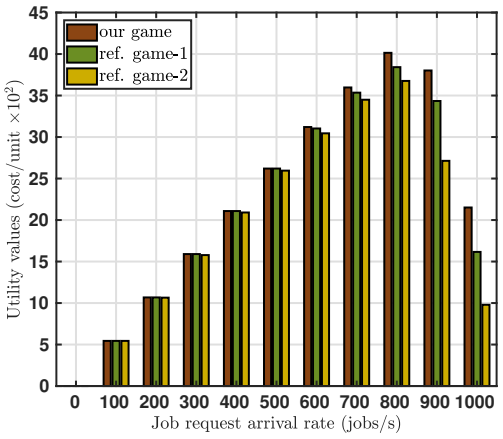
the PDF $f_{ij}(\varphi_{ij})$ around the pure strategy NE solution $\varphi^*_{ij} = 1 - \frac{1}{\lambda_i}\left[\mu_{ii} - \frac{1}{D_Q - t_{ui}}\right]$, such that it can meet $D_Q$ with the rest of the job requests by itself. Hence, the corresponding search spaces can be reduced accordingly. However, as each cloudlet is unaware about the load condition of its neighbouring cloudlets, an occasional feedback mechanism is required from the neighbouring cloudlets to indicate that whenever a cloudlet is completely or partially unable to process the job requests received from its neighbours. This implies that when $i^{\text{th}}$ cloudlet offloads $\varphi_{ij}\lambda_i$ job requests to $j^{\text{th}}$ cloudlet, it indicates that only $\hat{\psi}_{ij}\lambda_i$ jobs are processed, where $0 \leq \hat{\psi}_{ij} < \varphi_{ij}$. Therefore, in such cases, the $i^{\text{th}}$ cloudlet updates the PDF by using $\hat{\psi}_{ij}$ in (5.25) instead of $\varphi_{ij}$. Furthermore, when $i^{\text{th}}$ cloudlet is under-loaded and receives job requests from multiple cloudlets but can partially process the job requests i.e., $(t_{ui} + \frac{1}{\mu_{ii} - \lambda_i}) < D_Q$ but $(t_{uj} + \frac{1}{\mu_{ii} - \lambda_i - \sum_{i \neq j} \varphi_{ji}\lambda_j} + t_{ji}) \geq D_Q$ the remaining job processing capacity of the $i^{\text{th}}$ cloudlet, $(\mu_{ii} - \lambda_i - \frac{1}{D_Q - t_{uj} - t_{ji}})$ is distributed according to the ratio of $\varphi_{ji}\lambda_j$. The proposed algorithm is summarised in Algorithm 5.2.

## 5.7 Results and Discussions

In this section, we investigate various behavioural aspects of the proposed load balancing strategy through numerical evaluations. For this purpose, we consider a set of 10 neighbouring cloudlets from same as well as different service providers. In this work, we consider average processing rate $\mu_{ii}$ varies within 1000-1500 jobs/s and incoming job request to each cloudlet $\lambda_i$ varies within 0-15000 jobs/s. The QoS latency target considered is $D_Q = 10$ msec. Due to the wireless and TDM-PON interfaces between mobile users and cloudlets, we consider the average value of $t_{ui}$ as 2 msec. The intermediate transmission latency between neighbouring cloudlets $t_{ij}$ varies within 0.5-1 msec. The optimal values of proportionality price factors $\Omega_1$, $\Omega_2$, and $\Omega_3$ can be determined by studying the market equilibrium conditions for providing cloud-based services [182]. In actual practice, sometimes the proper price factors are also determined by applying the "multiple criteria decision-making theory" [183]. However, in this work we arbitrarily choose $\Omega_1 = 5 \times 10^3$, $\Omega_2 = 8 \times 10^2$, and $\Omega_3 = 1 \times 10^4$, such that our necessary game design condition $\Omega_2 \geq$




(a) $\mu_{ii}$ = 1000 jobs/s and $\lambda_i$ variance = 0-400 jobs/s, $\forall i \in \mathcal{C}$.

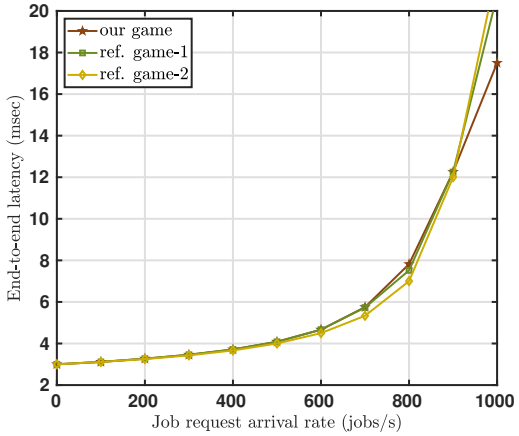



(b) $\mu_{ii}$ = 1000 jobs/s and $\lambda_i$ variance = 0-400 jobs/s, $\forall i \in \mathcal{C}$.

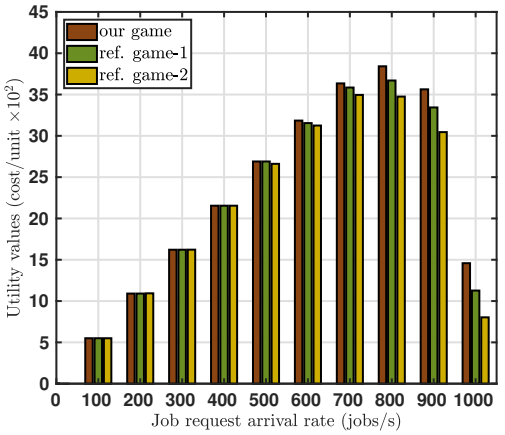
Figure 5.6: Comparison of end-to-end latency and utility values of competing cloudlets with high variance (0-400 jobs/s) in incoming job request arrival rates among neighbouring cloudlets with our proposed game and other baseline games.

$\Omega_3 \left[\max\{t_{ui}\} + \frac{1}{\max\{\mu_{ii}\}} + \max\{t_{ij}\} - D_Q\right], \forall i, j \neq i \in \mathcal{C}$ is satisfied. Moreover, for our gradient projection algorithm to compute NE of the load balancing game, we choose a step size $\omega = 0.1$, and a tolerance limit $\epsilon = 10^{-4}$.

In Fig. 5.6a, we compare average end-to-end latency of all the participating cloudlets against job request arrival rate with our currently proposed game and games proposed in [83] (labelled as "ref. game-1") and [84] (labelled as "ref. game-2"), respectively. In this case, we consider a high variance in job request arrival rates among under and overloaded cloudlets (within 0-400 jobs/s) and the service rates of all the cloudlets are $\mu_{ii}$ = 1000 jobs/s. We see that the ref. game-1 performs best when the load condition is low or moderate as it always tries to minimise end-to-end latency. Under these conditions, our proposed game as well as ref. game-2 performs slightly poorer as these models do not require the cloudlets to offload anything. However, ref. game-2 allows the cloudlets to offload job requests after reaching a certain threshold in incoming job requests and their latency performance begins to improve. Nonetheless, under high load condition, when all the cloudlets become sufficiently overloaded, our game also allows the cloudlets to strategically offload some job requests and latency performance becomes relatively better than ref. game-1 and ref. game-2. This is because it is ensured in our game that all the




(a) $\mu_{ii}$ = 1000 jobs/s and $\lambda_i$ variance = 0-200 jobs/s, $\forall i \in \mathcal{C}$.




(b) $\mu_{ii}$ = 1000 jobs/s and $\lambda_i$ variance = 0-200 jobs/s, $\forall i \in \mathcal{C}$.

Figure 5.7: Comparison of end-to-end latency and utility values of competing cloudlets with moderate variance (0-200 jobs/s) in incoming job request arrival rates among neighbouring cloudlets with our proposed game and other baseline games.

under-loaded cloudlets meet the QoS latency target $D_Q$. The over-loaded cloudlets may exceed $D_Q$, but they are allowed to offload job requests to the maximum extent possible. On the contrary, ref. game-1 becomes infeasible in high load condition as some of the cloudlets start to violate explicit latency constraints and ref. game-2 appears to overload the under-loaded cloudlets by uncontrolled offloading of the job request. Next, Fig. 5.6b shows a comparison among average utility values of all the participating cloudlets against job request arrival rate with our game, ref. game-1, and ref. game-2. It is clear that with our game, the average economic utility values of the cloudlets are relatively better than both ref. game-1 and ref. game-2 under all the network scenario.

Similarly, Fig. 5.7a shows a comparison between average end-to-end latency performance and Fig. 5.7b shows a comparison among average utility values of all the participating cloudlets against job request arrival rate with our game, ref. game-1, and ref. game-2. In this case, we consider a moderate variance in job request arrival rates (within 0-200 jobs/s) among under-loaded and overloaded cloudlets and the service rates of all the cloudlets are $\mu_{ii}$ = 1000 jobs/s. Note that both the plots show similar behaviour as in Fig. 5.6a and Fig. 5.6b. However, since the difference between under-loaded and overloaded cloudlets in job request arrival rates is lower

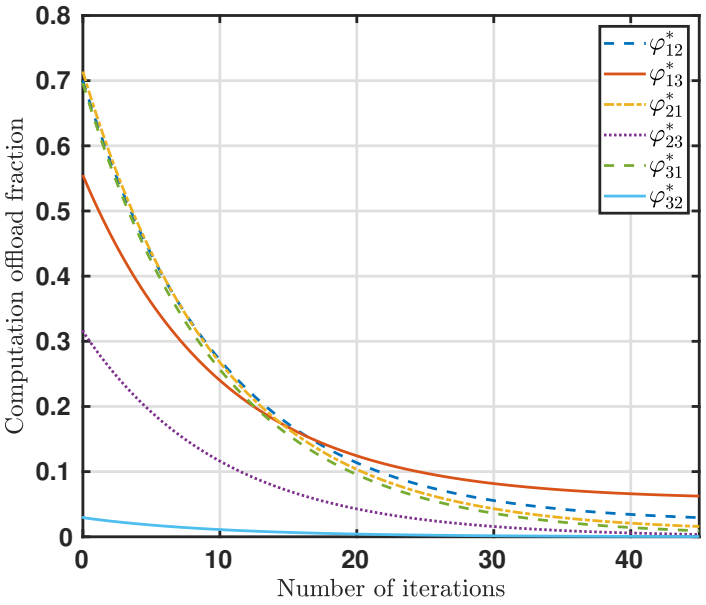



Figure 5.8: Computation offload decision of cloudlets versus (parallel update) iterations achieved using Algorithm 5.1. We consider three cloudlets with $\mu_{ii} = 1000$ jobs/s and $\lambda_1 = 970$ jobs/s, $\lambda_2 = 820$ jobs/s, and $\lambda_3 = 700$ jobs/s.

than the previous case, the scope for overloaded cloudlets to offload job requests is also lower. Hence, the average end-to-end latency overshoots to a relatively higher value in an overload condition and the average utility gained are also lower.

In Fig. 5.8, we present the convergence rate of Algorithm 5.1 with three neighbouring cloudlets (for not making the figure unnecessarily overcrowded). We consider an average processing rate $\mu_{ii} = 1000$ jobs/s for each cloudlet and their respective incoming job requests are $\lambda_1 = 970$ jobs/s, $\lambda_2 = 820$ jobs/s, and $\lambda_3 = 700$ jobs/s. In this case, Cloudlet-1 is overloaded but Cloudlet-2 and Cloudlet-3 are under-loaded. Hence, Cloudlet-1 offloads certain fraction of its total incoming job requests to the under-loaded cloudlets, i.e., $\varphi_{12} = 0.0219$ and $\varphi_{13} = 0.0569$. As Cloudlet-2 is the more loaded than Cloudlet-3, the computation offload fraction from Cloudlet-1 to Cloudlet-3 is more than that of Cloudlet-2.

In Fig. 5.9, we observe the convergence properties of the proposed reinforcement learning algorithm. We consider two neighbouring cloudlets, Cloudlet-1 and Cloudlet-2 with intermediate transmission latency $t_{ij} = 1$ msec, trying to meet a

QoS requirement of $D_Q = 10$ msec. We also consider that $\mu_{11} = \mu_{22} = 1000$ jobs/s and $\lambda_1 = 970$ jobs/s, $\lambda_2 = 800$ jobs/s, such that Cloudlet-1 is overloaded, but Cloudlet-2 is under-loaded. Therefore, to meet a QoS latency of $D_Q = 10$ msec, Cloudlet-1 needs to offload $0.098 \times \lambda_1$ job requests to Cloudlet-2, whereas Cloudlet-2 does not need to offload anything (from analytical solution). From the PDFs of both the cloudlets also, we observe that the most preferable decision for Cloudlet-2 is not to offload and Cloudlet-1 prefers to offload around 8-9% of its total incoming job requests. As we choose the learning and spreading parameters as $\Theta = 0.9$ and $\sigma = 0.01$, respectively, we see that Algorithm 5.2 converges to the expected utility and reward values for both the cloudlets in nearly 1000 iterations. Note that, instead of searching over the whole strategy space, we considered only the most likely strategies that the cloudlets should possibly consider by using our understanding from the underlying load balancing game. Moreover, by using these techniques, Algorithm 5.2 can perform 100% accurately when all cloudlets are under-loaded without much exploration.

It is interesting to note that in the previously considered scenario, even faster convergence to NE is possible by increasing the value of learning rate parameter $\Theta$,

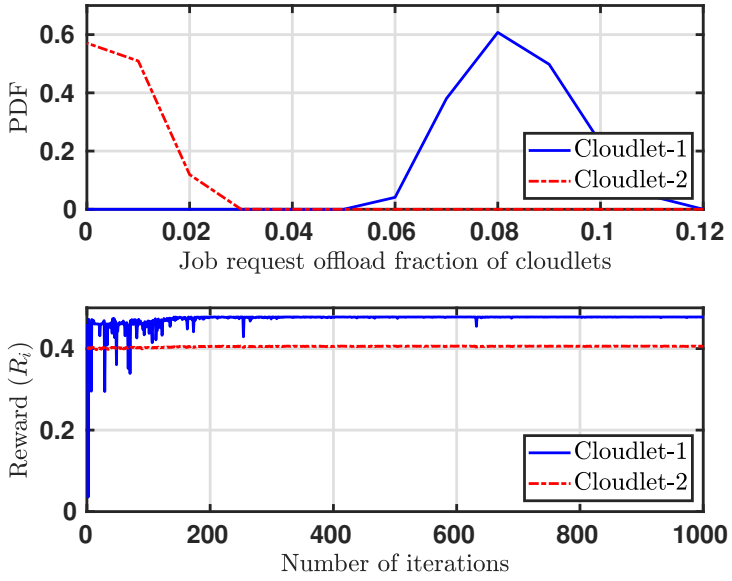



Figure 5.9: Independently learning NE of the non-cooperative load balancing game with Algorithm 5.2 by two cloudlets with $\mu_{11} = \mu_{22} = 1000$ jobs/s, $\lambda_1 = 970$ jobs/s, $\lambda_2 = 800$ jobs/s, and $\Theta = 0.9$, $\sigma = 0.01$.

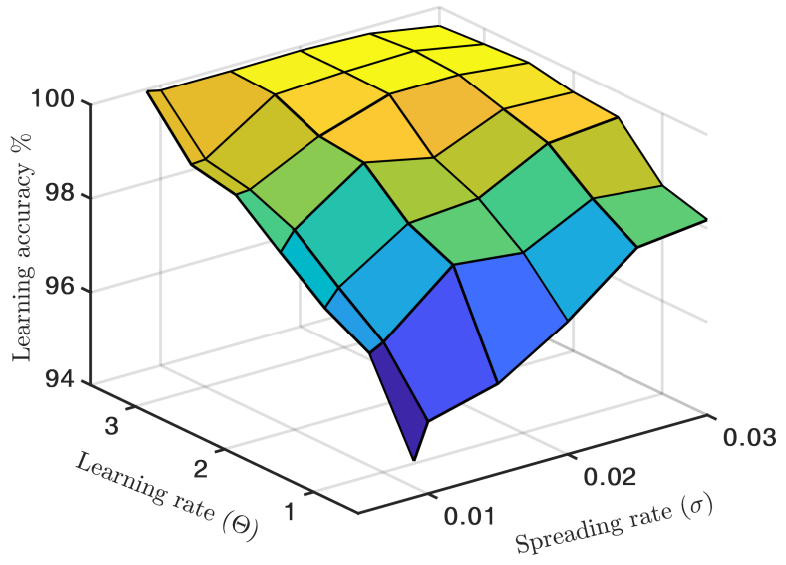



Figure 5.10: Average NE learning accuracy of Algorithm 5.2 over 1000 time-slots of duration 10 ms against variation of learning rate parameter $\Theta$ and spreading rate parameter $\sigma$.

but the learning process may become unstable. Therefore, to study the performance of Algorithm 5.2, we plot the average learning accuracy within 1000 iterations in Fig. 5.10 against variation of $\Theta$ and $\sigma$. As we consider 1000 iterations, i.e., 1000 time-slots of duration 10 msec (same as $D_Q$), hence it is expected that received job request rates remain stationary at least for 10 seconds. From this plot, we observe that the NE learning accuracy increases with $\Theta$, but if we simultaneously increase $\sigma$ also, then accuracy performance starts to slightly decrease as exploration increases.

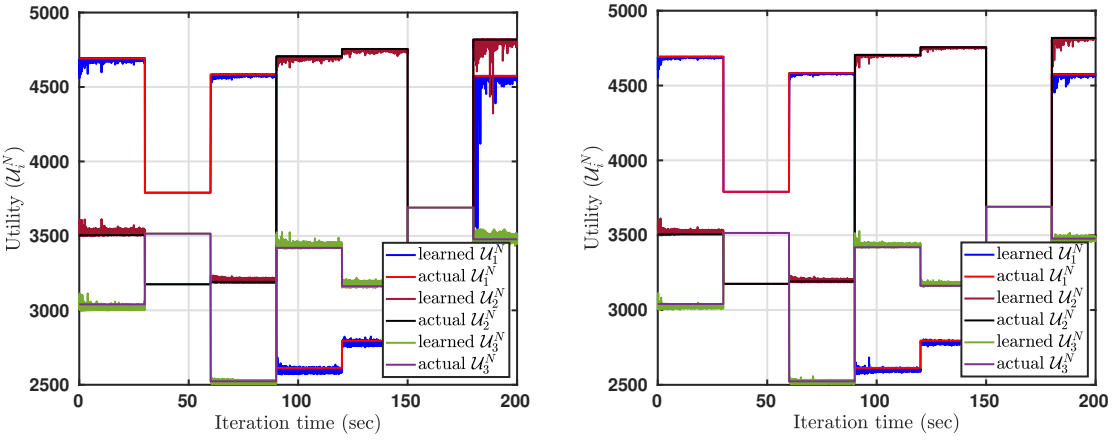
Now, we consider a scenario where three competing cloudlets are present with $\mu_{11} = \mu_{22} = \mu_{33} = 1000$ jobs/s and intermediate transmission latencies $t_{12} = t_{21} = 0.5$ msec, $t_{23} = t_{32} = 0.7$ msec, and $t_{31} = t_{13} = 0.9$ msec. Moreover, the job request arrival rates $\lambda_i$ to each of these cloudlets randomly change every 30 sec and hence, the utilities of each $i^{\text{th}}$ cloudlet $\mathcal{U}_i^N$ changes accordingly. Firstly, Fig. 5.11a presents a comparison between the learned utilities and actual utilities (by solving $\mathcal{P}$) with a small learning rate $\Theta = 0.9$ and big spreading rate $\sigma = 0.03$. Thus, we can observe that Algorithm 5.2 takes some time to learn the actual utility values after the $\lambda_i$ values change and some oscillation persists due to more exploration. Secondly, Fig. 5.11b presents another comparison between the learned utilities and actual utilities

with a big learning rate $\Theta = 4.0$ and small spreading rate $\sigma = 0.01$. With these parameters, we see that Algorithm 5.2 learns the actual utility values relatively faster after $\lambda_i$ values change, but there is less oscillation afterwards.

From the previous results, we found that it is essential to choose the learning rate and spreading rate parameters in such a way so that a proper balance between exploration and exploitation is maintained against the stationarity time of job request arrival rates. Thus, in Fig. 5.12 we plot the average NE learning accuracy against the stationarity time of the job request arrival rate. We vary the stationarity time from 5 sec to 150 sec and also tune $\Theta$ from 0.5 to 3 with $\sigma = 0.01$ in Fig. 5.12a, and tune $\sigma$ from 0.009 to 0.030 with $\Theta = 0.9$ in Fig. 5.12b. We observe in general, that the performance of our proposed Algorithm 5.2 increases as the stationarity of the job request arrival rate increases because the algorithm is given more time-slots to exploit and explore the search space.

## 5.8 Conclusions

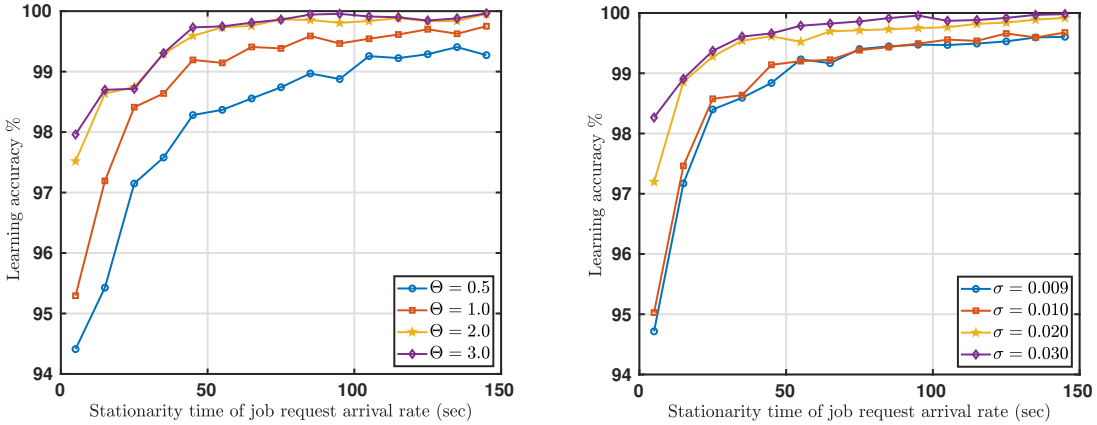
In this chapter, we have proposed a novel economic and non-cooperative game-theoretic model for low-latency applications among multiple competitive cloudlets




(a) Average NE learning accuracy with $\Theta = 0.9$ and $\sigma = 0.03$.

(b) Average NE learning accuracy with $\Theta = 4.0$ and $\sigma = 0.01$.

Figure 5.11: Average NE learning accuracy of Algorithm 5.2 for load balancing game among three cloudlets with 30 second stationarity time of job request arrival rates ($\lambda_i$) to all cloudlets.




(a) Average NE learning accuracy with $\sigma = 0.01$.

(b) Average NE learning accuracy with $\Theta = 0.9$.

Figure 5.12: Average NE learning accuracy of Algorithm 5.2 against stationarity time of job request arrival rates ($\lambda_i$) to all cloudlets with variations of learning rate $\Theta$ and spreading rate $\sigma$.

from same as well as different service providers. We have designed a very efficient method to solve the game and have proposed a very fast-converging gradient projection algorithm to compute the pure-strategy NE computation offload strategy among multiple cloudlets. Moreover, we have proposed a centralised incentive mechanism that computes the unique NE load balancing strategies of the cloudlets under the supervision of a neutral mediator. This mechanism ensures that the truthful revelation of private information to the mediator is a weakly-dominant strategy for both the under-loaded and overloaded cloudlets. We have also shown that our proposed game allows the cloudlets to maximise their utilities better than some recently proposed game-theoretic load balancing frameworks. Followed by we have designed a distributed continuous-action reinforcement learning automata-based algorithm to facilitate the competing cloudlets to learn their NE job request offload strategies independently, without exchanging any control information with neighbouring cloudlets. We have improved the convergence rate of the proposed reinforcement learning algorithm by adapting the basic characteristics of the underlying game. Through extensive simulation, we have shown the dependency of NE learning accuracy of our proposed algorithm on learning rate and spreading rate parameters, i.e., exploration and exploitation and derived various seminal insights on the performance and adaptability of our proposed reinforcement algorithm for non-cooperative

load balancing against dynamic job request arrival process in both overloaded and under-loaded network conditions.

## 5.9 Appendices

### 5.9.1 Appendix A

**Load Balancing Problem among $N \geq 2$ Cloudlets**

The complete job request offloading strategy space of all cloudlets is defined as a matrix $\Phi = (\Phi_1^T, \Phi_2^T, \ldots, \Phi_N^T)^T \subset \mathbb{R}^{N \times N}$, where $\boldsymbol{\varphi}_i = (\varphi_{i1}, \varphi_{i2}, \ldots, \varphi_{iN}) \in \Phi_i \subset \mathbb{R}^N$, $\varphi_{ij} \in \Phi_{ij} = [0, 1] \subset \mathbb{R}$, and $\sum_{j=1}^{N} \varphi_{ij} = 1, \forall i \in \mathcal{C}$. Each $\varphi_{ij}$ denotes the fraction of job requests $i^{\text{th}}$ cloudlet offloads to its $j^{\text{th}}$ neighbouring cloudlet. Due to the non-homogeneous service rates of the neighbouring cloudlets, all the received job requests at each $i^{\text{th}}$ cloudlet are served with different service rates, e.g., $\varphi_{ii}\lambda_i = (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i$ jobs/s are served with service rate $\mu_{ii}$ job/s and $\sum_{j \neq i} \varphi_{ji}\lambda_j$ jobs/s are served with service rate $\mu_{ji}$ jobs/s. Therefore, to compute the overall processing and queuing latency of the received job requests at each cloudlet, we need to use a multi-dimensional Markov chain for $M/M/1$ queues [184]. For this analysis, we make the following assumptions:

- Each $i^{\text{th}}$ cloudlet is in state $(m_{i1}, m_{i2}, \ldots, m_{iN})$ at each time-slot, where $m_{ij}$ denotes the independent job requests received from $j^{\text{th}}$ cloudlet.

- The detailed balance equations for each $i^{\text{th}}$ cloudlet hold for all the pairs of adjacent states $(m_{i1}, \ldots, m_{ij}, \ldots, m_{iN})$ and $(m_{i1}, \ldots, m_{ij} + 1, \ldots, m_{iN})$,

$$\begin{aligned} &\varphi_{ji}\lambda_j \mathbb{P}_i(m_{i1}, \ldots, m_{ij}, \ldots, m_{iN}) \\ &\qquad = \mu_{ji}\mathbb{P}_i(m_{i1}, \ldots, m_{ij} + 1, \ldots, m_{iN}), \forall i, j \in \mathcal{C}. \end{aligned}$$

- The stationary state probability distribution of each $i^{\text{th}}$ cloudlet can be expressed in the following *product form*,

$$\mathbb{P}_i(m_{i1}, m_{i2}, \ldots, m_{iN}) = \mathbb{P}_{i1}(m_{i1})\mathbb{P}_{i2}(m_{i2}) \ldots \mathbb{P}_{iN}(m_{iN}).$$

- The number of job requests received from $j^{\text{th}}$ neighbouring cloudlet by each $i^{\text{th}}$ cloudlet follows the *geometric distribution*, $\mathbb{P}_i(m_{ij}) = \rho_{ji}^{m_{ij}}(1-\rho_{ji})$, where $\rho_{ji} = \frac{\varphi_{ji}\lambda_j}{\mu_{ji}}$.

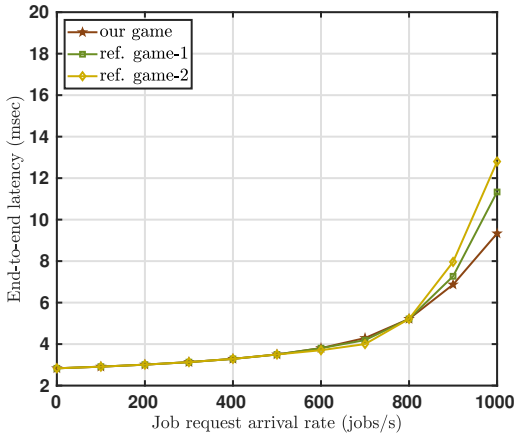
With the above mentioned assumptions, we derive the following closed form expression for average number of job requests served by each $i^{\text{th}}$ cloudlet:

$$\begin{aligned}\mathbb{M}_i &= \sum_{m_{i1}=0}^{\infty}\sum_{m_{i2}=0}^{\infty}\cdots\sum_{m_{iN}=0}^{\infty}\left\{(m_{i1}+\cdots+m_{iN})\prod_{j=1}^{N}\rho_{ji}^{m_{ij}}(1-\rho_{ji})\right\} \\ &= \frac{\sum_{j=1}^{N}\left\{\rho_{ji}\prod_{k=1,k\neq j}^{N}(1-\rho_{ki})\right\}}{\prod_{j=1}^{N}(1-\rho_{ji})}. \end{aligned} \tag{5.26}$$

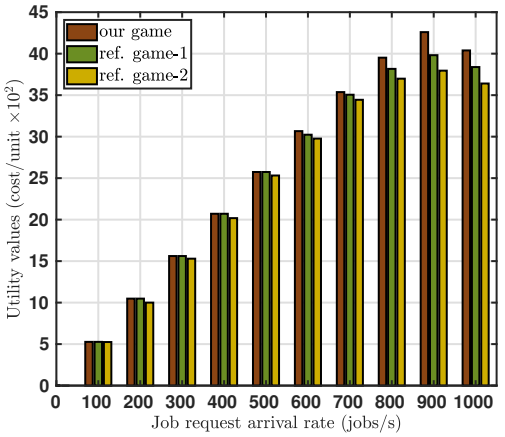
Therefore, by using Little's theorem [184] and (5.26), we get the overall processing and queuing latency of the job requests at $i^{\text{th}}$ cloudlet as follows:

$$\begin{aligned}\mathbb{T}_i(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) &= \frac{1}{\varphi_{ii}\lambda_i + \sum_{j\neq i}\varphi_{ji}\lambda_j}\left(\frac{\sum_{j=1}^{N}\left\{\rho_{ji}\prod_{k=1,k\neq j}^{N}(1-\rho_{ki})\right\}}{\prod_{j=1}^{N}(1-\rho_{ij})}\right) \\ &= \frac{1}{\varphi_{ii}\lambda_i + \sum_{j\neq i}\varphi_{ji}\lambda_j}\left(\frac{\sum_{j=1}^{N}\left\{\varphi_{ji}\lambda_j\prod_{k=1,k\neq j}^{N}(\mu_{ki}-\varphi_{ki}\lambda_k)\right\}}{\prod_{j=1}^{N}(\mu_{ji}-\varphi_{ij}\lambda_j)}\right). \end{aligned} \tag{5.27}$$

Now, we present a comparison between the average end-to-end latency performance in Fig. 5.13a and comparison among average utility values of all the participating cloudlets in Fig. 5.13b with our game, ref. game-1, and ref. game-2. In this case, we keep the job request arrival rates of all the cloudlets equal but vary their service rates. Thus, to calculate the processing latency of each cloudlet, we need to use the expression in (5.27) and this creates a very general load balancing scenario. At first, we consider a high variance (1000-1500 jobs/s) of service rates among neighbouring cloudlets and observe similar patterns of the graphs as before, but the end-to-end latency and utility values are relatively better as the service rates of some cloudlets are much higher than the average job request arrival rate. We present similar graphs in Fig. 5.14a and in Fig. 5.14b, but we consider a moderate variance (1000-1200 jobs/s) of service rates among neighbouring cloudlets. As a consequence, the latency and utility performance is slightly poorer than the previous

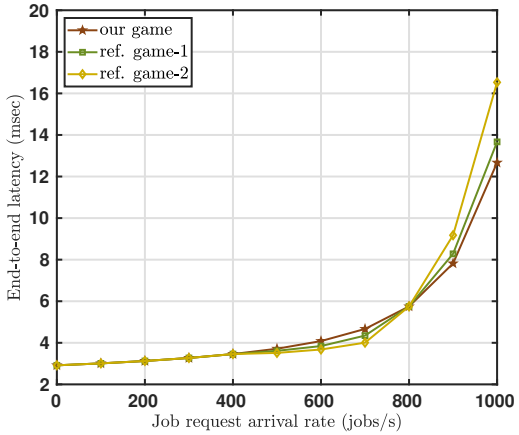



(a) $\mu_{ii}$ variance = 1000-1500 jobs/s and $\lambda_i$ is same $\forall i \in \mathcal{C}$.

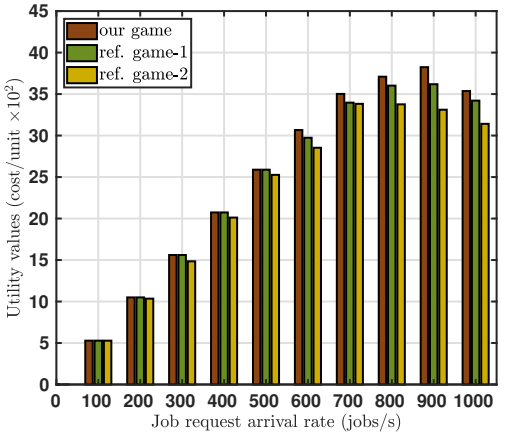



(b) $\mu_{ii}$ variance = 1000-1500 jobs/s and $\lambda_i$ is same $\forall i \in \mathcal{C}$.

Figure 5.13: Comparison of end-to-end latency and utility values of competing cloudlets with high variance (1000-1500 jobs/s) in service rates among neighbouring cloudlets and same job request arrival rates with our proposed game and other baseline games.




(a) $\mu_{ii}$ variance = 1000-1200 jobs/s and $\lambda_i$ is same $\forall i \in \mathcal{C}$.




(b) $\mu_{ii}$ variance = 1000-1200 jobs/s and $\lambda_i$ is same $\forall i \in \mathcal{C}$.

Figure 5.14: Comparison of end-to-end latency and utility values of competing cloudlets with moderate variance (1000-1200 jobs/s) in service rates among neighbouring cloudlets and same job request arrival rates with our proposed game and other baseline games.

graphs as the under-loaded cloudlets have lesser room to receive job requests from overloaded cloudlets.

### 5.9.2 Appendix B

#### A Detailed Proof of Lemma 4.1

*Proof.* Recall that in the game $\Gamma$, the utility function of each $i^{\text{th}}$ cloudlet is denoted by $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}), \forall i \in \mathcal{C}$ is defined as follows:

$$\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \sum_{j=1, j\neq i}^{N} \gamma_{ji}\varphi_{ji}\frac{\lambda_j}{\mu_{ii}} - \Omega_2 \sum_{j=1, j\neq i}^{N} \gamma_{ij}\varphi_{ij}\frac{\lambda_i}{\mu_{jj}}$$
$$- \Omega_3 \left[ \left(1 - \sum_{j=1, j\neq i}^{N} \varphi_{ij}\right) \frac{\lambda_i}{\mu_{ii}} \max\left\{0, \left(t_{ui} + \frac{1}{\mu_{ii} - (1 - \sum_{j\neq i}\varphi_{ij})\lambda_i - \sum_{j\neq i}\varphi_{ji}\lambda_j} - D_Q\right)\right\} \right.$$
$$\left. + \sum_{j=1, j\neq i}^{N} \varphi_{ji}\frac{\lambda_j}{\mu_{ii}} \max\left\{0, \left(t_{uj} + \frac{1}{\mu_{ii} - (1 - \sum_{j\neq i}\varphi_{ij})\lambda_i - \sum_{j\neq i}\varphi_{ji}\lambda_j} + t_{ji} - D_Q\right)\right\} \right]. \tag{5.28}$$

From this definition, we can observe that when $i^{\text{th}}$ cloudlet is able to meet the QoS target latency $D_Q$, or $(t_{uj} + \frac{1}{\mu_{ii} - (1 - \sum_{j\neq i}\varphi_{ij})\lambda_i - \sum_{j\neq i}\varphi_{ji}\lambda_j} + t_{ji} - D_Q) \leq 0$, then the utility function can be interpreted as an affine function as follows:

$$\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \sum_{j=1, j\neq i}^{N} \gamma_{ji}\varphi_{ji}\frac{\lambda_j}{\mu_{ii}} - \Omega_2 \sum_{j=1, j\neq i}^{N} \gamma_{ij}\varphi_{ij}\frac{\lambda_i}{\mu_{jj}}. \tag{5.29}$$

Nonetheless, when $i^{\text{th}}$ cloudlet is unable to satisfy the QoS target latency $D_Q$, or $(t_{ui} + \frac{1}{\mu_{ii} - (1 - \sum_{j\neq i}\varphi_{ij})\lambda_i - \sum_{j\neq i}\varphi_{ji}\lambda_j} - D_Q) \geq 0$, then the utility function should be interpreted as a non-linear function as follows:

$$\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}) = \Omega_1 \frac{\lambda_i}{\mu_{ii}} + \Omega_2 \sum_{j=1, j\neq i}^{N} \gamma_{ji}\varphi_{ji}\frac{\lambda_j}{\mu_{ii}} - \Omega_2 \sum_{j=1, j\neq i}^{N} \gamma_{ij}\varphi_{ij}\frac{\lambda_i}{\mu_{jj}}$$
$$- \Omega_3 \left[ \left(1 - \sum_{j=1, j\neq i}^{N} \varphi_{ij}\right) \frac{\lambda_i}{\mu_{ii}} \left\{t_{ui} + \frac{1}{\mu_{ii} - (1 - \sum_{j\neq i}\varphi_{ij})\lambda_i - \sum_{j\neq i}\varphi_{ji}\lambda_j} - D_Q\right\} \right.$$
$$\left. + \sum_{j=1, j\neq i}^{N} \varphi_{ji}\frac{\lambda_j}{\mu_{ii}} \left\{t_{uj} + \frac{1}{\mu_{ii} - (1 - \sum_{j\neq i}\varphi_{ij})\lambda_i - \sum_{j\neq i}\varphi_{ji}\lambda_j} + t_{ji} - D_Q\right\} \right]. \tag{5.30}$$

If (5.30) represents a concave function, then the sufficient condition is that its Hessian matrix should be a positive semi-definite matrix [154]. However, we find that both the diagonal and non-diagonal elements of this matrix are equal, as summarised

below with $i \neq j, k \in \mathcal{C}$:

$$\frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{ij}^2} = \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{ij} \partial \varphi_{ik}} = \frac{-2\Omega_3 \lambda_i^2}{(\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \varphi_{ji} \lambda_j)^3} \leq 0. \tag{5.31}$$

Clearly, this implies that the Hessian matrix of (5.29) is not a semi-definite matrix and we need to extend our analysis by evaluating the bordered Hessian matrix [158]. Note that the $r^{\text{th}}$ order bordered Hessian matrix of $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}), \forall i \in \mathcal{C}$, where $r = 1, 2, \ldots, N$ is written as follows:

$$\mathcal{H}_B = \begin{bmatrix} 0 & \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i1}} & \cdots & \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{ir}} \\ \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i1}} & \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i1}^2} & \cdots & \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i1} \partial \varphi_{ir}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{ir}} & \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{ir} \partial \varphi_{i1}} & \cdots & \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{ir}^2} \end{bmatrix}, \tag{5.32}$$

whose the bordered elements are written as follows:

$$\frac{\partial \mathcal{U}_i^N}{\partial \varphi_{ij}} = -\Omega_2 \gamma_{ij} \frac{\lambda_i}{\mu_{jj}} + \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[ t_{ui} + \frac{\mu_{ii}}{(\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i - \sum_{j \neq i} \varphi_{ji} \lambda_j)^2} - D_Q \right]. \tag{5.33}$$

Now, we observe that the determinant values of the bordered Hessian matrix (5.32), denoted by $D_r$, are negative when $r$ is odd and are positive when $r$ is even, $\forall r = 1, 2, \ldots, N$, as follows:

$$D_1 = 0 - \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i1}} \right)^2 = - \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i1}} \right)^2 \leq 0, \tag{5.34}$$

$$\begin{aligned} D_2 &= 0 - \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i1}} \right) \left[ \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i1}} \right) \left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i2}^2} \right) - \left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i1} \partial \varphi_{i2}} \right) \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i2}} \right) \right] \\ &\quad + \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i2}} \right) \left[ \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i1}} \right) \left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i2} \partial \varphi_{i1}} \right) - \left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i1}^2} \right) \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i2}} \right) \right] \\ &= - \left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i1} \partial \varphi_{i2}} \right) \left[ \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i1}} \right) - \left( \frac{\partial \mathcal{U}_i^N}{\partial \varphi_{i2}} \right) \right]^2 \geq 0, \end{aligned} \tag{5.35}$$

and so on. Note that, the inequalities in (5.34)-(5.35) hold true, because $\left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i1}^2} \right) = \left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i2}^2} \right) = \left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i1} \partial \varphi_{i2}} \right) = \left( \frac{\partial^2 \mathcal{U}_i^N}{\partial \varphi_{i2} \partial \varphi_{i1}} \right)$ (from (5.31)) and $[\mu_{ii} - (1 - \sum_{j \neq i} \varphi_{ij})\lambda_i -$

$\sum_{j\neq i}\varphi_{ji}\lambda_j] \geq 0, \forall i, j \neq i \in \mathcal{C}$. Therefore, we can conclude that (5.30) is a "quasi-concave function" of $\boldsymbol{\varphi}_i$ [158], and in general, we can conclude that the utility functions of each $i^{\text{th}}$ cloudlet $\mathcal{U}_i^N(\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_{-i}), \forall i \in \mathcal{C}$ are quasi-concave functions of $\boldsymbol{\varphi}_i$. Hence proved. □

### 5.9.3 Appendix C

**Detailed Calculations to Find the Unique NE of the Non-cooperative Load Balancing Game**

In this section, we show detailed calculations for finding the NE of the non-cooperative load balancing game against different network load conditions.

**Case-1:** $\left[\left(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i}\right) < D_Q, \left(t_{uj} + \frac{1}{\mu_{jj}-\lambda_j}\right) < D_Q\right]$

In this case, both the cloudlets are under-loaded and their Lagrangian function can be written from $\boldsymbol{\mathcal{P}^u}$ as follows:

$$\begin{aligned}\mathcal{L}_i^u = \Omega_1\frac{\lambda_i}{\mu_{ii}} + \Omega_2\gamma_{ji}\varphi_{ji}\frac{\lambda_j}{\mu_{ii}} - \Omega_2\gamma_{ij}\varphi_{ij}\frac{\lambda_i}{\mu_{jj}} + \alpha_i\varphi_{ij} + \beta_i(1-\varphi_{ij}) \\ - \xi_i\left(t_{uj} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} + t_{ji} - D_Q\right),\end{aligned} \tag{5.36}$$

where $\alpha_i, \beta_i, \xi_i \geq 0$ are Lagrange multipliers corresponding to the respective constraints. Thus, the necessary first-order KKT conditions (FOC) $\forall i, j \neq i \in \mathcal{C}$ are derived as follows:

$$\frac{\partial\mathcal{L}_i^u}{\partial\varphi_{ij}} = -\Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}} + \alpha_i - \beta_i + \xi_i\left[\frac{\lambda_i}{(\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j)^2}\right] = 0. \tag{5.37}$$

In addition to this, the complementary slackness conditions (CSC) $\forall i, j \neq i \in \mathcal{C}$ are written as follows:

$$\alpha_i\varphi_{ij} = 0, \tag{5.38}$$

$$\beta_i(1-\varphi_{ij}) = 0, \tag{5.39}$$

$$\xi_i\left(t_{uj} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} + t_{ji} - D_Q\right) = 0. \tag{5.40}$$

Note that both the cloudlets have sufficient computational resources to meet the QoS latency target and hence, $(t_{uj} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j} + t_{ji} - D_Q) < 0$. Therefore, from (5.40) we get $\xi_i^* = \xi_j^* = 0$. At first, we consider that $0 < \varphi_{ij} < 1$, $0 < \varphi_{ij} < 1$ and hence, $\alpha_i = \beta_i = \alpha_j = \beta_j = 0$ from (5.38)-(5.39). However, with these values of Lagrange multipliers, we cannot solve the equations from (5.37) and find any values for $\varphi_{ij}$ and $\varphi_{ji}$. Thus, our initial considerations cannot lead to a valid NE solution.

Again, if we consider $\varphi_{ij} = \varphi_{ji} = 1$, then $\alpha_i = \alpha_j = 0$ and $\beta_i > 0$, $\beta_j > 0$ from (5.38)-(5.39). With these values, solve the equations from (5.37) and find that $\beta_i = -\Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}}$, $\beta_j = -\Omega_2\gamma_{ji}\frac{\lambda_j}{\mu_{ii}}$. Clearly, both of these values are negative and hence, $\varphi_{ij} = \varphi_{ji} = 1$ cannot be a valid NE solution.

Next, we consider $\varphi_{ij} = 1$, $\varphi_{ji} = 0$ such that $\alpha_i = 0$, $\beta_i > 0$ and $\alpha_j > 0$, $\beta_j = 0$ from (5.38)-(5.39). With these values, we solve the equations from (5.37) and find that $\beta_i = -\Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}}$ and $\alpha_j = \Omega_2\gamma_{ji}\frac{\lambda_j}{\mu_{ii}}$. As $\beta_i$ is negative, hence $\varphi_{ij} = 1$, $\varphi_{ji} = 0$ cannot be a valid NE solution. Similarly, if we consider $\varphi_{ij} = 0$, $\varphi_{ji} = 1$, then we obtain $\alpha_i = \Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}}$, $\beta_i = 0$, $\alpha_j = 0$, and $\beta_j = -\Omega_2\gamma_{ji}\frac{\lambda_j}{\mu_{ii}}$. This is also not a valid NE solution as $\beta_j$ is a negative number.

Finally, we consider $\varphi_{ij} = \varphi_{ji} = 0$ such that $\alpha_i > 0$, $\beta_i = 0$ and $\alpha_j > 0$, $\beta_j = 0$ from (5.38)-(5.39). Solving the equations from (5.37) with these values, we find that $\alpha_i = \Omega_2\gamma_{ij}\frac{\lambda_i}{\mu_{jj}}$ and $\alpha_j = \Omega_2\gamma_{ji}\frac{\lambda_j}{\mu_{ii}}$. Therefore, none of the Lagrange multipliers are negative and the "unique NE solution" corresponding to this case is $\varphi_{ij}^* = \varphi_{ji}^* = 0$. Intuitively, this implies that the cloudlets do not offload any job requests.

**Case-2:** $\left[\left(t_{ui} + \frac{1}{\mu_{ii}-\lambda_i}\right) \geq D_Q, \left(t_{uj} + \frac{1}{\mu_{jj}-\lambda_j}\right) \geq D_Q\right]$

In this case, both the cloudlets are overloaded and hence, the corresponding Lagrangian function from $\boldsymbol{\mathcal{P}^o}$:

$$\begin{aligned}
\mathcal{L}_i^o = {} & \Omega_1\frac{\lambda_i}{\mu_{ii}} + \Omega_2\gamma_{ji}\varphi_{ji}\frac{\lambda_j}{\mu_{ii}} - \Omega_2\gamma_{ij}\varphi_{ij}\frac{\lambda_i}{\mu_{jj}} \\
& - \Omega_3\left[(1-\varphi_{ij})\frac{\lambda_i}{\mu_{ii}}\left(t_{ui} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j} - D_Q\right)\right. \\
& \left.+\varphi_{ji}\frac{\lambda_j}{\mu_{ii}}\left(t_{uj} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j} + t_{ji} - D_Q\right)\right] \\
& + \alpha_i\varphi_{ij} + \beta_i(1-\varphi_{ij}) + \xi_i\left(t_{ui} + \frac{1}{\mu_{ii}-(1-\varphi_{ij})\lambda_i-\varphi_{ji}\lambda_j} - D_Q\right). \qquad (5.41)
\end{aligned}$$

Again, the necessary FOC and CSC $\forall i, j \neq i \in \mathcal{C}$ are derived as follows:

$$\frac{\partial \mathcal{L}_i^o}{\partial \varphi_{ij}} = -\Omega_2 \gamma_{ij} \frac{\lambda_i}{\mu_{jj}} + \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[ t_{ui} + \frac{\mu_{ii}}{(\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j)^2} - D_Q \right] + \alpha_i - \beta_i - \xi_i \left[ \frac{\lambda_i}{(\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j)^2} \right] = 0, \tag{5.42}$$

$$\alpha_i \varphi_{ij} = 0, \tag{5.43}$$

$$\beta_i (1 - \varphi_{ij}) = 0, \tag{5.44}$$

$$\xi_i \left( t_{ui} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q \right) = 0. \tag{5.45}$$

As both the cloudlets are overloaded, therefore, in this case, $(t_{ui} + \frac{1}{\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q) > 0$ and (5.45) yields $\xi_i^* = \xi_j^* = 0$. At first, we consider that $0 < \varphi_{ij} < 1$, $0 < \varphi_{ij} < 1$ and hence, $\alpha_i = \beta_i = \alpha_j = \beta_j = 0$ from (5.43)-(5.44). With these values of Lagrange multipliers, we find a system of non-linear equation as follows:

$$-\Omega_2 \gamma_{ij} \frac{\lambda_i}{\mu_{jj}} + \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[ t_{ui} + \frac{\mu_{ii}}{(\mu_{ii} - (1-\varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j)^2} - D_Q \right] = 0, \tag{5.46}$$

$$-\Omega_2 \gamma_{ji} \frac{\lambda_j}{\mu_{ii}} + \Omega_3 \frac{\lambda_j}{\mu_{jj}} \left[ t_{uj} + \frac{\mu_{jj}}{(\mu_{jj} - (1-\varphi_{ji})\lambda_j - \varphi_{ij}\lambda_i)^2} - D_Q \right] = 0. \tag{5.47}$$

However, this is an inconsistent system of equations that does not have any solution and hence, we cannot find any valid NE solution. Again, if we consider $\varphi_{ij} = \varphi_{ji} = 1$, then $\alpha_i = \alpha_j = 0$ and $\beta_i > 0$, $\beta_j > 0$ from (5.38)-(5.39). With these values, solve the equations from (5.37) and find that,

$$\beta_i = -\Omega_2 \gamma_{ij} \frac{\lambda_i}{\mu_{jj}} + \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[ t_{ui} + \frac{\mu_{ii}}{(\mu_{ii} - \lambda_j)^2} - D_Q \right], \tag{5.48}$$

$$\beta_j = -\Omega_2 \gamma_{ji} \frac{\lambda_j}{\mu_{ii}} + \Omega_3 \frac{\lambda_j}{\mu_{jj}} \left[ t_{uj} + \frac{\mu_{jj}}{(\mu_{jj} - \lambda_i)^2} - D_Q \right]. \tag{5.49}$$

We observe that both of these values are negative based on the necessary game design condition $\Omega_2 \geq \Omega_3 \left[ \max\{t_{ui}\} + \frac{1}{\max\{\mu_{ii}\}} + \max\{t_{ij}\} - D_Q \right], \forall i, j \neq i \in \mathcal{C}$ mentioned in Proposition 5.1. Therefore, $\varphi_{ij} = \varphi_{ji} = 1$ cannot be a valid NE solution.

Next, we consider $\varphi_{ij} = 1$, $\varphi_{ji} = 0$ such that $\alpha_i = 0$, $\beta_i > 0$ and $\alpha_j > 0$, $\beta_j = 0$ from (5.38)-(5.39). With these values, we solve the equations from (5.37) and find that,

$$\beta_i = -\Omega_2 \gamma_{ij} \frac{\lambda_i}{\mu_{jj}} + \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[ t_{ui} + \frac{\mu_{ii}}{(\mu_{ii} - \lambda_j)^2} - D_Q \right], \tag{5.50}$$

$$\alpha_j = \Omega_2 \gamma_{ji} \frac{\lambda_j}{\mu_{ii}} - \Omega_3 \frac{\lambda_j}{\mu_{jj}} \left[ t_{uj} + \frac{\mu_{jj}}{(\mu_{jj} - \lambda_i)^2} - D_Q \right]. \tag{5.51}$$

Although $\alpha_j$ is positive, but $\beta_i$ is negative and hence, $\varphi_{ij} = 1$, $\varphi_{ji} = 0$ cannot be a valid NE solution. Similarly, if we consider $\varphi_{ij} = 0$, $\varphi_{ji} = 1$, then we obtain the following:

$$\alpha_i = \Omega_2 \gamma_{ij} \frac{\lambda_i}{\mu_{jj}} - \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[ t_{ui} + \frac{\mu_{ii}}{(\mu_{ii} - \lambda_j)^2} - D_Q \right], \tag{5.52}$$

$$\beta_j = -\Omega_2 \gamma_{ji} \frac{\lambda_j}{\mu_{ii}} + \Omega_3 \frac{\lambda_j}{\mu_{jj}} \left[ t_{uj} + \frac{\mu_{jj}}{(\mu_{jj} - \lambda_i)^2} - D_Q \right]. \tag{5.53}$$

This implies that $\varphi_{ij} = 0$, $\varphi_{ji} = 1$ is also not a valid NE solution as $\beta_j$ is a negative number. Finally, we consider $\varphi_{ij} = \varphi_{ji} = 0$ such that $\alpha_i > 0$, $\beta_i = 0$ and $\alpha_j > 0$, $\beta_j = 0$ from (5.38)-(5.39). Solving the equations from (5.37) with these values, we find the following:

$$\alpha_i = \Omega_2 \gamma_{ij} \frac{\lambda_i}{\mu_{jj}} - \Omega_3 \frac{\lambda_i}{\mu_{ii}} \left[ t_{ui} + \frac{\mu_{ii}}{(\mu_{ii} - \lambda_j)^2} - D_Q \right], \tag{5.54}$$

$$\alpha_j = \Omega_2 \gamma_{ji} \frac{\lambda_j}{\mu_{ii}} - \Omega_3 \frac{\lambda_j}{\mu_{jj}} \left[ t_{uj} + \frac{\mu_{jj}}{(\mu_{jj} - \lambda_i)^2} - D_Q \right]. \tag{5.55}$$

Clearly, both the Lagrange multipliers are positive and hence, the "unique NE solution" corresponding to this case is $\varphi_{ij}^* = \varphi_{ji}^* = 0$, which implies that both the cloudlets do not to offload any job requests to each other.

**Case-3:** $\left[ \left( t_{ui} + \frac{1}{\mu_{ii} - \lambda_i} \right) \geq D_Q, \left( t_{uj} + \frac{1}{\mu_{jj} - \lambda_j} \right) < D_Q \right]$

In this case, we consider that $i^{\text{th}}$ cloudlet is overloaded but $j^{\text{th}}$ cloudlet is underloaded. Thus, $i^{\text{th}}$ cloudlet needs to offload some job requests to $j^{\text{th}}$ cloudlet to meet the QoS target latency $D_Q$, as long as $j^{\text{th}}$ cloudlet does not exceed $D_Q$. This implies that $0 < \varphi_{ij}^* < 1$ and $\varphi_{ji}^* = 0$. Moreover, we get $(t_{ui} + \frac{1}{\mu_{ii} - (1 - \varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} - D_Q) = 0$, $\xi_i^* > 0$ and $(t_{uj} + \frac{1}{\mu_{ii} - (1 - \varphi_{ij})\lambda_i - \varphi_{ji}\lambda_j} + t_{ji} - D_Q) \leq 0$, $\xi_j^* = 0$. Considering the FOC

(5.42) and CSC (5.43)-(5.45) for $i^{\text{th}}$ cloudlet and FOC (5.37) and CSC (5.38)-(5.40) for $j^{\text{th}}$ cloudlet, we get the following unique NE solution:

$$\varphi_{ij}^{*} = 1 - \frac{1}{\lambda_i}\left[\mu_{ii} - \frac{1}{D_Q - t_{ui}}\right] \leq \frac{1}{\lambda_i}\left[\mu_{jj} - \lambda_j - \frac{1}{D_Q - t_{ui} - t_{ij}}\right]. \tag{5.56}$$

Along with this, we get the Lagrange multiplier values $\alpha_i^* = 0$, $\beta_i^* = 0$, $\alpha_j^* = 0$, and $\beta_j^* = 0$. It is interesting to note from (5.56) that for the overloaded $i^{\text{th}}$ cloudlet, the computation offloading decision is not entirely controlled by itself. As long as the under-loaded $j^{\text{th}}$ cloudlet can process the entire extra load from $i^{\text{th}}$ cloudlet, $\varphi_{ij}^* = 1 - \frac{1}{\lambda_i}\left[\mu_{ii} - \frac{1}{D_Q - t_{ui}}\right]$ is acceptable. However, when $j^{\text{th}}$ cloudlet cannot process the entire extra load, then the offload fraction is not allowed to exceed the upper-bound, and hence, $\varphi_{ij}^* = \frac{1}{\lambda_i}\left[\mu_{jj} - \lambda_j - \frac{1}{D_Q - t_{ui} - t_{ij}}\right]$. Therefore, from the above analysis we showed that we can compute a unique NE for our proposed load balancing game among competing cloudlets.

# Chapter 6

# Edge-Artificial Intelligence for Human-to-Machine Applications over Fibre-Wireless Access Networks

> "We've all heard that we have to learn from our mistakes, but I think it's more important to learn from successes. If you learn only from your mistakes, you are inclined to learn only errors."
>
> Norman Vincent Peale

## 6.1 Introduction

The Tactile Internet (TI) envisions a telecommunication network that supports and empowers human users to immersively control and manipulate both real and virtual remote things or machines [88]. Within the next few decades, the Internet-of-Things (IoT) will play an essential role in our daily lives as well as in industrial manufacturing (Industry 4.0). In this technical evolution process of IoT, enabling the interaction among machines and humans over the Internet appears as a fundamental requirement. Recently, the IEEE P1918.1 standards working group is created for

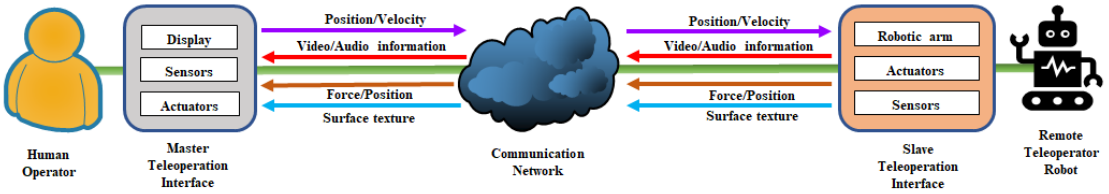



Figure 6.1: A schematic diagram of bidirectional haptic communication system with a human operator at the master teleoperation interface and a teleoperator robot at the slave teleoperation interface.

standardisation of TI and some primary use cases considered are teleoperation, immersive virtual/augmented reality, and industrial automotive control. The authors of [15] indicated that there is an overlap in the concepts and technologies related to IoT, 5G and TI, such as ultra-low-latency (1-10 ms) and high-reliability (99.999%) communication channels, high-bandwidth, and secure network infrastructure with integrated edge-computing intelligence. Most of these applications involve remote immersion that is achieved through bilateral exchange of multi-modal information, e.g., a combination of audio, video, and haptic information over the Internet with reaction latencies of 100 ms, 10 ms, and 1 ms, respectively [88].

The latency requirements of haptic communications are dependent on the intensity of control dynamics involved in the application scenario. For applications with "low dynamics" like remote surgery and "intermediate dynamics" like a collaboration of users in virtual or real environments, it is possible to involve human users in a closed global control loop between the human users and the teleoperators, thus leading to human-to-machine (H2M) applications [185]. A typical H2M communication system consists of a "master" and a "slave" subsystem that interact over a network, as shown in Fig. 6.1. A human operator at the master subsystem end generates and transmits control signals i.e., position and velocity data through motion sensors, over a communication channel. In return, the slave subsystem receives that data and response back with force reflection/feedback of the remote environment, in the form of kinesthetic or vibrotactile force feedback data [186].

Note that the ultra-low latency communication between local master-slave pairs residing within the same wireless coverage area can be easily supported by advanced wireless technology such as 5G, WiFi, or Bluetooth. Nonetheless, data between

remote master-slave pairs need to traverse through the optical front/back-haul segments [187]. Thus, it becomes challenging to meet the stringent latency requirements of H2M applications without being limited by the master-slave distance. This challenge can be overcome by preempting the haptic feedback from slave devices [128] and transmit this feedback from an intermediate artificial intelligence (AI)-enhanced H2M server. The AI-enhanced H2M server forecasts the haptic samples much before the control signals reach to the slave device, which reduces the end-to-end closed-loop latency to a great extent. Therefore, With this approach, the master receives the haptic feedback samples much quicker than the round-trip time of the master-slave pair and thus, remote H2M communications that meet stringent latency requirements can be deployed without being limited by the master-slave distance.

Recently, the authors of [133] proposed an edge sample forecast module that relies on historical data to forecast haptic feedback to the master. However, with this model, user experience can be significantly impacted by dynamic H2M applications where a certain diversity is present among various haptic feedback samples. Thus, in this paper, we propose an Event-based HAptic feedback SAmple Forecast (EHASAF) module that exploits a two-stage AI model consisting of an artificial neural network (ANN) unit followed by a reinforcement learning (RL) unit to forecast haptic feedback to the master. The ANN-based supervised learning unit decides when the master controller should start receiving haptic feedback samples and the RL unit ensures that the proper values of the haptic feedback samples are delivered. Thus, the master controller device receives proper haptic feedback samples within the expected quality of experience (QoE) time ($D_Q$). Moreover, the transmission of redundant haptic feedback samples is also reduced to a great extent.

The rest of this chapter is organised as follows. Section 6.2 provides a background review on supervised machine learning and artificial neural network. In Section 6.3, we discuss our considered network architecture to facilitate remote H2M applications. In Section 6.4, the details of our experimental setup, the ANN-based binary classifier unit and the RL unit are presented. In Section 6.5, the performance of the proposed EHASAF module is evaluated. Finally, in Section 6.6, our primary observations and achievements by using the proposed EHASAF module are summarised.

## 6.2 Background on Classification by Supervised Machine Learning Methods

Classification is a supervised learning method in machine learning and statistics, in which a set of given data is used as input for training or learning and new observations are classified with this learning. The objective of a classifier is to learn a mapping from a set of given inputs $x$ to outputs $y$, where $y \in \{1, 2, \ldots, C\}$, with $C$ being the number of classes [188]. The input space is thus divided into decision regions, the frontiers of which are referred to as decision boundaries or decision surfaces. If $C = 2$, then we call it "binary classification" (in this case, we can consider $y \in \{0, 1\}$) and if $C > 2$, then we call it "multi-class classification". Some common examples of classification problems are e-mail spam filtering, speech recognition, handwriting recognition, biometric identification, document classification, to name a few. Next, we briefly discuss a few commonly used classification methods.

### Naive Bayes Classifier

This is a classification method based on Bayes's theorem, which assumes that predictors have independence. In a nutshell, the existence of a specific feature in a class is considered to be unrelated to any other feature. Also if these features are mutually related or contingent on other features, all of these properties contribute to the likelihood independently. The model of Naive Bayes is simple to construct and particularly useful for large sets of data. Naive Bayes is known for its efficiency along with the simplicity, which sometimes outperforms highly sophisticated classification methods.

### Nearest Neighbor

The $k$-nearest-neighbors algorithm is a non-parametric supervised classification algorithm. A set of labelled points are given as inputs to the classifier, which uses them to learn how to label other points. For any new point, the classifier checks other points that are labelled nearest to this new point (the nearest neighbours)

and is classified by assigning the most frequent label among the $k$ training samples nearest to this point.

### Logistic Regression

It is a statistical approach in which one or more independent variables are used to evaluate a data set to derive a conclusion. The outcome is calculated by a dichotomous variable (with only two possible outcomes). Logistic regression aims to determine the most suitable model for describing the relation between the dichotomous interest feature and a set of independent (predictor or explanatory) variables. This is better than other classifications such as the nearest neighbour, as the factors which lead to classification are also quantitatively clarified.

### Decision Tree

Decision tree builds a tree structure in the context of classification or regression models. This splits the data set into smaller and smaller subsets while simultaneously creating a related decision tree. The final result is a tree with leaf nodes and decision nodes. There are two or more branches of a decision node and a leaf node is a division or decision. The highest decision node in a tree or the best predictor is known as the root node. All categorical and numerical data can be treated by decision trees.

### Artificial Neural Networks

A neural network is made up of layered units (neurons) that transform an input vector into a specific output, as shown in Fig. 6.2. Every unit receives an input, applied a function to it (sometimes not linear) and transferred the output to the next layer. Networks are commonly described as feed-forward (a unit feeds its output to all units on the next layer), but input on the previous level is not available. Signals from one unit to another are weighted, and these weightings are adjusted during the training stage to adjust the neural network to the specific problem [189].

In this case, the problem of learning is formulated as a $f$ loss index minimisation. In general, the loss index consists of an error and regularisation terms. The error

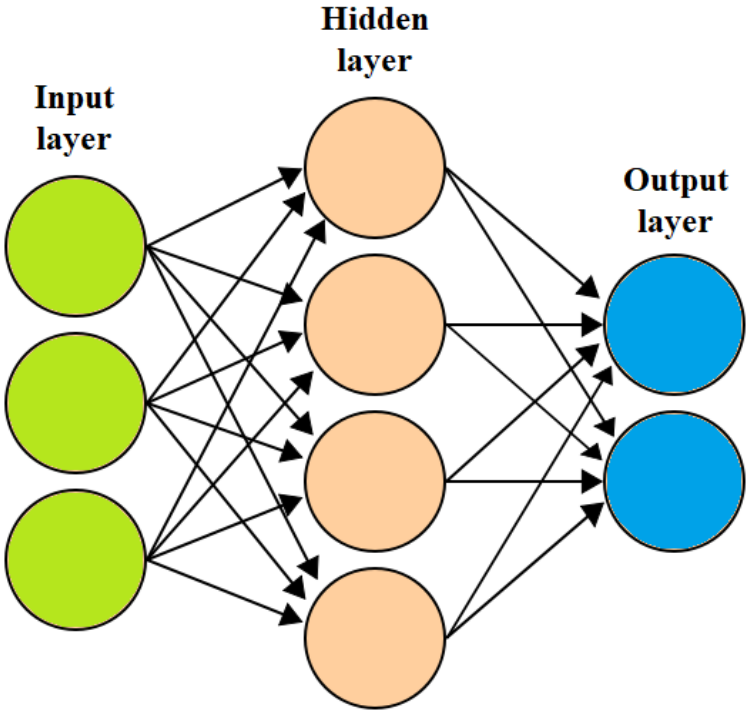



Figure 6.2: A schematic diagram of artificial neural networks.

term determines how the data set relates to the neural network. By controlling the effective complexity of the neural network, the regularisation concept is used to avoid over-fitting. The loss function depends on the adaptive parameters of the neural network (biases and synaptic weights). We can combine all these variables into a single $n$-dimensional weight vector of $\boldsymbol{w}$ and the loss function $f$ is minimum at $\boldsymbol{w}^*$. We can compute the first and second-order derivatives of the loss function at any arbitrary point in the search space. The first-order derivatives are grouped into the gradient vector which can be written as,

$$\nabla_i f(\boldsymbol{w}) = \frac{\partial f}{\partial \boldsymbol{w}_i}, \forall i = 1, \ldots, n.$$

Similarly, the second-order derivatives of the loss function can be grouped together in the Hessian matrix,

$$\boldsymbol{H}_{i,j} f(\boldsymbol{w}) = \frac{\partial^2 f}{\partial w_i \partial w_j}, \forall i, j = 1, \ldots, n.$$

In general, the loss function is a nonlinear function of the parameters. As a result,

closed training algorithms for the minima can not be found. Instead, we consider a search through the space parameter consisting of a succession of steps. At each step, the loss will decrease by adjusting the parameters of the neural network. In this way, to train a neural network, we start with a vector parameter (often randomly chosen). Then we generate a sequence of parameters so that each iteration of the algorithm reduces the loss function. The training algorithm will stop when a required condition, or criteria for stopping, is met. Some commonly used algorithms to train neural networks are gradient descent, Newton method, conjugate gradient algorithm, Quasi-Newton method, and Levenberg-Marquardt algorithm, to name a few [189].

## 6.3 Human-to-Machine Network Architecture

Fig. 6.3 illustrates an example of network architecture where different combinations of local and remote H2M master-slave pairs are connected over FiWi access networks. The wireless access points (WAP) are integrated with optical network units (ONUs) of a time division multiplexed (TDM) or wavelength division multiplexed (WDM) passive optical network (PON) as fibre backhaul network. The ONUs in a typical PON is situated at a distance of 10-20 km from the central office (CO) and the distance is nearly 100 km for long-reach PON. For local H2M teleoperation (shown by the purple circle), the intermediate distance between master and slave devices is very short, usually, a few metres and hence, wireless technologies like 5G new radio, WiFi, or Bluetooth can support them. However, when we consider remote H2M teleoperation scenarios like intra-PON master-slave devices (shown by the green circle) and inter-PON master-slave devices (shown by the pink circle), then relying on optical front/back-haul segments becomes essential [190].

It is important to note that the end-to-end propagation latency still presents a bottleneck for remote H2M teleoperation scenarios as light cannot travel faster than $2 \times 10^8$ m/s within an optical fibre. If each of the master and slave devices is connected to an ONU of a 100 km long-reach PON, then only the end-to-end propagation latency becomes 1 ms. Nonetheless, such remote H2M teleoperations can be made feasible with various AI-enhanced edge-computing servers to predict bandwidth demands or forecast haptic samples, indicated as “H2M server” in Fig.

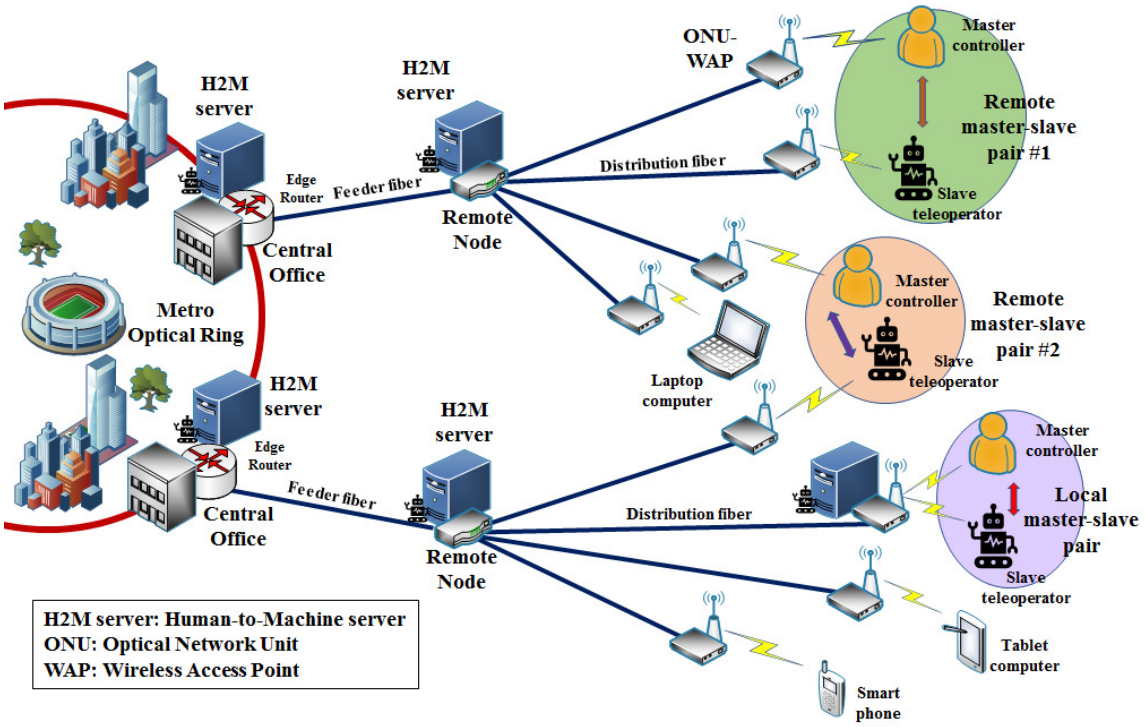



Figure 6.3: Local and remote teleoperation master-slave pairs using H2M servers over TDM or WDM-PON based FiWi networks.

6.3. These H2M servers can be installed at the network edge to support a group of ONUs, or at the remote node or CO, depending on given cost, bandwidth, and latency constraints.

## 6.4 System Model

In this section, we briefly describe our virtual reality (VR) based H2M experimental setup and the statistical characteristics of control and haptic feedback signals. Then we discuss the EHASAF module consisting of a supervised learning algorithm implemented by an ANN unit to predict if haptic feedback samples required to be forecasted, followed by an RL algorithm to forecast proper haptic feedback samples to the master device.

### 6.4.1 Experimental Setup and Network Statistics

In any H2M experimental setup, the hardware design parameters like sampling frequency, the number and type of sensors and actuators determine the volume of input

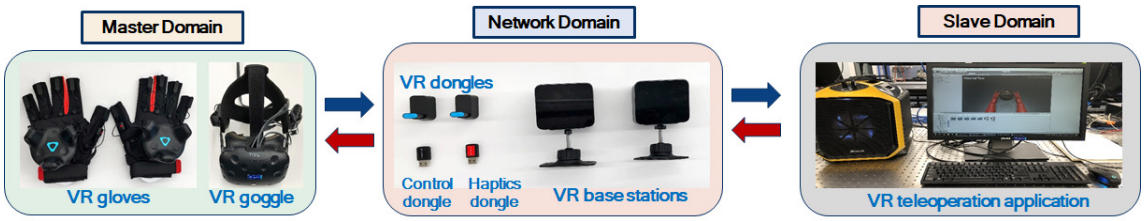
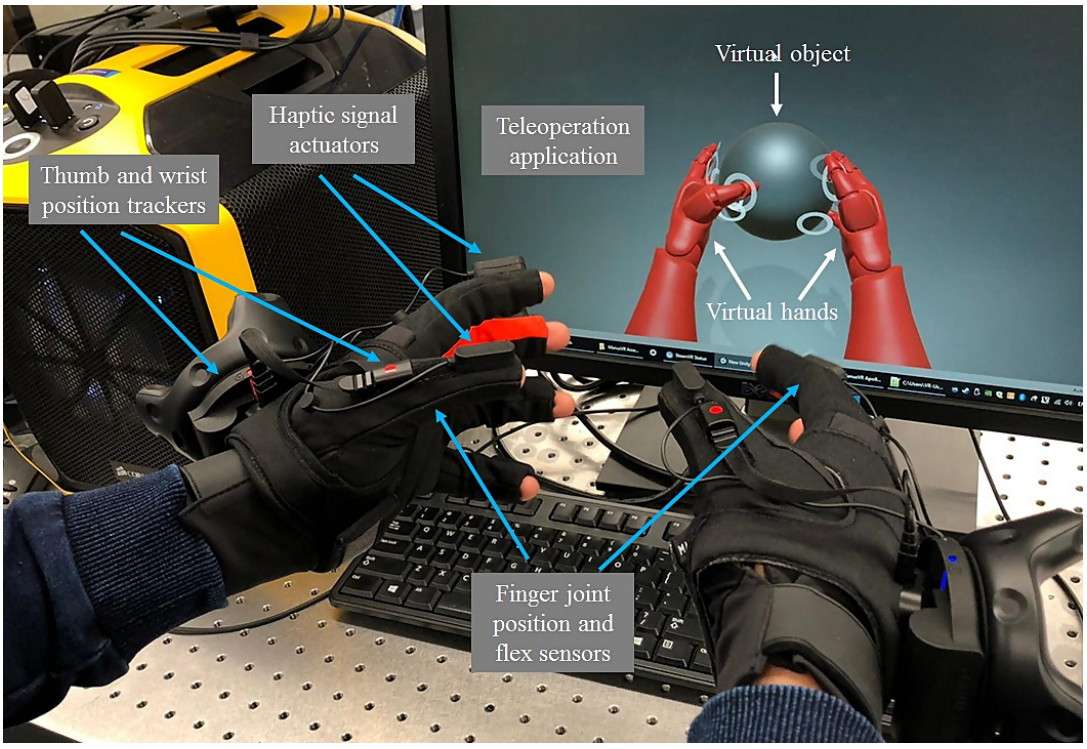


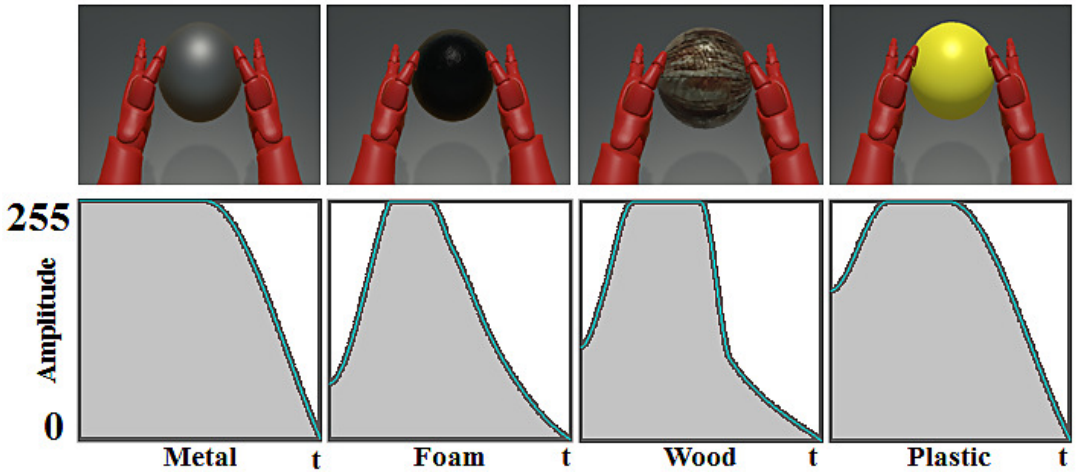

Figure 6.4: The hardware and software components of master, network interface, and slave domains for virtual reality based experimental setup.

and output data [191]. The slave haptic subsystem can either be a physical device interacting with a remote physical environment or a virtual pointer like a virtual hand, operating in a virtual environment. A key difference between physical and virtual environments is that the laws governing a physical environment are continuous, while a virtual one is discrete. Although it is not possible to reproduce a physical environment completely, simulated worlds have the only advantage of allowing connectivity with multiple users to communicate with each other in a hassle-free virtual space over a local network or the Internet in some situations. To study the characteristics of the control signals from a master device and the corresponding haptic signals from a slave device, we created a VR based H2M experimental setup. Various hardware and software components of this experimental setup are shown in Fig. 6.4. The master domain consists of VR gloves and VR google. Various wireless dongles like VR dongles, control signals dongle, haptic feedback dongle, and infrared base stations are the primary elements of the network domain. The slave domain is a desktop computer where the VR teleoperation application is executed.

In Fig. 6.5a, we show an instance of performing the VR based teleoperation experiment. Each of the VR gloves in the master domain has two orientation sensors on the thumb and wrist with 9 degrees-of-freedom, and five flexible sensors on five fingers for tracking movements and applied forces [192]. The sensor sampling rate is 200 Hz and the control signals are transmitted over a wireless interface to the computer where a VR application for touching a virtual ball (the slave device) is run. The wireless interface is a customised Bluetooth interface that can support a maximum master-slave distance of 30 metres [192]. Quaternion and Euler's angles are used to record the thumb, wrist, and finger joints orientation data [193]. Moreover, two flex sensors per finger record the normalised tension on each



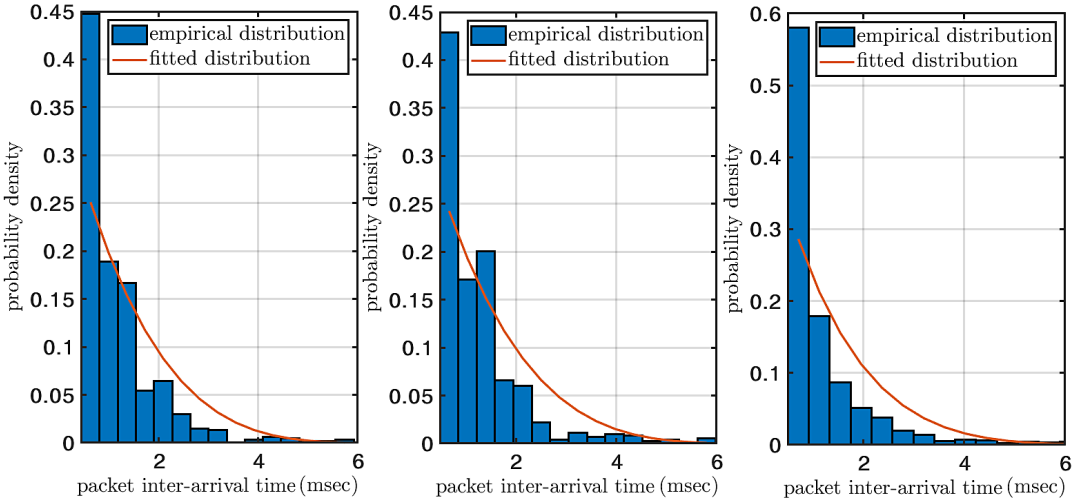

(a) Various components of VR based H2M teleoperation experimental setup.



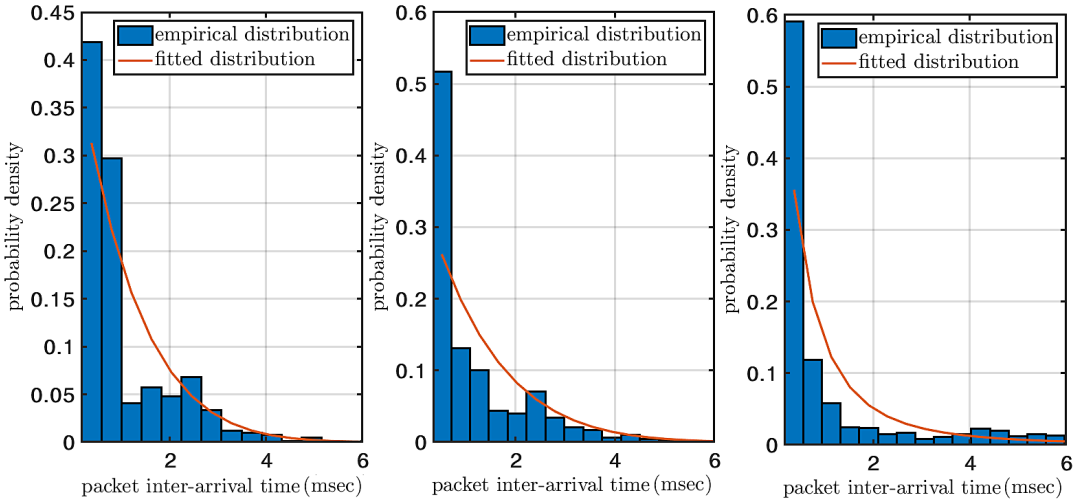

(b) Haptic feedback samples for different types of material.

Figure 6.5: Different components of VR based H2M experimental setup and haptic feedback samples corresponding to different types of material.

finger. Hence, each instance of control signal from either hand contains a total of $(4 \times 2) + (5 \times 5 \times 3) + (5 \times 2) = 93$ elements. When any finger touches the virtual ball and depending on the type of material of the ball, e.g., metal, foam, wood, or plastic, the VR application sends different haptic feedback samples to the haptic actuators of the corresponding finger. The haptic feedback samples, with amplitude values that lie within 0 to 255, as shown in Fig. 6.5b, are transmitted to the master




(a) Control signals for grab a ball, a cube, and a circular cube, respectively.




(b) Haptic feedback for grab a ball, a cube, and a circular cube, respectively.

Figure 6.6: Distribution fitting corresponding to empirical histograms of timestamps of control signals and haptic feedback from different tests.

device within 100 ms. The signal latency of the haptic feedback is 10 ms and the maximum force felt is 0.9 gram-force [192].

To characterise the traffic pattern of control signals and haptic feedback of VR based H2M teleoperation, we perform three different test scenarios in our experimental setup, e.g., grabbing a virtual ball, grabbing a virtual cube, and grabbing a virtual circular cube (cube with circular facets). The histograms obtained from the timestamps of control signals for all three test scenarios are shown in Fig. 6.6a. Similarly, the histograms obtained from the timestamps of haptic feedback for all three test scenarios are shown in Fig. 6.6b. Note that we scaled these histograms

from a mean of 10 ms to 1 ms for compatibility with standard H2M teleoperation QoE requirements. We observe that all these distributions well fit a generalised Pareto distribution [194]. Recall that the probability density function $f_X(\cdot)$ of a generalized Pareto distribution $X$ with shape parameter $k \neq 0$, scale parameter $\sigma$, and threshold parameter $\theta$ is given by:

$$f_X(x|k,\sigma,\theta) = \left(\frac{1}{\sigma}\right)\left(1 + k\frac{(x-\theta)}{\sigma}\right)^{-1-\frac{1}{k}}, \tag{6.1}$$

for $\theta < x$, if $k > 0$, or for $\theta < x < (\theta - \frac{\sigma}{k})$ if $k < 0$. For $k = 0$, the density becomes an *exponential distribution* as follows:

$$f_X(x|0,\sigma,\theta) = \left(\frac{1}{\sigma}\right)\exp\left(-\frac{(x-\theta)}{\sigma}\right), \tag{6.2}$$

for $\theta < x$. The optimal distribution parameter values in terms of goodness-of-fit, empirical and fitted means and variances are summarized in Table 6.1. We also ensure through *Chi-square goodness-of-fit (GoF)* test that the distribution fittings are accurate with a 5% significance level.

### 6.4.2 AI-based Haptic Feedback Sample Forecast Module

We consider the case in which H2M server is situated between the master and slave devices and hence, the network is divided into two logical segments, i.e., master-H2M server (MH) and H2M server-slave (HS), as shown in Fig. 6.7. We denote the latency between master and H2M server as $t_{MH}$ and the latency between the H2M server and slave as $t_{HS}$. Hence, the total end-to-end latency between control signal generation at the master device and reception of the corresponding haptic signal is $D_{MS} = 2 \times (t_{MH} + t_{HS})$. However, if $D_{MS} > D_Q$, then the user experience degrades. To overcome this issue, we install the proposed EHASAF module in the H2M server that acts as a proxy of the slave device.

**(a) EHASAF - ANN unit:** The first stage of the EHASAF module is an ANN unit corresponding to each of the thumb, index, middle, ring, and baby fingers of both hands (as shown in Fig. 6.8) and detects each fingers actions, i.e., whether it is going to touch the virtual object or not. A typical ANN can be viewed as

Table 6.1: Generalised Pareto distribution fitting corresponding to control signals and haptic feedback timestamp histograms

| | **Grab a ball** | | **Grab a cube** | | **Grab a circular cube** | |
|---|---|---|---|---|---|---|
| | **Control signal** | **Haptic feedback** | **Control signal** | **Haptic feedback** | **Control signal** | **Haptic feedback** |
| $k$ | -0.2017<br>[-0.2390, -0.1644] | -0.1145<br>[-0.1426, -0.08633] | -0.1999<br>[-0.2395, -0.1603] | -0.1100<br>[-0.1312, -0.08883] | -0.1422<br>[-0.1768, -0.1077] | +0.4618<br>[0.3904, 0.5332] |
| $\sigma$ | +0.001391<br>[0.001280, 0.001512] | +0.001150<br>[0.001099, 0.001204] | +0.001456<br>[0.001342, 0.001579] | +0.001306<br>[0.001251, 0.001364] | +0.001332<br>[0.001242, 0.001428] | +0.00073<br>[0.00067, 0.00079] |
| $\theta$ | 0 | 0 | 0 | 0 | 0 | 0 |
| **Histogram mean** | 0.001183615615615 | 0.001034569892473 | 0.001235497267759 | 0.001182086891942 | 0.001178701740812 | 0.001271528073916 |
| **Distribution mean** | 0.001158180137842 | 0.001032717179577 | 0.001213301782568 | 0.001176983051996 | 0.001166483139090 | 0.001356732115846 |
| **Histogram variance** | $5.9957417984 \times 10^{-07}$ | $5.9957417984 \times 10^{-07}$ | $7.00931861 \times 10^{-07}$ | $9.9095864475 \times 10^{-07}$ | $7.6875969728 \times 10^{-07}$ | $3.2054223432 \times 10^{-06}$ |
| **Distribution variance** | $9.5572064513 \times 10^{-07}$ | $8.6776193696 \times 10^{-07}$ | $1.05159692 \times 10^{-06}$ | $1.1354124842 \times 10^{-06}$ | $1.0592440674 \times 10^{-06}$ | $2.4129239902 \times 10^{-05}$ |
| **Chi-square GoF (5 %)** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

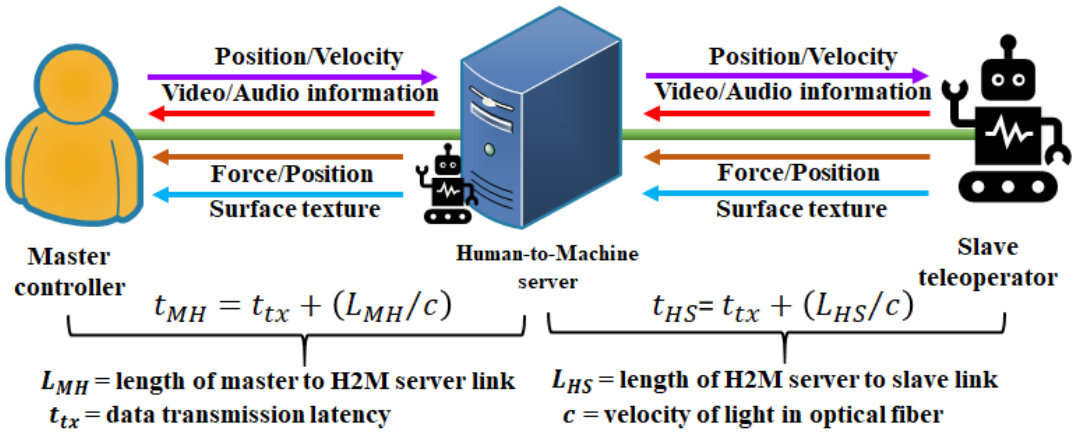



Figure 6.7: Logical links between master, H2M server, and slave devices.

*a universal approximator, which can approximate any linear/non-linear function to any arbitrary degree of accuracy* [195]. Therefore, an ANN can be used for function approximation, regression analysis, and data classification. Usually, there are multiple hidden layers in an ANN, consisting of a network of artificial neurons to represent a linear combination of certain parameterized nonlinear functions. During the training phase, the weights of the interconnections among the neurons are determined based on the trained data. Once the network is fully trained, the ANN can be used for producing a suitable output based on any input data. From our VR based H2M experiments, the wrist and thumb coordinates and the rotation and tension of each finger are considered as inputs to the ANN. The ANN corresponding to each finger uses supervised learning algorithms to act as a *binary classifier*. These ANNs try to minimize the mean-squared error (MSE) in the classification [196] and the outputs of each ANN are "touch" or "no-touch." Thus, the training data for each ANN corresponding to each finger can be considered a vector of $(5 \times 3) + 2 = 17$ elements and we denote it by $\boldsymbol{x}$. The output of the classifier is denoted by a variable $y \in \{\text{touch, no-touch}\}$ and $\boldsymbol{w}$ denotes the weight parameter vector. Therefore, for a set of input vectors $\{\boldsymbol{x}_n\}$, where $n = 1, \ldots, N$ and a corresponding set of labels $\{t_n\}$, we minimize the MSE function as follows:

$$\mathbb{E}(w) = \frac{1}{2} \sum_{n=1}^{N} \{y(\boldsymbol{x}_n, \boldsymbol{w}) - t_n\}^2 . \tag{6.3}$$

As our multi-layer ANN uses forward-propagation and backward-propagation

methods to implement this binary classifier, the time-complexity is given by $\mathcal{O}(n_0 n_1 + n_1 n_2 + \cdots + n_i n_{i+1} + \dots)$, where $n_i$ is the number of neurons in the $i^{\text{th}}$ layer [197]. This complexity can be further improved by some specialized training algorithms like the *Levenberg-Marquardt algorithm*, which does not compute the exact Hessian matrix but, computes the gradient vector and the Jacobian matrix [198]. With a cost function like MSE, that has the form of a sum of squares, the Hessian matrix can be approximated as $H = J^T J$ and the gradient can be computed as $g = J^T e$, where $J$ is the Jacobian matrix that contains first derivatives of the network errors with respect to the weights and biases, and $e$ is a vector of network errors. With this approximation of the Hessian matrix, the Levenberg-Marquardt algorithm reduces to the following Newton-like update rule:

$$x^{(k+1)} = x^{(k)} - [J^T J + \mu I]^{-1} J^T e. \tag{6.4}$$

When the scalar $\mu$ is equal to zero, then (6.4) is the same as Newtons method but, when $\mu > 0$, then (6.4) reduces to a gradient descent algorithm with a small step size. As Newtons method is very fast and highly accurate near an error minimum, we aim to shift towards Newtons method as soon as possible. To achieve this, we decrease $\mu$ after each successful step and increase after a step if there is an increase in the MSE cost function. In this way, the MSE cost function is always reduced at each iteration of the algorithm. Note that, this complexity is applicable only for the training or calibration phase and does not affect the run-time complexity of the system. If it is a no-touch event, then no immediate haptic feedback is required to be transmitted to the master. However, when any finger touches the object, the EHASAF module starts to generate haptic samples every $(D_Q - t_{MH})$ interval. Therefore, the master device receives a haptic feedback sample after every $D_Q$ interval, satisfying both user experience and network latency constraints.

**(b) EHASAF - RL unit:** If there are different types of material involved, it is important to correctly forecast the corresponding haptic feedback samples. Hence, we implement an RL unit that uses a customized *linear reward-inaction algorithm* for this purpose [199]. When a finger touch is detected by the ANN at $i^{\text{th}}$ time-slot, the RL units randomly chooses a material $(m_k^{(i)})$, where $k \in \{1, \dots, K\}$, and

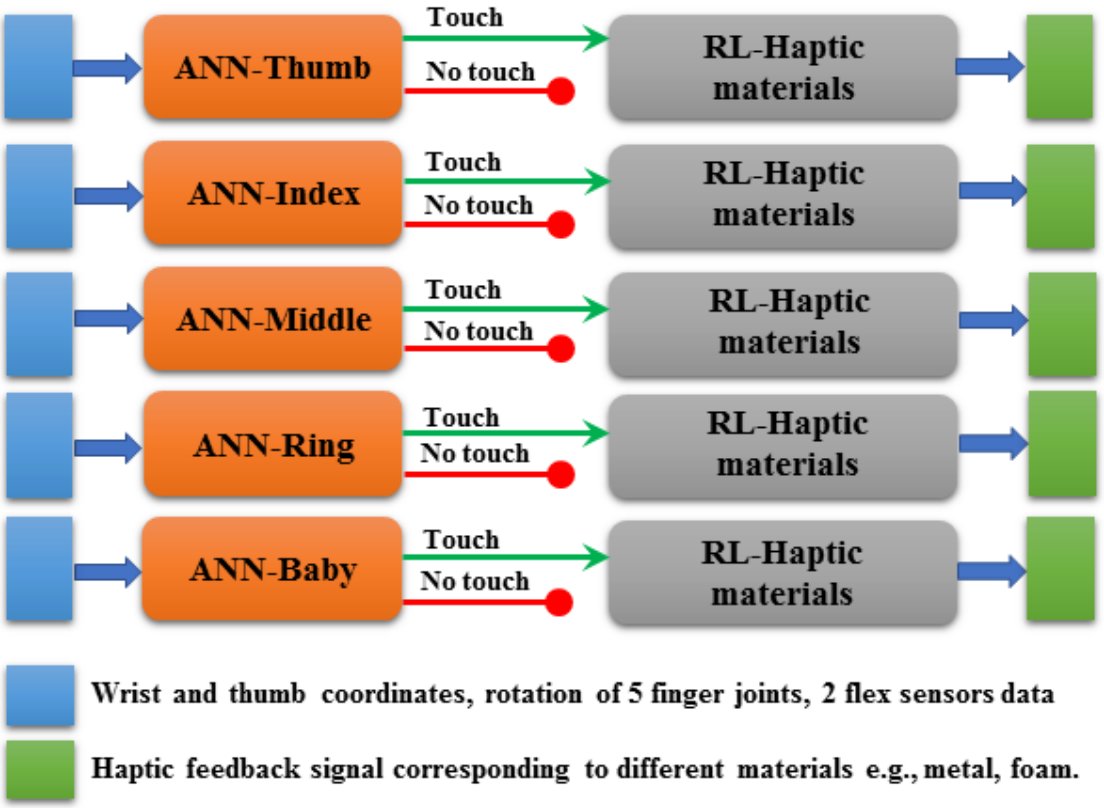



Figure 6.8: AI-based EHASAF module with cascaded ANN and RL units.

forecasts the first haptic feedback sample. To forecast the haptic feedback sample correctly, initially, the RL unit uses its prior knowledge of haptic feedback profiles of all the materials. After every $(t_{MH} + 2 \times t_{HS})$ interval, the H2M server receives the actual haptic feedback sample from the slave device and computes the reward $(r^{(i)})$, which is then normalized error in haptic sample forecasting. With this reward value, the RL unit updates its probability distribution for choosing a haptic material in the $(i+1)^{\text{th}}$ timeslot. When $(2 \times D_Q - t_{MH}) \geq (t_{MH} + 2 \times t_{HS})$, then the H2M server receives the haptic feedback sample before generating the next haptic sample and the RL unit works at its best; otherwise, the RL unit takes more time to detect the actual material and the user experience consequently degrades. As we are forecasting haptic feedback samples, hence we are gaining an additional time buffer of $D_Q$ interval. A summary of the working principle of the RL unit, if a touch event is detected by ANN at the $i^{\text{th}}$ time-slot, is given in the following:

(i) The RL unit randomly chooses a material $(m_k^{(i)})$ from all possible $K$ materials.

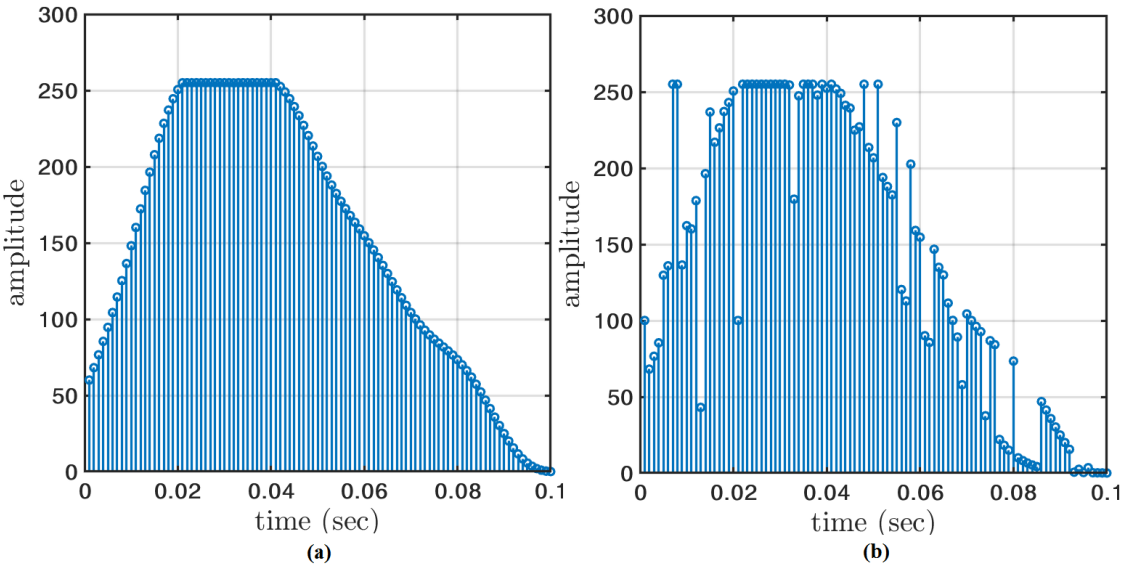



Figure 6.9: (a) Actual haptic feedback samples and (b) RL unit forecasted haptic feedback samples.

(ii) We denote the haptic feedback sample chosen by the RN unit at $i^{\text{th}}$ timeslot by $h_{HM}^{(i)}$ and the actual haptic sample generated at the slave device by $h_{SM}^{(i)}$. Thus, we define the value of the reward received at the H2M server as $r^{(i)} = |h_{SM}^{(i)} - h_{HM}^{(i)}|/255$, such that $0 \leq r^{(i)} \leq 1$.

(iii) The probability distribution of the RL unit for choosing a material in $(i+1)^{\text{th}}$ timeslot is updated according to the following update rule:

$$\boldsymbol{p}_m^{(i+1)} = \boldsymbol{p}_m^{(i)} + \alpha r^{(i)}(\boldsymbol{e}_k^{(i)} - \boldsymbol{p}_m^{(i)}), \tag{6.5}$$

where $0 \leq \alpha \leq 1$ is the *learning rate parameter* and $\boldsymbol{e}_k^{(i)}$ is an *indicator vector* with unity at the $k^{\text{th}}$ index.

Note that (6.5) is a linear reward-inaction automaton and this algorithm shows very strong convergence properties in practice. As it is an RL-based algorithm, it relies on commonly used *exploration* and *exploitation* mechanisms [199]. A smaller value of $\alpha$ indicates more weight on exploration whereas a larger value of $\alpha$ indicates more weight on exploitation. As an example case, we consider the four haptic feedback profiles shown in Fig. 6.5b that has nearly 75% degree of similarity based on the mutual correlation among themselves. Fig. 6.9a shows the actual haptic feedback samples generated by the slave devices. If we choose $\alpha = 1$, i.e., aggressive

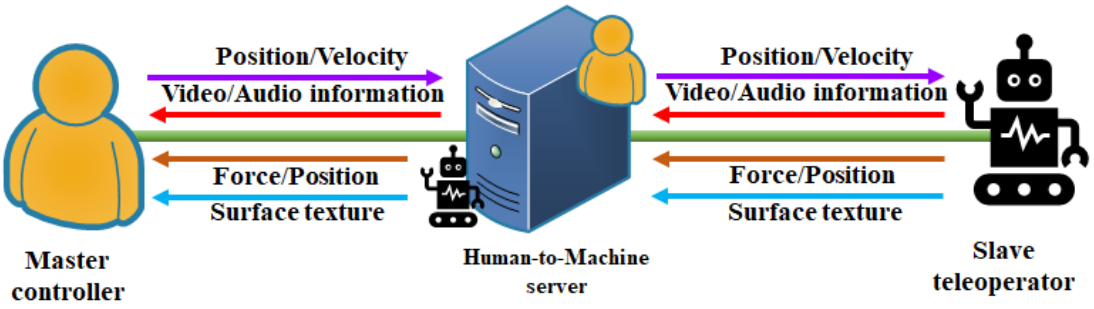



Figure 6.10: Logical links between master, H2M server, and slave devices for the bi-directional haptic control and feedback samples forecasting scheme.

exploitation and less exploration, then an instance of the RL unit forecasted haptic feedback samples are shown in Fig. 6.9b. In this case, the forecasted samples are 92.59% accurate to the actual haptic feedback samples.

### 6.4.3 Bi-directional Haptic Control and Feedback Samples Forecasting

To further reduce the end-to-end closed-loop latency between the master and slave devices, we extend the idea of our EHASAF module to a bi-directional haptic control and feedback samples forecasting module. Fig. 6.10 shows a logical interconnection between master, H2M server, and slave devices, where the H2M server acts as a proxy of the master for the slave device as well as a proxy of the slave for the master device. The master device sends the control signals to H2M server only, instead of all the way long to the slave device. When the first-stage ANN of the EHASAF module detects a touch, it uses the second-stage RL unit to forecast the haptic feedback samples to the master device as earlier. In addition, it also starts to forecast control signal samples to the slave device. Clearly, with this scheme, the H2M server receives the haptic feedback samples much earlier, i.e., at an interval of $2 \times t_{HS}$. Hence, H2M teleoperation applications can be deployed over a much longer distance because as long as $(2 \times D_Q - t_{MH}) \geq 2 \times t_{HS}$, each haptic feedback sample is properly received from the slave device by the H2M server before generating the next haptic feedback sample and the best performance is obtained from the RL unit.

## 6.5 Performance Evaluation

In this section, we evaluate the performance of our proposed EHASAF module. From our experimental setup, we collected several instances of both control signal and haptic feedback data of all the experiments i.e., grabbing a virtual ball, a virtual cube, and a circular cube of different materials with both hands. The training data for each ANN corresponding to each finger contains $(5 \times 3) + 2 = 17$ vectors. We implemented an ANN in MATLAB that contains 2 hidden layers with 10 and 5 nodes, respectively, and used the Levenberg-Marquardt training method for training because of its faster convergence rate as compared to conventional gradient descent algorithms [198]. In Fig. 6.11, we show the space and time consumed by various standard training algorithms while training the proposed ANN with our experimental dataset. The computer contains a 64 bit Intel core i7-6600U CPU with 2.60 GHz clock frequency and 16 GB RAM. We considered Levenberg-Marquardt, BFGS quasi-Newton, resilient backpropagation, scaled conjugate gradient, conjugate gradient with Powell/Beale restarts, one step secant, variable learning rate gradient descent algorithms.

Next, Fig. 6.12 shows the accuracy of the binary classification performed by the ANN with the control data from 800 instances of "grabbing a virtual ball" test

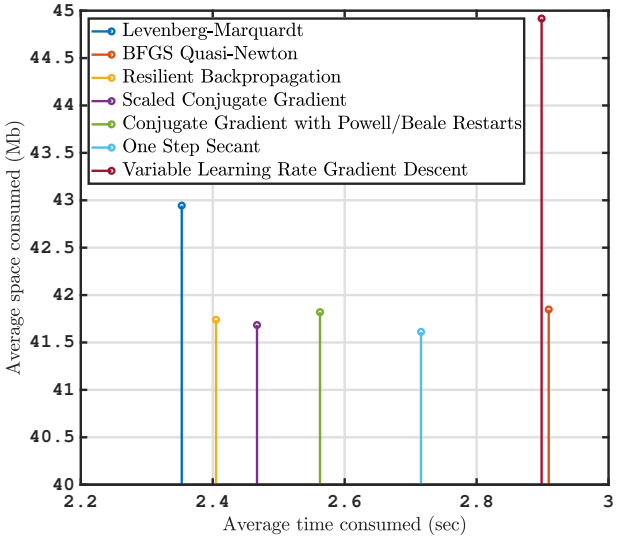



Figure 6.11: Space and time consumed while training our ANN with various training algorithms.

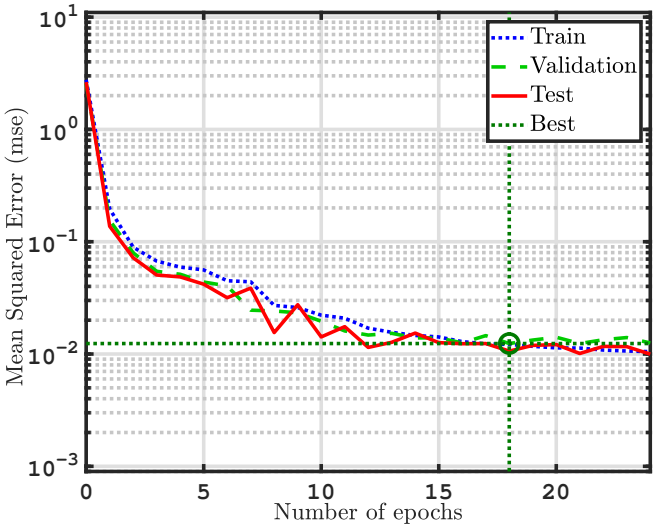



Figure 6.12: Accuracy of binary classification implemented by ANN.

scenario. The mean square error decreases gradually with the number of epochs. We used 70% of the data for training, 15% data for validation, and 15% for testing to achieve a prediction accuracy of $\sim$ 99%. A similar degree of accuracy was also observed in the other test scenarios.

When the ANN detects a finger touching the virtual ball, the EHASAF module starts to generate and forecast haptic feedback samples. With only one material involved in the testing, the feedback is 100% correct, but if there are multiple materials to choose, then the accuracy of the forecasted haptic feedback samples decreases. In

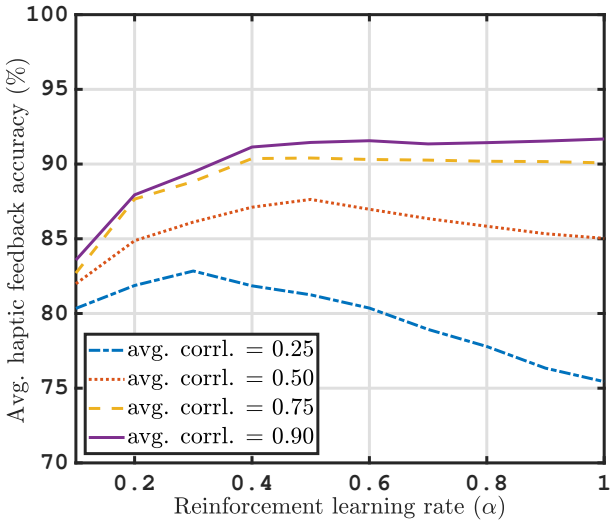



Figure 6.13: RL forecasted haptic feedback accuracy vs. parameter $\alpha$.

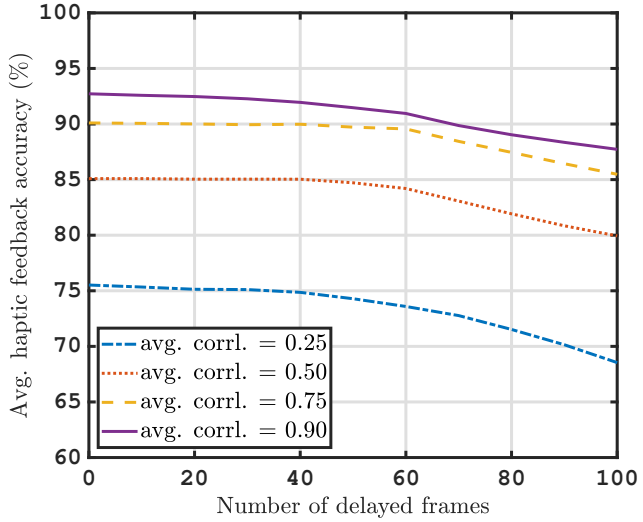



Figure 6.14: RL forecasted haptic feedback accuracy vs. frame delay.

Fig. 6.13, we consider that the learning rate parameter ($\alpha$) varies from 0.1 to 1.0 and four sets of data that have an average degree of similarity in terms of average mutual correlation coefficient 0.25, 0.50, 0.75, and 0.90, respectively. The total duration of each haptic feedback burst is 100 ms and consequent samples are forecast after every 1 ms interval. We observe that the average haptic feedback sample forecasting accuracy increases as the degree of similarity increases. It is also interesting to observe that, when the degree of similarity is lower, more exploration provides relatively better accuracy, whereas when the degree of similarity is higher, then a better accuracy is achieved through more exploitation.

In order to extract the best performance from the RL unit, at every iteration, each actual haptic feedback sample from the slave device must reach the H2M server before the generation of the next haptic feedback sample. If we can ensure that $(D_Q - t_{HM}) \geq (t_{MH} + 2 \times t_{HS})$, then the H2M server receives the haptic feedback samples before generating the next haptic sample and the RL unit works at its best; otherwise, the RL unit takes more time to detect the actual material and the user experience consequently degrades. Thus, in Fig. 6.14 we observe the impact of delay in receiving haptic feedback samples from the slave device to H2M server, where each number of delayed frames implies 1 ms delay. Nonetheless, we observe that the accuracy of the RL unit does not degrade very drastically, rather gradually. This implies that the EHASAF module is quite robust in terms of handling delay in packet

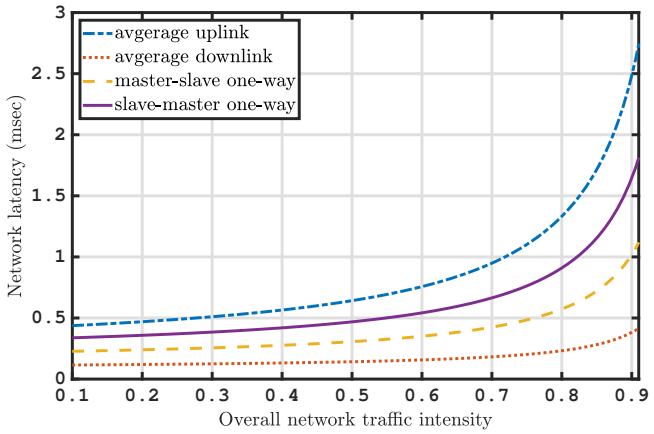



Figure 6.15: End-to-end network latency over fibre backhaul vs. overall traffic intensity of a 20 km 1G-EPON with 16 ONUs.

arrivals. Even when the network is overloaded and the master-slave packet delay is significant, the EHASAF module still ensures that the master receives reasonably accurate haptic feedback samples within every $D_Q$ interval.

To investigate the end-to-end latency between master and slave devices over a TDM-PON based fibre backhaul, we consider a 20 km 1G-EPON with 16 ONUs. We consider that a master device is directly connected to an ONU and a slave device is connected to another ONU. The remaining 14 ONUs are transmitting Poisson distributed background traffic. We use optimal distribution fitting parameter values from Section 6.4 to characterise the control signal and haptic feedback traffics. The master device transmits only control signals with a packet size of 100 B and a bandwidth of 16 Mbps is required for both the gloves. On the other hand, the slave device transmits 360° video and audio data along with haptic feedback [200]. The video data has a packet size of 1.5 kB and requires a bandwidth of 30 Mbps (average) to 60 Mbps (maximum). The audio data has a packet size of 100 B and requires a bandwidth of 512 Kbps. The haptic feedback data has a packet size of 80 B and requires a bandwidth of 1.28 Mbps. With this configuration, the intermediate distance between the master and slave devices is $20 + 20 = 40$ km and the end-to-end propagation latency is 0.2 ms. We consider that the H2M server is situated at the CO and hence, its distance from both the master and slave is 20 km with a propagation latency of 0.1 ms. For computing the average uplink and downlink network latency of the fibre backhaul, we refer to the analytical expressions derived

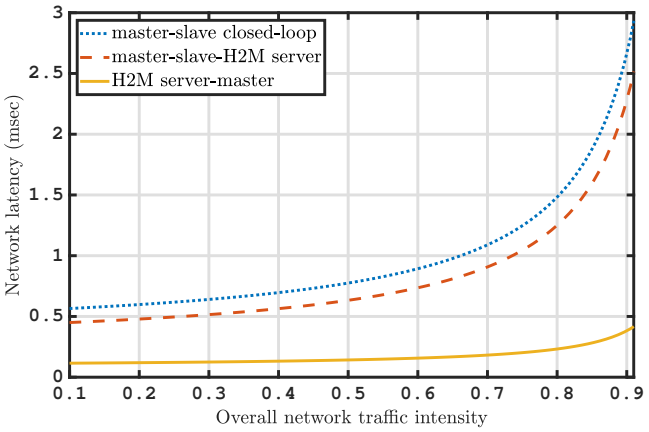



Figure 6.16: Comparison of closed-loop master-slave network latency vs. master-slave-H2M server latencies over fibre backhaul.

in [201] and whereby the authors previously verified the accuracy of these expressions through network simulations.

In Fig. 6.15, we show the average uplink and downlink latencies of all the ONUs. We also show the one-way latencies of the master-slave and slave-master paths in this figure. The results indicate that when the overall network traffic intensity, i.e., background plus H2M traffic intensity is more than 80%, then the slave to master latency exceeds 1 ms. Furthermore, in Fig. 6.16 we compare the closed-loop master-slave latency with the latency of master to slave and slave to H2M server. We observe that the closed-loop master-slave latency exceeds 1 ms when the overall network traffic intensity is higher than 65%. In this case, remote H2M teleoperation can not be implemented because the haptic feedback samples from the slave device will reach later than 1 ms to the master device. However, the latency between the H2M server and the master device remains below 0.5 ms even with 90% overall network traffic intensity. Therefore, our results indicate that we can still implement remote H2M teleoperation by using our proposed EHASAF module in the H2M server at the CO.

## 6.6 Conclusions

In this chapter, we have fitted generalised Pareto distribution with optimal parameter values to the histograms obtained from the timestamps of control signals

and haptic feedback from three different VR based H2M teleoperation test scenarios. To deploy remote H2M teleoperation, we have proposed the idea of installing AI-enhanced H2M server between master and slave devices that supports the preemption of haptic feedback from control signals. Moreover, we have proposed a two-stage AI-based EHASAF module to forecast haptic feedback samples from the H2M server to the master device. The first stage of the EHASAF module is a supervised learning unit that implements a binary classifier by using an ANN to detect if haptic sample forecasting is necessary with $\sim$ 99% accuracy. The second stage of the EHASAF module is a reinforcement learning unit that ensures $\sim$ 90% accuracy in the forecasted haptic feedback samples with four different types of material and average mutual correlation coefficient of 0.9. We have also ensured that the EHASAF module is robust in terms of delayed packet arrivals. Finally, by using analytical expressions for fibre backhaul latency, we have shown the feasibility of deploying remote H2M teleoperation over a 20 km 1G-EPON with 16 ONUs by using our proposed EHASAF module in an intermediate H2M server, even under heavy traffic intensity.

# Chapter 7

# Conclusions and Future Directions

"Reasoning draws a conclusion, but does not make the conclusion certain, unless the mind discovers it by the path of experience."

Roger Bacon

## 7.1 Introduction

The key factor behind the successful implementation of the three emerging dimensions of modern-day Internet viz., nomadicity, embeddedness, and ubiquity is the deployment of distributed intelligence across the global infrastructure [202]. The very foundation of this is mobile cloud computing technology that facilitated the usage of communication and computation-intensive applications in mobile devices by compensating their limitations in the battery, memory, and computational resources. Companies like Amazon, Microsoft, Google, IBM, Unisys, to name a few, started to offer cloud services as a utility like water, electricity, and gas, which inaugurated an entirely new business model. It is with the proliferation of low-latency applications such as augmented reality, online gaming, and tactile internet requiring an end-to-end latency constraint of 1-100 ms that the evolution of edge-computing solution like cloudlets was expedited [14]. The research interests of this thesis lie in the analysis and modelling of latency-sensitive networking protocols and systems with sub-millisecond latency requirements while using optimisation theory, game theory, and artificial intelligence like machine learning and reinforcement learning.

## 7.2 Summary of Contributions

In Chapter 1, we discussed the genesis of "cloudlet computing" technology from its precursor mobile cloud computing technology and its relevance for low-latency applications. We also summarised our research objectives and novel contributions and provided a brief outline of the thesis in this chapter. In Chapter 2, firstly we reviewed some of the state-of-the-art optical, wireless, and fibre-wireless access network technology standards. We thoroughly explored the networking features of TDM-PON, WDM-PON, LTE-A, WiFi, mmWave, 5G NR, R&F, RoF, and C-RAN standards. Secondly, we discussed the importance of cloudlets for the deployment of low-latency TI applications. Next, we identified some of the primary research problems in cloudlet computing networks e.g., cloudlet placement over existing access network infrastructures, job request assignment from mobile devices to cloudlets, and load balancing among neighbouring cloudlets. In this regard, we also discussed some of the recently published research works to address these problems. Furthermore, we discussed the characteristics and networking challenges of recently evolved ultra-reliable and low-latency immersive TI applications.

In Chapter 3, we proposed a FiWi over TDM-PON based hybrid cloudlet placement framework, where cloudlets can be optimally placed in the field, RN or CO locations. In this framework, an MINLP model was developed to evaluate the optimal cost of cloudlet placement. We showed that the installation of more RN and CO cloudlets provides improved cost-optimal solution than the installation of field cloudlets alone. We showed that the field cloudlet framework can be derived as a subset of the hybrid cloudlet placement framework. In general, the results help us to understand that "cost-optimal cloudlet placement locations" and "computational resources per cloudlet" are scenario dependent, i.e., user density, network architecture, TDM-PON split-ratio and QoS requirements primarily dictate the optimal cloudlet placement. We also showed that the percentage of the incremental energy budget of the access networks due to active cloudlet installation is reasonably small.

In Chapter 4, we proposed a novel method to find lower bound of integer programming based cost optimisation frameworks for TDM-PON based hybrid cloudlet placement architecture, where cloudlets are allowed to be optimally placed in the

field, at RN or CO locations. This technique very easily resolves the scalability issues in our previously designed MINLP based frameworks. To solve the cloudlet placement problem, we used KKT conditions and numerical techniques for system of nonlinear equations, and no other heuristic algorithms. We also performed a parametric analysis and observed the dependency of cloudlet deployment cost on network parameters e.g., user density, network architecture, TDM-PON split-ratio and QoS requirements. At the network planning stage, this method to find a lower bound of integer programming based cost optimisation frameworks appears to be a very much user-friendly tool to the cloudlet service providers for developing fundamental insights on the cloudlet deployment and making a first-hand estimate for the total cost.

In Chapter 5, we proposed a novel economic and non-cooperative game-theoretic model for low-latency applications among multiple competitive cloudlets from same as well as different service providers. We designed a very efficient method to solve the game and proposed a very fast-converging gradient projection algorithm to compute the pure-strategy NE computation offload strategy among multiple cloudlets. Moreover, we propose a centralised incentive mechanism that computes the unique NE load balancing strategies of the cloudlets under the supervision of a neutral mediator. Followed by we designed a distributed continuous-action reinforcement learning automata-based algorithm to facilitate the competing cloudlets to learn their NE job request offload strategies independently, without exchanging any control information with neighbouring cloudlets. Through extensive simulation, we showed the dependency of NE learning accuracy of our proposed algorithm on learning rate and spreading rate parameters, i.e., exploration and exploitation and derived various seminal insights on the performance and adaptability of our proposed reinforcement algorithm for non-cooperative load balancing against dynamic job request arrival process in both overloaded and under-loaded network conditions.

In Chapter 6, we fitted generalised Pareto distribution with optimal parameter values to the histograms obtained from the timestamps of control signals and haptic feedback from three different VR based H2M teleoperation test scenarios. To deploy remote H2M teleoperation, we proposed the idea of installing AI-enhanced H2M

server between master and slave devices that supports the preemption of haptic feedback from control signals. Moreover, we proposed a two-stage AI-based EHASAF module to forecast haptic feedback samples from the H2M server to the master device. We also ensured that the EHASAF module is robust in terms of delayed packet arrivals. Finally, by using analytical expressions for fibre backhaul latency, we showed the feasibility of deploying remote H2M teleoperation over a 20 km 1G-EPON with 16 ONUs by using our proposed EHASAF module in an intermediate H2M server, even under heavy traffic intensity.

## 7.3 Future Research Directions

In this section, we discuss some of the potential future research directions, that we can extend from the works done in this thesis.

### 7.3.1 Dynamic Job Request Assignment from Mobile Devices to Cloudlets and Related Problems

In Chapter 3 we proposed an MINLP based framework for cloudlet placement and in Chapter 4 we proposed analytical frameworks to obtain generalised insights on cloudlet deployment strategies. However, after the cloudlets are placed and mobile users start to offload job requests, it is very important to efficiently allocate job requests from mobile devices to cloudlets so that the QoS requirements of the mobile users are satisfied. This problem can be addressed in several ways, either by using some state-of-art job request assignment algorithms or by designing some optimisation or game-theoretic framework. As we are mainly considering low-latency applications, we must ensure that the proposed algorithms converge sufficiently fast. Therefore, some heuristic algorithms can be designed to solve the formulated job request assignment problem. In addition, there are scopes to formulate some related problems while considering network economics aspects. If different users are subscribed to different types of subscription plans, then the dynamic allocation of computational resources among mobile users leads to an interesting research problem where auction-based approaches can prove to be highly useful. It is also important

to note that in general, communication network and cloudlet service providers are different. Hence, the economic interaction among the communication network and cloudlet service providers leads to many interesting research problems.

### 7.3.2 Cooperative Game-Theoretic Load Balancing among Cloudlets

In Chapter 5, we proposed economic and non-cooperative game-theoretic frameworks for load balancing among neighbouring cloudlets from same as well different service providers. However, sometimes different cloudlet service providers may wish to cooperate and share computation load among each other. In such frameworks, different cloudlets are allowed to bargain among themselves to decide a price for offloading job request and the NE computation offloading are strategies will depend on these prices. In this context, we can also explore various polynomial-time algorithms that can be implemented very efficiently for faster convergence. For example, a group of under-loaded and over-loaded cloudlets can be categorised as two different class of nodes in a bipartite graph and a suitable matching algorithm can be adapted to make the load balancing decisions.

### 7.3.3 Advanced AI-based approaches for Low-Latency Applications

We studied some experimental data obtained from VR based H2M teleoperation experiments in Chapter 6. However, we observed that the intensities of control signals and haptic feedback are largely dependent on the type of actions performed, e.g., for very delicate actions like cutting a soft tissue with a knife, the volume of traffic generated is larger compared to regular actions like grabbing or punching some object. Therefore, an extensive experimental study is required against various use cases and model the respective network traffic characteristics. Along with this, we need to design more sophisticated classification models for identifying various complex gestures from the control signals and haptic feedback data and predict user actions. Furthermore, we observed that state-of-the-art H2M applications can not

be deployed beyond more than 40 km, even with optical networks. Hence, we need to explore various predictive models by using machine learning and reinforcement learning techniques to enhance the range beyond this limit.

### 7.3.4 Theoretical Analysis of Low-Latency Networks using Network Calculus

Network calculus is a recently developed mathematical tool to theoretically analyse the performance guarantees in digital electronic and computer communication networks [203]. This tool is very much useful for theoretically modelling vital constraints in communication networks like link capacity, congestion control, background traffic, to name a few. Moreover, various traffic arrival and departure processes can also be modelled by using network calculus. Therefore, we can use this efficient theoretical tool to propose some suitable models for understanding the characteristics of network traffic generated from low-latency TI and H2M applications. As these characteristics are mostly unknown to researchers, these theoretical models can be cross-validated through the experimental results.

## 7.4 Conclusions

In this thesis, we proposed a hybrid cost-optimal cloudlet placement framework over existing fibre-wireless access networks based on mixed-integer non-linear programming. In addition, we designed an analytical framework with some closed-form expressions that can provide a quick first-hand estimation of cloudlet deployment cost depending on mobile user density, network architecture, and latency requirements. To address the load balancing problem among neighbouring cloudlets, we proposed an economic and non-cooperative load balancing game for low-latency applications. In this context, firstly we proposed a centralised incentive mechanism for elicitation of truthful information from the competing cloudlets by a central mediator to compute the unique Nash equilibrium load balancing strategies of the cloudlets. Secondly, we proposed a continuous-action reinforcement learning automata-based

algorithm, which allows each cloudlet to independently compute the Nash equilibrium in a completely distributed network setting. Furthermore, we studied experimental data on traffic characteristics of control signals and haptic feedback from a virtual reality-based human-to-machine teleoperation and proposed a two-stage AI module (artificial neural network-based binary classifier unit and reinforcement learning unit) for pre-empting and forecasting haptic feedback samples to the human operator before the actual haptic feedback arising from the slave teleoperator device.

# Bibliography


[1] Mark Weiser. The computer for the 21st century. *Scientific American*, 265(3): 66–75, September 1991.

[2] R V D Meulen. Gartner says 8.4 billion connected "Things" will be in use in 2017, up 31 percent from 2016. *Gartner Newsroom*, February 2017. URL http://www.gartner.com/newsroom/id/3598917.

[3] Cisco Visual Networking Index. Global mobile data traffic forecast update, 2016-2021, 2017. URL http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf.

[4] Abdelsalam Helal, Bert Haskell, Jeffery L. Carter, Richard Brice, Darrell Woelk, and Marek Rusinkiewicz. *Introduction to Mobile Computing*, pages 1–11. Springer US, Boston, MA, 2002. ISBN 978-0-306-47301-2. doi: 10.1007/0-306-47301-1_1.

[5] K. Kumar and Y. H. Lu. Cloud Computing for Mobile Users: Can Offloading Computation Save Energy? *Computer*, 43(4):51–56, April 2010. ISSN 0018-9162. doi: 10.1109/MC.2010.98.

[6] Peter Mell and Timothy Grance. The NIST Definition of Cloud Computing, September 2011. URL http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf.

[7] Ashkan Yousefpour, Caleb Fung, Tam Nguyen, Krishna Kadiyala, Fatemeh Jalali, Amirreza Niakanlahiji, Jian Kong, and Jason P. Jue. All One Needs to Know About Fog Computing and Related Edge Computing Paradigms: A

Complete Survey. *Journal of Systems Architecture*, 2019. ISSN 1383-7621. doi: 10.1016/j.sysarc.2019.02.009.

[8] Mahadev Satyanarayanan, Paramvir Bahl, Ramón Caceres, and Nigel Davies. The Case for VM-Based Cloudlets in Mobile Computing. *IEEE Pervasive Computing*, 8(4):14–23, October 2009. ISSN 1536-1268.

[9] Mahadev Satyanarayanan. The Role of Cloudlets in Hostile Environments. In *Proceeding of the Fourth ACM Workshop on Mobile Cloud Computing and Services*, MCS '13, pages 1–2, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2072-6. doi: 10.1145/2482981.2483793.

[10] S. J. Lederman and R. L. Klatzky. Haptic perception: A tutorial. *Attention, Perception, & Psychophysics*, 71(7):1439–1459, Oct 2009. ISSN 1943-393X. doi: 10.3758/APP.71.7.1439. URL https://doi.org/10.3758/APP.71.7.1439.

[11] M. Simsek, A. Aijaz, M. Dohler, J. Sachs, and G. Fettweis. 5G-Enabled Tactile Internet. *IEEE Journal on Selected Areas in Communications*, 34(3):460–473, March 2016. ISSN 0733-8716. doi: 10.1109/JSAC.2016.2525398.

[12] E. Wong, M. Pubudini Imali Dias, and L. Ruan. Predictive Resource Allocation for Tactile Internet Capable Passive Optical LANs. *Journal of Lightwave Technology*, 35(13):2629–2641, July 2017. ISSN 0733-8724. doi: 10.1109/JLT.2017.2654365.

[13] Rajesh Balan, Jason Flinn, M. Satyanarayanan, Shafeeq Sinnamohideen, and Hen-I Yang. The Case for Cyber Foraging. In *Proceedings of the 10th Workshop on ACM SIGOPS European Workshop*, EW 10, pages 87–92, New York, NY, USA, 2002. ACM. doi: 10.1145/1133373.1133390.

[14] K. Dolui and S. K. Datta. Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing. In *2017 Global Internet of Things Summit (GIoTS)*, pages 1–6, June 2017. doi: 10.1109/GIOTS.2017.8016213.

[15] M. Maier, M. Chowdhury, B. P. Rimal, and D. P. Van. The tactile internet: vision, recent progress, and open challenges. *IEEE Communications Magazine*, 54(5):138–145, May 2016. ISSN 0163-6804. doi: 10.1109/MCOM.2016.7470948.

[16] Thomas Starr, John M. Cioffi, and Peter J. Silverman. *Understanding Digital Subscriber Line Technology*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999. ISBN 0-13-780545-4.

[17] Jerry A. Hausman, J. Gregory Sidak, and HalJ. Singer. Cable Modems and DSL: Broadband Internet Access for Residential Customers. *American Economic Review*, 91(2):302–307, May 2001. doi: 10.1257/aer.91.2.302.

[18] K. Maxwell. Asymmetric digital subscriber line: interim technology for the next forty years. *IEEE Communications Magazine*, 34(10):100–106, Oct 1996. ISSN 1558-1896. doi: 10.1109/35.544330.

[19] J. M. Cioffi, V. Oksman, J. . Werner, T. Pollet, P. M. P. Spruyt, J. S. Chow, and K. S. Jacobsen. Very-high-speed digital subscriber lines. *IEEE Communications Magazine*, 37(4):72–79, April 1999. ISSN 1558-1896. doi: 10.1109/35.755453.

[20] E. Wong. Next-Generation Broadband Access Networks and Technologies. *Journal of Lightwave Technology*, 30(4):597–608, Feb 2012. ISSN 0733-8724. doi: 10.1109/JLT.2011.2177960.

[21] Glen Kramer. *Ethernet Passive Optical Networks*. McGraw-Hill, 03 2005. ISBN 0-07-144562-5.

[22] T. Koonen. Fiber to the Home/Fiber to the Premises: What, Where, and When? *Proceedings of the IEEE*, 94(5):911–934, May 2006. ISSN 1558-2256. doi: 10.1109/JPROC.2006.873435.

[23] C. Lee, W. V. Sorin, and B. Y. Kim. Fiber to the Home Using a PON Infrastructure. *Journal of Lightwave Technology*, 24(12):4568–4583, Dec 2006. ISSN 1558-2213.

[24] E. Harstead. Future bandwidth demand favors TDM PON, not WDM PON. In *2011 Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference*, pages 1–3, March 2011.

[25] R. Jirachariyakool, N. Sra-ium, and S. Lerkvaranyu. Design and implement of GPON-FTTH network for residential condominium. In *2017 14th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 1–5, July 2017.

[26] IEEE. IEEE Standard for Ethernet. *IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015)*, pages 1–5600, Aug 2018. ISSN null. doi: 10.1109/IEEESTD.2018.8457469.

[27] Stefania Sesia, Issam Toufik, and Matthew Baker. *LTE, The UMTS Long Term Evolution: From Theory to Practice*. Wiley Publishing, 2009. ISBN 0470697164, 9780470697160.

[28] Arunabha Ghosh, Jun Zhang, Jeffrey G. Andrews, and Rias Muhamed. *Fundamentals of LTE*. Prentice Hall Press, Upper Saddle River, NJ, USA, 1st edition, 2010. ISBN 0137033117, 9780137033119.

[29] A. Ghosh, R. Ratasuk, B. Mondal, N. Mangalvedhe, and T. Thomas. LTE-advanced: next-generation wireless broadband technology [Invited Paper]. *IEEE Wireless Communications*, 17(3):10–22, June 2010. ISSN 1558-0687. doi: 10.1109/MWC.2010.5490974.

[30] Erik Dahlman, Stefan Parkvall, and Johan Skld. *4G: LTE/LTE-Advanced for Mobile Broadband*. Academic Press, second edition, 2014. ISBN 9780124199859.

[31] Erik Dahlman, Stefan Parkvall, and Johan Skld, editors. *4G LTE-Advanced Pro and The Road to 5G*. Academic Press, third edition edition, 2016. ISBN 978-0-12-804575-6.

[32] Jeanette Wannstrom. LTE-Advanced, June 2013. URL https://www.3gpp.org/technologies/keywords-acronyms/97-lte-advanced.

[33] P. S. Henry and Hui Luo. WiFi: what's next? *IEEE Communications Magazine*, 40(12):66–72, Dec 2002. ISSN 1558-1896. doi: 10.1109/MCOM.2002.1106162.

[34] Aruna Balasubramanian, Ratul Mahajan, and Arun Venkataramani. Augmenting Mobile 3G Using WiFi. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 209–222, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-985-5. doi: 10.1145/1814433.1814456.

[35] L. Uryvsky, S. Osypchuk, and B. Shmigel. The 802.11 protocols usage for wireless systems construction with flexible architecture. In *2016 13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET)*, pages 918–921, Feb 2016. doi: 10.1109/TCSET.2016.7452225.

[36] M. D. Aime, G. Calandriello, and A. Lioy. Dependability in Wireless Networks: Can We Rely on WiFi? *IEEE Security Privacy*, 5(1):23–29, Jan 2007. ISSN 1558-4046. doi: 10.1109/MSP.2007.4.

[37] T. S. Rappaport, Y. Xing, G. R. MacCartney, A. F. Molisch, E. Mellios, and J. Zhang. Overview of Millimeter Wave Communications for Fifth-Generation (5G) Wireless NetworksWith a Focus on Propagation Models. *IEEE Transactions on Antennas and Propagation*, 65(12):6213–6230, Dec 2017. ISSN 1558-2221. doi: 10.1109/TAP.2017.2734243.

[38] T.E. Bogale, X. Wang, and L.B. Le. mmWave communication enabling techniques for 5G wireless systems: A link level perspective. In *mmWave Massive MIMO*, pages 195 – 225. Academic Press, 2017. ISBN 978-0-12-804418-6. doi: https://doi.org/10.1016/B978-0-12-804418-6.00009-1.

[39] X. Wang, L. Kong, F. Kong, F. Qiu, M. Xia, S. Arnon, and G. Chen. Millimeter Wave Communication: A Comprehensive Survey. *IEEE Communications Surveys Tutorials*, 20(3):1616–1653, thirdquarter 2018. ISSN 2373-745X. doi: 10.1109/COMST.2018.2844322.

[40] Sarang Han, Soonbae Ji, Inseok Kang, Sung Chul Kim, and Cheolwoo You. Millimeter wave beamforming receivers using a Si-based OBFN for 5G wireless communication systems. *Optics Communications*, 430:83 – 97, 2019. ISSN 0030-4018. doi: https://doi.org/10.1016/j.optcom.2018.08.031.

[41] J. Thompson, X. Ge, H. Wu, R. Irmer, H. Jiang, G. Fettweis, and S. Alamouti. 5G wireless communication systems: prospects and challenges [Guest Editorial]. *IEEE Communications Magazine*, 52(2):62–64, February 2014. ISSN 1558-1896. doi: 10.1109/MCOM.2014.6736744.

[42] P. Popovski, . Stefanovi, J. J. Nielsen, E. de Carvalho, M. Angjelichinoski, K. F. Trillingsgaard, and A. Bana. Wireless Access in Ultra-Reliable Low-Latency Communication (URLLC). *IEEE Transactions on Communications*, 67(8):5783–5801, Aug 2019. ISSN 1558-0857. doi: 10.1109/TCOMM.2019.2914652.

[43] J. Sachs, L. A. A. Andersson, J. Arajo, C. Curescu, J. Lundsj, G. Rune, E. Steinbach, and G. Wikstrm. Adaptive 5G Low-Latency Communication for Tactile InternEt Services. *Proceedings of the IEEE*, 107(2):325–349, Feb 2019. ISSN 1558-2256. doi: 10.1109/JPROC.2018.2864587.

[44] Martin Maier, Navid Ghazisaidi, and Martin Reisslein. The Audacity of Fiber-Wireless (FiWi) Networks. In Chonggang Wang, editor, *AccessNets*, pages 16–35, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. ISBN 978-3-642-04648-3.

[45] A. M. Zin, M. S. Bongsu, S. M. Idrus, and N. Zulkifli. An overview of radio-over-fiber network technology. In *International Conference On Photonics 2010*, pages 1–3, July 2010. doi: 10.1109/ICP.2010.5604429.

[46] N. J. Gomes, P. P. Monteiro, and A. Gameiro. *Introduction to Radio over Fiber*, pages 61–89. Wiley, 2012. ISBN null. doi: 10.1002/9781118306017.ch4.

[47] M. Maier and B. P. Rimal. Invited paper: The audacity of fiber-wireless (FiWi) networks: Revisited for clouds and cloudlets. *China Communications*, 12(8):33–45, August 2015. ISSN 1673-5447. doi: 10.1109/CC.2015.7224704.

[48] C. I, J. Huang, R. Duan, C. Cui, J. Jiang, and L. Li. Recent Progress on C-RAN Centralization and Cloudification. *IEEE Access*, 2:1030–1039, 2014. ISSN 2169-3536. doi: 10.1109/ACCESS.2014.2351411.

[49] Tingting Duan, Min Zhang, Zhilong Wang, and Chuang Song. Inter-bbu control mechanism for load balancing in c-ran-based bbu pool. In *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, pages 2960–2964, Oct 2016. doi: 10.1109/CompComm.2016.7925239.

[50] H. Rastegarfar, T. Svensson, and N. Peyghambarian. Optical layer routing influence on software-defined C-RAN survivability. *IEEE/OSA Journal of Optical Communications and Networking*, 10(11):866–877, Nov 2018. ISSN 1943-0639. doi: 10.1364/JOCN.10.000866.

[51] Akamai. Cloudlets - akamai Specifications, May 2015. URL http://www.akamai.com/cloudlets.

[52] S. Clinch, J. Harkes, A. Friday, N. Davies, and M. Satyanarayanan. How close is close enough? Understanding the role of cloudlets in supporting display appropriation by mobile users. In *2012 IEEE International Conference on Pervasive Computing and Communications*, pages 122–127, March 2012. doi: 10.1109/PerCom.2012.6199858.

[53] Kiryong Ha, Padmanabhan Pillai, Grace Lewis, Soumya Simanta, Sarah Clinch, Nigel Davies, and Mahadev Satyanarayanan. The Impact of Mobile Multimedia Applications on Data Center Consolidation. In *Proceedings of the 2013 IEEE International Conference on Cloud Engineering*, IC2E '13, pages 166–176, Washington, DC, USA, 2013. IEEE Computer Society. ISBN 978-0-7695-4945-3. doi: 10.1109/IC2E.2013.17.

[54] Y. Li and W. Wang. The unheralded power of cloudlet computing in the vicinity of mobile devices. In *2013 IEEE Global Communications Conference (GLOBECOM)*, pages 4994–4999, Dec 2013. doi: 10.1109/GLOCOMW.2013.6855742.

[55] Kiryong Ha, Padmanabhan Pillai, Wolfgang Richter, Yoshihisa Abe, and Mahadev Satyanarayanan. Just-in-time provisioning for cyber foraging. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '13, pages 153–166, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1672-9. doi: 10.1145/2462456.2464451. URL http://doi.acm.org/10.1145/2462456.2464451.

[56] Wenlu Hu, Brandon Amos, Zhuo Chen, Kiryong Ha, Wolfgang Richter, Padmanabhan Pillai, Benjamin Gilbert, Jan Harkes, and Mahadev Satyanarayanan. The case for offload shaping. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, HotMobile '15, pages 51–56, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3391-7. doi: 10.1145/2699343.2699351. URL http://doi.acm.org/10.1145/2699343.2699351.

[57] G. Lewis, S. Echeverra, S. Simanta, B. Bradshaw, and J. Root. Tactical Cloudlets: Moving Cloud Computing to the Edge. In *2014 IEEE Military Communications Conference*, pages 1440–1446, Oct 2014. doi: 10.1109/MILCOM.2014.238.

[58] Y. Jiang. A Survey of Task Allocation and Load Balancing in Distributed Systems. *IEEE Transactions on Parallel and Distributed Systems*, 27(2):585–599, Feb 2016. ISSN 1045-9219. doi: 10.1109/TPDS.2015.2407900.

[59] Byung-Gon Chun, Sunghwan Ihm, Petros Maniatis, Mayur Naik, and Ashwin Patti. CloneCloud: Elastic Execution Between Mobile Device and Cloud. In *Proceedings of the Sixth Conference on Computer Systems*, EuroSys '11, pages 301–314, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0634-8. doi: 10.1145/1966445.1966473.

[60] Eduardo Cuervo, Aruna Balasubramanian, Dae-ki Cho, Alec Wolman, Stefan Saroiu, Ranveer Chandra, and Paramvir Bahl. Maui: Making smartphones last longer with code offload. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 49–62, New

York, NY, USA, 2010. ACM. ISBN 978-1-60558-985-5. doi: 10.1145/1814433.1814441.

[61] Roelof Kemp, Nicholas Palmer, Thilo Kielmann, and Henri Bal. Cuckoo: A Computation Offloading Framework for Smartphones. In Martin Gris and Guang Yang, editors, *Mobile Computing, Applications, and Services*, pages 59–79, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-29336-8.

[62] K. Li and J. Nabrzyski. Virtual Machine Placement in Cloudlet Mesh. *Journal of Communications and Networks*, 20(3):266–278, June 2018. ISSN 1229-2370. doi: 10.1109/JCN.2018.000039.

[63] A. Ceselli, M. Premoli, and S. Secci. Mobile Edge Cloud Network Design Optimization. *IEEE/ACM Transactions on Networking*, 25(3):1818–1831, June 2017. ISSN 1063-6692. doi: 10.1109/TNET.2017.2652850.

[64] A. Ceselli, M. Premoli, and S. Secci. Mobile Edge Cloud Network Design Optimization. *IEEE/ACM Transactions on Networking*, 25(3):1818–1831, June 2017. ISSN 1063-6692. doi: 10.1109/TNET.2017.2652850.

[65] Z. Xu, W. Liang, W. Xu, M. Jia, and S. Guo. Capacitated cloudlet placements in Wireless Metropolitan Area Networks. In *2015 IEEE 40th Conference on Local Computer Networks (LCN)*, pages 570–578, Oct 2015. doi: 10.1109/LCN.2015.7366372.

[66] Z. Xu, W. Liang, W. Xu, M. Jia, and S. Guo. Efficient Algorithms for Capacitated Cloudlet Placements. *IEEE Transactions on Parallel and Distributed Systems*, 27(10):2866–2880, Oct 2016. ISSN 1045-9219. doi: 10.1109/TPDS.2015.2510638.

[67] M. Jia, J. Cao, and W. Liang. Optimal Cloudlet Placement and User to Cloudlet Allocation in Wireless Metropolitan Area Networks. *IEEE Transactions on Cloud Computing*, 5(4):725–737, Oct 2017. ISSN 2372-0018. doi: 10.1109/TCC.2015.2449834.

[68] Q. Fan and N. Ansari. Cost Aware Cloudlet Placement for Big Data Processing at the Edge. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2017. doi: 10.1109/ICC.2017.7996722.

[69] X. Sun and N. Ansari. PRIMAL: PRofIt Maximization Avatar pLacement for mobile edge computing. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2016. doi: 10.1109/ICC.2016.7511131.

[70] S. H. S. Newaz, A. F. Y. Mohammed, G. M. Lee, and J. K. Choi. Energy efficient and latency aware TDM-PON for local customer internetworking. In *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 1184–1189, May 2015. doi: 10.1109/INM.2015.7140464.

[71] S. H. S. Newaz, W. Susanty binti Haji Suhaili, G. M. Lee, M. R. Uddin, A. F. Y. Mohammed, and J. K. Choi. Towards realizing the importance of placing fog computing facilities at the central office of a PON. In *2017 19th International Conference on Advanced Communication Technology (ICACT)*, pages 152–157, Feb 2017. doi: 10.23919/ICACT.2017.7890075.

[72] A. H. Helmy and A. Nayak. Integrating Fog With Long-Reach PONs From a Dynamic Bandwidth Allocation Perspective. *IEEE Journal of Lightwave Technology*, 36(22):5276–5284, Nov 2018. ISSN 0733-8724. doi: 10.1109/JLT.2018.2873161.

[73] D. T. Hoang, D. Niyato, and P. Wang. Optimal Admission Control Policy for Mobile Cloud Computing Hotspot with Cloudlet. In *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 3145–3149, April 2012. doi: 10.1109/WCNC.2012.6214347.

[74] S. Joilo and G. Dn. A Game Theoretic Analysis of Selfish Mobile Computation Offloading. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, May 2017. doi: 10.1109/INFOCOM.2017.8057148.

[75] S. Ahn, J. Lee, S. Park, S. H. S. Newaz, and J. K. Choi. Competitive Partial Computation Offloading for Maximizing Energy Efficiency in Mobile Cloud

Computing. *IEEE Access*, 6:899–912, 2018. ISSN 2169-3536. doi: 10.1109/ACCESS.2017.2776323.

[76] H. Cao and J. Cai. Distributed Multiuser Computation Offloading for Cloudlet-Based Mobile Cloud Computing: A Game-Theoretic Machine Learning Approach. *IEEE Transactions on Vehicular Technology*, 67(1):752–764, Jan 2018. ISSN 0018-9545. doi: 10.1109/TVT.2017.2740724.

[77] M. Liu and Y. Liu. Price-Based Distributed Offloading for Mobile-Edge Computing With Computation Capacity Constraints. *IEEE Wireless Communications Letters*, 7(3):420–423, June 2018. ISSN 2162-2337. doi: 10.1109/LWC.2017.2780128.

[78] L. Liu, S. Chan, G. Han, M. Guizani, and M. Bandai. Performance Modeling of Representative Load Sharing Schemes for Clustered Servers in Multiaccess Edge Computing. *IEEE Internet of Things Journal*, 6(3):4880–4888, June 2019. ISSN 2327-4662. doi: 10.1109/JIOT.2018.2879513.

[79] M. Jia, W. Liang, Z. Xu, and M. Huang. Cloudlet Load Balancing in Wireless Metropolitan Area Networks. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, April 2016. doi: 10.1109/INFOCOM.2016.7524411.

[80] Satish Penmatsa and Anthony T. Chronopoulos. Game-theoretic Static Load Balancing for Distributed Systems. *Journal of Parallel and Distributed Computing*, 71(4):537 – 555, 2011. ISSN 0743-7315. doi: https://doi.org/10.1016/j.jpdc.2010.11.016.

[81] X. Zhou, K. Wang, W. Jia, and M. Guo. Reinforcement learning-based adaptive resource management of differentiated services in geo-distributed data centers. In *2017 IEEE/ACM 25th International Symposium on Quality of Service (IWQoS)*, pages 1–6, June 2017. doi: 10.1109/IWQoS.2017.7969161.

[82] R. Beraldi, A. Mtibaa, and H. Alnuweiri. Cooperative Load Balancing Scheme for Edge Computing Resources. In *2017 Second International Conference on*

*Fog and Mobile Edge Computing (FMEC)*, pages 94–100, May 2017. doi: 10.1109/FMEC.2017.7946414.

[83] L. Chen, S. Zhou, and J. Xu. Computation Peer Offloading for Energy-Constrained Mobile Edge Computing in Small-Cell Networks. *IEEE/ACM Transactions on Networking*, 26(4):1619–1632, Aug 2018. ISSN 1063-6692. doi: 10.1109/TNET.2018.2841758.

[84] C. Liu, K. Li, and K. Li. A Game Approach to Multi-servers Load Balancing with Load-Dependent Server Availability Consideration. *IEEE Transactions on Cloud Computing*, pages 1–1, 2018. doi: 10.1109/TCC.2018.2790404.

[85] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai. A Survey on Low Latency Towards 5G: RAN, Core Network and Caching Solutions. *IEEE Communications Surveys Tutorials*, 20(4):3098–3130, Fourth quarter 2018. ISSN 1553-877X. doi: 10.1109/COMST.2018.2841349.

[86] O. N. C. Yilmaz, Y. . E. Wang, N. A. Johansson, N. Brahmi, S. A. Ashraf, and J. Sachs. Analysis of ultra-reliable and low-latency 5G communication for a factory automation use case. In *2015 IEEE International Conference on Communication Workshop (ICCW)*, pages 1190–1195, June 2015. doi: 10.1109/ICCW.2015.7247339.

[87] C. Campolo, A. Molinaro, G. Araniti, and A. O. Berthet. Better Platooning Control Toward Autonomous Driving : An LTE Device-to-Device Communications Strategy That Meets Ultralow Latency Requirements. *IEEE Vehicular Technology Magazine*, 12(1):30–38, March 2017. ISSN 1556-6080. doi: 10.1109/MVT.2016.2632418.

[88] P. Schulz, M. Matthe, H. Klessig, M. Simsek, G. Fettweis, J. Ansari, S. A. Ashraf, B. Almeroth, J. Voigt, I. Riedel, A. Puschmann, A. Mitschele-Thiel, M. Muller, T. Elste, and M. Windisch. Latency Critical IoT Applications in 5G: Perspective on the Design of Radio Interface and Network Architecture. *IEEE Communications Magazine*, 55(2):70–78, February 2017. ISSN 1558-1896. doi: 10.1109/MCOM.2017.1600435CM.

[89] A. Taleb Zadeh Kasgari, W. Saad, and M. Debbah. Human-in-the-Loop Wireless Communications: Machine Learning and Brain-Aware Resource Management. *IEEE Transactions on Communications*, 67(11):7727–7743, Nov 2019. ISSN 1558-0857. doi: 10.1109/TCOMM.2019.2930275.

[90] M. Dohler, T. Mahmoodi, M. A. Lema, M. Condoluci, F. Sardis, K. Antonakoglou, and H. Aghvami. Internet of skills, where robotics meets AI, 5G and the Tactile Internet. In *2017 European Conference on Networks and Communications (EuCNC)*, pages 1–5, June 2017. doi: 10.1109/EuCNC.2017.7980645.

[91] M. Condoluci, T. Mahmoodi, E. Steinbach, and M. Dohler. Soft Resource Reservation for Low-Delayed Teleoperation Over Mobile Networks. *IEEE Access*, 5:10445–10455, 2017. ISSN 2169-3536. doi: 10.1109/ACCESS.2017.2707319.

[92] Heather Culbertson, Samuel Schorr, and Allison Okamura. Haptics: The Present and Future of Artificial Touch Sensation. *Annual Review of Control, Robotics, and Autonomous Systems*, 1, 05 2018. doi: 10.1146/annurev-control-060117-105043.

[93] E. Ruffaldi, A. Di Fava, C. Loconsole, A. Frisoli, and C. A. Avizzano. Vibrotactile feedback for aiding robot kinesthetic teaching of manipulation tasks. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 818–823, Aug 2017. doi: 10.1109/ROMAN.2017.8172397.

[94] Shan Luo, Joao Bimbo, Ravinder Dahiya, and Hongbin Liu. Robotic Tactile Perception of Object Properties: A Review. *Mechatronics*, 48, 12 2017. doi: 10.1016/j.mechatronics.2017.11.002.

[95] E. Steinbach, M. Strese, M. Eid, X. Liu, A. Bhardwaj, Q. Liu, M. Al-Jaafreh, T. Mahmoodi, R. Hassen, A. El Saddik, and O. Holland. Haptic Codecs for the Tactile Internet. *Proceedings of the IEEE*, 107(2):447–470, Feb 2019. ISSN 1558-2256. doi: 10.1109/JPROC.2018.2867835.

[96] C. Pacchierotti, S. Sinclair, M. Solazzi, A. Frisoli, V. Hayward, and D. Prattichizzo. Wearable Haptic Systems for the Fingertip and the Hand: Taxonomy, Review, and Perspectives. *IEEE Transactions on Haptics*, 10(4):580–600, Oct 2017. ISSN 2334-0134. doi: 10.1109/TOH.2017.2689006.

[97] Susan J Lederman and Roberta L Klatzky. Hand movements: A window into haptic object recognition. *Cognitive Psychology*, 19(3):342 – 368, 1987. ISSN 0010-0285. doi: https://doi.org/10.1016/0010-0285(87)90008-9.

[98] S Lederman and Roberta Klatzky. Haptic Perception: A Tutorial. *Attention, perception & psychophysics*, 71:1439–59, 10 2009. doi: 10.3758/APP.71.7.1439.

[99] S. Okamoto, H. Nagano, and Y. Yamada. Psychophysical Dimensions of Tactile Perception of Textures. *IEEE Transactions on Haptics*, 6(1):81–93, First 2013. ISSN 2334-0134. doi: 10.1109/TOH.2012.32.

[100] C. Richard and M. R. Cutkosky. Friction modeling and display in haptic applications involving user performance. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, volume 1, pages 605–611 vol.1, May 2002. doi: 10.1109/ROBOT.2002.1013425.

[101] Jeremy Fishel and Gerald Loeb. Bayesian Exploration for Intelligent Identification of Textures. *Frontiers in neurorobotics*, 6:4, 06 2012. doi: 10.3389/fnbot.2012.00004.

[102] L. Kaim and K. Drewing. Exploratory Strategies in Haptic Softness Discrimination Are Tuned to Achieve High Levels of Task Performance. *IEEE Transactions on Haptics*, 4(4):242–252, October 2011. ISSN 2334-0134. doi: 10.1109/TOH.2011.19.

[103] Christopher Richard, Mark Cutkosky, Rumel Mark, and advisor. On the identification and haptic display of friction. *Proc. Haptic Interfaces Virtual Environ. Teleoper. Syst.*, 01 1999.

[104] M. Strese, Y. Boeck, and E. Steinbach. Content-based surface material retrieval. In *2017 IEEE World Haptics Conference (WHC)*, pages 352–357, June 2017. doi: 10.1109/WHC.2017.7989927.

[105] Cagatay Basdogan and Ayam Srinivasan. Haptic Rendering in Virtual Environments. *Handbook Virtual Environments*, 1:117–134, 08 2002.

[106] A. M. Okamura, M. R. Cutkosky, and J. T. Dennerlein. Reality-based models for vibration feedback in virtual environments. *IEEE/ASME Transactions on Mechatronics*, 6(3):245–252, Sep. 2001. ISSN 1941-014X. doi: 10.1109/3516.951362.

[107] K. J. Kuchenbecker, J. Fiene, and G. Niemeyer. Improving contact realism through event-based haptic feedback. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):219–230, March 2006. ISSN 2160-9306. doi: 10.1109/TVCG.2006.32.

[108] T. Aujeszky, G. Korres, and M. Eid. Measurement-Based Thermal Modeling Using Laser Thermography. *IEEE Transactions on Instrumentation and Measurement*, 67(6):1359–1369, June 2018. ISSN 1557-9662. doi: 10.1109/TIM.2017.2785138.

[109] H. Choi, S. Cho, S. Shin, H. Lee, and S. Choi. Data-driven thermal rendering: An initial study. In *2018 IEEE Haptics Symposium (HAPTICS)*, pages 344–350, March 2018. doi: 10.1109/HAPTICS.2018.8357199.

[110] S. Okamoto and Y. Yamada. Perceptual properties of vibrotactile material texture: Effects of amplitude changes and stimuli beneath detection thresholds. In *2010 IEEE/SICE International Symposium on System Integration*, pages 384–389, Dec 2010. doi: 10.1109/SII.2010.5708356.

[111] M. Dulik and L. Ladanyi. Surface detection and recognition using infrared light. In *2014 ELEKTRO*, pages 159–164, May 2014. doi: 10.1109/ELEKTRO.2014.6847893.

[112] S. Shin and S. Choi. Geometry-based haptic texture modeling and rendering using photometric stereo. In *2018 IEEE Haptics Symposium (HAPTICS)*, pages 262–269, March 2018. doi: 10.1109/HAPTICS.2018.8357186.

[113] G. Campion and V. Hayward. Fundamental limits in the rendering of virtual haptic textures. In *First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics Conference*, pages 263–270, March 2005. doi: 10.1109/WHC.2005.61.

[114] W. McMahan, J. M. Romano, A. M. Abdul Rahuman, and K. J. Kuchenbecker. High frequency acceleration feedback significantly increases the realism of haptically rendered textured surfaces. In *2010 IEEE Haptics Symposium*, pages 141–148, March 2010. doi: 10.1109/HAPTIC.2010.5444665.

[115] S. Choi and K. J. Kuchenbecker. Vibrotactile Display: Perception, Technology, and Applications. *Proceedings of the IEEE*, 101(9):2093–2104, Sep. 2013. ISSN 1558-2256. doi: 10.1109/JPROC.2012.2221071.

[116] L. A. Jones and A. Singhal. Perceptual dimensions of vibrotactile actuators. In *2018 IEEE Haptics Symposium (HAPTICS)*, pages 307–312, March 2018. doi: 10.1109/HAPTICS.2018.8357193.

[117] D. A. Lawrence. Stability and transparency in bilateral teleoperation. *IEEE Transactions on Robotics and Automation*, 9(5):624–637, Oct 1993. ISSN 2374-958X. doi: 10.1109/70.258054.

[118] J. E. Colgate and G. Schenkel. Passivity of a class of sampled-data systems: application to haptic interfaces. In *Proceedings of 1994 American Control Conference - ACC '94*, volume 3, pages 3236–3240 vol.3, June 1994. doi: 10.1109/ACC.1994.735172.

[119] J. J. Abbott and A. M. Okamura. Effects of position quantization and sampling rate on virtual-wall passivity. *IEEE Transactions on Robotics*, 21(5):952–964, Oct 2005. ISSN 1941-0468. doi: 10.1109/TRO.2005.851377.

[120] N. Diolaiti, G. Niemeyer, F. Barbagli, and J. K. Salisbury. Stability of Haptic Rendering: Discretization, Quantization, Time Delay, and Coulomb Effects. *IEEE Transactions on Robotics*, 22(2):256–268, April 2006. ISSN 1941-0468. doi: 10.1109/TRO.2005.862487.

[121] E. Steinbach, S. Hirche, M. Ernst, F. Brandi, R. Chaudhari, J. Kammerl, and I. Vittorias. Haptic Communications. *Proceedings of the IEEE*, 100(4): 937–956, April 2012. ISSN 1558-2256. doi: 10.1109/JPROC.2011.2182100.

[122] V. Nitsch and B. Frber. A Meta-Analysis of the Effects of Haptic Interfaces on Task Performance with Teleoperation Systems. *IEEE Transactions on Haptics*, 6(4):387–398, Oct 2013. ISSN 2334-0134. doi: 10.1109/TOH.2012.62.

[123] Adrin Mora and Antonio Barrientos. An experimental study about the effect of interactions among functional factors in performance of telemanipulation systems. *Control Engineering Practice*, 15(1):29 – 41, 2007. ISSN 0967-0661. doi: https://doi.org/10.1016/j.conengprac.2006.02.017.

[124] Toktam Mahmoodi and Srini Seetharaman. *On using a SDN-based control plane in 5G mobile networks*. Wireless World Research Forum (WWRF), 2014.

[125] A. C. Morales, A. Aijaz, and T. Mahmoodi. Taming Mobility Management Functions in 5G: Handover Functionality as a Service (FaaS). In *2015 IEEE Globecom Workshops (GC Wkshps)*, pages 1–4, Dec 2015. doi: 10.1109/GLOCOMW.2015.7414151.

[126] M. Amani, T. Mahmoodi, M. Tatipamula, and H. Aghvami. Programmable policies for data offloading in lte network. In *2014 IEEE International Conference on Communications (ICC)*, pages 3154–3159, June 2014. doi: 10.1109/ICC.2014.6883806.

[127] I. Giannoulakis, E. Kafetzakis, G. Xylouris, G. Gardikis, and A. Kourtis. On the applications of efficient NFV management towards 5G networking. In *1st International Conference on 5G for Ubiquitous Connectivity*, pages 1–5, Nov 2014. doi: 10.4108/icst.5gu.2014.258133.

[128] M. A. Lema, T. Mahmoodi, and M. Dohler. On the Performance Evaluation of Enabling Architectures for Uplink and Downlink Decoupled Networks. In *2016 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6, Dec 2016. doi: 10.1109/GLOCOMW.2016.7848982.

[129] A. A. Dyumin, L. A. Puzikov, M. M. Rovnyagin, G. A. Urvanov, and I. V. Chugunkov. Cloud computing architectures for mobile robotics. In *2015 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EIConRusNW)*, pages 65–70, Feb 2015. doi: 10.1109/EIConRusNW.2015.7102233.

[130] E. Shin and G. Jo. Uplink frame structure of short TTI system. In *2017 19th International Conference on Advanced Communication Technology (ICACT)*, pages 827–830, Feb 2017. doi: 10.23919/ICACT.2017.7890208.

[131] W. Tarneberg, M. Karaca, A. Robertsson, F. Tufvesson, and M. Kihl. Utilizing Massive MIMO for the Tactile Internet: Advantages and Trade-Offs. In *2017 IEEE International Conference on Sensing, Communication and Networking (SECON Workshops)*, pages 1–6, June 2017. doi: 10.1109/SECONW.2017.8011041.

[132] ITU. Itu-t technology watch report: Tactile internet, 2014. URL http://www.itu.int/oth/T2301000023/en.

[133] M. Maier and A. Ebrahimzadeh. Towards Immersive Tactile Internet experiences: Low-latency FiWi Enhanced Mobile Networks with Edge Intelligence [Invited]. *IEEE/OSA Journal of Optical Communications and Networking*, 11 (4):B10–B25, April 2019. ISSN 1943-0639. doi: 10.1364/JOCN.11.000B10.

[134] Z. Pang, L. Sun, Z. Wang, E. Tian, and S. Yang. A Survey of Cloudlet Based Mobile Computing. In *2015 International Conference on Cloud Computing and Big Data (CCBD)*, pages 268–275, Nov 2015. doi: 10.1109/CCBD.2015.54.

[135] Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990. ISBN 0716710455.

[136] Katta G. Murty and Santosh N. Kabadi. Some NP-complete problems in quadratic and nonlinear programming. *Mathematical Programming*, 39(2): 117–129, Jun 1987. ISSN 1436-4646. doi: 10.1007/BF02592948.

[137] R. C. Jeroslow. There Cannot Be Any Algorithm for Integer Programming with Quadratic Constraints. *Oper. Res.*, 21(1):221–224, February 1973. ISSN 0030-364X. doi: 10.1287/opre.21.1.221. URL http://dx.doi.org/10.1287/opre.21.1.221.

[138] Duan Li and Xiaoling Sun. *Mixed-Integer Nonlinear Programming*, pages 373–395. Springer US, Boston, MA, 2006. ISBN 978-0-387-32995-6. doi: 10.1007/0-387-32995-1_13.

[139] E. Wong, C. M. Machuca, and L. Wosinska. Survivable Hybrid Passive Optical Converged Network Architectures Based on Reflective Monitoring. *Journal of Lightwave Technology*, 34(18):4317–4328, Sept 2016. ISSN 0733-8724. doi: 10.1109/JLT.2016.2593481.

[140] B. P. Rimal, D. Pham Van, and M. Maier. Cloudlet Enhanced Fiber-Wireless Access Networks for Mobile-Edge Computing. *IEEE Transactions on Wireless Communications*, 16(6):3601–3618, June 2017. ISSN 1536-1276. doi: 10.1109/TWC.2017.2685578.

[141] Xiang Sun and Nirwan Ansari. Latency Aware Workload Offloading in the Cloudlet Network. *IEEE Communications Letters*, 21(7):1481–1484, July 2017. ISSN 1089-7798. doi: 10.1109/LCOMM.2017.2690678.

[142] Chee-Hock Ng and Boon-Hee Soong. *Single-Queue Markovian Systems*, pages 103–140. John Wiley & Sons, Ltd, 2008. ISBN 9780470994672. doi: 10.1002/9780470994672.ch4.

[143] M. Jia, J. Cao, and W. Liang. Optimal Cloudlet Placement and User to Cloudlet Allocation in Wireless Metropolitan Area Networks. *IEEE Transactions on Cloud Computing*, 5(4):725–737, Oct 2017. ISSN 2168-7161. doi: 10.1109/TCC.2015.2449834.

[144] Ulrich Faigle, Walter Kern, and Georg Still. *Integer Programming*, pages 173–196. Springer Netherlands, Dordrecht, 2002. ISBN 978-94-015-9896-5.

[145] Convex Over and Under ENvelopes for Nonlinear Estimation (Couenne), 2015. URL https://projects.coin-or.org/Couenne.

[146] William P. Pierskalla. The Multidimensional Assignment Problem. *Operations Research*, 16(2):422–431, 1968. ISSN 0030364X, 15265463.

[147] Pierre Bonami, Mustafa Kilinç, and Jeff Linderoth. Algorithms and software for convex mixed integer nonlinear programs. *Mixed integer nonlinear programming*, pages 1–39, 2012.

[148] Pietro Belotti, Jon Lee, Leo Liberti, Francois Margot, and Andreas Wachter. Branching and Bounds Tightening techniques for Non-convex MINLP. *Optimization Methods Software*, 24(4-5):597–634, August 2009. ISSN 1055-6788. doi: 10.1080/10556780903087124.

[149] A. Spencer, J. Gill, and L. Schmahmann. Urban or Suburban? Examining The Density of Australian Cities in a Global Context. In *The State of Australian Cities Conference 2015*, pages 9–11, Dec 2015.

[150] Stephen Matthews and Daniel M. Parker. Progress in Spatial Demography. *Demographic Research*, S13(10):271–312, 2013. doi: 10.4054/DemRes.2013.28.10.

[151] cnet. Lenovo ThinkStation P900 Workstation Specifications, 2017. URL https://www.lenovo.com/gb/en/workstations/p-series/ThinkStation-P900/p/33TS3TPP900.

[152] B. Skubic, J. Chen, J. Ahmed, L. Wosinska, and B. Mukherjee. A Comparison of Dynamic Bandwidth Allocation for EPON, GPON, and Next-generation TDM PON. *IEEE Communications Magazine*, 47(3):S40–S48, March 2009. ISSN 0163-6804. doi: 10.1109/MCOM.2009.4804388.

[153] S. Mubeen, P. Nikolaidis, A. Didic, H. Pei-Breivold, K. Sandstrm, and M. Behnam. Delay Mitigation in Offloaded Cloud Controllers in Industrial

IoT. *IEEE Access*, 5:4418–4430, 2017. ISSN 2169-3536. doi: 10.1109/ACCESS.2017.2682499.

[154] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.

[155] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. ISBN 0521833787.

[156] Dimitri P. Bertsekas, Angelia Nedic, and Asuman E. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific Publications, 2003. ISBN 1-886529-45-0.

[157] C. Jiang, Y. Chen, Q. Wang, and K. J. R. Liu. Data-Driven Stochastic Scheduling and Dynamic Auction in IaaS. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, Dec 2015. doi: 10.1109/GLOCOM.2015.7417023.

[158] Ronald E. Miller. *Optimization: Foundations and Applications*. John Wiley & Sons, Inc., 1999. ISBN 9781118032930.

[159] Wikipedia. List of Centroids, 2019. URL https://en.wikipedia.org/wiki/List_of_centroids.

[160] Q. Fan and N. Ansari. Cost Aware cloudlet Placement for Big Data Processing at the Edge. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2017. doi: 10.1109/ICC.2017.7996722.

[161] S. Mondal, G. Das, and E. Wong. Efficient Cost-Optimization Frameworks for Hybrid Cloudlet Placement over Fiber-Wireless Networks. *IEEE/OSA Journal of Optical Communications and Networking*, 11(8), August 2019.

[162] Sourav Mondal, Goutam Das, and Elaine Wong. Cost-Optimal Cloudlet Placement Frameworks over Fiber-Wireless Access Networks for Low-Latency Applications. *Journal of Network and Computer Applications*, 138:27 – 38, 2019. ISSN 1084-8045. doi: https://doi.org/10.1016/j.jnca.2019.04.014.

[163] X. Meng, W. Wang, and Z. Zhang. Delay-Constrained Hybrid Computation Offloading With Cloud and Fog Computing. *IEEE Access*, 5:21355–21367, 2017.

[164] H. Zhang, Y. Xiao, S. Bu, D. Niyato, F. R. Yu, and Z. Han. Computing Resource Allocation in Three-Tier IoT Fog Networks: A Joint Optimization Approach Combining Stackelberg Game and Matching. *IEEE Internet of Things Journal*, 4(5):1204–1215, Oct 2017. ISSN 2327-4662. doi: 10.1109/JIOT.2017.2688925.

[165] D. T. Nguyen, L. B. Le, and V. Bhargava. Price-based Resource Allocation for Edge Computing: A Market Equilibrium Approach. *IEEE Transactions on Cloud Computing*, pages 1–1, 2018. ISSN 2168-7161. doi: 10.1109/TCC.2018.2844379.

[166] A. G. Tasiopoulos, O. Ascigil, I. Psaras, S. Toumpis, and G. Pavlou. On-path Cloudlet Pricing for Low Latency Application Provisioning. In *2018 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, pages 31–36, June 2018. doi: 10.1109/LANMAN.2018.8475057.

[167] Theodore Groves and John O Ledyard. Optimal Allocation of Public Goods: A Solution to the "Free Rider" Problem. *Econometrica*, 45(4):783–809, May 1977.

[168] Andrea Schaerf, Yoav Shoham, and Moshe Tennenholtz. Adaptive Load Balancing: A Study in Multi-agent Learning. *Journal of Artificial Intelligence Research*, 2(1):475–500, May 1995. ISSN 1076-9757.

[169] Tatiana Tatarenko. *Game-Theoretic Learning and Distributed Optimization in Memoryless Multi-Agent Systems*. Springer International Publishing, Cham, 2017. ISBN 978-3-319-65479-9. doi: 10.1007/978-3-319-65479-9_2.

[170] Zhu Han, Dusit Niyato, Walid Saad, Tamer Baar, and Are Hjrungnes. *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications*. Cambridge University Press, USA, 1st edition, 2012. ISBN 0521196965.

[171] M. A. L. Thathachar and P. S. Sastry. *Networks of Learning Automata: Techniques for Online Stochastic Optimization*. Springer-Verlag, Berlin, Heidelberg, 2003. ISBN 1402076916.

[172] Abdel Rodrguez, Peter Vrancx, Ricardo Grau, and Ann Now. A Reinforcement Learning Approach to Coordinate Exploration with Limited Communication in Continuous Action Games. *The Knowledge Engineering Review*, 31(1):7795, 2016. doi: 10.1017/S026988891500020X.

[173] Yoav Shoham and Kevin Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, New York, NY, USA, 2008. ISBN 0521899435.

[174] Chayan Bhar, Nilesh Chatur, Atri Mukhopadhyay, Goutam Das, and Debasish Datta. Designing a Green Optical Network Unit Using ARMA-based Traffic Prediction for Quality of Service-aware Traffic. *Photonic Network Communication*, 32(3):407–421, December 2016. ISSN 1387-974X. doi: 10.1007/s11107-016-0671-y.

[175] F. Tang, Z. M. Fadlullah, B. Mao, and N. Kato. An Intelligent Traffic Load Prediction-Based Adaptive Channel Assignment Algorithm in SDN-IoT: A Deep Learning Approach. *IEEE Internet of Things Journal*, 5(6):5141–5154, Dec 2018. ISSN 2327-4662. doi: 10.1109/JIOT.2018.2838574.

[176] IBM Cloud pricing, 2015. URL https://www.ibm.com/cloud/pricing.

[177] Nolan McCarty and Adam Meirowitz. *Political Game Theory: An Introduction*. Analytical Methods for Social Research. Cambridge University Press, 2007. doi: 10.1017/CBO9780511813122.

[178] J. Szp and Ferenc Forg. *Introduction to the Theory of Games*, chapter 4, pages 41–59. Springer Dordrecht, 1985. ISBN 978-94-010-8796-4. doi: 10.1007/978-94-009-5193-8_14.

[179] Y. Narahari. *Game Theory and Mechanism Design*. World Scientific Publishing Company Pte. Limited, 2014. ISBN 9789814525046, 9814525049.

[180] Richard S. Sutton and Andrew G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition, 1998. ISBN 0262193981.

[181] V. V. Phansalkar, P. S. Sastry, and M. A. L. Thathachar. Absolutely Expedient Algorithms for Learning Nash Equilibria. *Proceedings Mathematical Sciences*, 104(1):279–294, Feb 1994. ISSN 0973-7685. doi: 10.1007/BF02830892.

[182] Y. Feng, B. Li, and B. Li. Price Competition in an Oligopoly Market with Multiple IaaS Cloud Providers. *IEEE Transactions on Computers*, 63(1):59–73, Jan 2014. ISSN 0018-9340. doi: 10.1109/TC.2013.153.

[183] Jyrki Wallenius, Peter C. Fishburn, Stanley Zionts, James S. Dyer, Ralph E. Steuer, and Kalyanmoy Deb. Multiple Criteria Decision Making, Multiattribute Utility Theory: Recent Accomplishments and What Lies Ahead. *Management Science*, 54(7):1336–1349, 2008. ISSN 00251909, 15265501.

[184] Dimitri P. Bertsekas and Robert G. Gallager. *Delay Models in Data Networks*, chapter 3, pages 149–270. Prentice Hall, 1992. ISBN 9780132009164.

[185] K. Antonakoglou, X. Xu, E. Steinbach, T. Mahmoodi, and M. Dohler. Toward Haptic Communications Over the 5G Tactile Internet. *IEEE Communications Surveys Tutorials*, 20(4):3034–3059, Fourth quarter 2018. ISSN 2373-745X. doi: 10.1109/COMST.2018.2851452.

[186] Jae-young Lee and Shahram Payandeh. *Haptic Teleoperation Systems: Signal Processing Perspective*. Springer Publishing Company, Incorporated, 2015. ISBN 3319195565, 9783319195568.

[187] L. Ruan, M. P. I. Dias, M. Maier, and E. Wong. Understanding the Traffic Causality for Low-Latency Human-to-Machine Applications. *IEEE Networking Letters*, 1(3):128–131, Sep. 2019. ISSN 2576-3156. doi: 10.1109/LNET.2019.2919712.

[188] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012. ISBN 0262018020.

[189] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

[190] J. A. Hernandez, R. Sanchez, I. Martin, and D. Larrabeiti. Meeting the Traffic Requirements of Residential Users in the Next Decade with Current FTTH Standards: How Much? How Long? *IEEE Communications Magazine*, 57(6): 120–125, June 2019. ISSN 1558-1896. doi: 10.1109/MCOM.2018.1800173.

[191] Zhanat Kappassov, Juan-Antonio Corrales, and Vronique Perdereau. Tactile sensing in dexterous robot hands Review. *Robotics and Autonomous Systems*, 74:195 – 220, 2015. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2015.07.015.

[192] MANUS VR virtual reality gloves, 2019. URL https://manus-vr.com/.

[193] Ken Shoemake. Animating Rotation with Quaternion Curves. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH 85, page 245254, New York, NY, USA, 1985. Association for Computing Machinery. ISBN 0897911660. doi: 10.1145/325334.325242.

[194] Isabel Fraga Alves and Cláudia Neves. *Extreme Value Distributions*, pages 493–496. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-04898-2. doi: 10.1007/978-3-642-04898-2_246.

[195] K. Hornik, M. Stinchcombe, and H. White. Multilayer Feedforward Networks Are Universal Approximators. *Neural Netw.*, 2(5):359366, July 1989. ISSN 0893-6080.

[196] Michael A. Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2018.

[197] Lihua Ruan, Maluge P. I. Dias, and Elaine Wong. Enhancing latency performance through intelligent bandwidth allocation decisions: a survey and comparative study of machine learning techniques. *J. Opt. Commun. Netw.*, 12(4):B20–B32, Apr 2020. doi: 10.1364/JOCN.379715.

[198] H. Liu. On the Levenberg-Marquardt Training Method for Feed-forward Neural Networks. In *2010 Sixth International Conference on Natural Computation*, volume 1, pages 456–460, Aug 2010. doi: 10.1109/ICNC.2010.5583151.

[199] Kumpati S. Narendra and Mandayam A. L. Thathachar. *Learning Automata: An Introduction*. Prentice Hall Englewood Cliffs, N.J, 1989. ISBN 0134855582.

[200] S. Friston and A. Steed. Measuring Latency in Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics*, 20(4):616–625, April 2014. ISSN 2160-9306. doi: 10.1109/TVCG.2014.30.

[201] B. P. Rimal, D. P. Van, and M. Maier. Mobile Edge Computing Empowered Fiber-Wireless Access Networks in the 5G Era. *IEEE Communications Magazine*, 55(2):192–200, February 2017. ISSN 0163-6804. doi: 10.1109/MCOM.2017.1600156CM.

[202] Leonard Kleinrock. An Internet vision: the invisible global infrastructure. *Ad Hoc Networks*, 1(1):3 – 11, 2003. ISSN 1570-8705. doi: https://doi.org/10.1016/S1570-8705(03)00012-X.

[203] Jean-Yves Le Boudec and Patrick Thiran. *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Springer-Verlag, Berlin, Heidelberg, 2001. ISBN 354042184X.