Adaptive Reinforcement Learning for Heterogeneous Network Selection

by

Duong Duc Nguyen

BSc (Hons) in Electronic Communication Systems,
University of Plymouth, UK, 2008.
MSc in Mobile and Personal Communications,
King's College London, UK, 2009.

Thesis submitted for the degree of

Doctor of Philosophy

in

School of Electrical and Electronic Engineering,
Faculty of Engineering, Computer and Mathematical Sciences
The University of Adelaide, Australia

June, 2018

Supervisors:

Prof Langford Barton White, School of Electrical & Electronic Engineering

Prof Cheng-Chew Lim, School of Electrical & Electronic Engineering

Dr Hung Xuan Nguyen, Teletraffic Research Centre

THE UNIVERSITY *of* ADELAIDE

Contents

Abstract

Next generation 5G mobile wireless networks will consist of multiple technologies for devices to access the network at the edge. One of the keys to 5G is therefore the ability for device to intelligently select its Radio Access Technology (RAT). Current fully distributed algorithms for RAT selection although guaranteeing convergence to equilibrium states, are often slow, require high exploration times and may converge to undesirable equilibria. In this dissertation, we propose three novel reinforcement learning (RL) frameworks to improve the efficiency of existing distributed RAT selection algorithms in a heterogeneous environment, where users may potentially apply a number of different RAT selection procedures. Although our research focuses on solutions for RAT selection in the current and future mobile wireless networks, the proposed solutions in this dissertation are general and suitable to apply for any large scale distributed multi-agent systems.

In the first framework, called RL with Non-positive Regret, we propose a novel adaptive RL for multi-agent non-cooperative repeated games. The main contribution is to use both positive and negative regrets in RL to improve the convergence speed and fairness of the well-known regret-based RL procedure. Significant improvements in performance compared to other related algorithms in the literature are demonstrated.

In the second framework, called RL with Network-Assisted Feedback (RLNF), our core contribution is to develop a network feedback model that uses network-assisted information to improve the performance of the distributed RL for RAT selection. RLNF guarantees no-regret payoff in the long-run for any user adopting it, regardless of what other users might do and so can work in an environment where not all users use the same learning strategy. This is an important implementation advantage as RLNF can be implemented within current mobile network standards.

In the third framework, we propose a novel adaptive RL-based mechanism for RAT selection that can effectively handle user mobility. The key contribution is to leverage forgetting methods to rapidly react to the changes in the radio conditions when users move. We show that our solution improves the performance of wireless networks and converges much faster when users move compared to the non-adaptive solutions.

Another objective of the research is to study the impact of various network models on the performance of different RAT selection approaches. We propose a unified benchmark to compare the performances of different algorithms under the same computational environment. The comparative studies reveal that among all the important network parameters that influence the performance of RAT selection algorithms, the number of base stations that a user can connect to has the most significant impact. This finding provides some guidelines for the proper design of RAT selection algorithms for future 5G. Our evaluation benchmark can serve as a reference for researchers, network developers, and engineers.

Overall, the thesis provides different reinforcement learning frameworks to improve the efficiency of current fully distributed algorithms for heterogeneous RAT selection. We prove the convergence of the proposed reinforcement learning procedures using the differential inclusion (DI) technique. The theoretical analyses demonstrate that the use of DI not only provides an effective method to study the convergence properties of adaptive procedures in game-theoretic learning, but also yields a much more concise and extensible proof as compared to the classical approaches.

Statement of Originality

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I give consent to this copy of my thesis when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

I acknowledge that copyright of published works contained within this thesis resides with the copyright holder(s) of those works.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

Signed

Date

Acknowledgments

First and foremost, I would like to convey my deep gratitude to my principal supervisor, **Professor Langford B. White** for his constant support throughout my PhD candidature at the University of Adelaide. He was the one who brought me to the field of game theory. He has not only inspired in me the passion for research but also motivated me to explore new knowledge. To me, he is a wonderful supervisor whose advice I always trust.

Second, I would like to mention my co-supervisors, **Dr Hung X. Nguyen** and **Professor Cheng-Chew Lim**. I am thankful to have Dr Hung X. Nguyen as my co-supervisor. Working under his supervision, I gained and started to build up my research skills and experiences. I am very grateful to him for patiently helping me to improve my writing skills. I am also strongly indebted to Professor Cheng-Chew Lim, who has offered me much useful feedback and advice on my research works.

This research would not be possible without the generous financial help in the form of the Beacon of Enlightenment PhD Scholarship from the University of Adelaide. I also would like to acknowledge the Da Nang People's Committee who allowed me to continue my PhD study in Australia after four year sponsoring my BSc and MSc degrees in the UK.

In addition, I wish to express my gratitude to my wonderful colleagues and staffs at the School of Electrical and Electronic Engineering (especially my friend George Stamatescu) who have given me a friendly second home for the past three and a half years. I also would like to thank my Vietnamese friends, including Long, Thanh, My, Min, Loc, Huyen, Nghia and Nguyen, for helping me balance my academic and social life.

Last but not least, I wish to give my endless love and deep appreciation to my wife Dung, my daughter Sola and my parents who have always been by my side in any situations and supported me unconditionally. Their love has guided me through this challenge journey.

Duong Duc Nguyen
April 2018

Thesis Conventions

The following conventions have been adopted in this thesis:

Typesetting

This document was compiled using LaTeX2e. TeXstudio were used as text editor interfaced to LaTeX2e. Inkscape was used to produce schematic diagrams and other drawings.

Referencing

Referencing and citation style in this thesis are based on the Institute of Electrical and Electronic Engineers (IEEE) Transaction style.

System of units

The units used in this thesis are based on the International System of Units (SI units).

Spelling

The Australian English spelling is adopted in this thesis.

Publications

Journal Articles

1. **D. D. Nguyen**, H. X. Nguyen and L. B. White, "Reinforcement Learning With Network-Assisted Feedback for Heterogeneous RAT Selection," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 6062-6076, Sep. 2017.

Conference Articles

1. **D. D. Nguyen**, L. B. White and H. X. Nguyen, "Adaptive Multiagent Reinforcement Learning with Non-positive Regret," *29th Australasian Joint Conference on Artificial Intelligence*, Hobart, Australia, 2016, pp. 29-41.

2. **D. D. Nguyen**, H. X. Nguyen and L. B. White, "Performance of Adaptive RAT Selection Algorithms in 5G Heterogeneous Wireless Networks," *2016 IEEE 26th International Telecommunication Networks and Applications Conference (ITNAC)*, Dunedin, New Zealand, 2016, pp. 70-75.

3. H. X. Nguyen, T. Pham, K. Hoang, **D. D. Nguyen** and E. Parsonage, "A Prototype of Policy Defined Wireless Access Networks," *2016 IEEE 26th International Telecommunication Networks and Applications Conference (ITNAC)*, Dunedin, New Zealand, 2016, pp. 101-106.

Papers under Review or Revision

1. **D. D. Nguyen**, H. X. Nguyen and L. B. White, "Evaluating Performance of RAT Selection Algorithms for 5G HetNets," submitted to *IEEE Access*, under revision.

2. **D. D. Nguyen**, L. B. White and H. X. Nguyen, "Adaptive Reinforcement Learning With Controlled Use of Forgetting Factor for Dynamic RAT Selection," submitted for publication, under review.

List of Acronyms

3GPP	Third Generation Partneship Project
5G	Fifth Generation of Broadband Cellular Network Technology
AP	Access Point
ARLFF	Adaptive Reinforcement Learning With Forgetting Factor
BS	Base Station
CDMA	Code Division Multiple Access
CDF	Cumulative Distribution Function
CE	Correlated Equilibrium
CODIPAS	Combined fully Distributed Payoff and Strategy
CQI	Channel Quality Indicator
DI	Differential Inclusion
EDGE	Enhanced Data Rates for GSM Evolution
ERL	Enhanced Reinforcement Learning
eNB	Evolved Node Base Station
FDMA	Frequency Division Multiple Access
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HetNets	Heterogeneous Wireless Networks
HSS	Highest Signal Strength
LSH	Local Search Heuristic
LTE	Long-Term Evolution

Mbps	Megabits per second
MMS	Multimedia Messaging Service
NE	Nash Equilibrium
ODE	Ordinary Differential Equation
PHY	Physical layer
RM	Regret Matching
RSG	RAT Selection Game
RSS	Received Signal Strength
RL	Reinforcement Learning
RAT	Radio Access Technology
RLNF	Reinforcement Learning With Network-Assisted Feedback
SDN	Software Defined Networking
SNR	Signal-to-Noise Ratio
TDMA	Time Division Multiple Access
UE	User Equipment
UMTS	Universal Mobile Telecommunications System
WiFi	Wireless Fidelity, Wireless Internet
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network

List of Figures

List of Tables

Chapter 1

Introduction

THIS first chapter highlights the significance and motivation of the research presented in this thesis, which focuses on the design of new adaptive reinforcement learning based algorithms for radio access technology (RAT) selection using the application of game theory. It first provides the background of RAT selection techniques in heterogeneous wireless networks and adaptive procedures in game-theoretic learning. The objectives of the thesis are then presented, followed by an overview of the thesis structure and its contributions.

1.1 Background and Motivation

1.1.1 5G Heterogeneous Wireless Networks

The rapid developments of wireless technologies enable people to access to the Internet as well as connect to people easily from anywhere at any time. Figure 1.1 and Table 1.1 illustrate the evolution of cellular networks from the first generation (1G) towards the existing 4G networks. This section describes the evolution of cellular and wireless communication networks. The development and key challenges in achieving 5G are also presented.

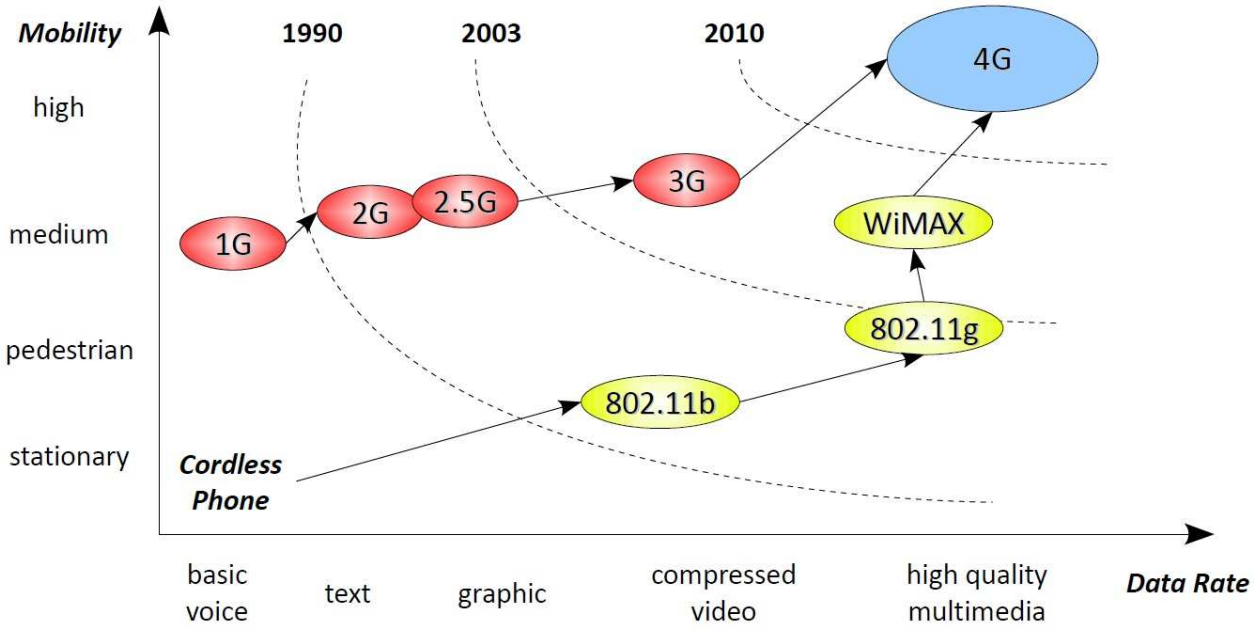

Figure 1.1. Evolution of wireless communication network [1].

The 1G networks, which are almost analogue systems and are mainly used for voice services, offer data rates around 2.4 Kbps and use frequency division multiple access (FDMA) for radio transmission. The later 2G networks, which are well known as the Global System for Mobile communications (GSM), are digital systems and offer data rates up to 9.6 Kbps. 2G systems use Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA) for radio transmission and are mainly used for voice services and slow data transmission. GSM (2G) was introduced in 1990 and then evolved with newer technologies including GPRS (2.5G) and EDGE (2.75G). Instead of using TDMA/FDMA,

Table 1.1. Mobile wireless communication technologies before 5G

Generation	Data rate	Technology	Multiplexing	Service
1G	2.4 Kbps	AMPS	FDMA	Voice only
2G	9.6 Kbps	GSM	TDMA/CDMA	Voice data
2.5G	114 Kbps	GPRS		Voice data, MMS, Internet
2.75G	400 Kbps	EDGE		
3G	384 Kbps	UMTS	CDMA	High speed access to voice, data, video services
3.5G	2 Mbps	HSDPA		
3.75G	30 Mbps	HSPA+		High speed internet, multimedia
4G	100 Mbps	LTE	OFDMA	High speed applications, mobile TV, real time streaming
	128 Mbps	WiMAX		

3G networks use Code Division Multiple Access (CDMA) for radio transmission and offer high data rate of 2Mbps. Later 3G releases, denoted by 3.5G and 3.75G, provide data rate of several Mbps and can support various applications such as wireless voice telephony, mobile Internet access and video calls. The current 4G networks, which are Long term evolution (LTE) and Worldwide Interoperability for Microwave Access (WiMAX) systems, provide the same features as 3G with much faster data transfer (up to 100 Mbps) than previous generations.

With the 4G systems being rolled out worldwide, 5G mobile and wireless technologies are emerging into research fields. 5G networks, compared to 4G, are expected to support diverse requirements of various applications and services in the future. To do so, 5G will be capable of interconnecting both new radio access technologies (RATs) and most of existing wireless technologies (i.e., LTE, WiMAX, UMTS, GSM and WiFi, femto, etc) [2]. Heterogeneous wireless networks (HetNets) that consist of multiple wireless access technologies are therefore the key components of future 5G networks [3]. Current research reveals that the major requirements for future 5G HetNets include [4–7]:

- Capacity: provides significant performance gains in system capacity;
- Data rate: is higher than 1Gbps at low mobility and 100Mbps at high mobility;
- Latency: is less than 1 millisecond to support real time mobile control;

- Connectivity: allows massive number of simultaneously connected users;
- Other factors: provides higher energy efficiency and reduces network cost.

1.1.2 Radio Access Technology (RAT) Selection

As mentioned above, HetNets, which are the coexistence and interworking of multiple wireless access technologies (such as WiFi, 3G, 4G and potential 5G technologies), are expected to be the key enablers of future 5G. In these networks, mobile devices with multiple radio interfaces are able to switch to the most suitable RATs among the available alternatives. Deciding which technology, and which individual base station (BS) supporting that technology mobile users should connect to, is known as the RAT selection problem [8], and is a topic of much on-going work within the LTE-WLAN interworking framework of the Third Generation Partneship Project (3GPP) [9] and in 5G research [10–12]. Choosing the appropriate RAT a wireless device connects to for good performance is vital but non-trivial. In the following, we discuss the key challenges to the RAT selection mechanisms for future 5G networks, which is also the main motivation of this research.

Nowadays with the growing demand for mobile data, cellular service operators need a solution to successfully handle mobile traffic demand to continue deliver high-quality services to subscribers. WiFi is currently the attractive offload solution for mobile operators because of its low cost, simple architecture and ability for quick deployment. Figure 1.2 illustrates the overview of interworking scenarios between cellular and WiFi networks. There are several other reasons for WiFi being a favourite to offload data traffic, include:

- Service extension to non-cellular devices: Most modern electronic devices, from laptops to tablets and cameras, contain WiFi radio. Integrating WiFi into cellular networks will allow the mobile operator to extend data services to non-cellular devices.
- Cost-effective additional capacity: WiFi networks operate in unlicensed spectra. Therefore, end users also see value in offloading to WiFi since it allows them to gain access to the Internet without incurring transport cost.
- Improving user experiences: New WiFi standard like Hotspot 2.0 [13] will enable seamless, secure roaming, improved network discovery and selection for the user.

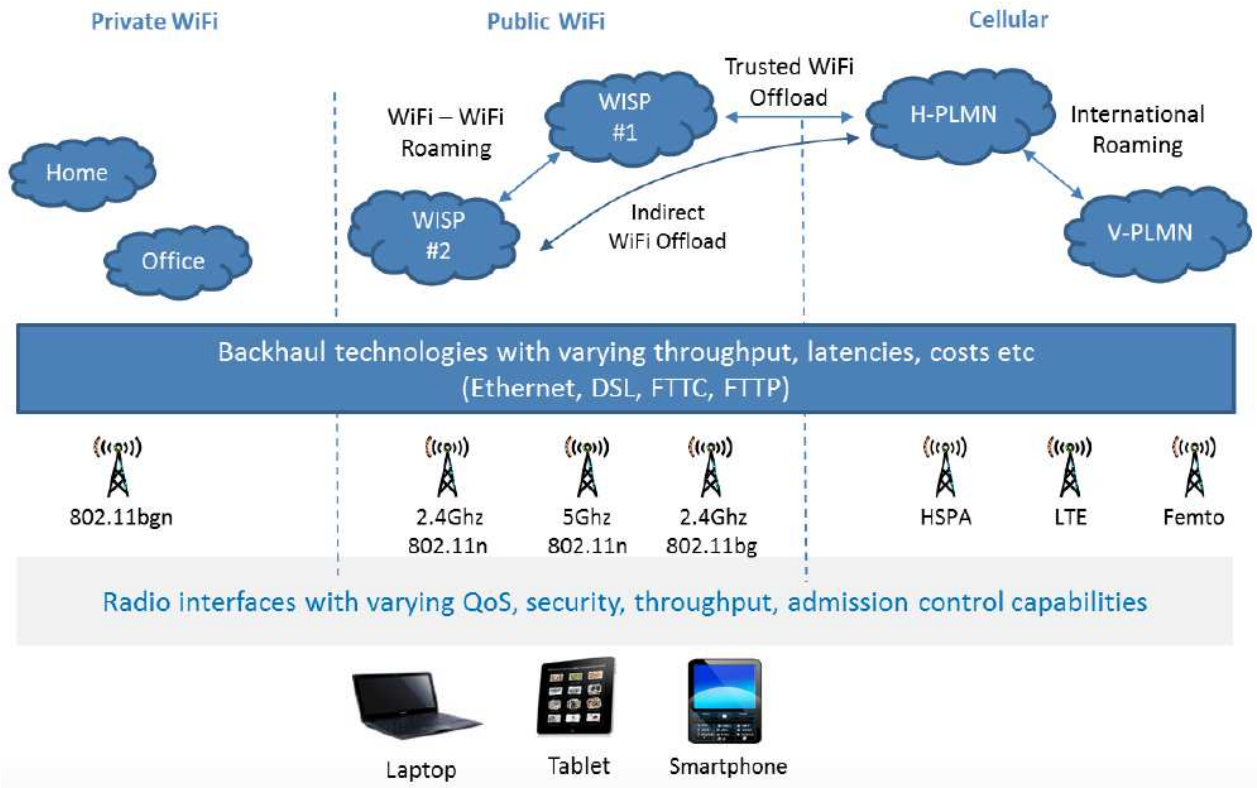

Figure 1.2. Overview of interworking scenarios between cellular and WiFi networks [14]

One key challenge to consider when integrating WiFi into cellular network in future 5G is the ability to manage user association, avoid unnecessary handover and maintain seamless connectivity when handing over between WiFi and other cellular access technologies such as UMTS, WiMAX and LTE. As a result, an intelligent RAT selection technique, that allows users to smartly choose which network to connect with for optimal performance, is needed. In addition, as more and more devices are capable of operating on multiple RAT types, intelligent RAT selection becomes more important.

It is expected that the optimal RAT association can be achieved when network selection decisions are made based on information available from both operator policies and user preferences. According to this, there are several important factors that need to take into account to enhance the RAT selection process, including:

Network conditions

Since users generally have no information on the global network condition, their RAT selection decisions may be in no user's long-term interest, causing performance degradation

and sometimes oscillation or instability [15]. Thus, real-time network load condition of the access network, where the load could be the number of connected users or the current consumption of the available resources (for example, bandwidth or uplink interference), is an important factor that could be used to improve the RAT decisions [5, 14].

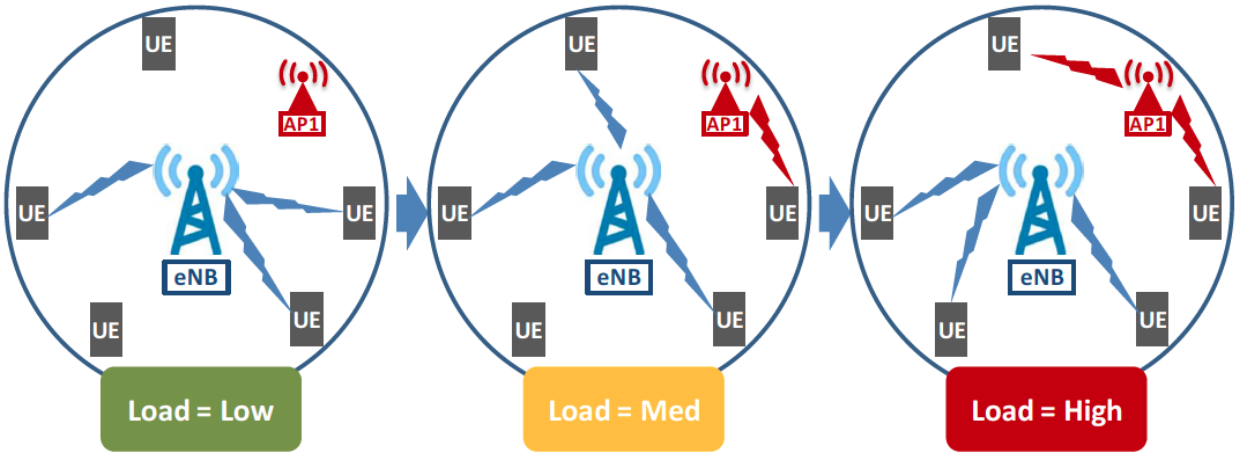

Figure 1.3. Example of RAT association based on cellular load conditions [14]

Figure 1.3 illustrates an example of RAT association based on the information from cellular load conditions. Assuming in future 5G scenarios, a unique network operator controls both cellular and WiFi networks in a given area. When the cellular network is not congested (Load = Low), the network operator prefers to serve their customers via the cellular BS. As the load increases (Load = Med), that operator may want to start steering some of the users towards the WiFi network. As the cellular network is in heavy load condition (Load = High), that operator may want to steer even more users towards the WiFi BS.

User mobility

Another factor, which is also very important to the RAT selection problem, is handling user mobility [2]. Mobility of users occurs frequently and makes the data rate unstable. Thus, it is important to take into account the user mobility when designing RAT selection mechanism in order to maintain optimal performance when network condition changes due to user's movements [1, 16, 17]. In future 5G, not only the network is responsible for handling user mobility, but also the user can make final RAT choice among different potential RATs based on its mobility situation as the user knows when it moves.

1.1.3 Game Theory Based Solution for RAT Selection

Game theory is a mathematical tool to model the interaction of decision makers with conflicting interests. It is mainly used in economics to model competition between companies. Nowadays, game theory is applied to a wide range of areas such as: biology, politics, law, psychology, computer science and engineering. Recently, its application was introduced in wireless networks, especially in wireless sensor networks [18], cognitive radio networks [19] and ad-hoc networks [20], to solve many related problems such as power control, resource allocation, medium access control, and cell selection [21].

The main components of a game are: the set of players, the set of actions, the set of payoffs and also the information sets available to the players [22]. In any game, each individual player tries to choose a suitable strategy from its actions set with the aim of maximising its payoff. The combination of best strategies for all players is known as an equilibrium. A strategy can be seen as a rule for choosing an action, and can be either pure (deterministic), or mixed (stochastic). In a pure strategy, a player chooses an action deterministically from its actions set; whereas in a mixed strategy, a player chooses an action randomly according to a probability distribution function (PDF) on its actions set.

Nash equilibrium (NE) and correlated equilibrium (CE) are the two commonly studied solution concepts in game theory. CE is an optimality concept introduced by Aumann [23] that generalises the NE. It is relevant to probabilistic games, namely where strategies are determined probabilistically. CE models possible correlation or co-ordination between players' actions compared to the usual strategic equilibrium of Nash, where all players act independently. The game is said to have reached a CE if each player does not benefit from choosing any other probability distribution over its actions, provided that all the other players do likewise. When each player chooses their action independently of the other players, or without any implicit co-ordination mechanism, a CE is also a NE. Some games can have more than one NE or might not have a NE [24]. Whereas, CE is proven to exist for any finite games with bounded payoffs [25].

Game theory has been used to model and analyse the cooperative or non-cooperative interaction between users and/or networks for RAT selection problem in wireless networks (for a survey refer to [38]). In a cooperative game, players collaborate in order

Table 1.2. Summary of game theory approaches on RAT selection

Category	Game model	Game type	RAT	Reference
Users versus Users	Evolutionary Game	Non-cooperative	WLAN	[26, 27]
	Bayesian Game	Non-cooperative	WLAN, CDMA, WiMAX	[28]
	Congestion Game	Non-cooperative	WLAN	[29]
Users versus Networks	Auction Game	Non-cooperative	HSDPA, WLAN	[30–32]
	Repeated Game	Cooperative	Not specified	[33]
Networks versus Networks	Strategic Game	Non-cooperative	WiMAX, WLAN, 4G	[34–36]
	Strategic Game	Non-cooperative	Not specified	[37]

to achieve mutual advantage. Unlike the cooperative game, players make decision independently with only aim at maximising their own benefit in a non-cooperative game. Different game models (Strategic game, Bayesian game, Auction game, etc.) are used to model the cooperative or non-cooperative game under different scenarios (users versus users, users versus network, and networks versus networks) [39]. Most of related works formulate the RAT selection problems as non-cooperative games, while only few works look at cooperative behaviour. Table 1.2 summaries a comparison of the state-of-the-art of game theoretic solution for RAT selection.

In this thesis, we focus on applying game theoretic approach to find correlated equilibrium as a solution for the multi-user RAT selection problem in HetNets. The RAT selection problem could be considered as a repeated game, in which mobile users are the players and the users' actions correspond to the selection of RATs. Users select their associated RATs with the objective of maximising their long-run average payoffs (througputs). The main advantage of reaching a CE is from the fact that by allowing the players to coordinate their actions, a CE can provide a balance between the non-cooperative solution (where all the players work independently but may yield poor performance) and the fully cooperative solution (which requires coordination between players but can be highly efficient). In fact, the set of CE is more natural than the set of NE in decentralised adaptive learning environments since the common history observed by all players can serve as a natural coordination mechanism [40, 41].

1.1.4 Multi-agent Reinforcement Learning for RAT Selection Games

In this research, we are interested in developing novel adaptive learning procedures that can compute a correlated equilibrium solution. Several adaptive algorithms can be used to achieve convergence to a stable CE in a repeated game, including regret matching in [40] and its fully distributed variant — a reinforcement learning (RL) based regret minimisation algorithm in [41]. To guarantee convergence, the regret matching algorithm in [40] and most existing distributed RAT selection algorithms [15, 42–47] require that all users have a global knowledge of the network including the payoff function, and the selection histories of the other users. From these, they are able to determine their own payoffs (throughputs) given other users' choices. This assumption implies that each user knows the instantaneous throughputs of the other users. The guaranteed convergence therefore comes at the cost of increased complexity, signaling and communication load.

In contrast, the fully distributed reinforcement leaning algorithm, such as the algorithm in [41], does not require a user to know anything about the other users. Indeed, the users do not even need to know that they are parts of a RAT selection "game". Each user learns about the "game" by observing only its own achieved payoffs. Over time, using only this information, a user can rationally choose the best course of actions to maximise its utility. Under mild conditions of finite payoffs and of unchanged network conditions, the RL-based regret minimisation algorithm in [41] is guaranteed to converge to a stable set of correlated equilibria. Despite this very attractive property, this RL-based algorithm in [41] and other conventional RL-based algorithms, however, suffer from the problems of slow convergence, and of convergence to sub-optimal equilibria due to the lack of knowledge on global network traffic, making them unsuitable for RAT selection in real networks where the environment can change quickly [48].

In this thesis, we address these critical issues and provide solutions to improve the efficiency of existing distributed solutions based on reinforcement learning mechanism for RAT selection. Furthermore, we not only focus on developing advanced reinforcement learning frameworks that lead users' behaviour converge to a set of correlated equilibria, but also study the interaction between user and network in order to obtain the best trade-off between satisfying user preferences and optimising overall network utility at the same time, as these approaches have not been fully addressed in the literature.

1.2 Thesis Objectives

The main objectives of the research are to develop advanced reinforcement learning frameworks to improve the efficiency of existing distributed solutions based on conventional reinforcement learning mechanism for heterogeneous RAT selection. Three novel frameworks are proposed in this thesis with the aim of overcoming the above-mentioned challenges in Section 1.1 facing future 5G systems. In particular, our proposed solutions aim to meet expected requirements of both users' demand and network's objective in speeding up the RAT selection process; improving per-user data rate, system fairness and overall network performance; supporting user mobility and a large number of connected devices. A detailed list of contributions for each proposed frameworks will be shown in the next section as an overview of the thesis. Throughout the thesis, both theoretical and experimental analyses are provided to validate the effectiveness of the proposed solutions.

Another objective of the research is to investigate the impact of different classes of network topology and bandwidth allocation models on the performance of various approaches to the RAT selection problem. Understanding the performance and limitation of different RAT selection solutions under various network models is important for their deployment. Based on the thorough comparative study conducted, advantages and disadvantages of the different RAT selection solutions are discussed. A unified simulation benchmark for comparing different algorithms under the same computational environment is proposed, which provides a useful tool to evaluate the effect of different network parameters on the performances of RAT selection algorithms. Recommendations for the proper design and evaluation of RAT selection algorithms for future 5G networks are also provided as a reference for further studies on this topic.

1.3 Thesis Overview and Original Contributions

The thesis consists of six chapters and one appendix. The topic and original contributions in each chapter are described in details as follows.

- Chapter 1 provides the background and motivation for the research presented in this thesis, and gives an overview of its original contributions to knowledge.

- Chapter 2 studies the problem of multi-agent non-cooperative repeated games. We propose a novel adaptive reinforcement learning framework that adopts both positive and negative regret measures in reinforcement learning to improve the convergence speed and fairness of the well-known RL-based regret minimisation procedure. We prove theoretically that the empirical distribution of the joint action of all learning agents converges to the set of correlated equilibrium. Simulation results are conducted to confirm the robustness and superiority of the proposed algorithm, especially in a large-scale distributed non-cooperative multi-agent system.

- Chapter 3 develops a network feedback model that uses limited network-assisted information to obtain fast convergence, low overhead and competitive user fairness and network utility for the RAT selection process. We prove theoretically that a fully distributed algorithm developed within this framework is guaranteed to converge to a set of correlated equilibria. We perform extensive simulation with realistic network scenarios to demonstrate the improved performance of our algorithm compared to other existing related algorithms. More importantly, our framework can flexibly support a wide range of network-assisted feedback and guarantees, at a theoretical level, no-regret in achieving average per-user throughput for any user adopting it, irrespective of the behaviour of other users. Thus, this solution is highly efficient to use in a heterogeneous environment, where users may potentially apply a number of different RAT selection procedures to select their associated wireless networks. This is an important implementation contribution as our solution can be implemented within current mobile network standards.

- Chapter 4 first conducts a brief overview of existing RAT selection algorithms and the different network models that were used to evaluate these works. Based on the algorithms' attributes, we classify them into centralised, distributed and hybrid based approaches. We then combine these different network models to build a unified benchmark for evaluating RAT selection algorithms in a 5G environment. Our benchmark covers a wide range of network models from throughput, connectivity between users and BSs, BS deployment, and mobility. We implement the representative algorithms of different approaches and cross compare them in our benchmark.

From the comprehensive experiments conducted, we illustrate how the different network parameters, such as link density (the number of BSs that a user sees), user density (the number of users per BS) and bandwidth distribution (the distribution of link bandwidth between BSs and users) could impact the performance of these algorithms. Importantly, our thorough comparative study reveals that RAT selection algorithms should be evaluated on a range of network model parameters, especially the number of BSs available to a user, to fully understand their limitations. Our findings and the unified evaluation benchmark in this chapter contribute as a reference for more effective design of RAT selection algorithms for future 5G.

- Chapter 5 addresses the challenge of handling user mobility for RAT selection by proposing a solution that takes user mobility into account. Previous solutions on distributed RAT selection cannot converge fast enough and hence perform poorly in networks with high mobility. The key contribution of this chapter is to develop a new adaptive reinforcement learning framework that leverages benefit of forgetting properties to rapidly react to the changes in the network due to various mobility situations of mobile users. In our solution, instead of using a constant forgetting factor for all users, we use an adjustable forgetting factor for a different user. Using our learning technique, a user can adaptively identify the change in the network when it moves and effectively re-select its appropriate associated RAT in order to quickly adapt to the fluctuations of its throughput due to its mobility. We prove theoretically that the proposed algorithm guarantees the long-term achievable throughput for any user adopting it no worse than choosing any fixed BS, no matter how the other user may do; and converges almost surely to the set of correlated equilibria when all users apply it. Using simulation with realistic network settings, we demonstrate the adaptability and performance improvement of our adaptive learning scheme compared with non-adaptive solutions under different user mobility models, including random mobility and group mobility scenarios.

- Chapter 6 concludes this thesis, by summarising its major results, and provides some possible directions for future works and improvements.

Chapter 2

Reinforcement Learning With Non-positive Regret

THIS chapter proposes a novel adaptive reinforcement learning (RL) procedure for multi-agent non-cooperative repeated games. Most existing regret-based algorithms only use positive regrets in updating their learning rules. In this chapter, we adopt both positive and negative regrets in reinforcement learning to improve its convergence behaviour. We prove theoretically that the empirical distribution of the joint play converges to the set of correlated equilibrium. Simulation results demonstrate that our proposed procedure outperforms the standard regret-based RL approach and a well-known state-of-the-art RL scheme in the literature in terms of both computational requirements and system fairness. Further experiments show that the performance of our solution is robust to variations in the total number of agents in the system; and that it can achieve markedly better fairness performance when compared to other relevant methods, especially in a large-scale multiagent system.

2.1 Introduction

Reinforcement learning (RL) is a popular adaptive procedure used in distributed system and has been widely studied in artificial intelligence (AI) research areas (for a survey on recent developed RL algorithms refer to [49]). A RL procedure [41, 48, 50–52] does not require the agents to know anything about the entire environment, except their local information. Each agent learns about the environment by observing its own payoffs. Over time, using only this information, it can rationally choose the best course of actions to maximise its objective utility (payoff). Under mild conditions of finite payoffs and of stationary environment, an RL procedure is guaranteed to converge to a set of stable equilibria.

Despite this very attractive property, RL procedure applied in multiagent settings suffers from two well-known problems of slow convergence and of convergence to sub-optimal equilibrium points that yield unfair resource allocation or inefficient utilisation of available resources, especially in a distributed system with a very large number of agents [50]. Another challenge of RL-based algorithms is the inefficiency of exploration. Since agents running RL procedure do not have a global knowledge of the whole system, they often require a high exploration time in order to converge to a stable equilibrium. In many applications, these behaviours can result in undesirable outcomes such as [48, 53].

This chapter introduces a new RL procedure that follows the regret-based principles [40, 41] to overcome the disadvantage of slow speed and inefficient convergence of standard RL solutions. The notion of regret has been explored both in game theory and computer science [40, 41, 54, 55]. Regret measures reflect how much worse in payoffs that an agent would experience if choosing other options instead of its current selection. In our problem formulation, we consider a multiagent non-cooperative repeated game with restricted information for the agents. Each agent only observe its own payoffs and knows neither its payoff function, which depends on the other agents' (unknown) actions, nor the information on the other agents in the game. The goal of every agent is to guarantee no-regret in the long-term (average) payoffs.

Unlike most the existing regret-based algorithms that use only positive parts of regret measures to update the play probability and completely ignore negative regrets, we propose to use both positive and negative regrets to accelerate the convergence of the RL

procedure. Our new approach is motivated by the observation that incorporation of negative regrets can help the agent to "explore" the environment more extensively as positive regrets decrease than the standard RL algorithm. The fact is that considering negative regrets can help agents make more "good" decisions by reducing unnecessary explorations on the actions that result in poor performances. Thus, more effective exploration has crucial impact on the convergence speed as well as the performance of the learning outcome.

However, since there is a negative impact on average performance by including actions with negative regrets, our approach weighs the impact of negative regrets on the probability distribution of actions in a manner that ensures (i) that actions with large (magnitude) negative regrets contribute less to the probability of choosing those actions than those with small (magnitude) negative regrets and (ii) that the contribution of negative regrets decreases to zero over time.

The main contribution of this chapter are as follows:

1. *A Novel Adaptive Multiagent Reinforcement Learning Procedure:* We propose a novel fully distributed RL procedure that uses both positive and negative regret measures to improve convergence speed and fairness of the well-know regret-based RL procedure. We show that our solution is suitable for large-scale distributed multiagent systems.

2. *Our proof methodology:* We prove the convergence of our proposed procedure using differential inclusion (DI) technique. DI is a powerful theoretical framework that derived from the expected motion of a stochastic process. This chapter demonstrates that the use of DI technique is particularly suitable to study the convergence behaviours of the regret based schemes and adaptive procedures in game theory, and provide a much more concise and extensible proof as compared to the classical approaches.

2.2 Background

This section reviews the background and notation used in this chapter.

2.2.1 Game Model

We consider a game with A players denoted by the set $\{1, \ldots, A\}$ for some (finite) integer $A \geq 2$. Each player a has its set of actions (moves) $\mathcal{S}_a = \{1, \ldots, m\}$, where m is the number of action of player a. For notational simplicity, we assume that m is the same for all players. The set of all possible moves is the Cartesian product $\mathcal{S} = \Pi_{a=1}^{A} \mathcal{S}_a$. We view the game from the point of view of player one. Let $\mathcal{I} = S_1$ denote the set of moves of player one and $\mathcal{L} = \mathcal{S} \setminus \mathcal{S}_1$ the set of moves of all other players. Denote by X, the set of all probability mass functions (pmf) on $\mathcal{I}$ and Y the set of pmf on $\mathcal{L}$. Let Z denote the set of pmf on $\mathcal{S}$, then $X \times Y$ is a subset of $\mathcal{Z}$ comprised of all pmf of the form $z = (x, y)$ where $x \in X$ and $y \in Y$, i.e. all pmf where the probability of the action of player one and the actions of all other players taken together, are statistically independent.

Let $U : \mathcal{S} \to \mathbb{R}$ denote the payoff achieved by player one when the overall action taken by all players is $s \in \mathcal{S}$. We represent a strategy in the form $s = (i, \ell)$ where i is the action of player one and ℓ is the action of all other players. We will consider the general formulation of game where users apply mixed strategies over the possible selection set $\mathcal{S}$. Under randomised actions with overall probability (pmf) $z \in Z$, the payoff obtained by player one is defined by extending the domain of definition of U to Z according to

$$U(z) = \sum_{k \in \mathcal{S}} z(k)\, U(k) \ . \tag{2.1}$$

The multiagent game model then can be denoted by $\mathcal{G} = (\mathcal{A}, (\mathcal{S}_a)_{a \in \mathcal{A}}, (U_a)_{a \in \mathcal{A}})$.

2.2.2 Equilibrium States

In this chapter, we are interested in a popular notion of rationality that generalises the Nash equilibrium called correlated equilibrium. It is an optimality concept introduced by Aumann [23]. It models possible correlation or co-ordination between players compared to the usual strategic equilibrium of Nash, where all players act independently. Correlated equilibrium is relevant to the probabilistic game, namely where strategies are determined probabilistically. Denote by ψ, a probability distribution defined in $\mathcal{S}$, the ψ is said to be a correlated equilibrium for the game $\mathcal{G}$ if for every player $a \in \mathcal{A}$, and for every pair of

action $j, k \in \mathcal{S}_a$, it holds that

$$\sum_{s \in \mathcal{S}: i=j} \psi(s)(U(k, \ell) - U(s)) \leq 0. \tag{2.2}$$

A correlated equilibrium results if each player does not benefit from choosing any other action, provided that all other players do likewise. When each player chooses their action independently of the other players, a correlated equilibrium is also a Nash equilibrium. We denote the set of correlated equilibria by CE.

2.2.3 Regret-based Reinforcement Learning

A fully distributed procedure that can be used to reach the CE solution is the regret-based RL procedure [41]. The key idea of this method is to adjust the player's play probability proportional to the "regrets" for not having played other actions. Specifically, for any two actions $j \neq k \in \mathcal{I}$ at any time n, the regret of player one for not playing k every time it played j is

$$[B_n]_{j,k} = \frac{1}{n} \sum_{t \leq n: i_t = j} U(k, \ell_t) - \frac{1}{n} \sum_{t \leq n: i_t = j} U(j, \ell_t). \tag{2.3}$$

This is the change in time average payoff that player one would have achieved if it substituted a given action j each time it was played in the past, with another action k. Since player one only knows his set of actions and his own payoffs, he cannot compute the first term. Thus, the regret in (2.3) needs to be replaced by an estimate that can be computed on the basic of the available information, as

$$[B_n]_{j,k} = \frac{1}{n} \sum_{t \leq n: i_t = k} \frac{p_t(j)}{p_t(k)} U(s_t) - \frac{1}{n} \sum_{t \leq n: i_t = j} U(s_t),$$

where, p_t denotes the play probabilities at time t, i.e., $p_t(k)$ is the probability of choosing k at time t and $U(s_t) = U(i_t, \ell_t)$ denotes the payoff at time t.

If $i_n = j$ is the action chosen by player one at time n, then the probability distribution that it chooses an action at time $n+1$ is defined recursively as [41]

$$p_{n+1}(k) = \begin{cases} \left(1 - \dfrac{\delta}{n^\gamma}\right) \min\left\{ \dfrac{[B_n]^+_{j,k}}{\mu}, \dfrac{1}{m} \right\} + \dfrac{\delta}{n^\gamma} \dfrac{1}{m}, & k \neq j, \\ 1 - \displaystyle\sum_{k' \neq j} p_{n+1}(k'), & k = j, \end{cases} \tag{2.4}$$

with the initial play probabilities at $t = 1$ uniformly distributed over the set of possible actions; $\mu > 2mG$ is a constant, m is the cardinality of the set $\mathcal{I}$ and G is an upper bound on $|U(s)|$ for all $s \in \mathcal{S}$; $0 < \delta < 1$ and $0 < \gamma < 1/4$. We use the notation $[B_n]^+_{j,k} := \max([B_n]_{j,k}, 0)$. By using $[B_n]^+_{j,k}$ in (2.4), the RL algorithm in [40] completely ignores negative regrets $[B_n]_{j,k} < 0$.

It is proven in [41] that if all players chooses their actions according to (2.4), the empirical distribution of all strategies played until time n, which is given by

$$z_n(s) = \frac{1}{n} \sum_{t=1}^{n} \mathbb{1}_{\{s_t = s\}},$$

converges almost surely as $t \to \infty$ to the CE set of the game $\mathcal{G}$. Note that this does not imply convergence to a specific point on CE set, but that the solution approaches the CE set.

The main drawback of this standard regret-based reinforcement learning procedure is that although guaranteeing convergence to the set of CE, it often requires long convergence time and sometime converges to an undesirable equilibrium (i.e. poor fairness). These issues motivate the reinforcement learning with non-positive regret in the next section.

2.3 Algorithm

In this section, we describe our proposed multiagent reinforcement procedure.

2.3.1 Reinforcement Learning with Non-positive Regret

The RL procedure in Section 2.3 does not use any negative regrets in determining the probability of plays. However, as discussed in Section 1, negative regrets contain information that could improve the performance of the learning procedure. We propose to complement the regret-based RL in [41] by taking into account additional negative regrets in updating the learning rule. To determine the probability distribution of its action at the next stage $n + 1$, agent uses both its positive and negative parts of the time average regrets

as follow

$$p_{n+1}(k) = \begin{cases} \delta_n \dfrac{1}{m}, & \text{if } k \neq j \text{ and } [B_n]_{j,k} = 0 \\ (1-\delta_n)\ \dfrac{[B_n]^+_{j,k}}{\sum_k [B_n]^+_{j,k}} + \delta_n \dfrac{1}{m}, & \text{if } k \neq j \text{ and } [B_n]_{j,k} > 0 \\ (1-\delta_n)\ \dfrac{1}{n^\alpha} \dfrac{\left([B_n]^-_{j,k}\right)^{-1}}{\sum_k \left([B_n]^-_{j,k}\right)^{-1}} + \delta_n \dfrac{1}{m}, & \text{if } k \neq j \text{ and } [B_n]_{j,k} < 0 \\ 1 - \sum_{k' \neq j}\ p_{n+1}(k'), & \text{if } k = j \end{cases} \tag{2.5}$$

where $\delta_n = \delta/n^\gamma$ for $0 < \delta \ll 1$ and $0 < \gamma < 1/2$; and $0 < \alpha \leq 1$. We use the notation $[B_n]^-_{j,k} := \min([B_n]_{j,k}, 0)$.

Our main insight here is that the negative regrets should be included in the update procedure to ensure that when n is small the algorithm keep exploring different solutions, including the solution that yields negative regret, to speed up the convergence. However, as the algorithm progresses, the negative regrets reduce to zero and the positive regrets become the dominant factors in determining the playing probabilities. We prove that our new RL algorithm converges almost surely to the CE set and show in simulations that this learning strategy provides very fast convergence toward equilibrium states.

2.3.2 Discussion on The Algorithm

We discuss in detail here the major differences between our solution and the standard regret-based RL approach [41]. The main novelty in our approach is in the formula to update the play probability.

(a) Firstly, we do not use a constant proportional factor μ as in (2.4), but normalise the vector of regrets to get a probability vector. The reason for doing this is to avoid being dependent on the appropriate choice of some arbitrarily large enough parameter μ. As discussed in [41], a higher value of μ results in a smaller probability of switching to another action and thus leads to a slower speed of convergence.

(b) Secondly, in our solution, not only positive regrets but also negative values are contributing to the update procedure of the player. In particular, the play probability is proportional to the positive regret and is proportional to the inverse of the negative regret.

This choice of play probability allows the action that yields larger positive regret to get a higher probability to be selected in the next state, while the action that yields larger negative regrets to receive a lower probability to be used in the future.

(c) Thirdly, in the standard approach, it is difficult to determine an appropriate $0 < \delta < 1$ in (2.4). A large δ will lead the convergence to a large distance from the CE set hence lead to lower total utility. However, small δ means to discourage the exploration processes, and agents tend to perform the same action and thus will cause slow convergence. In our proposed approach, the choice of δ is much simpler: we only need to set $0 < \delta \ll 1$. A much smaller value of δ not only improves the convergence rate but also reduces the instability properties caused by inaccurate estimates of regrets in the standard RL solution. The key point here is that δ can be taken smaller to still obtain a similar amount of "exploration" due to the inclusion of the negative regret terms.

(d) Lastly, the negative regrets vanish in the play probability as the time step goes to infinity due to the inclusion of $1/n^{\alpha}$ in the play probability for negative regrets in (2.5). This means that the agent no longer considers the selection that yields negative regret after sufficiently exploring all the potential options. Using negative regrets after the exploration phase would reduce the achievable payoffs.

2.3.3 Convergence Analysis

Theorem 2.1. *If an agent (i.e. player one) uses the proposed procedure, its time average conditional regret is guaranteed to approach the set of non-positive regrets in the payoff space almost surely, provided that other agents do likewise.*

We now provide a brief overview of the proof. We use the differential inclusion (DI) framework in [56] to prove our Theorem. DI is a generalisation of ordinary differential equation that is particularly suitable to study the asymptotic trajectory of the iterative process in game-theoretic learning, especially when the information available to a player is "restricted". Standard approach in game theory such as Blackwell's approachability theorem used in [40, 41], however, cannot be trivially extended to prove the convergence of the proposed algorithm and will require a significant number of additional steps to

handle the modifications of the play probabilities p_n. The use of DI technique yields a considerably simpler and shorter proof as compared to the classical approach in [41].

Proof. Please refer to Appendix 2.A. □

Theorem 2.2. *If all agents follow the proposed procedure, the empirical distribution of joint play of all agents $z_n(s)$ converges almost surely as $t \to \infty$ to the set of correlated equilibria in the action space, for finite payoffs.*

Proof. Please refer to Appendix 2.B. □

2.4 Evaluation

In this section, we evaluate the performance of our proposed algorithm using a well-known multiagent Prisoner's Dilemma game (also known as the Tragedy of the Commons) [57]. Let's consider the game in which multiple agents $(A \geq 200)$ compete for a limited common resource. Each agent has to make a binary decision – "yes" or "no" that models the agent's decision of using the common resource or not, respectively. The agent that does not use the resource gets a fixed payoff. All the agents using the resource get the same payoff. Consequently, the more agents decided to use the resource, the smaller the obtainable payoff per agent; and when the number of agents sharing the resource is higher than a certain threshold, it is better for the others not to use the resource. A simple utility function reflecting this game can be expressed as follows:

$$U = \begin{cases} 1 & \text{if agent decision is "no",} \\ 101 - \eta & \text{if agent decision is "yes".} \end{cases}$$

with η being the number of agents making the same "yes" decision.

To evaluate the performance of our solution, we analyse the two metrics:

- Convergence speed (iterations): number of iterations to convergence. A fast convergence is preferable.

- System fairness index [58], which is derived as

$$J = \frac{(\sum_{a=1}^{A} x_a)^2}{A \times \sum_{a=1}^{A} x_a^2}, \tag{2.6}$$

  where x_a is the average payoff of user a and A is the number of agents. Notes that $J = 1$ is the best fairness of the system, which guaranteeing the same payoff among the agents.

It can be seen that this game has two pure Nash equilibrium points when either 99 or 100 agents use the common resource. Any solutions that yield the average number of resource agents between 99 and 100 will be in the set of correlated equilibria. Among them, the equilibrium point when $\eta = 100$ provides the best system fairness since all agents will receive the same payoff of 1.

We compare our proposed algorithm with three other algorithms:

- CODIPAS-RL in [48]: Agents learn both the expected payoff and the strategies in order to make decisions. This is a popular state-of-the-art reinforcement learning algorithm and has been shown to be superior to the conventional RL scheme such as Q-learning [48].
- Regret-based RL in [41]: Agents update their play probability proportional only to the estimates of "positive regret" for not having played other options.
- Our proposed algorithm: Agents update their learning rules by considering both positive and negative regrets for not choosing other options.
- Exhaustive Search: A centralised controller with complete information of the game considers all possible associations involving all agents and assigns agents decisions in a way to maximise the system fairness. We use this algorithm as a benchmark since it leads to the highest performance in fairness.

Figs. 2.1 and 2.2 show, respectively, the evolution of average number of agents using the resource (resource agents) and the system fairness index for the game with 200 agents. With the same initial probabilities, we observed that our proposed algorithm achieves the

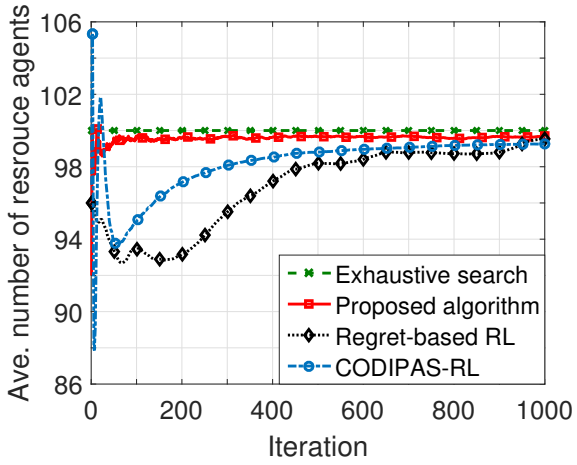

Figure 2.1. Evolution of average number of resource agents by different algorithms.

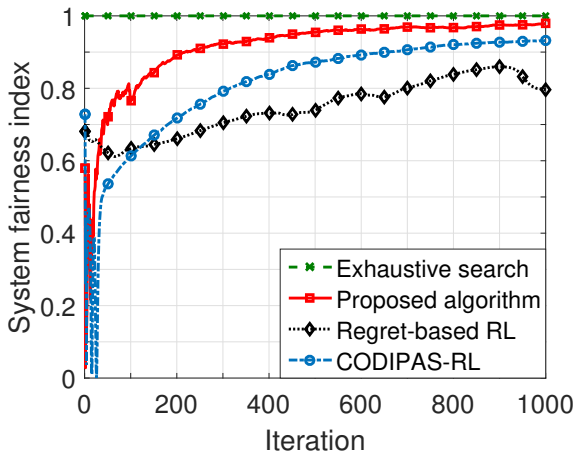

Figure 2.2. Evolution of system fairness index by different algorithms.

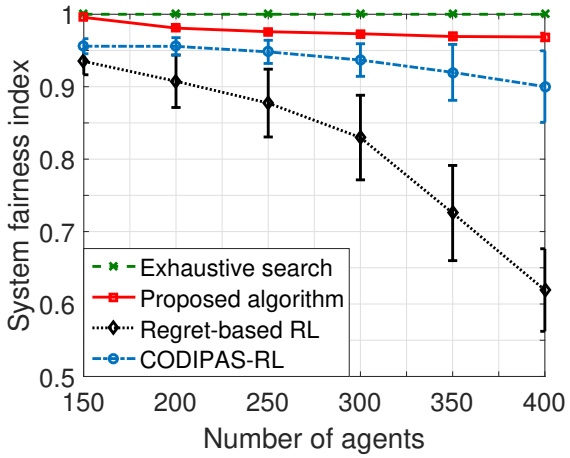

Figure 2.3. Comparison of fairness between algorithms for the same number of iterations.

fastest convergence speed among all the reinforcement learning algorithms. Our algorithm converges to equilibrium states in a very small number of iterations (less than 150 iterations), where as it requires a longer time to converge for both CODIPAS-RL (up to 400 iterations) and Regret-based RL (up to 900 iterations), especially the later. In fairness metric, our algorithm also leads to the highest system fairness index under the same number of iterations, as compared to the other RL schemes. The Regret-based RL scheme performs poorest due to its slow convergence speed.

To further study the impact of the total number of agents in the game on algorithms performance, we vary the agent number from 150 to 400 and measure the performances of all algorithms in fairness metric. The result is shown in Fig. 2.3. As we can see, proposed algorithm is quite robust in achieving system fairness to the change of the agent number. Increasing the total learning agents slightly reduces the system fairness index in our solution, but considerably bring down system fairness in other approaches, especially the Regret-based RL approach and when the total number of agents is very large.

2.5 Conclusion

We studied the problem of multiagent repeated games. We develop a fully distributed reinforcement learning procedure that takes advantage of both positive and negative regrets to speed up the learning process and improve the efficiency of the well-known regret-based reinforcement learning. Simulation results show that our solution is highly efficient with fast convergence speed and good fairness performance; and is more robust to the total number of agents in the system than other reinforcement learning algorithms. In our future research, we will study the rate of convergence of our algorithm and compare its performances on a broader set of benchmarks. As further work in this direction, a reinforcement learning framework for finding the global optimal solution in distributed multiagent system is still an open problem. Investigating the impact of irrational agents on the learning outcome is another challenging problem to consider.

Appendices of Chapter 2

2.A Proof of Theorem 2.1

Let $C : Z \to \mathbb{R}^{m\times m}$ be defined by

$$[C(z)]_{j,k} = \sum_{\ell\in\mathcal{L}} z(j,\ell)\,(U(k,\ell) - U(j,\ell))\ ,$$

which is the expected regret for player one when substituting action k for action j under the joint distribution z of actions. Suppose we consider player one playing some action i

with probability one, then

$$[C(z^i)]_{j,k} = \sum_{\ell \in \mathcal{L}} \mathbb{1}_{\{i=j\}}\, y_\ell \left(U(k,\ell) - U(j,\ell)\right) = \mathbb{1}_{\{i=j\}} \left(U(k,y) - U(j,y)\right).$$

Since player one cannot compute the first term as it only has access to the payoffs corresponding to actions it actually took, following [41], define an estimate of this term by

$$\tilde{U}(k,y)\ \mathbb{1}_{\{i=j\}} = \frac{p(j)}{p(k)}\, U(k,y)\ \mathbb{1}_{\{i=k\}}.$$

which is computed from the regrets associated with the alternative action k weighted proportional to the relative probabilities of player one choosing action j versus k when those actions were actually taken. The associated pseudo regret matrix at stage n is now

$$\tilde{C}_n(j,k) = \frac{p_n(j)}{p_n(k)}\, U(k,y_n)\ \mathbb{1}_{\{i_n=k\}} - U(j,y_n)\ \mathbb{1}_{\{i_n=j\}}.$$

Thus, we have

$$\begin{aligned}
\mathbf{E}\left\{\tilde{C}_n(j,k)|h_{n-1}\right\} &= p_n(k)\frac{p_n(j)}{p_n(k)}U(k,y_n) - p_n(j)\ U(j,y_n) \\
&= p_n(j)\left(U(k,y_n) - U(j,y_n)\right) \\
&= \mathbf{E}\left\{C_n(j,k)|h_{n-1}\right\},
\end{aligned}$$

where h_{n-1} is the action history of the game until stage $n-1$.

It can be seen that $C_n(j,k)$ and $\tilde{C}_n(j,k)$ are each bounded by $2mG/\delta_n$. The limit sets of the pair processes C_n and $\tilde{C}_n$ also coincide since they both have the same conditional expected values (see [41] for more details and discussions). Then Theorem 7.3 of [56] can be applied and thus the two processes exhibit the same asymptotic behaviour.

The average regret at stage n is thus a matrix B_n defined by

$$B_n(j,k) = \frac{1}{n}\sum_{t=1}^{n}\left[\frac{p_t(j)}{p_t(k)}U(k,y_t)\ \mathbb{1}_{\{i_t=k\}} - U(j,y_t)\ \mathbb{1}_{\{i_t=j\}}\right].$$

Hence, the discrete dynamics

$$\bar{B}_{n+1} - \bar{B}_n = \frac{1}{n+1}\ \left(B_{n+1} - \bar{B}_n\right)$$

is a discrete stochastic approximation of the DI

$$\dot{\mathbf{w}} \in N(\mathbf{w}) - \mathbf{w} \ \text{(with } w = B_n\text{)}. \tag{2.7}$$

Now for $j \neq k$, define the matrix sequence

$$[M_n]_{j,k} = \begin{cases} 0, & \text{if } [B_n]_{j,k} = 0 \\ \dfrac{[B_n]^+_{j,k}}{\sum_k [B_n]^+_{j,k}}, & \text{if } [B_n]_{j,k} > 0 \\ \dfrac{1}{n^\alpha} \dfrac{\left([B_n]^-_{j,k}\right)^{-1}}{\sum_k \left([B_n]^-_{j,k}\right)^{-1}}, & \text{if } [B_n]_{j,k} < 0 \end{cases} \tag{2.8}$$

We set $[M_n]_{j,j} = 1 - \sum_{k \neq j} [M_n]_{j,k}$, which takes value in $[0,1]$ by virtue of (2.8). Thus M_n is a transition probability matrix on $\mathcal{S}$. So there is a probability vector μ_n such that $M_n^T \mu_n = \mu_n$.

The "non-positive regret set" $D^1 \subset \mathbb{R}^{m \times m}$ for player one is defined by

$$D^1 = \left\{ g \in \mathbb{C}^{m \times m} : g(j,k) \leq 0, \forall (j,k) \right\}.$$

Evidently, D^1 is a closed, convex subspace of $\mathbb{R}^{m \times m}$. Define the Lyapunov function $P(w) = \frac{1}{2}\|w\|^2$, with $\nabla P(w) = w$. Then P satisfies the following properties and thus is a potential function for D^1:

- P is continuously differentiable;
- $P(w) = 0 \Leftrightarrow w \in D^1$;
- $\langle \nabla P(w), w \rangle > 0$ for all $w \notin D^1$.

Let $\varphi : \mathbb{R}^{m \times m} \to 2^X$ given by

$$\varphi(w) = \begin{cases} (1 - \delta_n)\, \mu(w) + \dfrac{\delta_n}{m}, & w \notin D^1 \\ X, & w \in D^1 \end{cases} \tag{2.9}$$

where $\mu(w)$ denotes a probability vector computed from the matrix $w = B_n$ according to the process above. Define a correspondence N on $\mathbb{R}^{m \times m} \setminus D^1$ by

$$N(w) = C(\varphi(w) \times Y)$$

so that φ is N-adapted, which means $N(w)$ contains all resulting average regrets.

According to Lyapunov theory, to prove the approachability of w to D^1, we need then to show that for any $w \in \mathbb{R}^{m\times m} \setminus D^1$ and some positive constant β,

$$\frac{d}{dt}P(w) = \langle \nabla P(w), \dot{w}\rangle \in \langle \nabla P(w), N(w) - w\rangle \leq -\beta P(w),$$

meaning that we need the following result

$$\langle \nabla P(w), \theta - w\rangle \leq -\beta P(w)$$

for all $\theta \in N(w)$ and some constant $\beta > 0$ (see [56] for details).

Suppose $w \notin D^1$, let $\theta = \mathbf{E}\left\{\tilde{C}(\varphi(w), y)|h_{n-1}\right\}$, with $y \in Y$, which means

$$[\theta]_{j,k} = \varphi_j(w)\ \left(U(k,y) - U(j,y)\right).$$

Then consider

$$\begin{aligned}
\langle \nabla P(w), \theta\rangle =& \sum_{j,k}^{m} \nabla P_{jk}(w)\ \varphi_j(w)\ \left(U(k,y) - U(j,y)\right)\\
=&(1-\delta_n)\sum_{j,k}\nabla P_{jk}(w)\ \mu_j(w)\ \left(U(k,y) - U(j,y)\right)\\
&+\frac{\delta_n}{m}\sum_{j,k}\nabla P_{jk}(w)\ \left(U(k,y) - U(j,y)\right)\\
=&(1-\delta_n)\sum_{j}U(j,y)\left(\sum_{k}\mu_k(w)\ \nabla P_{kj}(w) - \mu_j(w)\ \sum_{k}\nabla P_{jk}(w)\right)\\
&+\frac{\delta_n}{m}\sum_{j,k}\nabla P_{jk}(w)\ \left(U(k,y) - U(j,y)\right).
\end{aligned} \tag{2.10}$$

In the second line we substituted for $\varphi_j(w)$ from (2.9), and in the last line we collected together all terms containing $U(j,y)$.

Let $\mu_j(w)$ be such an invariant measure. Suppose that for every $j = 1,\ldots,m$, it holds that

$$\mu_j(w)\ \sum_{k}\nabla P_{jk}(w) = \sum_{k}\mu_k(w)\ \nabla P_{kj}(w),$$

then the first term in the sum in (2.10) is equal to zero. Therefore, noting that the payoff function $|U(.)|$ is bounded by G, we obtain

$$\langle \nabla P(w), \theta\rangle = \frac{\delta_n}{m}\sum_{j,k}\nabla P_{jk}(w)\ \left(U(k,y) - U(j,y)\right) \leq ||\nabla P(w)||\ \frac{2G\delta_n}{m}. \tag{2.11}$$

Next, using $P(w) = \|w\|^2/2$ and $\nabla P(w) = w$, it can be show that

$$\langle \nabla P(w), w \rangle = \langle w, w \rangle = ||w||^2 = 2P(w). \tag{2.12}$$

Therefore, it follows, using (2.11) and (2.12), that given $\epsilon > 0$, $||w|| \geq \epsilon$, one can choose $\delta_n > 0$ small enough such that

$$\begin{aligned}\langle \nabla P(w), \theta - w \rangle &= \langle \nabla P(w), \theta \rangle - \langle \nabla P(w), w \rangle \\ &\leq ||\nabla P(w)|| \, \frac{2G\delta_n}{m} - 2P(w) \leq -P(w).\end{aligned}$$

Consequently,

$$\frac{d}{dt} P\left(w(t)\right) \leq -P\left(w(t)\right),$$

so that

$$P\left(w(t)\right) \leq P\left(w(0)\right) e^{-t}.$$

This implies that $P(w(t))$ goes to zero at exponential rate and the set D^1 is a global attractor for the DI (2.7). Hence, the time average regret B_n and its corresponding regret C_n will then approach D^1. This completes the proof.

2.B Proof of Theorem 2.2

The proof follows from how the "regret" measure is defined. Recall that

$$\begin{aligned}[C(z_n)]_{j,k} &= \sum_{\ell \in \mathcal{L}} z_n(j, \ell_n) \left(U(k, \ell_n) - U(j, \ell_n)\right) \\ &= \sum_{s_n \in S : i_n = j} z_n(s_n) \left(U(k, \ell_n) - U(s_n)\right),\end{aligned}$$

where $s_n = (i_n, \ell_n)$ is the joint play made at stage n. On any convergent subsequence $\lim_{n\to\infty} z_n \to \Pi$, we get

$$\lim_{n\to\infty} [C(z_n)]_{j,k} = \sum_{s_n \in S : i_n = j} \Pi(s_n) \left(U(k, \ell_n) - U(s_n)\right) \leq 0.$$

Next, comparing with the definition of CE as in (2.2) completes the proof.

Chapter 3

Reinforcement Learning With Network-Assisted Feedback

FUTURE wireless networks (e.g., 5G) will consist of multiple radio access technologies (RATs). In these networks, deciding which RAT users should connect to is not a trivial problem. Current fully distributed algorithms although guaranteeing convergence to equilibrium states, are often slow, require high exploration times and may converge to undesirable equilibria. To overcome these limitations, this chapter develops a network feedback model that uses limited network-assisted information to obtain fast convergence, low overhead, small number of RAT switching, and competitive user fairness and network utility for the RAT selection process. We prove theoretically that a fully distributed algorithm developed within this framework is guaranteed to converge to a set of correlated equilibria. Our framework guarantees convergence in self-play even when only a single user applies the algorithm. Simulation results demonstrate that our solution (1) is highly efficient and outperforms the other existing related algorithms; and (2) can flexibly support a wide range of network-assisted feedback. The simulations demonstrate the effectiveness of our solution in a heterogeneous environment where users may potentially apply a number of different RAT selection procedures.

3.1 Introduction

To cope with the exponential growth of mobile traffic, network operators are continuously looking for ways to leverage spectrum across available radio access technologies (RATs) [59]. Multiple wireless network architectures (e.g., LTE, UMTS, WiFi, femto, etc) are being deployed concurrently in the current and next generation wireless networks [3]. At the same time, mobile devices are increasingly equipped with multiple RATs that can connect to and choose among the different base stations (BSs) with different access technologies. Deciding which technology, and which individual BS supporting that technology mobile users should connect to, is known as the RAT selection problem [8], and is a topic of much current research in LTE and 5G [10].

RAT selection is often addressed in the literature by using either a network-centric or a user-centric approach. In a network-centric approach [60–62], a centralised controller assigns BSs to users in a service area. This approach is suitable in a software defined networking environment where a controller has a complete logical view of the network [62]. It, however, requires collaboration between all wireless networks and users – exchanging significant communication overheads. When the networks are run by competing operators, such close collaboration may not at all be possible. A user-centric approach can overcome this problem by implementing the network-selection algorithms at the user side [15, 42–48, 63]. When intelligence is pushed to the network edge, rational users select their RAT in order to selfishly maximise their utility. However, as users have no information on BS load conditions, their decisions may be in no user's long-term interest, causing performance degradation and sometimes oscillation or instability. To guarantee convergence, most existing distributed RAT selection algorithms [15, 42–47] require that all users know the selection histories of other users, and are able to determine their own throughputs given other users' choices. This assumption implies that each user knows the instantaneous rates of the other users. The guaranteed convergence therefore comes at the cost of increased complexity, signalling and communication load.

To reduce the communication overheads, a fully distributed algorithm such as a reinforcement learning (RL) based algorithm [41, 48, 63] can be used. This algorithm does not require the users to know anything about other users. Indeed, users do not even need to know that they are parts of a RAT selection "game". Each user learns about the game

by observing only its own achieved payoffs. Over time, using only this information, a user can rationally choose the best course of actions to maximise its utility. Under mild conditions of finite payoffs and of unchanged network conditions, the RL-based regret minimisation algorithm in [41] is guaranteed to converge to a stable set of equilibria. We refer to the algorithm [41] as *Hart's RL-based algorithm* throughout this chapter. Despite this attractive property, Hart's RL-based algorithm suffer from problems of slow convergence, and of convergence to socially sub-optimal equilibria, making them unsuitable for RAT selection in real networks where the environment can change quickly [48].

One of the promising ideas to overcome the shortcomings of the Hart's RL-based algorithm is to use external information to aid users in their estimation of the game [53]. In engineering systems such as wireless networks, such external information is often readily available at the network BSs. We therefore propose to use such information to improve distributed RAT selections. A real challenge is to design a method that guarantees fast convergence and good performance, while signalling and processing burden remains acceptable. To achieve this balance, in our solution mobile users select their RAT depending on their individual observations, as well as feedback provided by the network. By tuning the network information, operators can also influence user decisions to achieve their objectives and avoid undesirable network states.

Our main contributions in this chapter are as follows:

1. *A Network Feedback Model:* We develop a network feedback model that uses network-assisted information to improve the performance of the Hart's RL-based algorithm in [41] for RAT selection. We show that our framework can be applied to multiple types of feedback. To our best knowledge, this is the first work that introduces network-assisted information in a RL-based algorithm for distributed RAT selection. Our framework accommodates a heterogeneous environment, where not all users have the same learning strategy and the same utility function. In practice, different users pursue different objectives and thus may use different learning strategies or utility functions. Our solution guarantees no-regret payoff in the long run for any user adopting it, irrespective of the behaviour of other users. Using our self-learning technique, any independent user can individually interface with networks to obtain the desired feedback and implement a no-regret based strategy. This adaptive

scheme does not require any modification of the current mobile network standards and can be easily implemented in software running on a end-user device.

2. *A Novel Fully Distributed RAT Selection Algorithm:* Using our framework, we develop a fully distributed algorithm which computes a correlated equilibrium solution. If all the users follow our algorithm, the empirical distribution of joint actions is guaranteed to converge to a set of correlated equilibria (CE), which are generalised Nash equilibria (NE).

3. *Comprehensive Practicality Study:* We perform extensive simulations with realistic network scenarios to evaluate our algorithm. Simulations demonstrate that our solution is highly efficient with fast convergence and low overheads. Our solution achieves competitive, if not better performance, both in fairness and utility, as well as per-user RAT switching, compared to state-of-the-art algorithms. A thorough evaluation of adaptive RAT selection algorithms including the one presented in this chapter is provided in [64].

The rest of this chapter is organised as follows. In Section 3.2, we discuss the related work. In Section 3.3, we present our RAT selection game model. We formally propose our reinforcement learning with network-assisted feedback in Sections 3.4. The evaluation is presented in Section 3.5. Finally, we conclude the chapter in Section 3.6.

3.2 Related Work

This section discusses the major differences between our solution and the most recent distributed RAT selection schemes.

3.2.1 Game Theory Applications in RAT Selection

Game theory is a mathematical tool to model the interaction of decision makers with conflicting interests, and has been widely used to both design, and to study the dynamics of network selection problems in wireless networks (for a survey refer to [65]). Most related works formulate the problems as non-cooperative games and propose iterative

procedures that converge to NE [42, 43, 48]. Unfortunately, most algorithms that aim to reach NE do not always guarantee convergence [66]. Substantial modifications of Nash-based algorithms are often required to achieve guaranteed behaviours for RAT selection games [15, 42–47]. A hysteresis mechanism, where a user changes its RAT only if its expected throughput is higher than a threshold or if a network controller allows the move [43], is used in [42–44] to guarantee convergence to NE. Authors in [15, 45–47] propose a network-assisted scheme, where additional knowledge of the network conditions is broadcast to all users, to aid them in their decisions.

Only a number of previous works [45, 53] consider the situation where players achieve co-ordination between their strategies, either directly or indirectly, in order to get better payoffs at the correlated equilibria. A CE is a generalised Nash equilibrium where each player chooses their actions based on their common knowledge of the game's history [23]. By allowing the players to coordinate their actions, a CE can provide a balance between the non-cooperative solution (where all the players work independently but may yield poor performance) and the fully cooperative solution (which requires coordination between players but can be highly efficient). In fact, the set of CE is more natural than the set of NE in decentralised adaptive learning environments since the common history observed by all players can serve as a natural coordination mechanism [41].

Several distributed algorithms can be used to achieve convergence to stable CE in a RAT game, including regret matching in [40] and its fully distributed variant – a reinforcement learning based regret minimisation algorithm in [41]. In Hart's RL-based algorithm, a user learns to make optimal decisions directly from its own past rewards without requiring any extra information. Contrary to the uncertainty of algorithms that aim to achieve convergence to NE, the Hart's RL-based algorithm in [41] converge to the set of CE almost surely. The main drawback of the Hart's RL-based algorithm in [41] is that although guaranteeing convergence to the CE set, it often requires long convergence time and can converge to a sub-optimal equilibrium. By this, we mean an outcome that yields lower payoffs, unfair resource allocation, or inefficient utilisation of available resources [53].

There are several possible approaches to theoretically analyse the convergence of reinforcement learning based algorithms. A method based on direct analysis was developed in [41]. Majority of subsequent proofs have been based on the stochastic approximation technique (i.e. averaging theory), such as the one used in [48]. More recently, Benam *et.*

al. [56] use the theory of differential inclusion (DI) to prove the convergence of adaptive procedures used in game theory. The proofs in this chapter are an application of [56] to RAT selection games.

The DI based stochastic approximation method is a generalisation of ordinary differential equation approach used in standard adaptive systems. DI is particularly suitable to study the asymptotic trajectory of the iterative process in game-theoretic learning where the information available to a player is inaccurate or missing. It provides a rich set of theoretical tools that allows us to study the convergence behaviour of multiple game settings including games with imperfect rewards that must be estimated from noisy observations, and when the strategies of the other plays are unknown. DI has been used in [52, 67, 68] for RL-based algorithms but to the best of our knowledge, this is the first work that this method is applied in RL procedure in which the "external" information is incorporated in the decision rule. The use of DI technique yields a considerably simpler and shorter proof as compared to the classical approach in [41].

3.2.2 Using External Feedback to Improve RAT Selection

There have been several RAT selection algorithms proposed that use some form of network feedback [15, 42–47]. In all of these approaches, the network runs a centralised algorithm to determine the controllable parameters (such as users' instantaneous rate [42–44], network suggestions [43, 46], traffic loads [47], quality of services [15] and offered bandwidths and costs [45]) for each user. Each BS then broadcasts these parameters to all the users in their coverage area, including those that are not actively served by it. The high amount of information exchange, excessive signalling and communication load all contribute to make these approaches unattractive in practice.

Several attempts have been made to ensure that the signalling overheads among BSs and users is kept at the minimum level by using RL-based algorithms [48, 63]. Two problems with these approaches are slow and arbitrary convergence [53]. Another major issue is that a very high number of RAT switching per-user is required due to the lack of information on global network load conditions. This is because each user must try many different actions in order to develop an understanding of the global structure of the RAT "game".

Our solution in this chapter follows the regret-based principles with significant modifications to accelerate convergence speed, reduce exploration times and avoid undesirable equilibria. We show in this chapter, using extensive simulations, that:

1. The overall signalling overheads of our algorithm are significantly less than those in [15, 42–48, 63], which are the state-of-the-art RAT selection algorithms.
2. Our algorithm has a fast convergence rate with a small number of per-user RAT switchings, whilst achieving competitive performance both in fairness and utility.
3. Lastly, our algorithm is one of a few algorithms of which we are aware, that can flexibly support a wide range of feedback, which can be defined according to the network operators' policies. Existing algorithms [15, 42–44, 46, 47] do not inherently support objective functions that are not directly related to throughput, and may require significant modifications to incorporate other objective functions. This will be described in detail in Section 3.5.2.

3.3 RAT Selection Game Model

3.3.1 Heterogeneous Network Throughput Model

We consider a heterogeneous wireless network (HWN) consisting of M base stations (BSs) and N end-user equipments (users). We use BS to denote any network node that connects directly to users such as a base station in WCMDA/LTE network or an access point in WiFi. In this chapter, we are primarily interested in user downlink throughput as the utility and use the same models as in [42–44] for different RATs. We divide the throughput models into two subclasses.

Class-1 (Proportional-Fair Model)

under this class, each user obtains a different user-specific throughput which is a function of its instantaneous physical (PHY) rate and the number of users sharing the same BS. The throughput of a user (i.e., user A) choosing BS k is

$$\bar{U}_A^k = \frac{R_A^k}{n^k}, \tag{3.1}$$

where R_A^k is the PHY rate of user A on BS k and n^k is the number of users on k. This class is suitable to model time/bandwidth-fair access technologies such as 3G/4G networks.

Class-2 (Throughput-Fair Model)

under this class, all users connected to the same BS will have the same per-user throughput. The throughput of a user (i.e., user A) connected to BS k can be expressed as

$$\bar{U}_A^k = \left(\sum_{a=1}^{n^k} \frac{1}{R_a^k} \right)^{-1} . \tag{3.2}$$

This class is suitable for throughput-fair access technologies such as WiFi.

Realistic Throughput Model

Most existing works assume that the user knows its actual throughput in (3.1) and (3.2). By actual throughput, we mean the long-term average throughput that a user experiences on a wireless network. In reality, the actual throughput of each user is influenced by not only the link quality (i.e., the signal to noise ratio) but also many other factors such as traffic load and interference from the surrounding environment. Therefore, in practice, the user only knows its sampled throughput, not the actual value. The sampled value can be modelled as a random variable where the actual throughput given in (3.1) or (3.2) is the mean, which is computed at the network side. At any one time, depending on the number of users per base station, the distribution of traffic load and sampling technique, instantaneous throughput observed by the user may vary from the mean.

Assumption 3.1: To model the real user observed throughput, we follow the most recently proposed instantaneous throughput model in [44], where the user observed throughput is assumed to follow a Gaussian distribution. Other distribution could be used but is outside the scope of this chapter. Under the Gaussian assumption, the mean is equal to the actual throughput and the standard deviation is equal to the product of the noise value e and the actual throughput [44]. Thus, instantaneous throughput rate of a user A choosing BS k is a Gaussian random variable:

$$U_A^k \sim \mathsf{N}(\bar{U}_A^k, \sigma^2) ,$$

where $\sigma = e \times \bar{U}_A^k$ and $0 < e < 1$. In our solution, the network provides every user with the actual throughput $\bar{U}_A^k$ calculated at the BS (BS computed throughput) rather than the randomly fluctuating rates U_A^k observed by the user (user observed throughput).

3.3.2 Radio Access Technology Selection Model

In the following, we adopt the notation of [56]. We model the RAT selection as a repeated game where the players (mobile users) aim to maximise their long-run average payoffs (throughput). We consider a game with N players denoted by the set $\mathcal{N} = \{1, \ldots, N\}$ for some (finite) integer $N \geq 2$. Each player a has its set of finite actions $\mathcal{S}_a$ (set of available BSs) and we denote by $\mathcal{S} = \mathcal{S}_1 \times \cdots \times \mathcal{S}_N$, the set of all strategies for all players, i.e. the Cartesian product of all players' possible actions.

We view the game from the point of view of player A – a randomly selected player among the set of all players. Let $\mathcal{I} = \mathcal{S}_A$ denote the set of actions of player A and $\mathcal{L} = \mathcal{S} \setminus \mathcal{S}_A$ the set of actions of all other players. Denote by X, the set of all probability mass functions (pmf) on $\mathcal{I}$ and Y the set of pmf on $\mathcal{L}$. Let Z denote the set of pmf on $\mathcal{S}$, then $X \times Y$ is a subset of Z comprised of all pmf of the form $z = (x, y)$ where $x \in X$ and $y \in Y$, i.e. all pmf of the probability of the action of player A and the actions of all other players taken together. The main notations that we use in this chapter are summarised in Table 3.1.

Let $U_A : \mathcal{S} \to \mathbb{R}$ denote the payoff achieved by player A when the overall action taken by all players is $s \in \mathcal{S}$. We represent a strategy in the form $s = (i, \ell)$ where i is the action of player A and ℓ is the action of all other players. We will consider the general formulation of the game where users apply mixed strategies over the possible selection set $\mathcal{S}$. Under randomised actions with overall probability (pmf) $z \in Z$, the payoff obtained by player A is defined as

$$U_A(z) = \sum_{s \in \mathcal{S}} z(s)\, U_A(s)\,.$$

The RAT selection game then can be denoted by $\mathcal{G} = (\mathcal{N}, (\mathcal{S}_A)_{A \in \mathcal{N}}, (U_A)_{A \in \mathcal{N}})$. In our game model, each player A knows only its set of actions (S_A) and its stream of payoffs (U_A) received in the past. Players are not aware of other players' actions and payoffs. Instead, players can observe the number of other players choosing the same action after each action, as explained later in Section 3.4.1.

Table 3.1. Summary of main notations used in Chapter 3

Symbol	Semantics
N	Number of users
M	Number of base stations
n^k	Number of users on base station k
R_A^k	Physical (PHY) rate of user A to BS k
$\bar{U}_A^k$	The actual throughput of user A choosing BS k
U_A^k	The instantaneous throughput of user A choosing BS k
$s = (i, \ell)$	The action taken by all players, where i is the action of player A and ℓ is the actions of the others
$U(s)$	The payoff achieved by player A when the overall action taken by all players is s
$z = (x, y)$	The probability of the action taken by all players, where x is the probability of action of player A and y is the probability of action of all other players except player A
Y_τ^k	The network-assisted feedback that BS k sends to user A at time τ
$\bar{U}_\tau^k$	The BS computed throughput that BS k sends to user A at time τ
n_τ^k	The number of users on BS k at time τ
$B_t(j,k)$	The user estimated regret in average payoff of player A up to time t for not playing k in stead of j
$Y_t(j,k)$	The network measured regret in average payoff for player A up to time t for not playing k in stead of j
$p_t(k)$	The probability of choosing BS k at time t by player A
$\bar{z}_t(s)$	The empirical distribution of join action s of all players until time t

In this chapter, we are interested in a popular notion of rationality that generalises the Nash equilibrium, known as a correlated equilibrium. CE is an optimality concept introduced by Aumann [23] and is proven to exist for any finite games with bounded payoffs [25]. It is relevant to probabilistic games, namely where strategies are determined probabilistically, and is a precise statement of rationality in this setting [23].

Definition 3.1. *A probability distribution ψ defined on $\mathcal{S}$ is said to be a correlated equilibrium for the game $\mathcal{G}$ if for every player $A \in \mathcal{N}$, and for every pair of action $j, k \in \mathcal{I}$, it holds that*[1]

$$\sum_{s \in \mathcal{S}: i=j} \psi(s)(U_A(k, \ell) - U_A(j, \ell)) \leq 0, \tag{3.3}$$

CE models possible correlation or co-ordination between players' actions compared to the usual strategic equilibrium of Nash, where all players act independently. A CE results if each player does not benefit from choosing any other probability distribution over its actions, provided that all the other players do likewise. When each player chooses their action independently of the other players, or without any implicit co-ordination mechanism, a CE is also a NE.

Definition 3.2. *A probability distribution ϕ defined in $\mathcal{S}$ is said to be a coarse correlated equilibrium for the game $\mathcal{G}$ if for every player $A \in \mathcal{N}$ and for every action $i \in \mathcal{I}$, it holds that*

$$\sum_{s \in S} \phi(s)(U_A(i, \ell) - U_A(s)) \leq 0. \tag{3.4}$$

A coarse correlated equilibrium (CCE) set or also know as the Hannan set is a generalisation of correlated equilibrium. The set of CE is contained in the Hannan set (and the two sets coincide when every player has at most two strategies). Moreover, the Hannan distributions that are independent across players are precisely the NE of the game. In a CCE, all players follow the learning rule. If a single player decides not to use the rule, it experiences a lower payoff.

3.3.3 Computing the Correlated Equilibria

A fully distributed algorithm that can be used to reach the CE solution is the RL-based regret minimisation procedure in [41]. The key idea of this method is to adjust the

[1] We write $\sum_{s \in \mathcal{S}: i=j}$ for the sum over all s in $\mathcal{S}$ whose i equals j. Similar notations are used elsewhere in the chapter.

player's action probability proportional to the "regrets" for not having played other actions. Specifically, for any two actions $j \neq k \in \mathcal{I}$ at any time t, the regret of player A for not playing k is

$$C_t(j,k) = \frac{1}{t} \sum_{\tau \leq t: i_\tau = j} U(k, \ell_\tau) - \frac{1}{t} \sum_{\tau \leq t: i_\tau = j} U(j, \ell_\tau) \,, \tag{3.5}$$

where i_τ denotes the action taken by player A at time τ (i.e. $i_\tau = j$ means player A selects BS j at time τ) and ℓ_τ denotes the actions of the others at time τ. This is the change in the average payoff that player A would observed if playing k instead of j every time it played j in the past. Note that the notations should have the subscript A to indicate that it refers to player A. Since we view the game from player A's point of view, we drop this subscript to keep the notation simple (thus, we write C_t and U in stead of $C_{A,t}$ and U_A, and so on). Similar notations are used in the rest of the chapter. Since player A only has access to the payoffs corresponding to actions it actually took, it cannot compute the first term. Thus, the regret in (3.5) needs to be replaced by an estimate that can be computed on the basis of the available information, via

$$B_t(j,k) = \frac{1}{t} \sum_{\tau \leq t: i_\tau = k} \frac{p_\tau(j)}{p_\tau(k)} U(k, \ell_\tau) - \frac{1}{t} \sum_{\tau \leq t: i_\tau = j} U(j, \ell_\tau) \,, \tag{3.6}$$

where p_τ denotes the play probabilities of player A at time τ (i.e., $p_\tau(k)$ is the probability of choosing k at time τ). This approximate regret measures the historical difference of the average payoff over the periods when k was used and the periods when j was used [41].

If $i_t = j$ is the action chosen by player A at time t, then the probability distribution that player A chooses an action at time $t+1$ is defined recursively as [41][2]

$$p_{t+1}(k) = \begin{cases} (1-\delta_t) \min\left\{ \dfrac{B_t^+(j,k)}{\mu}, \dfrac{1}{m} \right\} + \dfrac{\delta_t}{m} & \text{if } k \neq j, \\ 1 - \displaystyle\sum_{k' \neq j} p_{t+1}(k') & \text{if } k = j, \end{cases} \tag{3.7}$$

with the initial action probabilities at $t = 1$ uniformly distributed over the set of possible actions; $\mu > 2mG$ is a constant with m being the cardinality of the set $\mathcal{I}$ and G being an upper bound on $|U(s)|$ for all $s \in \mathcal{S}$; $\delta_t = \delta / t^\gamma$, $0 < \delta < 1$ and $0 \leq \gamma < 1/4$.

[2]We use the notation $x^+ := \max(x, 0)$ for a real number x throughout this chapter (e.g. $B_t^+(j,k) = \max(B_t(j,k), 0)$). The definition is extended to real vectors and matrices elementwise.

It is proven in [41] that if *every* player chooses their actions according to (3.7), then the empirical distribution of joint actions s of all players until time t, which is given by[3]

$$\bar{z}_t(s) = \frac{1}{t} \sum_{\tau=1}^{t} \mathbb{1}_{\{s_\tau = s\}} ,$$

converges almost surely as $t \to \infty$ to the set of CE of the game $\mathcal{G}$. Note that this does not imply convergence to a specific point on the CE set, but that the solution approaches the CE set.

3.3.4 Example of RAT Selection Game

We use the example in Fig. 3.1 to illustrate the concepts introduced so far in this chapter. In this example, there are two users and two RATs: WiFi (RAT1) and 4G (RAT2). User 2 is at the cell-center of RAT1 and has a good PHY rate of 54Mbps. User 1 is at the cell-edge location of RAT1 and so obtains a lower PHY rate of 6Mbps. Both users are located at similar distances from RAT2 and thus have the same PHY rate of 5.4Mbps. These PHY rates are also their obtained throughputs when connected alone to these RATs.

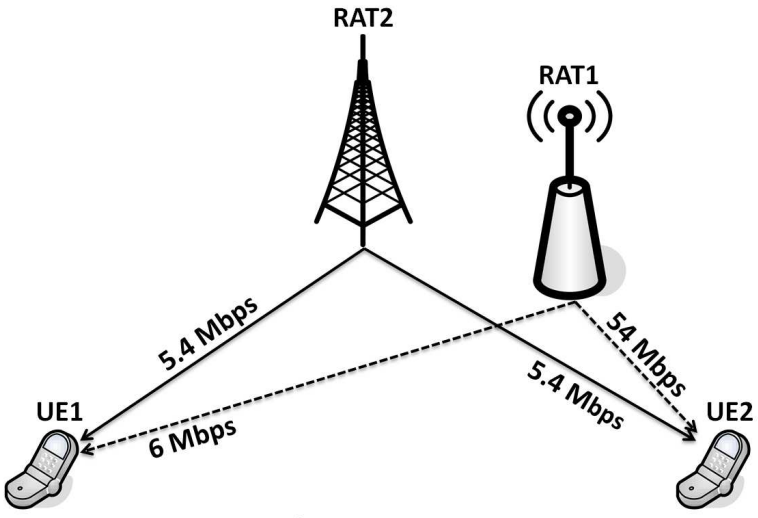

Figure 3.1. An example of RAT selection in a mixed 4G/WiFi network

The set of actions is denoted by $\mathcal{S} = \{(j,k) : j,k = 1,2\}$ where $s = (j,k)$ means that user 1 chooses RAT j and user 2 chooses RAT k. The payoff functions are the throughput

[3] Where $\mathbb{1}(.)$ denotes the indicator function.

Table 3.2. Payoff matrix for the RAT selection game

	$s_2 = 1$	$s_2 = 2$
$s_1 = 1$	(5.4, 5.4)	(6.0, 5.4)
$s_1 = 2$	(5.4, 54)	(2.7, 2.7)

obtained for each user. When both users connect to the WiFi access point, under Class-1 throughput model, they receive a very low throughput of $(1/6 + 1/54)^{-1} = 5.4$ Mbps as given by equation (3.2) for their WiFi connections. The 4G BS are assumed to use the time-fair protocol (Class-2 throughput model) which allows each user the same time duration to access to the network. When both users select RAT2, they receive the throughputs that are equal to half of their physical rates. Using equation (3.1), the throughput payoff is $5.4/2 = 2.7$ Mbps for each user. We summarise this game in Table 3.2.

Let p be a probability distribution on $\mathcal{S}$ with $p(j,k)$ denoting the joint probability that player 1 chooses RAT j and player 2 chooses RAT k, for $j,k = 1,2$. Substituting the payoffs from table 3.2, equation (3.3) yields the four linear inequalities

$$\begin{cases} p(1,1)\{5.4-5.4\} + p(1,2)\{2.7-6.0\} \le 0\,, \\ p(2,1)\{5.4-5.4\} + p(2,2)\{6.0-2.7\} \le 0\,, \\ p(1,1)\{5.4-5.4\} + p(2,1)\{2.7-54\} \le 0\,, \\ p(1,2)\{5.4-5.4\} + p(2,2)\{54-2.7\} \le 0\,. \end{cases} \Rightarrow \begin{cases} p(1,2) \ge 0 \\ p(2,2) \le 0 \\ p(2,1) \ge 0 \\ p(2,2) \le 0 \end{cases}$$

We also have the four inequalities $p(j,k) \ge 0$ for $j,k = 1,2$ and the equality $p(1,1) + p(1,2) + p(2,1) + p(2,2) = 1$ that defines a pmf. Then, a correlated equilibrium is a quadruple $(p(1,1),\ p(1,2),\ p(2,1),\ p(2,2))$ that satisfies:

$$\begin{cases} p(2,2) = 0\,, \\ p(1,1), p(1,2), p(2,1) \ge 0\,, \\ p(1,1) + p(1,2) + p(2,1) = 1\,. \end{cases}$$

Therefore, any solutions of the form $p(1,1) + p(1,2) + p(2,1) = 1$ will be in the set of CE. The corner points $p(1,1) = 1$, $p(1,2) = 1$ and $p(2,1) = 1$ are pure NE whilst the other solutions are mixed NE. Payoff pairs in these pure NE are, respectively, (5.4, 5.4), (6, 5.4) and (5.4, 54). Fig. 3.2 shows the set of all payoff allocations under correlated strategies and under correlated equilibria. The set of correlated strategies (light gray) is the set of all

possible combination of players' pure strategies; and the set of CE (dark gray), which is a super set of the NE set, is the triangle with these three NE as vertices.

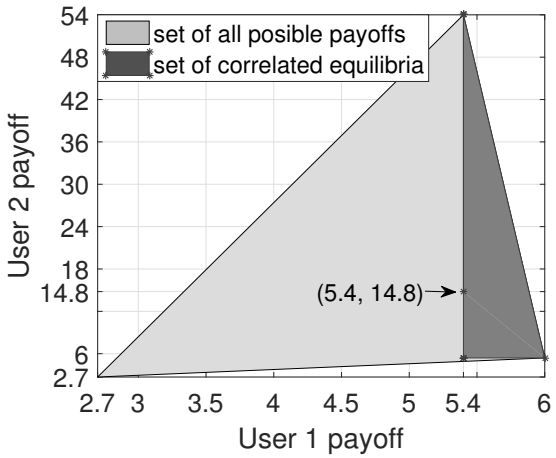

Figure 3.2. The set of correlated strategies and correlated equilibria in payoff space

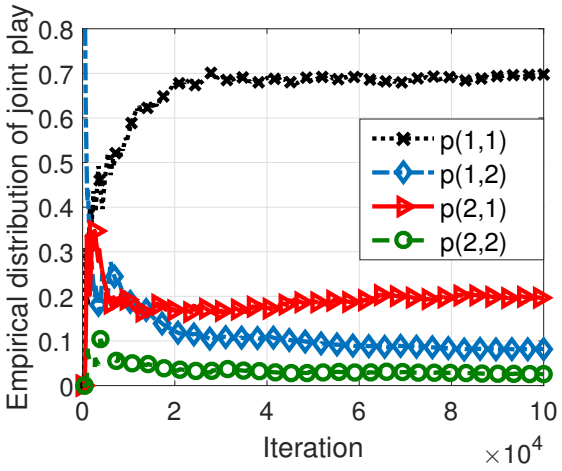

Figure 3.3. The empirical distribution of join play by Hart's RL-based algorithm

Limitations of the Hart's RL-based Algorithm in [41] for RAT selection: We implemented the Hart's RL-based algorithm in [41] and applied it to the RAT selection game in Fig. 3.1. We encountered the following three undesirable outcomes even on this simple example.

1. **Sub-optimal convergence:** Our implementation of the Hart's RL-based algorithm when applying to the above network leads to the CE point $(p(1,1) = 0.70,\ p(1,2) = 0.08,\ p(2,1) = 0.20,\ p(2,2) = 0.02)$ that yields a payoff pair of (5.4, 14.8) the majority of the time. This equilibrium is neither fair ((5.4, 5.4) provides the best system fairness) nor throughput efficient ((5.4, 54) yields the highest overall throughput, albeit unfairly).

2. **Slow convergence:** The algorithm takes at least 6,000 iterations to converge on a simple 2 base stations – 2 users network as shown in Fig. 3.3! This is a significant problem for RAT selection where network conditions can change quickly, breaking the implicit assumptions of stable environment, required for the RL-based algorithm to converge.

3. **High numbers of switching:** The algorithm also requires up to 400 RAT switchings per user to converge. This is another major constraint for real network implementation due to the challenge in providing seamless vertical handover between different RATs.

These issues of slow convergence, sub-optimal convergence and high numbers of switching of Hart's RL-based algorithm in [41] motivate the introduction of network-assisted feedback to the reinforcement learning based regret minimisation algorithm in the next section.

3.4 Algorithm

To overcome the limitations of the Hart's RL-based algorithm as observed above, we propose a feedback model that uses network-assisted information from the network base stations[4]. The main idea of our solution is to help users estimate their utilities more accurately using the limited information that is readily available at the BS. Using network feedback, the operators can also alter the trajectory of the algorithm. There have been several proposals for using network feedback to improve distributed RAT selection algorithms [15, 42–47], but not for RL-based ones[5]. We show empirically via simulation in Section 3.5.1 that our algorithm, by using little extra information, achieves a faster convergence rate to the CE set than existing distributed RAT selection algorithms including a recent proposed RL-based algorithm in [48].

3.4.1 Using Feedback to Update Network Measured Regret

The types of network feedback varys depending on the objectives of the network designers. In this chapter, we use two types of feedback: (1) BS computed throughput $\bar{U}_t^k$, which indicates the actual throughput that a user could receive from the BS k at time t; (2) and the number of users n_t^k, which is the number of users currently connected to the BS k at

[4]Note that the feedback model is a model not an algorithm and RL is an algorithm.

[5]Note that in general learning theory, RL-based algorithms, their convergence and approximations are well studied.

time t. Providing the actual achievable throughputs can help users make informed decisions that lead to better outcomes by exploiting the actions that yield higher throughputs. Knowing the number of concurrent users at each BS will help users avoid exploring selections that result in poor performances. However, these types of information are not directly available to the end-users.

Since most networks have up-to-date and accurate measurements of these metrics, we propose to use this information to improve the performance of the Hart's RL-based algorithm in [41] for RAT selection. A number of mechanisms to distribute additional feedback information from BS to users have been standardised and can be used for this purpose, including the logical communication channel in IEEE standard 1900.4 [69], and the Access Network Query Protocol (ANQP) in IEEE 802.11u standards [70]. These protocols allow users to query information about the capabilities of the network (such as throughput, packet error rate, available services) prior to performing the authentication process.

As explained in Section 3.3.1, users do not know their actual throughput. Their instantaneous estimations are often very noisy. By using network-assisted feedback, each user can estimate its obtainable throughput $U(k, \ell_t)$ if it switches to another BS k given its current action is $i_t = j \neq k$. The user then can compute network measured regrets $Y_t(j,k)$, which is a measure of the average regret for the user observed by the network at time t for not selecting other BS k instead of the actual BS j every time in the past, as follows.

Class-1 Throughput Estimation

Suppose $i_t = j$ is the action chosen by user A at time t. Using (3.1), the obtainable throughput if user A connects instead to BS k, is equal to R_A^k divided by $(n_t^k + 1)$, the total number of users sharing the BS k at time t, if user A joins.

$$U(k, \ell_t)|_{i_t=j\neq k} = \frac{R_A^k}{n_t^k + 1} \approx \frac{\sum_{\tau \leq t: i_\tau = k} (\bar{U}_\tau^k \times n_\tau^k)}{\nu_t^k \times (n_t^k + 1)}, \tag{3.8}$$

where $\nu_t^k = \sum_{\tau \leq t} \mathbb{1}_{\{i_\tau = k\}}$ counts how many times BS k has been chosen up to time t. R_A^k is obtained by taking the average of $(\bar{U}_\tau^k \times n_\tau^k)$ over ν_t^k – the periods when k was used.

Similarly, the number of users sharing the same BS k at time t can be estimated by taking the average number of users on k over the periods when k was used. That is,

$$n_t^k|_{i_t=j\neq k} = \frac{\sum_{\tau\leq t:i_\tau=k} n_\tau^k}{\nu_t^k}\,. \tag{3.9}$$

Replacing (3.9) into the denominator of (3.8), the estimate of $U(k,\ell_t)$ in (3.8) is then

$$\tilde{U}(k,\ell_t)|_{i_t=j\neq k} = \frac{\sum_{\tau\leq t:i_\tau=k}(\bar{U}_\tau^k \times n_\tau^k)}{\sum_{\tau\leq t:i_\tau=k}(n_\tau^k+1)}\,.$$

The BS observed regret measured at time t for class-1 RAT can be calculated as

$$Y_t(j,k) = \frac{1}{t}\sum_{\tau\leq t:i_\tau=j}\left(\tilde{U}(k,\ell_\tau) - \bar{U}_\tau^j\right) = \frac{1}{t}\sum_{\tau\leq t:i_\tau=j}\left(\frac{\sum_{\tau\leq t:i_\tau=k}(\bar{U}_\tau^k \times n_\tau^k)}{\sum_{\tau\leq t:i_\tau=k}(n_\tau^k+1)} - \bar{U}_\tau^j\right). \tag{3.10}$$

Class-2 Throughput Estimation

Suppose $i_t = j$ is the action chosen by user A at time t, then using (3.2), we obtain the throughput of user A if it connects to another BS k as

$$\begin{aligned} U(k,\ell_t)|_{i_t=j\neq k} &= \left(\sum_{a=1}^{n_t^k}\frac{1}{R_a^k} + \frac{1}{R_A^k}\right)^{-1} = \left[\left(\sum_{a=1}^{n_t^k}\frac{1}{R_a^k}\right)\left(1+\frac{(R_A^k)^{-1}}{\sum_{a=1}^{n_t^k}(R_a^k)^{-1}}\right)\right]^{-1} \\ &= \bar{U}_t^k\left[1+\frac{(R_A^k)^{-1}}{\sum_{a=1}^{n_t^k}(R_a^k)^{-1}}\right]^{-1} \approx \bar{U}_t^k\left[1-\frac{(R_A^k)^{-1}}{\sum_{a=1}^{n_t^k}(R_a^k)^{-1}}\right]. \end{aligned} \tag{3.11}$$

To obtain the final expression (3.11), in the last line we use the first order Taylor approximation $(1+x)^n \approx (1+nx)$ when $0 < x \ll 1$. This approximation is likely to hold as long as the number of users is large enough $n_t^k \gg 1$.

Assumption 3.2: To make the analysis simple, we assume that all the PHY rates R_a^k for all $a = 1,2,\ldots n_t^k$ to a BS k are independent and identical distributed with a uniform distribution $R_a^k \sim \mathsf{U}(\alpha,\beta)$, where α and β denote the minimum and maximum PHY rates of all users.

Since each R_a^k is independent and identically distributed, they have the same expected value. Thus, the obtainable throughput if user A connects to another BS k, can be calculated as

$$\tilde{U}(k,\ell_t)|_{i_t=j\neq k} \approx \bar{U}_t^k \left[1 - \frac{\mathbf{E}\left\{(R_A^k)^{-1}\right\}}{\sum_{a=1}^{n_t^k} \mathbf{E}\left\{(R_a^k)^{-1}\right\}}\right] = \bar{U}_t^k \left(1 - \frac{1}{n_t^k}\right).$$

Proposition 1. *The absolute error between the actual value $U(k,\ell_t)$ and the estimate $\tilde{U}(k,\ell_t)$ in our WiFi throughput estimation is bounded by*

$$\frac{G}{n_t^k}\left(\frac{\beta}{\alpha} - 1\right) \text{ , where } \beta \geq \alpha \,.$$

Proof. The absolute error is

$$\begin{aligned}
|U(k,\ell_t) - \tilde{U}(k,\ell_t)| &= \bar{U}_t^k \left| \left(1 - \frac{(R_A^k)^{-1}}{\sum_{a=1}^{n_t^k}(R_a^k)^{-1}}\right) - \left(1 - \frac{1}{n_t^k}\right)\right| = \bar{U}_t^k \left| \frac{1}{n_t^k} - \frac{(R_A^k)^{-1}}{\sum_{a=1}^{n_t^k}(R_a^k)^{-1}}\right| \\
&\leq G \ \max\left\{ \left| \frac{1}{n_t^k} - \frac{(1/\alpha)^{-1}}{n_t^k(1/\beta)^{-1}}\right| , \left| \frac{1}{n_t^k} - \frac{(1/\alpha)^{-1}}{n_t^k(1/\beta)^{-1}}\right| \right\} \\
&= \frac{G}{n_t^k} \ \max\left\{ \left|1 - \frac{\beta}{\alpha}\right| , \left|1 - \frac{\alpha}{\beta}\right| \right\} = \frac{G}{n_t^k}\left(\frac{\beta}{\alpha} - 1\right).
\end{aligned}$$

□

Accordingly, we can conclude that the absolute error will be zero when $\beta = \alpha$, which assumes all users on BS k have the same PHY rates. Otherwise, if the number of users sharing the same BS n_t^k is large enough $n_t^k \gg G(\beta/\alpha - 1)$, the absolute error is also very close to zero.

Similarly, replacing $\bar{U}_t^k$ by the average of $\bar{U}_\tau^k$ over ν_t^k and using (3.9), $\tilde{U}(k,\ell_t)$ is then

$$\tilde{U}(k,\ell_t)|_{i_t=j\neq k} = \frac{\sum_{\tau\leq t:i_\tau=k} \bar{U}_\tau^k}{\nu_t^k}\left(1 - \frac{\nu_t^k}{\sum_{\tau\leq t:i_\tau=k} n_t^k}\right) = \frac{\sum_{\tau\leq t:i_\tau=k} \bar{U}_\tau^k \,(n_\tau^k - 1)}{\sum_{\tau\leq t:i_\tau=k} n_\tau^k}.$$

The BS observed regret measured at time t for class-2 RAT can be calculated as

$$Y_t(j,k) = \frac{1}{t}\sum_{\tau\leq t:i_\tau=j}\left(\tilde{U}(k,\ell_\tau) - \bar{U}_\tau^j\right) = \frac{1}{t}\sum_{\tau\leq t:i_\tau=j}\left(\frac{\sum_{\tau\leq t:i_\tau=k} \bar{U}_\tau^k(n_\tau^k - 1)}{\sum_{\tau\leq t:i_\tau=k} n_\tau^k} - \bar{U}_\tau^j\right). \quad (3.12)$$

3.4.2 Reinforcement Learning With Network-Assisted Feedback

We propose to fundamentally complement the Hart's RL-based algorithm in [41] with external feedback from the network to aid users in their RAT selection. Let Y_τ^k be the network feedback that the BS k sends to its connected user A at time τ. In this chapter, the network feedback is a tuple $Y_\tau^k = (\bar{U}_\tau^k, n_\tau^k)$, where $\bar{U}_\tau^k$ is the BS computed per-user throughput at time τ and n_τ^k is the number of users on BS k at time τ. In our RLNF algorithm, the user then uses Y_τ^k to compute network measured regrets $Y_t(j,k)$ at time $t \geq \tau$.

Our main idea is to complement the user estimated regret B_t in [41] with the network observed assisted information Y_t at each time step t to speed up convergence towards the equilibria. We modify the probability of actions $p_{t+1}(k)$ in (3.7) with the combined regrets (B_t, Y_t) as in equation (3.13)

$$p_{t+1}(k) = \begin{cases} (1-\delta_t)\,\min\left\{\dfrac{B_t^+(j,k)}{\epsilon + \sum_k B_t^+(j,k)}, \dfrac{Y_t^+(j,k)}{\epsilon + \sum_k Y_t^+(j,k)}\right\} + \dfrac{\delta_t}{m} & \text{if } k \neq j\,, \\ 1 - \sum_{k' \neq j}\, p_{t+1}(k') & \text{if } k = j\,. \end{cases} \tag{3.13}$$

for any $0 < \epsilon \ll 1$. In order to implement this policy, each user needs 2 inputs: (1) the user observed throughputs $(U_1^k, \ldots, U_t^k)$ to compute user estimated regret $B_t(j,k)$ using equation (3.6); and (2) the network-assisted feedback $(Y_1^k, \ldots, Y_t^k)$ to compute network measured regret $Y_t(j,k)$. The exact procedure to compute $Y_t(j,k)$ from network feedbacks was explained in Section 3.4.1.

Our RLNF algorithm differs from the Hart's RL-based algorithm in [41] in the formula to update $p_{t+1}(k)$ in (3.13). Here, we make two changes to equation (3.7) in [41] for updating $p_{t+1}(k)$. First, $p_{t+1}(k)$ in (3.7) is a function of two inputs, i.e., $p_{t+1}(k) = f(\min\{B_t(j,k), m\})$, whereas in (3.13), we remove m in the min function and complement this function to take the extra information $Y_t(j,k)$ as $p_{t+1}(k) = f(\min\{B_t(j,k), Y_t(j,k)\})$. Thus, in our solution, not only the user observed regrets $B_t(j,k)$ but also the network measured regrets $Y_t(j,k)$ contribute to the update procedure of the user. Second, we do not use a constant proportionality factor μ as in (3.7), but normalise the vector of regret to get a probability vector. This is done to avoid needing to choose an appropriately large arbitrary parameter μ. As discussed in [41], a higher value of μ results in a smaller probability

of switching and thus leads to a slower speed of convergence. It is not clear to us that the proof in [41] of convergence of the Hart's RL-based procedure using (6) could be readily modified to include the form of normalisation we propose in (12).

There are three major terms in the formula (3.13). The first, $B_t^+(j,k)$, is the original regret as observed by the user in a manner similar to [41]. The second, $Y_t^+(j,k)$, is the extra "regret" observed at the BS. As the BS has a more complete view of the system than the individual users, $Y_t(j,k)$ is expected to take into account the information on network load conditions, which may not available under $B_t(j,k)$. Taking the minimum function of the two regrets guarantees that the sum $\sum_{k' \neq j} p_{t+1}^i(k')$ does not exceed 1. The last term, δ_t/m, is the weighted uniform distribution over $\mathcal{I}$ to guarantee that all probabilities at time $t+1$ are at least $\delta_t/m > 0$. This last term together with the scaling of the regrets by $(1-\delta_t)$ ensures that when t is small the algorithm explores different solutions to learn about the network environment. As the algorithm progresses, the regrets become the dominant factors in determining the selection probabilities.

Our algorithm for distributed RAT selection operated by each user is presented as follows.

Algorithm Reinforcement Learning with Network-Assisted Feedback (RLNF)

1: *Exploration:* At the beginning, each user A takes sequential actions to explore all available choices $j \in S_A$ in order to learn possible payoffs and feedback from potential RATs.
2: *Initialisation:* Generate random uniform probability $p_1(j)$ for all $j \in S_A$.
3: **for** $t = 1, 2, \ldots$ **do**
4: *Action Selection:* Select action $i_t = j$ according to the probability distribution $p_t(j)$.
5: *Feedback Exchange:* Obtain feedback Y_t^j from the corresponding base station j.
6: *Regret Update:* for all $k \neq j \in S_A$

- Update the user estimated regret $B_t(j,k)$,
- Update the network measured regret $Y_t(j,k)$.

7: *Strategy Update:* Update $p_{t+1}(k)$ using (3.13).
8: **end for**

3.4.3 Unconditional Variant of RLNF

Note that the set of coarse correlated equilibrium is strictly richer than the set of correlated equilibrium. We can modify p_{t+1} in (3.13) to ensure quick convergence towards the set of coarse correlated equilibria. The idea is that the action probability of every player is proportional to the "unconditional regrets" instead of "conditional regrets" as in equation (3.5), and actions with larger regret get larger probability increased (reinforced) while actions with smaller regrets get smaller probability increased (see [40,41] for more details and discussion on the difference between unconditional and conditional regrets). We call this new variant of RLNF the unconditional reinforcement learning with network-assisted feedback (URLNF).

At time t, the "unconditional regret" for not playing strategy $i \in \mathcal{I}$ every time in the past is defined as [41]

$$CH_t(i) = \frac{1}{t} \sum_{\tau \leq t: i_\tau = i} \frac{1}{p_t(i)} U(i, \ell_t) - \frac{1}{t} \sum_{\tau \leq t} U(s_t).$$

Similarly, this regret can be estimated based on the available information provided by the base stations. Then, the network observed unconditional regret for user A at interaction t for not choosing cellular base station i can be calculates as

$$YH_t(i) = \frac{1}{t} \sum_{\tau \leq t} \left(\frac{\sum_{\tau \leq t: i_\tau = i} (\bar{U}_\tau^i \times n_\tau^i)}{\sum_{\tau \leq t: i_\tau = i} (n_\tau^i + 1)} - \bar{U}_\tau^{i_\tau} \right).$$

or for not choosing Wi-Fi base station i

$$YH_t(i) = \frac{1}{t} \sum_{\tau \leq t} \left(\frac{\sum_{\tau \leq t: i_\tau = i} \bar{U}_\tau^i (n_\tau^i - 1)}{\sum_{\tau \leq t: i_\tau = i} n_\tau^i} - \bar{U}_\tau^{i_\tau} \right).$$

Player A then choose $i \in \mathcal{I}$ at time $t+1$ with probability

$$p_{t+1}(i) = (1 - \delta_t) \min \left\{ \frac{CH_t^+(i)}{\sum_{k \in \mathcal{I}} CH_t^+(k)}, \frac{YH_t^+(i)}{\sum_{k \in \mathcal{I}} YH_t^+(k)} \right\} + \frac{\delta_t}{m}.$$

At the outcome, the player A has no regret for following this learning rule instead of selecting any certain action in all previous time steps, irrespective of the behaviour of the other players. If every player plays according to this learning rule, then their empirical distribution of joint play will converge to the CCE set.

3.4.4 Convergence Properties

Theorem 3.1. *If a player (i.e. player A) uses RLNF algorithm, its time average regret is guaranteed to approach the set of non-positive regrets almost surely irrespective of the behaviour of the other players, for finite payoffs and positive and finite feedback.*

Proof. Please refer to Appendix 3.A for our proof which adopts the notation of [56]. □

Assumption 3.3: We assume that the payoffs are bounded and the network feedback $Y_\tau^k = (\bar{U}_\tau^k, n_\tau^k)$ is positive and finite for all τ, k. This assumption enables us to establish some convergence result for RLNF. In practice, all the payoffs and the feedback that we use (the number of users, the throughput) are finite and positive.

Theorem 3.2. *If all players follow RLNF algorithm, the empirical distribution of joint play of all players $\bar{z}_t(s)$ converges almost surely as $t \to \infty$ to the set of correlated equilibria.*

Proof. Please refer to Appendix 3.B. □

Remark 3.1: Contrary to most existing works that use the classical averaging theory for ordinary differential equations (ODEs) techniques to examines the convergence properties of their game algorithms [48], we use the differential inclusion (DI) framework in [56] to prove our Theorem. In our proof, if a single player uses the proposed procedure, its time average regret is guaranteed to approach its own set of non-positive regrets in the payoff space for any strategies of the other players. All players are required to follow the same algorithm in order to obtain the global convergence of the empirical distribution of joint actions of all players to the set of CE.

The following corollaries trivially follow from the proofs of Theorems 3.1 and 3.2 with small modifications for the construction of the probability vectors, and therefore omitted.

Corollary 1. *Class-1 RAT selection games with the BS observed regret update in* (3.10) *converges almost surely to the set of CE.*

Corollary 2. *Class-2 RAT selection games with the BS observed regret update in* (3.12) *converges almost surely to the set of CE.*

Theorem 3.3. *If a player (i.e. player A) follows URLNF algorithm, its long-run unconditional regrets is guaranteed to approach the set of non-positive regrets almost surely for any strategies of the other players. Moreover, if all players play URLNF, the empirical distribution of the joint play converges to the Hannan set of correlated actions yielding non-positive regrets.*

Proof. Please refer to Appendix 3.C. □

Remark 3.2: In the repeated game literature ([40, 41]), rather than the case "conditional" (or "external") regrets as considered in RLNF, the notion of "unconditional" (or "internal") regret involves a player reasoning about replacing each action played by a fixed strategy. In [41], it's shown that an RL procedure based on unconditional regrets converges to the *Hannan set* of global non-positive regrets if all players play that strategy. The Hannan set contains CE. Furthermore, it's argued in [56] that, in a heterogeneous system where some players may adopt different strategies, those players that use the unconditional regret based algorithm will themselves achieve non-positive unconditional regret. These approaches can be handled within the framework presented in this chapter with appropriate modifications as shown in our proof of Theorem 3.3 in the Appendix 3.C.

3.5 Evaluation

We consider a heterogeneous wireless network environment with 2 different RATs (WiFi and LTE) in a narrow square area of 150×150 meters. We assume that WiFi BSs and users are located within the coverage area of one macro LTE BS at the center of the network. We follow the same network model in [60], that reflect real world WiFi BSs and users distribution. In this model, the connectivity and bandwidth between BSs and users are determined by their geographical distribution. We divide the given geographic area into 9 smaller, non-overlapping square-shaped areas and randomly place a WiFi BS within the borders of each small area. We then place a random number of users (up to 20 – the maximum number of local users for each WiFi BS) for each WiFi BS within the area. A user is considered to be a local user to BSs that are located in the same area of its location and to be a non-local users to the rest of the BSs in the network. We assume that each WiFi BS allocates a certain portion of its bandwidth $(0 \leq \kappa \leq 1)$ to serve the non-local users

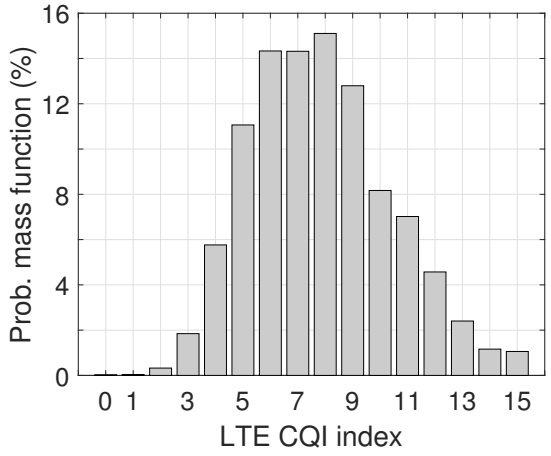

Figure 3.4. Example CQI distribution of a real-world LTE network.

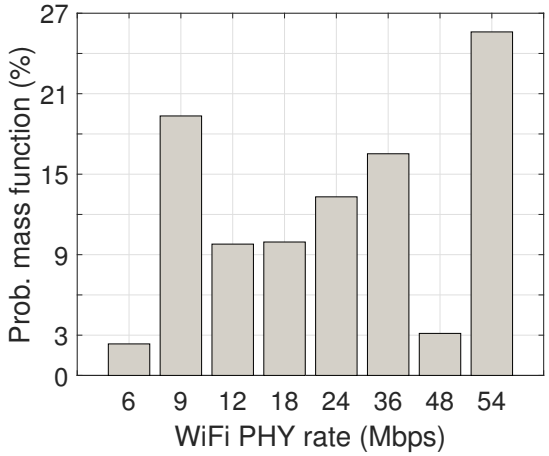

Figure 3.5. PHY rate distribution of a real-world WiFi network.

($\kappa = 1$ for the local users). The actual throughput of users A under the non-local BS k is equal to $\kappa \times \bar{U}_A^k$, where $\bar{U}_A^k$ is given in (3.2).

We use real network data from a tier-1 LTE operator in North America to simulate users' PHY rates to the macro LTE BS. In particular, we use the measured Channel Quality Indication (CQI) and map them to the possible data rates that a user can receive from a BS [71]. Fig. 3.4 shows an example CQI distribution of the real-world LTE BS from the dataset. In the simulation, maximum data rate of 35 Mbps per cell in LTE 20 MHz is assumed [71]. We then linearly divide the data rate into 16 different levels corresponding to the 16 CQI indexes. For example, the strongest CQI 15 correspond to the highest data rate of 35 Mbps and the median CQI 7 corresponds to 17.5 Mbps. A user's PHY rate to a BS is supposed to be unchanged over time.

In addition to LTE data, we also use the collected residential WiFi data [72] in setting up users' PHY rates to WiFi BSs. This data set provides traces of received signal strength (RSS) measurements of the WiFi BSs collected at the University of Colorado. These values are then converted to PHY rates based on Table III as follow. Fig 3.5 shows an example of PHY rate distribution of the WiFi network from the simulated dataset.

To evaluate the performance of our proposed RLNF algorithm, we compare the performance of the following four distributed algorithms for RAT selection:

Table 3.3. PHY rate and the RSS for IEEE 802.11g [73]

PHY (Mbps)	6	9	12	18	24	36	48	54
RSS (dBm)	-90	-84	-82	-80	-77	-73	-72	>-72

- RAT Selection Games (RSG) in [42–44]: All BSs broadcast their traffic information to all users. Thus, each user has the information on the number of other users on each BS and their PHY rates. At each iteration, user selects a BS that provides the highest throughput. This broadcasting assumption is similar to those in [15, 46, 47].

- Regret Matching (RM) in [45]: Users are assumed to have a global view of the network including the actions taken by other users and their historical PHY rates. Users apply the regret matching algorithm [40] to select their RATs.

- Combined Fully Distributed Payoff and Strategy Reinforcement Learning (CODIPAS) in [48]: Users learn and adapt their decisions based on their own observation of the rewards received from past experiences. At each iteration, using only this information, user selects the best available BS to maximise its utility. This is a state-of-the-art RL-based algorithm and has been shown to be superior to the Hart's RL-based scheme in [41].

- *Our Reinforcement Learning with Network-Assisted Feedback (RLNF)*: User data is not required to exchange among the users or the BSs. Each BS shares feedback only to its connecting users to assist them in their RAT selection decisions.

For comparison purposes, we use the following metrics:

- Total overheads (bits): amount of data exchanges between users and BSs. Lower overhead is preferable.

- Convergence time (iterations): required number of iterations to convergence. A fast convergence is desired since the wireless channel conditions may change quickly.

- Per-user switchings: maximum number of switchings required by all users to convergence. A small number of switching is desirable to minimise the cost for managing the vertical switching between RATs.

- Jain's fairness index, which is derived as

$$J = \frac{(\sum_{a=1}^{N} x_a)^2}{N \times \sum_{a=1}^{N} x_a^2},$$

  where x_a is the average throughput of user a and N is the number of users. Notes that the largest value 1 indicating the best fairness of the system, which guaranteeing the same throughput among all the users.

- System utility: sum of all users' average throughputs. Higher utilities benefit both mobile operators and service providers in offering higher bandwidth-services.

We would like to emphasise that users running our RLNF algorithm select their RAT by combining their individual observed throughput and the network feedback; whereas in all the other solutions, users make their RAT selection decisions based only on their own observations. For each network model and algorithm, the actual throughput $\bar{U}_A^k$ that a user A gets from the BS k depends on the other users that share the same BS, and is given in the equations (3.1) and (3.2). The instantaneous throughput U_A^k that a user A observes directly from its connecting BS k is a random number generated according to the Gaussian distribution $\mathsf{N}(\bar{U}_A^k, \sigma^2)$ with $\bar{U}_A^k$ mean and $e \times \bar{U}_A^k$ proportional standard deviation, where we assume the proportional noise factor is $e = 0.3$. This model and choice of parameters were used in [44].

We also set $\delta = 10^{-5} \ll 1$ and $\gamma = 0.1$ for all the simulations of RLNF algorithm. Note that large δ may cause the convergence to a large distance from the CE set. To compare the performance of different schemes versus the number of BSs, we fix the number of LTE BSs to 1 BS and vary the number of WiFi BSs from 2 BSs to 10 BSs. Thus the total BS number varies from 3 BSs to 11 BSs. All the results presented are averaged over 50 simulation runs. Each data point on the graphs is the average value shown with the standard deviation as an error bar.

3.5.1 Performance Comparison

We first compare our RLNF against existing algorithms in convergence behaviour. The feedback used in RLNF is the BS computed throughput and the number of connected

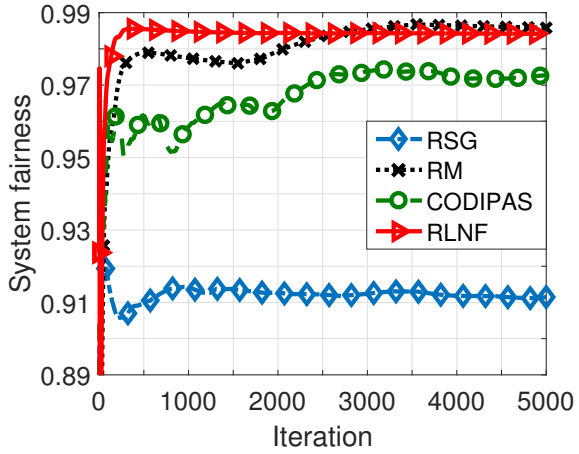

Figure 3.6. Evolution of system fairness index J for different schemes.

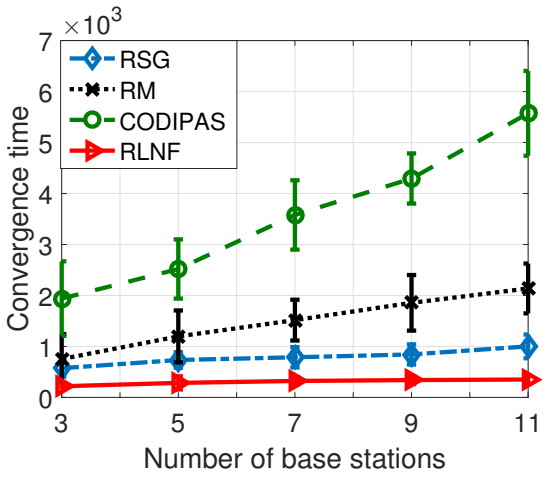

Figure 3.7. Convergence time comparison with varying number of BSs.

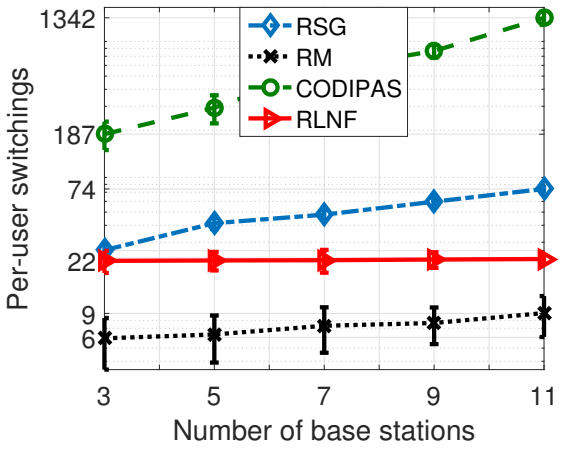

Figure 3.8. Evolution of total overheads by different schemes.

users. In our simulation, the BS computes the actual throughput for each user using equations (3.1) or (3.2) and provide them this number. Each BS also keeps track of the number of users currently connected to it and sends this information to its serving users.

Figs. 3.6, 3.7 and 3.8 show, respectively, the evolution of system fairness index, the convergence time versus number of BSs and the number of RAT switchings for each user (per-user switching) versus number of BSs by different algorithms. We observe that RLNF achieves the fastest convergence with a small number of per-user switchings among all algorithms. Our RLNF even outperforms the RM in convergence speed. It should be note

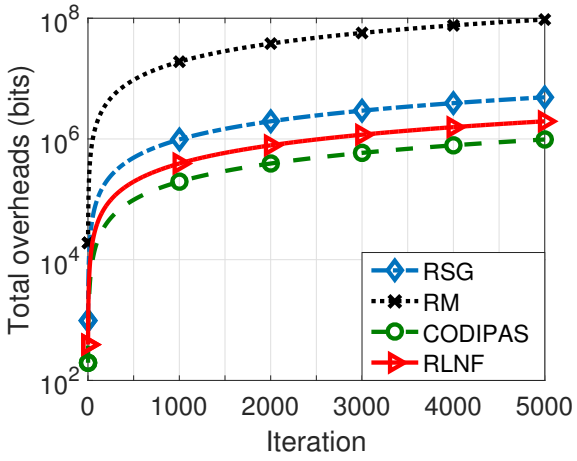

Figure 3.9. Total overheads comparison with varying number of BSs.

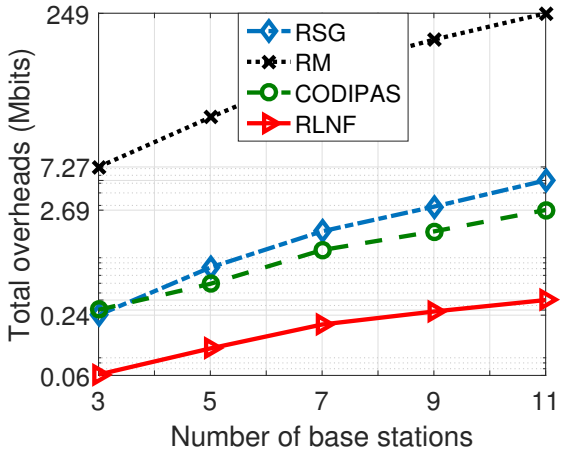

Figure 3.10. Per-user switchings comparison with varying number of BSs.

that in standard RM in [41], payoffs of the players (mobile users) are not noisy. But in our simulation, in order to reflect practical consideration of real-world network for RAT selection, users running our RLNF receive the actual throughputs via network-assisted feedback; whereas in all the other schemes including RM, users only observe their noisy payoffs from their instantaneous throughputs. The network feedback in RLNF is therefore more accurate than the user observed throughput in RM and hence user running RM may take a longer time to learn the throughput in order to converge. Although RM obtains a smaller number of per-user switchings than RLNF, it requires a longer time to converge and exchanges significant communication overheads as we explain later in Figs. 3.9 and 3.10. CODIPAS performs poorest in both convergence speed and per-user switchings metrics due to the lack of information on global network conditions.

We present, respectively, in Figs. 3.9 and 3.10 the total information exchange versus number of iterations and number of BSs across different algorithms in order to compare their overheads. We assume that 4 bits are used to represent the number of users or throughput. Let T be the number of iterations to convergence. The calculations of the information exchanges for each algorithm are summarised below.

- RSG: Each user obtains its payoff from its serving BS ($4 \times N$ bits). Each user also needs to receive the number of connecting users on Class-1 BSs and per-user

throughput on Class-2 BSs ($4 \times N \times (M-1)$ bits) in order to calculate its expected throughputs of joining the other $(M-1)$ BSs. The total overheads are thus $4NM \times$ convergence time (bits) $\sim O(TNM)$.

- RM: Each user obtains its payoff from its serving BS ($4 \times N$ bits). Each user also needs to know the PHY rates and actions taken by other $(N-1)$ users in each iteration ($8N(N-1)$ bits). The total overheads are thus $(8N^2 - 4N) \times$ convergence time (bits) $\sim O(TN^2)$.
- CODIPAS: Each user receives its payoff directly from its serving BS without requiring any extra communication. The total overheads are just $4N \times$ convergence time (bits) $\sim O(TN)$.
- RLNF: Apart from the BS computed per-user throughput (4 bits), user also requires the number of users sharing the same BS (4 bits) from its connecting BS. The total overheads are then $8N \times$ convergence time (bits) $\sim O(TN)$.

Fig. 3.9 shows that with the same number of iterations CODIPAS has the lowest overhead performance. However, as shown in Fig 3.10, RLNF is the best algorithm to minimise overheads. CODIPAS, despite using less information to make decisions, requires higher overheads due to its slower convergence speed, i.e., larger T. Both require an order of magnitude less information exchange than RSG and RM algorithms, especially the later, and when the number of users is large. The reason is that their complexity is linear whereas the complexity of RM is quadratic and the complexity of RSG depends also on the total number of BSs in the network.

We now compare the performances of the algorithms on system fairness and system utility. The fairness and utility results are shown in Figs. 3.11 and 3.12, respectively. As shown, RLNF achieves comparable fairness to RM. Both are better than the others in fairness metric. We also observe that RM, RLNF and CODIPAS achieve very good fairness indexes in all the cases compared to RSG. This can be explained by the fact that these three algorithms are designed to reach an efficient equilibrium points such as CE (RM and RLNF) or optimal-NE (CODIPAS) rather than converging to arbitrary NE as in RSG. Similarly, RLNF achieves very similar utility to the RM, and outperforms the other remaining algorithms.

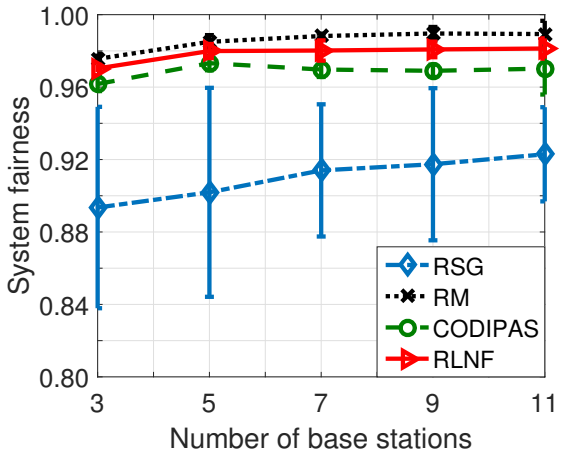

Figure 3.11. System fairness J for varying number of BSs.

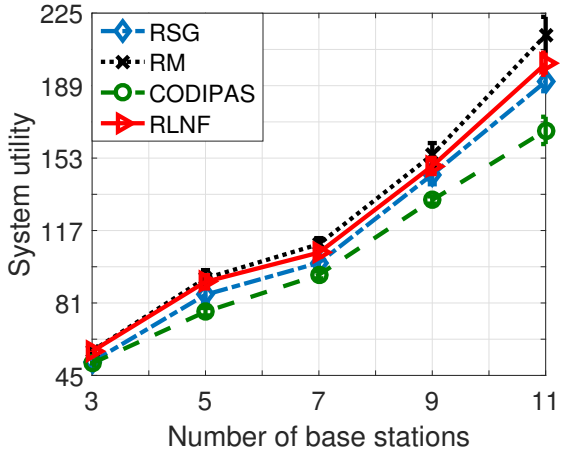

Figure 3.12. System utility comparison with varying number of BSs.

For further evaluation of the scalability of the algorithms, we study the impact of network size (total number of users in the network) on the performances of different algorithms. We fix the total number of BS in the network to 5 BSs (composed of 1 LTE BS and 4 WiFi BSs). We then vary the number of local users per WiFi BS from 10 users/BS to 50 users/BS resulting in increasing the total number of users in the network from 40 users to 200 users. Fig. 3.13 shows the scalability behaviour of the algorithms with respect to the size of the network. Further experiments presented in Fig. 3.13 demonstrate the robust performance and stability of our RLNF algorithm to variations in the network size as compared to relevant RAT selection schemes. Overall, RLNF achieves the fastest speed and lowest overheads, whilst guaranteeing competitive performance both in fairness and utility, as well as requiring a small number of per-user switchings as compared to the others.

3.5.2 Using Feedback to Change Convergence Points

One of the main difference between our RLNF and other game algorithms [15, 42–47] is that our solution can flexibly support a wide range of policy-defined feedback. Under our framework, network operators can influence user decisions to achieve their objectives by tuning their network feedback information. In the following, we show how to apply different feedback in RLNF to achieve different convergence points.

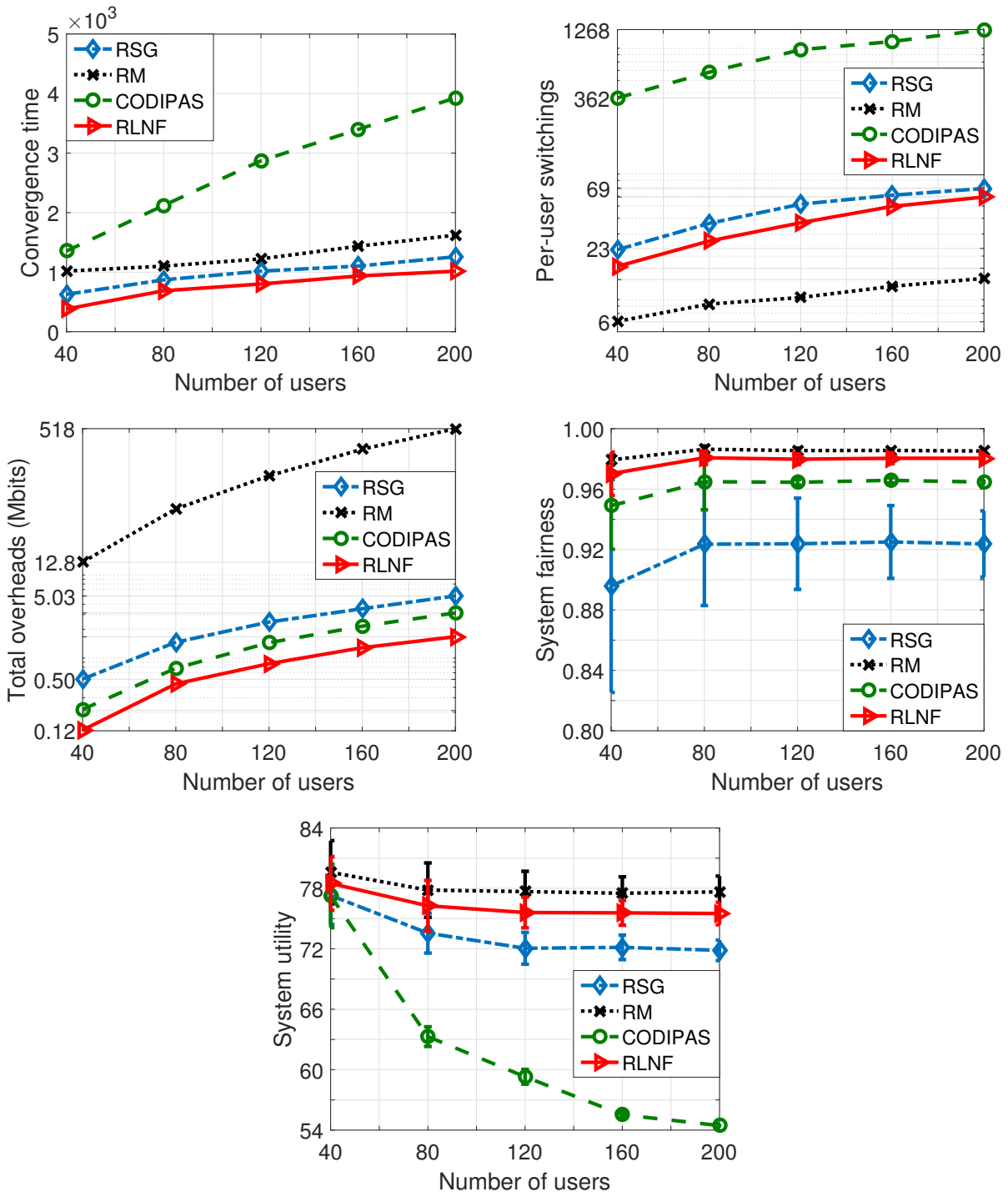

Figure 3.13. Performance comparison of the different algorithms for increasing size of the network (number of users).

As explained in Section 3.4, once the BS has computed per-user throughput, it can send the users this information to aid them in their RAT selections. This network-assisted information, however, does not need to be the actual value of per-user throughput, but can be functions of these throughputs. This type of feedback reflects information about the expected payoff that a user could receive from a BS. Let $\hat{U}_t^k = f(\bar{U}_t^k)$ be the feedback that the BS k sends to its connected users at time t. For simplicity, $\hat{U}_t^k$ can be defined as a function of $\bar{U}_t^k$ as below

$$\hat{U}_t^k = \begin{cases} \bar{U}_t^k\,(1+\gamma) & \text{if } R_a^k \geq \omega_1^k\,, \\ \bar{U}_t^k & \text{if } \omega_2^k < R_a^k < \omega_1^k\,, \\ \bar{U}_t^k\,(1-\gamma) & \text{if } R_a^k \leq \omega_2^k\,, \end{cases} \tag{3.14}$$

where $0 \leq \gamma \leq 1$ is some weighted parameter and $[\omega_1^k, \omega_2^k]$ are PHY rate thresholds defined by the network operator. Each network could use different γ and $[\omega_1^k, \omega_2^k]$ depends on its own policy. Note that the feedback $\hat{U}_t^k$ is equal to real actual throughput $\bar{U}_t^k$ when $\gamma = 0$.

The idea is that network feedback is tuned as a function of the user PHY rate. When user PHY rate on a BS k is higher than a threshold $R_a^k \geq \omega_1^k$, the network encourages that user to select BS k by putting more weight on the feedback throughput. In contrast, when user PHY rate is lower than the threshold $R_a^k \leq \omega_2^k$, the network discourages that user from selecting the BS by reducing the feedback throughput.

In this experiment, instead of using feedback in term of the real actual throughput, we vary the weighted parameter γ according to the feedback form as in equation (3.14); and measure the performance of RLNF in terms of fairness and utility. We also set the PHY rate thresholds $[\omega_1^k, \omega_2^k]$ of WiFi and LTE BSs to be [36Mbps, 12Mbps] and [24Mbps, 10Mbps], respectively. Figs. 3.14 and 3.15 illustrate the impact of different feedback mechanisms on fairness and utility.

As shown, increasing γ improves the total utility, however reduces the system fairness. The reason behind this observation is that increasing γ will encourage users to select the BSs that offer the higher PHY rates, which results in providing them with better throughputs. Therefore, the total utility increases. At the same time, this higher utility comes with a cost of increasing disparities of users' throughputs, which also results in bringing

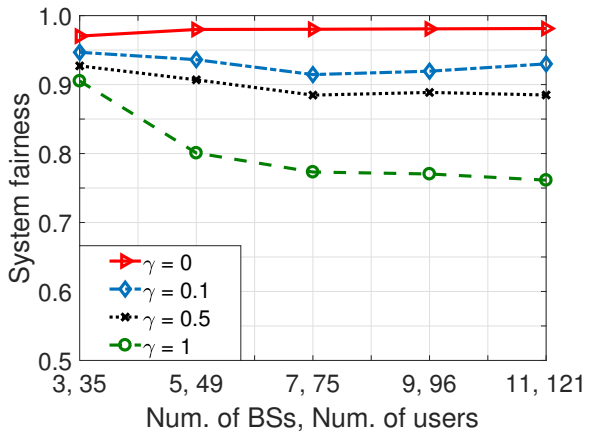

Figure 3.14. Impact of different feedback mechanism on system fairness.

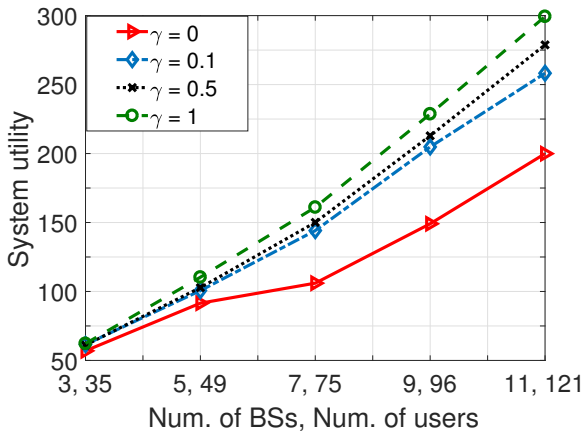

Figure 3.15. Impact of different feedback mechanism on system utility.

down the system fairness. Obviously, there is a trade-off between fairness and utility. Depending on different policies, different feedback mechanisms could be defined to meet the operator's goals.

3.5.3 Performance of RLNF in Heterogeneous Environment

Lastly, we investigate the case where users do not use the same learning rule. Particularly, we simulate a network in a complex heterogeneous situation where half of the users play a random fixed strategy and the others play a random strategy at each iteration, except only one user using an adaptive game algorithm. We randomly select one user among all the users and let that user applies our proposed RLNF. We then repeat the same simulation with different algorithms including RSG, RM and CODIPAS, respectively, for performance comparison purposes. The comparison of average throughput of the selected user running different learning algorithms is illustrated in Fig. 3.16.

As shown, RLNF achieves very close performance to RM scheme and outperforms the others. Note that RLNF does not use global information of the network (how many players are, their actions and payoffs) as required in RM. RLNF achieves faster convergence and exchange significant less overheads, especially for a network with large number of BSs. We observed that the average throughput of users running a random fixed strategy heavily depends on the BSs they select as well as their PHY rates on these BSs. User runs

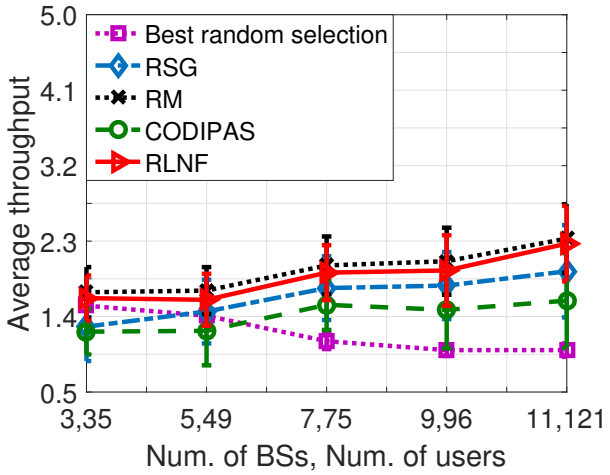

Figure 3.16. Average throughput performance by different schemes in a heterogeneous situation where users use different learning strategies

random selection strategy at each iteration also obtains a very poor throughput when the number of BSs is large. The long-run payoffs of the RLNF user, however, does not depend on either its selected BSs or its PHY rates on any BSs as well as the number of BSs in the network. The result implies that such a user has no regret nor does it lose by committing to use RLNF rather than playing any other strategies. This demonstrates the efficiency of using RLNF in real networks where each user often plays different RAT selection strategy according to its own preference.

3.6 Conclusion

We have studied the problem of RAT selection games in heterogeneous wireless networks. We have developed a new decentralised framework, called Reinforcement Learning with Network-Assisted Feedback (RLNF), that incorporates limited base station measurements in a user's RAT selection policy to achieve fast convergence to the set of correlated equilibria. Our RLNF, as compared to other algorithms, achieves faster convergence rate, lower signalling overheads with a small number of RAT switching per-user, whilst achieving competitive performance both in global network utility and user fairness. More importantly, by adopting an efficient feedback mechanism, RLNF enables mobile users to adapt their selection behaviours to various network feedback, resulting in behaviour that meets

operator objectives while providing users with good performance. Lastly, we show that our solution guarantees non-positive regret in the long-run for any user applying RLNF, regardless of what other users might do and so can work in an environment where other users may not use RLNF. This is an important implementation issue as RLNF can be implemented within current standards. We have demonstrated the improved performance of RLNF compared to other related algorithms using realistic simulations.

Appendices of Chapter 3

3.A Proof of Theorem 3.1

Let $C : Z \to \mathbb{R}^{m\times m}$ be defined by

$$[C(z)]_{j,k} = \sum_{\ell\in\mathcal{L}} z(j,\ell)\,(U(k,\ell) - U(j,\ell))\ ,$$

which is the expected regret for player A when substituting action k for action j under the joint distribution z of actions. Suppose we consider player A playing some action $i = j$ with probability one, then

$$\begin{aligned}[C(z^i)]_{j,k} &= \sum_{\ell\in\mathcal{L}} \mathbb{1}_{\{i=j\}}\, y_\ell\,(U(k,\ell) - U(j,\ell)) \\ &= \mathbb{1}_{\{i=j\}}\,(U(k,y) - U(j,y))\,.\end{aligned}$$

Since player A cannot compute the first term as it only has access to the payoffs corresponding to actions it actually took, following [41], define an estimate of this term by

$$\tilde{U}(k,y)\ \mathbb{1}_{\{i=j\}} = \frac{p(j)}{p(k)}\,U(k,y)\ \mathbb{1}_{\{i=k\}}\ ,$$

which is computed from the regrets associated with the alternative action k weighted proportional to the relative probabilities of player A choosing action j versus k when those actions were actually taken. The associated pseudo regret matrix at stage t is now

$$B_t(j,k) = \frac{p_t(j)}{p_t(k)}\,U(k,y_t)\ \mathbb{1}_{\{i_t=k\}} - U(j,y_t)\ \mathbb{1}_{\{i_t=j\}}\ .$$

Thus, we have

$$\begin{aligned}\mathbf{E}\left\{B_t(j,k)|h_{t-1}\right\} &= p_t(k)\frac{p_t(j)}{p_t(k)}U(k,y_t) - p_t(j)\ U(j,y_t)\\ &= p_t(j)\left(U(k,y_t) - U(j,y_t)\right)\\ &= \mathbf{E}\left\{C_t(j,k)|h_{t-1}\right\},\end{aligned}$$

where h_{t-1} is the action history of the game until stage $t-1$. It can be seen that $B_t(j,k)$ and $C_t(j,k)$ are each bounded by $2mG/\delta_t$. The limit sets of the pair processes B_t and C_t also coincide since they both have the same conditional expected values (see [41] for more details and discussion). Then Theorem 7.3 of [56] can be applied and thus the two processes exhibit the same asymptotic behaviour.

Let $\bar{B}_t(j,k) = \frac{1}{t}\sum_{\tau=1}^{t} B_\tau(j,k)$ be the time-average of $B_t(j,k)$. The average regret at stage t is thus a matrix B_t defined by

$$B_t(j,k) = \frac{1}{t}\sum_{\tau=1}^{t}\left[\frac{p_\tau(j)}{p_\tau(k)}U(k,y_\tau)\mathbb{1}_{\{i_\tau=k\}} - U(j,y_\tau)\mathbb{1}_{\{i_\tau=j\}}\right]$$

We then have the algebraic identity

$$\bar{B}_{t+1} - \bar{B}_t = \frac{1}{t+1}\ \left(B_{t+1} - \bar{B}_t\right)$$

holds. This result follows directly from the definition of average $\bar{B}_t$. Hence, the above discrete dynamics is a discrete stochastic approximation of the DI

$$\dot{\mathbf{w}} \in \hat{N}(\mathbf{w}) - \mathbf{w}\ \ (\text{with } w = B_t)\,, \tag{3.15}$$

where $\hat{N}$ is a mapping from $\mathbb{R}^m$ into the class of all subsets of $\mathbb{R}^m$ (called a *correspondence* on $\mathbb{R}^m$) that satisfies the various conditions outlined in Hypothesis 2.1 of [56] (see [56] for details).

Now define the matrix sequence

$$[M_t]_{j,k} = \min\left\{\frac{B_t^+(j,k)}{\epsilon + \sum_k B_t^+(j,k)}, \frac{Y_t^+(j,k)}{\epsilon + \sum_k Y_t^+(j,k)}\right\} \tag{3.16}$$

for $j \neq k$. We set $[M_t]_{j,j} = 1 - \sum_{k\neq j}[M_t]_{j,k}$ which is in $[0,1]$ by Assumption 3.3 and virtue of (3.16). Thus M_t is a transition probability matrix on $\mathcal{S}$. So there is a probability vector μ_t such that $M_t^T\,\mu_t = \mu_t$.

The "no positive regret set" $D \subset \mathbb{R}^{m\times m}$ for player A is defined by

$$D = \left\{ g \in \mathbb{C}^{m\times m} : g(j,k) \leq 0, \forall (j,k) \right\}.$$

Evidently, D is a closed, convex subspace of $\mathbb{R}^{m\times m}$. Define the Lyapunov function $P(w) = \frac{1}{2}\|w^+\|^2$, with $\nabla P(w) = w^+ \geq 0$. Then P satisfies the following properties:

(i) P is continuously differentiable ;

(ii) $P(w) = 0 \Leftrightarrow w \in D$;

(iii) $[\nabla P(w)]_i \geq 0$ for all $i = 1, \ldots, m$;

(iv) $\langle \nabla P(w), w \rangle > 0$ for all $w \notin D$.

Thus P is a potential function for D. Let $\Pi_D(w)$ be the convex projection onto D, then we have $w^+ = w - \Pi_D(w)$, and $\langle w^+, \Pi_D(w) \rangle = 0$. Let $\varphi : \mathbb{R}^{m\times m} \to 2^X$ given by

$$\varphi(w) = \begin{cases} (1-\delta_n)\mu(w) + \dfrac{\delta_n}{m} & \text{if } w \notin D^1 , \\ X & \text{if } w \in D^1 , \end{cases} \tag{3.17}$$

where $\mu(w)$ denotes a probability vector computed from w^+ according to the process above.

Define a correspondence $\hat{N}$ on $\mathbb{R}^{m\times m} \setminus D$ by $\hat{N}(w) = C(\varphi(w) \times Y)$ so that $\hat{N}(w)$ contains all resulting average regrets. According to Lyapunov theory, to prove the approachability of w to D, it suffices to show that for any $w \in \mathbb{R}^{m\times m} \setminus D$ and some constant $\lambda > 0$,

$$\frac{d}{dt}P(w) = \langle \nabla P(w), \dot{w} \rangle \in \langle \nabla P(w), N(w) - w \rangle \leq -\lambda P(w),$$

meaning $\langle \nabla P(w), \theta - w \rangle \leq -\lambda P(w)$ for all $\theta \in \hat{N}(w)$ (see [56] for details).

Suppose that $w \notin D$, let $\theta = \mathbf{E}\left\{ \tilde{C}(\varphi(w), y) | h_{n-1} \right\}$, with $y \in Y$, which means

$$[\theta]_{j,k} = \varphi_j(w)\ \left(U(k,y) - U(j,y) \right).$$

Then consider

$$\begin{aligned}\langle \nabla P(w), \theta \rangle &= \sum_{j,k}^{m} \nabla P_{jk}(w)\ \varphi_j(w)\ (U(k,y) - U(j,y)) \\ &= (1-\delta_t) \sum_{j,k} \nabla P_{jk}(w)\ \mu_j(w)\ (U(k,y) - U(j,y)) \\ &\quad + \frac{\delta_t}{m} \sum_{j,k} \nabla P_{jk}(w)\ (U(k,y) - U(j,y)) \\ &= (1-\delta_t) \sum_{j} U(j,y) \left(\sum_{k} \mu_k(w)\ \nabla P_{kj}(w) - \mu_j(w) \sum_{k} \nabla P_{jk}(w) \right) \\ &\quad + \frac{\delta_t}{m} \sum_{j,k} \nabla P_{jk}(w)\ (U(k,y) - U(j,y))\ . \end{aligned} \tag{3.18}$$

In the third line we substituted for $\varphi_j(w)$ from (3.17), and in the last line we collected together all terms containing $U(j,y)$.

Let $\mu_t = \mu(w)$ be such a measure. Suppose that for every $j = 1, \ldots, m$, it holds that

$$\mu_j(w) \sum_{k} \nabla P_{jk}(w) = \sum_{k} \mu_k(w)\ \nabla P_{kj}(w)\ ,$$

then the first term in (3.18) is equal to zero. Noting that the payoff function $|U(.)|$ is bounded by G using Assumption 3.3, then

$$\langle \nabla P(w), \theta \rangle = \frac{\delta_t}{m} \sum_{j,k} \nabla P_{jk}(w)\ (U(k,y) - U(j,y)) \leq ||\nabla P(w)||\ \frac{2G\delta_t}{m}. \tag{3.19}$$

Next, consider

$$\begin{aligned}\langle \nabla P(w), w \rangle &= \langle w^+, w \rangle = \langle w^+, w^+ + \Pi_D(w) \rangle = ||w^+||^2 \\ &= 2P(w) \ \ (\text{since } \langle w^+, \Pi_D(w) \rangle = 0). \end{aligned} \tag{3.20}$$

It follows that given $\epsilon > 0$, $||w^+|| \geq \epsilon$, one can choose $\delta_n > 0$ small enough such that

$$\begin{aligned}\langle \nabla P(w), \theta - w \rangle &= \langle \nabla P(w), \theta \rangle - \langle \nabla P(w), w \rangle \\ &\leq ||\nabla P(w)||\ \frac{2G\delta_t}{m} - 2P(w) \leq -P(w). \end{aligned}$$

Thus from Lyapunov theory, the set D is a global attractor for the DI (3.15). Hence, the regret B_t and its corresponding conditional regret C_t will then approach D. Note that

in our proof, Theorem 1 holds no matter what the other players do as long as all the payoffs are bounded. In other words, any user applying our RLNF will achieve "self-consistency" [41] (all its positive regrets approach zero in the long run). This completes the proof.

3.B Proof of Theorem 3.2

The proof follows immediately from how the "regret" measure is defined. Recall that

$$\begin{aligned}[C(z_t)]_{j,k} &= \sum_{\ell \in \mathcal{L}} z_t(j,\ell_t)\,(U(k,\ell_t) - U(j,\ell_t)) \\ &= \sum_{s_t \in S : i_t = j} z_t(s_t)\,(U(k,\ell_t) - U(j,\ell_t))\,,\end{aligned}$$

where $s_t = (i_t, \ell_t)$ is the joint action at time t. On any convergent subsequence $\lim_{t\to\infty} z_t \to \Pi$, we get

$$\lim_{t\to\infty}[C(z_t)]_{j,k} = \sum_{s_t \in S : i_t = j} \Pi(s_t)\,(U(k,\ell_t) - U(j,\ell_t)) \le 0\,.$$

Next, comparing with the definition of CE as in equation (3.3), the desired results follows.

3.C Proof of Theorem 3.3

We first define the *regret vector* $R : Z \to \mathbb{R}^m$ for player A

$$\begin{aligned}[R(z)]_i &= \sum_{s\in S} z(s)\,(U(i,\ell) - U(s)) = \sum_{\ell\in\mathcal{L}} y_\ell\, U(i,\ell) - \sum_{s\in S} z(s)\, U(s) \\ &= U(i,y) - U(z) = U(i,y) - U(x,y)\,,\end{aligned} \tag{3.21}$$

Thus the i-th component of $R(z)$ is the payoff that player A would achieve by playing action i with probability one, instead of the (generally randomised) action specified by x, whilst all other players play according to y.

We denote the set $D^1 = \{g \in \mathbb{R}^m : g_i \le 0, \forall\, i \in I\}$. So $R(z) \in D^1$ if and only if the vector of regrets corresponding to probability z are elements which are all non-positive.

Since player A cannot compute $U(i,y)$, define an estimate

$$\tilde{U}(i,y) = \frac{U(i,y)}{p_n(i)} \mathbb{1}_{\{i_n=i\}}.$$

The associated pseudo regret vector at stage n is now

$$\tilde{R}_n^i = \frac{U(i,y_n)}{p_n^i} \mathbb{1}_{\{i_n=i\}} - U(x_n,y_n).$$

Thus, we have

$$\begin{aligned}\mathbf{E}\left\{\tilde{R}_n^i | h_{n-1}\right\} &= p_n(i)\frac{U(i,y_n)}{p_n(i)} - \sum_{i=1}^{m} p_n(i)\, U(x_n,y_n) \\ &= U(i,y_n) - \sum_{i=1}^{m} p_n(i)\, U(x_n,y_n) = \mathbf{E}\left\{R_n^i | h_{n-1}\right\}.\end{aligned}$$

Similarly to the previous case in Appendix 3.A, we know that the two processes $\tilde{R}_n$ and R_n exhibit the same asymptotic behaviour. The average regret BH_n at stage n is now

$$BH_n(k) = \frac{1}{n}\sum_{t=1}^{n}\left[\frac{U(i,y_t)}{p_t(i)} \mathbb{1}_{\{i_t=i\}} - U(x_t,y_t)\right].$$

For every $i \in I$ define a probability vector μ_n such that

$$[\mu_n]_i = \min\left\{\frac{CH_n^+(j,k)}{\sum_k CH_n^+(j,k)}, \frac{YH_n^+(j,k)}{\sum_k YH_n^+(j,k)}\right\}.$$

Let construct a potential function P for the negative orthant D^1 of $\mathbb{R}^m$ that satisfies the properties (i)-(iv) in Appendix A and the following condition

$$\langle \nabla P(w), w\rangle \geq Q\, ||\nabla P(w)||\, ||w^+||. \tag{3.22}$$

for all $w \notin D^1$ and some positive constant Q.

We now define a strategy for player A based on its regret function. Define a correspondence $\varphi : \mathbb{R}^m \to 2^{\mathbb{R}^m}$ by

$$\varphi(w) = \begin{cases} (1-\delta_n)\mu(w) + \dfrac{\delta_n}{m} & w \notin D^1 \\ X & w \in D^1 \end{cases}, \text{ with } \mu(w) = \frac{\nabla P(w)}{\sum_{i=1}^m [\nabla P(w)]_i} \tag{3.23}$$

Thus $\varphi(w) \in X$ (i.e. is a pmf) whenever $w \notin D$ (we note from property (iii) $\nabla P(w)$ then has no negative components, and by property (iv), $\nabla P(w)$ then can't be identically zero).

Let $\theta = \mathbf{E}\left\{\tilde{R}(\varphi(w), y)|h_{n-1}\right\}$, using linearity of U, the following result holds

$$\langle \varphi(w), \theta \rangle = \left\langle \varphi(w), \mathbf{E}\left\{\tilde{R}(\varphi(w), y)|h_{n-1}\right\} \right\rangle = \sum_{i \in \mathcal{I}} [\varphi(w)]_i \left(U(i, y) - U(\varphi(w), y)\right) = 0.$$

Now, using $\nabla P(w) = ||\nabla P(w)||\, \mu(w)$, it can be show that

$$\begin{aligned} \langle \nabla P(w), \theta \rangle &= ||\nabla P(w)||\, \langle \mu(w), \theta \rangle = ||\nabla P(w)||\, \left\langle \frac{1}{1-\delta_n}\varphi(w) - \frac{\delta_n}{(1-\delta_n)m}, \theta \right\rangle \\ &\leq ||\nabla P(w)||\, \frac{2G\delta_n}{(1-\delta_n)m}. \end{aligned} \tag{3.24}$$

since $\langle \varphi(w), \theta \rangle = 0$ and $|\theta| = |U(i, y) - U(\varphi(w), y)| \leq 2G$. In the first line we substituted for $\mu(w)$ from (3.23).

It follows, using (3.22) and (3.24), that assuming $||w^+|| \geq \epsilon > 0$, one can choose $\delta_n > 0$ small enough such that

$$\begin{aligned} \frac{d}{dt}P(w) &= \langle \nabla P(w), \dot{w} \rangle = \langle \nabla P(w), \theta \rangle - \langle \nabla P(w), w \rangle \\ &\leq ||\nabla P(w)|| \left(\frac{2G\delta_n}{(1-\delta_n)m} - Q||w^+|| \right) \leq -\frac{1}{2}\, ||P(w)||\, Q\epsilon \; < 0\,. \end{aligned}$$

since the condition $\langle \nabla P(w), w \rangle > 0$ for $w \notin D^1$ implies $||\nabla P(w)|| \geq \kappa > 0$ on $||w^+|| \geq \epsilon$. So P is a Lyapunov function for the DI. This proves approachability of the player A regrets to the set D^1 (i.e. all regrets approach zero).

Finally, if all players use the same above procedure, we obtain the convergence of the empirical distribution of the joint actions of all players to approach the *Hannan set* of correlated actions yielding non-positive rewards. The result is immediate from the definition of the "regret" as in (3.21). On any convergent subsequence $\lim_{n\to\infty} z_n \to \Pi$, we get

$$\lim_{n\to\infty} [R(z_n)]_i = \sum_{s_n \in S} \Pi(s_n)\left(U(i, \ell_n) - U(s_n)\right) \leq 0\,.$$

Next, comparing with the definition of Hannan set as in (3.4) completes the proof.

Chapter 4

Performance of Heterogeneous RAT Selection Algorithms

NEXT generation 5G cellular networks will consist of multiple technologies for devices to access the network at the edge. One of the keys to 5G is therefore the ability for devices to intelligently select its Radio Access Technology (RAT). There have been several proposals for RAT selection in the last few years. Understanding the performance and limitation of these RAT selection solutions is important for their deployment in future 5G heterogeneous networks. In this chapter, we provide an overview of recent RAT selection algorithms and the different network models that were used to evaluate these works. We combine these different network models to build a benchmark for evaluating RAT selection algorithms in a 5G environment. We implement the representative algorithms of different approaches and cross compare them in our benchmark. From the experiments conducted, we illustrate how the different network parameters such as the number of base stations that a user sees and the available link bandwidths could impact the performance of these algorithms.

4.1 Introduction

The fifth generation of cellular communication (5G) is rapidly gaining momentum worldwide with commercial deployments scheduled for 2020. 5G is expected to offer a variety of novel technologies that can coexist with existing technologies such as 3G and 4G to support diverse requirements of the various applications and services in the future. Heterogeneous networks (HetNets) that consist of multiple wireless access technologies (e.g., LTE, WiMAX, UMTS, GSM and WiFi, femto, etc) are therefore the key components of future 5G networks [3]. In these networks, mobile devices with multiple radio access technologies (RATs) can connect to and choose among the different base stations (BSs) with different access technologies. Deciding which technology and which BS in that technology mobile users should connect to is known as the RAT selection problem and is a topic of much on-going work within the LTE-WLAN interworking framework of the Third Generation Partneship Project (3GPP) [9] and in 5G research [10–12].

There is now an extensive body of research on RAT selection solutions in HetNets [8, 15, 42–48, 60, 62–64, 74–93]. These solutions cover a wide ranges of solution paradigms from centralised to distributed, from one-shot to iterative game theoretic. Most of these works, however, concentrate mainly on developing novel RAT selection algorithms and testing them on specific network topologies or traces. Despite a number of recent surveys of RAT selection techniques [12], thorough comparative performance evaluation of these algorithms under different network settings have not been explored in the literature.

We provide in this chapter a benchmark for studying impact of various network models on the performances of RAT selection algorithms. We mainly focus on evaluating the state-of-the-art RAT selection algorithms under diverse and realistic network models to understand their strengths and limitation. Our benchmark covers a wide range of network models from throughput, connectivity between users and BSs, BS deployment, and mobility. Using this benchmark, we evaluate and cross-compare the performance of the RAT selection algorithms. We observe significant performance differences for all algorithms when we change the model parameters such as the number of BSs, the number of users and the probability that a link exists between a user and a BS. More interestingly, we find that the expected number of BSs per user has the most impact on the performance of RAT selection algorithms. Our study indicates that RAT selection algorithms should be

evaluated on a range of network model parameters, especially the number of BSs available to a user, to fully understand their limitations.

Our key contributions are:

1. *A taxonomy of existing RAT selection algorithms:* We conduct a brief survey of existing RAT selection algorithms and evaluation platforms in the literature. Based on their attributes, we classify the algorithms into centralised, distributed and hybrid based approaches. We then select and implement the representative algorithms from each group to evaluate their performances on multiple metrics including system fairness, total utility as well as convergence behavior.

2. *A unified benchmark for RAT selection algorithms:* We propose a unified benchmark for performance evaluation of RAT selection algorithms using realistic settings. We consider two particular classes of networks: (i) random graph based model which represents scenarios where users are distributed in the network independently of each other and (ii) geographical based model that reflects real world deployments. Our aim is to provide a simulation benchmark for comparing different approaches to the RAT selection problem. As far as we know, such a unified framework has not been proposed before.

3. *A thorough comparative study:* We provide the first comprehensive evaluation of the impact of different classes of network topology and bandwidth models on the performance of various RAT selection algorithms. For each such group, we investigate several aspects of the network model, including link density (the number of BSs that a user sees), user density (the number of users per BS) and bandwidth distribution (the distribution of link bandwidth between BSs and users) to highlight the impact of each of them on the algorithm performance.

4. *Software library for RAT selection:* We implement in Matlab a library of different RAT selection algorithms including the default association mechanism using highest signal strength, a centralised algorithm with local search, a wide range of game theoretic algorithms (regret matching, reinforcement learning, non-cooperative scheme, combined fully distributed payoff and strategy reinforcement learning). We make these libraries publicly available.

The rest of this chapter is organised as follows. Section 4.2 provides a thorough survey of current RAT selection techniques and evaluation platforms. In Section 4.3, we present a unified benchmark for performance evaluation of RAT selection algorithms. The comparative studies and discussions are presented in Section 4.4. Section 4.5 concludes this chapter.

4.2 RAT Selection Algorithms and Models

4.2.1 RAT Selection Algorithms

RAT selection algorithms can be divided into: (i) centralised (network controlled), (ii) distributed (user controlled), or (iii) hybrid (user controlled with network assistance) solutions. We present the most recent state-of-the-art works on the three different approaches. We use BS to denote any network node that connects directly to end users and offers radio access service such as a base station in LTE network or an access point in WiFi.

Centralised RAT Selection Approaches

In a centralised approach [60, 62, 74–81], all the decisions on which RAT a user connects to are made on the network side. In order to do this, all users need to report their local channel conditions to an authorised network controller. Based on this information, the controller calculates the optimal association of users to RATs with respect to a network objective, and then assigns BS to users. Using this centralised mechanism, service providers can maintain control of network operation to achieve some network related objectives such as network throughput maximization [60, 74, 75], load balancing optimisation [76, 77], user fairness enhancement [78], etc. Centralised approach gaining popularity due to the emergence of future software-defined wireless networks [62, 80, 81].

Centralised algorithms have been shown to be superior than distributed solutions in term of overall network throughput [82]. They, however, require collaboration between all wireless BSs and users – exchanging significant communication overheads, especially for ultra-dense network deployment [94]. Furthermore, different network operators pursue different network sharing strategies. Therefore, such close collaboration may not be possible across multiple networks.

Distributed RAT Selection Approaches

A distributed approach [45, 48, 63, 83–86] can overcome the problem of high communication overhead by implementing the RAT selection algorithms at the user side [12]. Most related distributed solutions are iterative game-based algorithms (for a survey refer to [38]). Distributed game-theoretic techniques can be classed into: partially distributed and fully distributed algorithms. A game-theoretic algorithm is considered to be partially distributed if each player (e.g., user) uses information about the other players in order to update its strategy. While using a fully distributed algorithm, players must be able to make decisions without knowledge of the other players (how many there are, their action and payoffs) [95].

In partially distributed solution such as [45,63], to guarantee convergence, all users are assumed to have a global knowledge of the network including the payoff function and the selection histories of other users. From these, they are able to determine their throughputs given other users' choices. This assumption implies that each user knows the instantaneous throughputs of the other users. The guaranteed convergence therefore comes at the cost of increased complexity, signaling and communication load.

In contrast, a fully distributed solution such as [48, 83–85] does not require the users to know anything about other users. Each user learns about the RAT selection "game" by observing only its own achieved payoffs. Despite this very attractive property, the conventional fully distributed algorithms in [48,83–85], however, suffer from the problems of slow convergence, and of convergence to sub-optimal equilibrium points due to the lack of knowledge on global network traffic [86].

Hybrid RAT Selection Approaches

In hybrid approaches [15, 42–44, 46, 47, 87–91], mobile users select their RAT depending on their individual observations as well as external information provided by the network. Several works such as [42–44,46,87] propose network-assisted schemes where some global knowledge of network is broadcast to every user in the network. Each user then uses these parameters to select the best BS that satisfies its utility requirements. These works however still require a large amount of additional information exchange between the users and the BSs.

To further reduce the signalling overhead by the broadcast technique, the works in [89–91] develop low-overhead distributed algorithms in which each BS shares limited feedback information only to its serving users to assist them in making RAT decision. The feedback sent to the users is related only to the local information of each BS such as the number of connecting users [89, 90], the achievable throughput offered by the BS [90], the BS traffic load [89] or the channel state condition between user and BS [91]. This approach reduces significantly the overheads in the network.

In these hybrid approaches, although the BS may provide some useful information, this knowledge is not guaranteed to be perfect or reflect the global condition of the network. Therefore, users will need to keep switching among the available BSs to discover how it would associate with the BSs to meet its objective. This leads to a high number of exploration times and results in a low per-user throughput.

4.2.2 Algorithms under Consideration

In the following, we discuss and compare the fundamental properties of the six representative algorithms of the classes reviewed above. We limit our discussion to two dominant RATs: WiFi and Cellular. We particularly focus on information input and the types of data exchanged between the users and the BSs. We summarise this discussion in Table 4.1.

One-shot algorithms:

- **Highest Signal Strength (HSS)** This is a fully distributed approach and is the current default user association mechanism in the 802.11 standard. Users have no information about the global network state. Based on their radio conditions, they randomly select a BS among the highest received signal strengths. In order to implement this algorithm, we assume a user randomly belongs to one of the two groups of users. That is either prefer WiFi network or prefer cellular network to mobile access with equal probabilities.

- **Local Search Heuristic (LSH) in [79]** This algorithm is based on a centralised approach, in which a controller searches for all possible associations between users and BSs. It assigns users to either WiFi or cellular BSs in a way to maximise the

Table 4.1. Summary of RAT selection algorithms under consideration in Chapter 4.

Algorithm	LSH	HSS	RM	RSG	ERL	CODIPAS
Learning-based	Centralised	Fully distributed	Partially distributed	Hybrid	Hybrid	Fully distributed
One-shot or Iterative	One-shot	One-shot	Iterative	Iterative	Iterative	Iterative
Infomation requirements	Global	Local	Global	Global	Local	Local
Data exchange among users	Yes	No	Yes	No	No	No
Knowledge of payoff function	Yes	No	Yes	Yes	No	No
External feedback from BSs	Yes	No	No	Yes	Yes	No
Convergence equilibra	No equilibrium	No equilibrium	CE	NE	CE	NE

sum of logs of the user's throughputs instead of the total users' individual throughput. This optimization method has been shown to significantly improve the overall network throughput while maintains the good fairness of user throughputs.

Iterative algorithms:

- **Regret Matching (RM) in [45]** The key idea of this partially distributed scheme is to adjust the user's action probability proportional to the "regrets" for not having chosen other actions. In this solution, users are assumed to have a global view of the network including the BSs selected by other users and their historical throughputs. Thus, each user is able to compute the regrets (the changes in average payoff) that it would have if selecting other BSs instead of its current BS. Users apply the RM procedure [40] that assures no regret in the long run to select their RATs. This algorithm converges to the set of correlated equilibrium (CE). CE is an optimality concept of game theory that models possible correlation between players compared to the usual strategic equilibrium of Nash, where all players act independently [40].

- **RAT Selection Games (RSG) in [44]** In this hybrid approach, the network runs a centralised algorithm to determine the global network traffic including the number of concurrent users on each BS and their physical (PHY) data rates. Each BS then broadcasts these assisted parameters to all users in its coverage area, including those that are not currently using it as a RAT point. Thus, each user can estimate its expected throughput if it switches to another BS. At each time step, each user selects a BS that provides the highest per-user throughput. This algorithm converges to a Nash equilibrium (NE) [44].

- **Enhanced Reinforcement Learning (ERL) in [90]** The main idea of this hybrid scheme is to help users estimate their payoffs more accurately using network-assisted feedback that are readily available at their associated BSs. In this solution, each BS shares the number of its concurrent users and the long-term achievable throughput (computed at the BS) that a user could receive to its serving users to aid them in their RAT selection decisions. From these feedback and its own observations, each user can estimate its obtainable throughput from all other target BSs and compute the network measured regrets, which indicate how much gain (or loss) in

average payoff if leaving the currently associated BS. Users then apply the ERL procedure in [90] that follows the regret-based principle [41] to select their RATs. This algorithm also guarantees convergence to the set of CE almost surely.

- **Combined Fully Distributed Payoff and Strategy Reinforcement Learning (CODIPAS) in [48]** In this fully distributed solution, users do not need to exchange their data to other users or BSs. Each user learns and adapts its RAT selection decisions only based on its own observation of the payoffs received from past experiences. At each time step, using only this local information, a user selects the best available BS to maximise its payoff. This algorithm guarantees convergence to a NE.

4.3 A Benchmark for RAT Selection Evaluation

4.3.1 Overview of Current Evaluation Platforms

Network topology

There are a large number of network topologies that have been used for wireless network simulations. Kauffmann *et. al.* [85] consider both a static and a dynamic topology. In static topology, users are assumed to be static and can only communicate to a fixed set of BSs. This kind of topology is easy to deploy but does not accurately reflect the actual networks. Dynamic topology demonstrates the more realistic scenario where users can join or leave the network at any time but increase the complexity of the simulation model.

Wang *et. al.* [87] evaluate their algorithm in a randomly deployed network, where both BSs and users are distributed according to a homogeneous Poisson Point Process in a a geographic region. In contrast, Ge *et. al.* [94] use a more complex heterogeneous topology, where BSs are sampled according to a non-homogeneous point process and therefore results in region with a very high density of BSs. The work in [60], instead, varied the BS density as well as user density to study the impact of these parameters on their algorithm.

To model the network under different deployment strategies in cellular network, Du *et. al.* [86] use three representative topologies scenarios including a chain-topology (treated as the roadside cellular network BS), a nestification-topology (represented the multimode

small cells deployment) and an overlapping-topology (reflected the conventional scenario of partially overlapping cells) to illustrate the applicability of their solution in many complex scenarios. Some other works such as [42, 44, 60, 87] validate their solutions in real-word networks by using the collected residential data traces via driven experiments.

Typically, most of the existing works evaluate their proposed algorithms on a selected network topology, often with a small number of BSs and full connectivity between users and BSs. These simple models may not reflect the realistic scenarios of future 5G ultra-dense heterogeneous network [94].

Bandwidth Allocation

Bandwidth allocation is a primary factor that significantly affects performance of wireless network. However, most of the prior works rely on simplifications such as uniform throughput among all clients and consider only a single class of throughput model. For example, the work in [63, 85, 86] assumes that all users connecting to the same BS are allocated with an equal amount of bandwidth. This assumption is suitable to model the throughput-fair access technologies in WLAN environment. Other works apply to a single class of RATs such as WiFi network in [45, 60] or cellular network in [46, 87]. Only a number of previous works [42–44, 64, 89, 90] look at heterogeneous network scenario where different RATs use different bandwidth allocation techniques. Those solutions that work with multiple RATs are more attractive due to the recent development of HetNets.

Recently, the throughput-fair and proportional-fair models in [42–44] as well as the service differentiation based throughput model in [46,86] are becoming popular. These works however ignore the fluctuating nature of the wireless channel by assuming that the users know the long-term average throughput that a user experiences on a wireless network. Unfortunately, in practice, for distributed or hybrid solutions, each user only knows its sampled throughput (instantaneous value), from which it infers the mean value. Inference from a limited number of samples always contains statistical errors. It is therefore important to take into account the statistical errors in evaluating RAT algorithms.

In the rest of this section, we propose a unified simulation model to evaluate and compare the performances of various RAT selection algorithms under the same network environment. With our model, one could also investigate the effect of varying various network parameters on the algorithm performance to fully understand its limitations.

4.3.2 Network Topology

We consider a wireless network consisting of M BSs and N users under two particular classes of networks: (i) random associations between users and BSs based on generic random graphs, which resembles the popular random Poisson point process for users distribution in wireless networks, and (ii) the correlated association models based on geographical distance, which better reflect real-word topology deployment.

Random Graph Based Model

Random graph is a popular mathematical tool to model the link connectivity and to study the scaling capacity of wireless networks [96]. Under a random graph model [96], users are assumed to be located within the coverage range of each BS (hence can potentially connect to that BS) independently of each other with a fixed probability. To generate the random topologies, we assign a probability p that a link (a connection from a user to a BS) is available for a certain user independently among all pairs (user, BS). We call *link density* as the expected number of BSs that a user sees (pM). Fig. 4.1 demonstrates the generic random graph scenario.

Geographical Based Model

In this scenario, the connectivity and bandwidth between BSs and users are determined by their geographical distances. We follow the same network models in [60, 87], that reflects real world BSs and users distribution. We consider a densely deployed networks, where a large number of small cells (e.g., Pico/Femto/WiFi BSs) are located within the coverage area of one macro BS in a narrow area [94]. We divide the given geographic area into smaller, non-overlapping square-shaped areas and randomly placed a BS within the borders of each small area. We then place a uniform random number of users (up to λ,

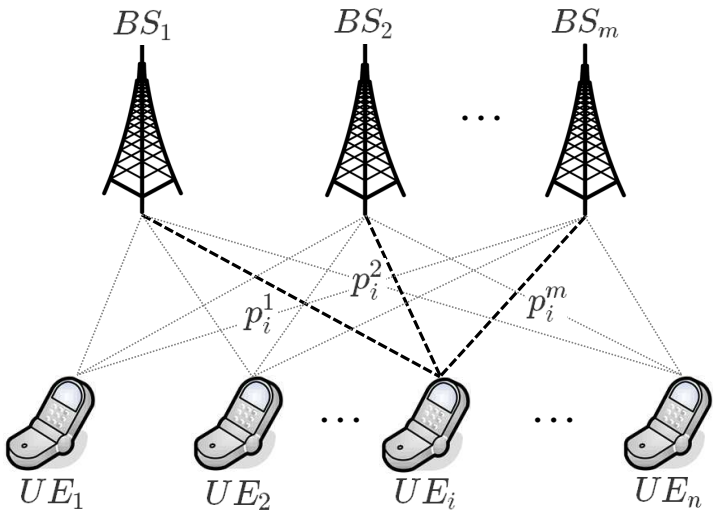

Figure 4.1. The scenario of BS and users in a random graph model.

the maximum number of users that a BS serves) for each BS within its area. A user is considered to be a local user to BSs that are located in the same area of its location and to be a non-local users to the rest of the BSs in the network. We assume that each BS can allocate a certain portion of its bandwidth $(0 \leq \alpha \leq 1)$, to serve other non-local users ($\alpha = 1$ for the local users).

4.3.3 Bandwidth Allocation

In this work, we are primarily interested in user downlink throughput and use the same throughput models as in [42–44] for different RATs.

Throughput-Fair Model

Under this model, in the long term, a set of users connected to the same BS receive the same per-user throughput. The throughput of user i connected to BS k is given by

$$\bar{\omega}_i^k = \left(\sum_{i'=1}^{n^k} \frac{1}{R_{i'}^k} \right)^{-1}, \tag{4.1}$$

where $R_{i'}^k$ is the PHY rate of user i' on BS k and n^k is the number of concurrent users on k. This model is suitable for throughput-fair access technologies such as WiFi.

Proportional-Fair Model

Under this model, each user obtains a different user-specific throughput which is a function of its PHY rate and the number of other users sharing the same BS. The throughput of user i choosing BS k can be expressed as follows

$$\bar{\omega}_i^k = \frac{R_i^k}{n^k}. \tag{4.2}$$

This model is suitably used to model time/bandwidth-fair access technologies such as 3G/4G cellular networks.

4.3.4 Instantaneous Throughput Model

Note that the throughputs given in the equations (4.1) and (4.2) are the mean (e.g., long term average) throughputs, which can be only computed at the network side. In distributed solutions, users only sample their instantaneous throughputs, not the mean values. At any one time, instantaneous throughput observed by the user may vary from the mean. This issue has been considered in [38], where the instantaneous achievable throughput of a user is modeled as a random variable.

In this chapter, we propose an instantaneous throughput model that can be efficiently implemented for computer simulations. In this model, we assume that user observed throughput follows a Gaussian distribution in which the mean is equal to the throughput computed by the network and the standard deviation is equal to the product of the noise e and the mean throughput $\bar{\omega}$. Thus, instantaneous throughput of a user i choosing a BS k is a Gaussian random variable:

$$\omega_i^k \sim \mathcal{N}(\bar{\omega}_i^k, \sigma_i^2),$$

where $\sigma_i = e \times \bar{\omega}_i^k$ and $e \in (0, 1)$. This instantaneous throughput model incorporates more practical considerations of real-world networks for RAT selection.

4.4 Comparative Studies

We perform comparative studies of the six algorithms in Section 4.2.2 under different network models in Section 4.3.2. We first use synthetic data to simulate a HetNet environment where users are located in the coverage of two different RATs: WiFi and LTE. For the

Table 4.2. PHY rates in WiFi and LTE BSs

Base station	WiFi	LTE
Good radio condition	48 Mbps	16.6 Mbps
Normal radio condition	24 Mbps	12.2 Mbps
Bad radio condition	9 Mbps	7.4 Mbps

sake of simplicity, we assume that half of the BSs are LTE base stations while the others are WiFi access points. Each user has three possible radio conditions to each BS, namely good, normal or bad. A user's PHY rate to a BS is supposed to be unchanged over time. For each pair of BS k and user i, we assign the good/normal/bad PHY rate to R_i^k with equal probabilities of 1/3. These PHY rates when connected alone to these BSs are listed in Table 4.2. We will relax these assumption by using real network data in Section 4.4.2.

For each network model and algorithm, the mean achievable throughput $\bar{\omega}$ a user gets depends on the other users that share the same BS, and is given in the equations (4.1) and (4.2). The instantaneous throughput ω that user observes individually from its BS is a random number generated according to the Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ with $\bar{\omega}$ mean and $0.3\,\bar{\omega}$ proportional standard deviation, where we assume the proportional noise factor is $e = 0.3$. All the results presented are averaged over 10 simulation runs.

In order to compare the algorithms in term of efficiency and fairness, we perform numerical tests on the following metrics:

- System utility: sum of all users' average throughputs. Higher utilities benefit both operators (higher offered bandwidth) and end-users (better per-user throughput).
- Jain's fairness index, which is derived as

$$J = \frac{(\sum_{i=1}^{N} x_i)^2}{N \times \sum_{i=1}^{N} x_i^2}, \tag{4.3}$$

  where x_i is the average throughput of user i and N is the number of users. Notes that J reaches the largest value 1 indicating the best fairness of the system, which guaranteeing the same throughput among the users.

To compare the iterative algorithms in convergence performances, we consider the following metrics:

- Total overheads (bits): amount of data exchanges between users and BSs. Lower overhead is preferable.
- Convergence time (iterations): required number of iterations to convergence. A fast convergence is desired since the wireless channel conditions change quickly.
- Per-user switchings: maximum number of switchings required by all users to convergence. A small number of switching is desirable to minimize the cost for managing the vertical switching between RATs.

4.4.1 Random Graph Based Model

We first report our results for the random graph case. We vary p from 0 to 1 and measure the performance of RAT selection algorithms in system fairness and system utility. Figures 4.2 and 4.3 illustrate the impact of link density on the performances of the six algorithms in Section 4.2 for two different BS numbers.

Impact of link density on system fairness

Our first observation is that all iterative algorithms are robust and obtain very good fairness performance as compared to that of one-shot algorithms, especially when the link density is large $(pM > 4)$. Among iterative algorithms, RM (requiring global network information) achieves the best performances. Both regret-based algorithms (RM and ERL) achieve better fairness than the others. This can be explained by the fact that they both are designed to reach efficient CE points [40, 41] rather than converging to arbitrary NE solutions as in RSG and CODIPAS. LSH performs poorer than all the iterative algorithms in term of fairness, with a maximum of 0.85 for a low link density of 6, since it aims to maximise network throughput not fairness.

We explain this observation by the following proposition.

Proposition 2. *For any constant $\epsilon > 0$, under the random BS and user association model, any user can connect to at least one BS with probability $1 - \epsilon$ if*

$$pM \geq M\left(1 - \sqrt[M]{\epsilon}\right),$$

where M is the number of BSs.

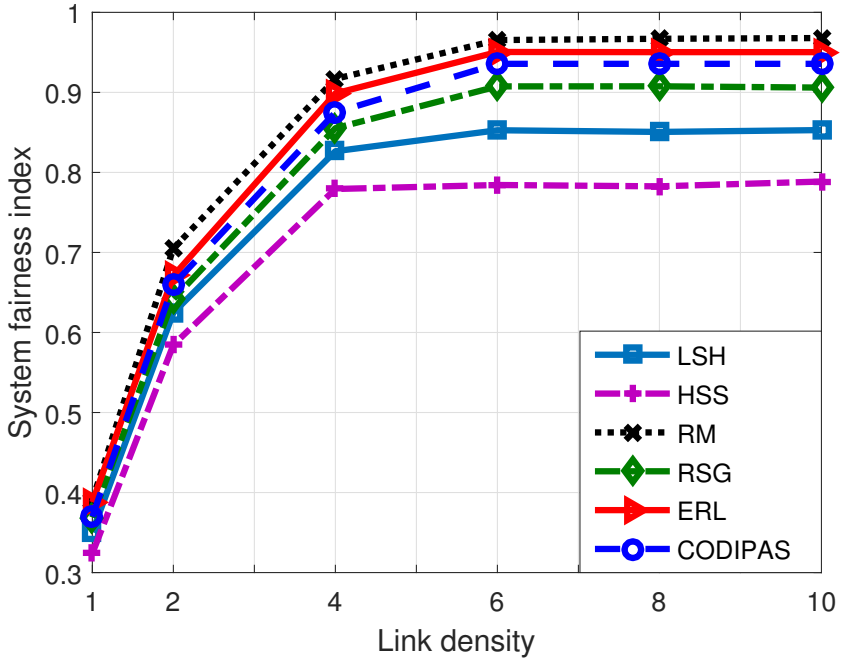

Figure 4.2. Impact of link density on fairness for network with 150 users and 10 BSs.

Proof. Let random variable X_i denotes the number of BSs that user i sees. Thus, X_i is considered as a binomial random variable with parameter (M, p). The probability for X_i to be l is given by

$$Pr\left[X_i = l\right] = \left(\begin{array}{c} M \\ l \end{array} \right) p^l \left(1-p\right)^{M-l}$$

The probability that user i can see at least one BS can be calculated as

$$Pr\left[X_i \geq 1\right] = 1 - Pr\left[X_i = 0\right] = 1 - \left(1-p\right)^M \tag{4.4}$$

In order to achieve this with high probability, we want

$$Pr[X_i \geq 1] \geq 1 - \epsilon \tag{4.5}$$

for all user i. Where ϵ is a pre-determined reliability threshold. For example, to guarantee a 99% confidence interval, we set $\epsilon = 1 - 0.99 = 0.01$. Thus, from (4.4) and (4.5), we have

$$1 - \left(1-p\right)^M \geq 1 - \epsilon \;\Leftrightarrow\; pM \geq M\left(1 - \sqrt[M]{\epsilon}\right)$$

This completes the proof. □

With $M = 10$ and let $\epsilon = 0.01$, we obtain $pM \geq 3.69$ from Proposition 1. We can see that the analytical result match the simulation result reasonable well.

The above formulation means, under probability condition, a user can associate with at least one BS when its link density is higher than a certain threshold value. Accordingly, a distributed iterative algorithm, which aims at maintaining maximum fairness among users, can be used to obtain a high system fairness index at an equilibrium point.

This observation has yielded a primary insight about the impact of link density on the fairness performance of iterative algorithms. That is an iterative game algorithm can achieve very good fairness performance when the link density is large enough (for example in a densely deployed networks). However, under this scenario, the increase in link density does not help to bring much higher performance in system fairness and therefore can result in wasting network resources.

Impact of link density on system utility

In term of utility, the one-shot algorithms achieve much higher performance than the iterative algorithms. Interestingly, when the link density is 18 in our simulation, the system utility reaches its highest value. Thus, even with higher link density, the centralised LSH algorithm could not bring better system utility. Also, when the link density is large enough $(pM > 24)$, the distributed HSS algorithm can achieve similar performance as the centralised one. Thus, in such a densely deployed network, we do not even need a centralised solution in order to maximise overall network throughput.

Among iterative game algorithms, RM that uses global information of the network also achieves highest performance in utility. ERL (using only assisted feedback from local BS) performs poorer than RSG (using network-assisted information from all BSs) when increasing the link density of the network. CODIPAS, which requires the least amount of network information, has the poorest utility. Again, when the link density reaches 18, the game algorithms could not improve much performance in utility metric.

In any network deployment scenarios, it is important to have mechanisms for associating users to BS so that the available network resources is efficiently used. From the perspective of a single user, increasing network density is always beneficial for increasing individual data rate. However, this might not be optimal from a network-wide viewpoint.

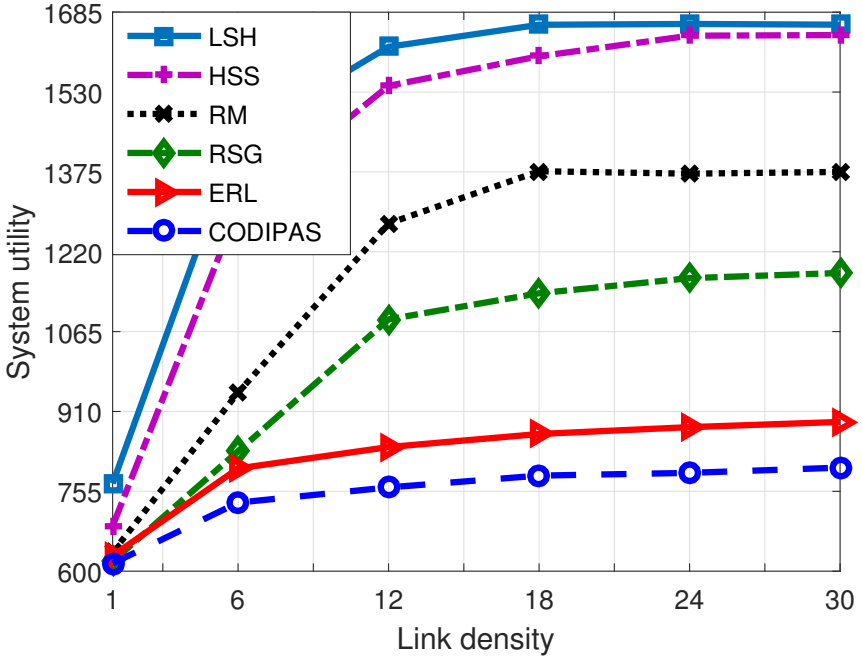

Figure 4.3. Impact of link density on utility for network with 150 users and 30 BSs.

In the following, we explore the answer to the question what is the condition for maximizing network throughput. We show in Proposition 2 that even a centralised algorithm (which has complete information regarding the network) could not bring better network throughput when the link density reaches a certain value.

Proposition 3. *For any constant $\epsilon > 0$, under the random BS and user association model, total throughput is maximised if*

$$pM \geq \frac{M}{\beta}\left(1 - \sqrt[M]{\epsilon}\right),$$

where M is the number of BSs, β is a probability that a user obtains a good radio condition to a BS.

Proof. For simplicity, we assume that a user can obtain a good radio condition to a BS with a fixed probability of β. It is obvious that the network can obtain the maximise throughput if every user can see at least one BS that offers the highest PHY rate (meaning that every user can potentially connect to at least one BS with good radio condition). The probability that a link with good radio condition is available for a certain user is $p\beta$. Let random variable Y_i denotes the number of BSs with good radio conditions that user i can sees.

According to binomial distribution,

$$Pr\left[Y_i = l\right] = \left(\begin{array}{c} M \\ l \end{array} \right) (p\beta)^l (1 - p\beta)^{M-l}$$

The probability that user i can see at least one BS with good radio condition can be calculated as

$$Pr\left[Y_i \geq 1\right] = 1 - Pr\left[Y_i = 0\right] = 1 - (1 - p\beta)^M \tag{4.6}$$

Similarly, to achieve this with high probability, we want

$$Pr[Y_i \geq 1] \geq 1 - \epsilon \tag{4.7}$$

for all user i. Thus, from (4.6) and (4.7), we have

$$1 - (1 - p\beta)^M \geq 1 - \epsilon \;\Leftrightarrow\; pM \geq \frac{M}{\beta}\left(1 - \sqrt[M]{\epsilon}\right)$$

When this condition is satisfied, a solution that maximises the sum of throughput of all the users can be implemented by using a centralised algorithm, such as LSH. The network then can achieve its maximise throughput and hence higher link density does not necessarily provide higher aggressive throughput. This completes the proof. □

Let $\beta = 1/3$ according to the simulation setting, Theorem 2 is satisfied for the condition of $pM \geq 11.07$. This result again matches with what we observe in the simulation.

In summary, we observe similar trend in the evolution of performance for all algorithms with varying link densities. When the number of link density is small, increasing the link density of the network brings significant difference in algorithm performance both in fairness and utility. As the link density reaches a certain threshold, which is 4 in terms of fairness as in Fig. 4.2 and 18 in terms of utility as in Fig. 4.3 in our simulation, all the algorithms reach their limits. Thus, higher link density does not necessarily provide higher performance in either fairness or utility and therefore can result in wasted network resources. This implies that neither the number of BSs nor the probability that a link exists between a user and a BS has significant effect on the performance of RAT selection algorithms. It is the link density that makes the difference.

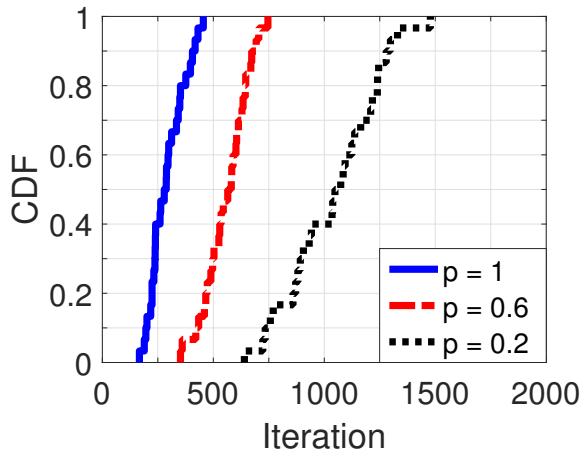

Figure 4.4. Impact of p on convergence time of Regret Matching.

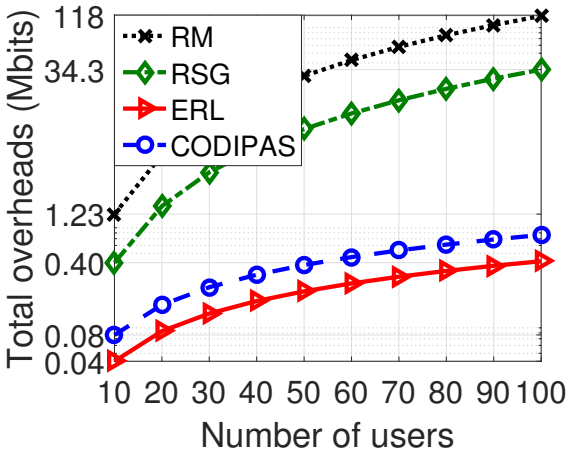

Figure 4.5. Convergence performance comparison in term of total overheads

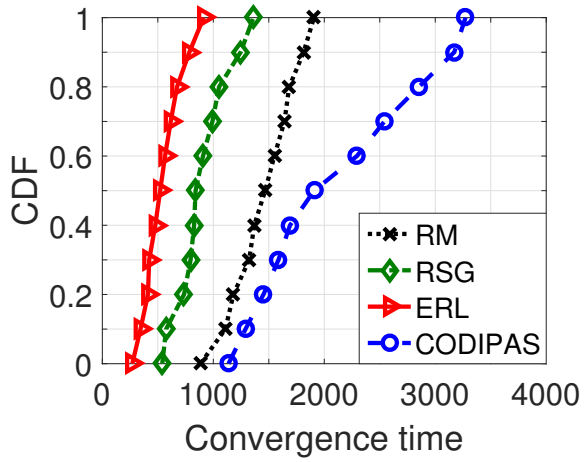

Figure 4.6. Convergence performance comparison in term of convergence time.

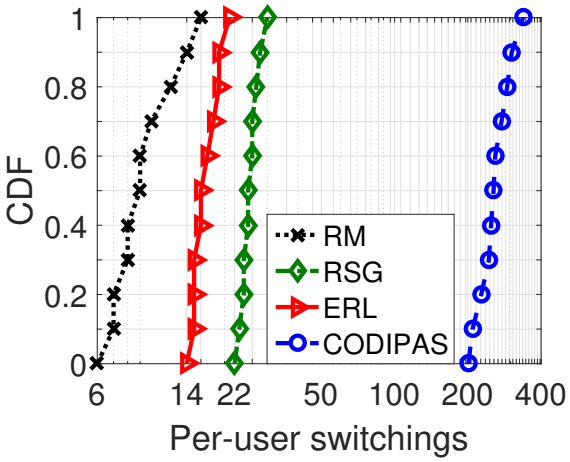

Figure 4.7. Convergence performance comparison in term of per-user switching.

Performance comparison of iterative algorithms

The probability p that a link is available between a user and a BS, however, significantly affects the convergence rate of iterative game-based algorithms. Fig. 4.4 shows the impact of p on the convergence speed of the RM scheme. As p increases, the convergence rate improves rapidly. The same observations on the impact of p apply to the other schemes (RSG, ERL and CODIPAS). This confirms the well-known fact of using iterative game-based algorithms: the more information you have, the better the solution.

We now fix $pM = 4$ and measure the performances of the four iterative algorithms in term of total overheads. We present in Fig. 4.5 the amount of data exchange between users and BSs across different algorithms in order to compare their overheads. We assume that 4 bits are used to represent the real-valued SNR, PHY rate and number of users or throughput. The calculations of the information exchanges for each algorithm are summarised below, where τ is the number of iterations to convergence.

- RM: Each user is required to report its SNR to its connecting BS and obtains its payoff as well as its PHY rate ($12 \times A$ bits). Each user also needs to know the PHY rates and actions taken by other $(A-1)$ users in each iteration ($8A(A-1)$ bits). The total overheads are thus $(8A^2 + 4A) \times$ convergence time (bits) $\sim O(\tau A^2)$.

- RSG: Each user is required to report its SNR to its connecting BS and obtains its payoff ($8 \times A$ bits). Each user then needs to receive the PHY rates of all the users from each BSs ($4 \times A^2$ bits). The total overheads are thus $(4A^2 + 8A) \times$ convergence time (bits) $\sim O(\tau A^2)$.

- ERL: Apart from the mean achievable throughput (4 bits), user also requires the number of users sharing the same BS (4 bits) from its connecting BS. The total overheads are then $8A \times$ convergence time (bits) $\sim O(\tau A)$.

- CODIPAS: Each user receives its payoff directly from its associated BS. The total overheads are just $4A \times$ convergence time (bits) $\sim O(\tau A)$.

As shown in Fig. 4.5, the best algorithm to minimize overheads is ERL. CODIPAS, despite using less information to make a decision, requires higher overheads due to its slower convergence speed, i.e., larger τ. Both ERL and CODIPAS require an order of magnitude less information exchange than RSG and RM algorithms, especially when the number of users is large. The reason is that their complexity is linear whereas the complexity of RSG and RM algorithms is quadratic.

Figs. 4.6 and 4.7 compare the algorithms in term of convergence time and per-user switchings. We observe that ERL achieves the fastest convergence rate among all algorithms. ERL even outperforms RM and RSG. This can be explained by the fact that the network

feedback in ERL is more accurate than the user observed throughput in RM and RSG. Although RM obtain a smaller number of per-user switchings than the others, it requires a longer time to converge and exchanges significant higher overheads as we explained earlier in Fig. 4.5. CODIPAS performs poorest in both speed and per-user switchings metrics due to the lack of information on global network conditions.

4.4.2 Geographical Based Model

To accurately emulate real-work network deployment, we consider an HetNet environment where WiFi BSs and users are located within the coverage area of one macro LTE BS at the center of the network. We use real network data, in particular the measured CQI, from a tier-1 LTE operator to simulate user's PHY rates to the macro LTE BS. In addition to LTE data, we also use the received SNR collected from several WiFi BSs across a university campus, in setting up users' PHY rates to WiFi BSs. These values are then converted to a PHY data rate (which we assume to be constant over time) based on the mapping table of the corresponding technology, and are fed to our simulation. Simulation parameters of the WiFi and the LTE network are set according to [90]. Figs. 4.8 – 4.11 illustrate the impact of the user density (number of user per BS) and bandwidth distribution (portion of bandwidth to serve non-local user) on the algorithms performances using the geographical based model.

Impact of user density

In this setup, we fix the total number of BS in the network to 5 BSs (composed of 1 LTE BS and 4 WiFi BSs) and enable a share portion of bandwidth $\alpha = 0.3$ on each BS. We vary the user density from 10 to 50. The results are shown in Figs. 4.8 and 4.9. It can be seen that iterative algorithms are quite robust in achieving system fairness to the change of user density compared to LHS and HSS. However, increasing user density reduces the total utility of the network in all the algorithms. The reason is that rational users running RAT selection algorithms select their BS in order to maximise their own payoffs, which could bring down the payoffs of other users that connect to the same BS. For example, a user tries to connect to a BS of low PHY rate but could yield a high payoff when the number of users on that BS is small.

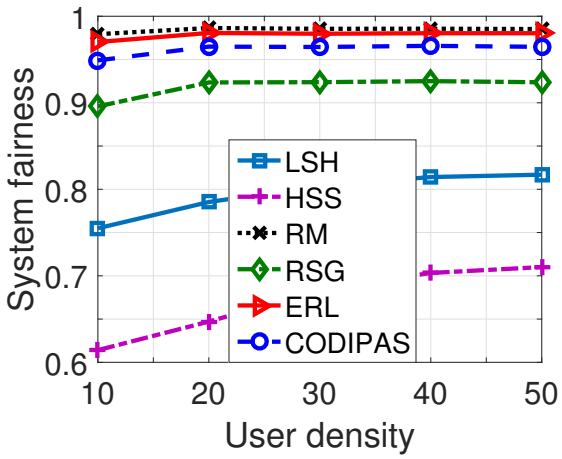

Figure 4.8. Impact of user density on system fairness.

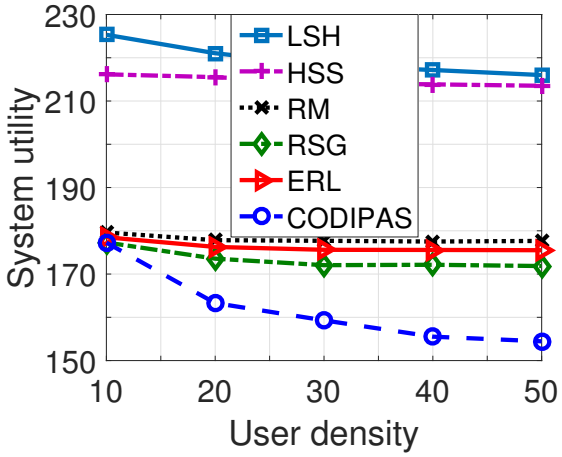

Figure 4.9. Impact of user density on system utility

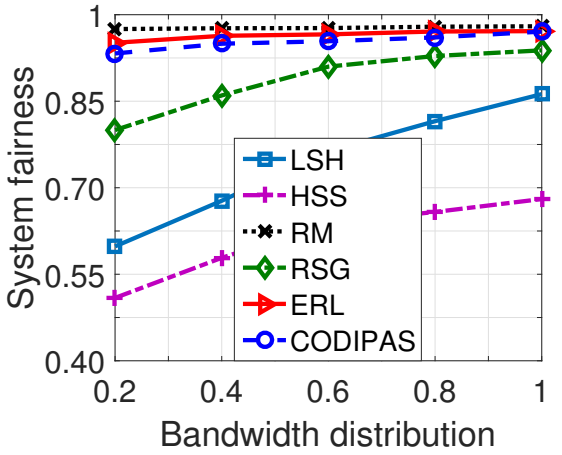

Figure 4.10. Impact of bandwidth distribution on system fairness.

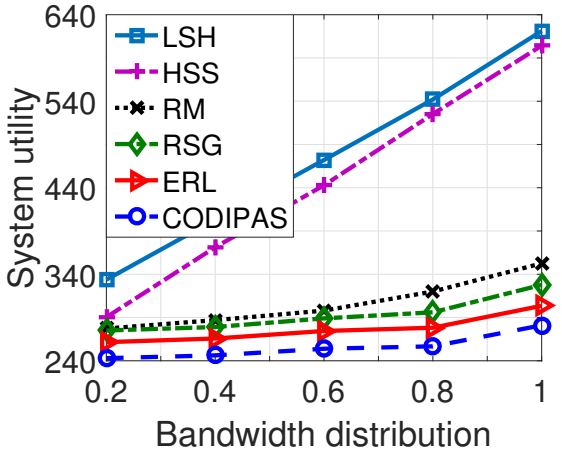

Figure 4.11. Impact of bandwidth distribution on system utility.

Impact of bandwidth distribution

In this setup, we fix the total number of BS in the network to 11 BSs (composed of 1 LTE BS and 10 WiFi BSs) and the user density to 20 users/BS. We vary the bandwidth distribution α on each BS from 0.2 to 1. The results are shown in Figs. 4.10 and 4.11. As shown, increasing α improves both utility and fairness. This can be explained by the fact that increasing α is equivalent to increasing BS density, users thus have more options to select their preferred BSs that offer the higher PHY rates, which also results in better per-user throughputs. Therefore, the overall utility and system fairness improve.

4.5 Conclusion

In this chapter, we start with a brief review of existing RAT selection algorithms and evaluation platforms. We then investigate the impact of different aspects of network models on the performances of representative algorithms from different approaches via a unified benchmark. Our aims are to compare the performance of various algorithms under the same computational environment and to investigate the effect of various network parameters such as link density, user density and link bandwidth distribution on their performances. The unified evaluation benchmark in this chapter can serve as a reference for researchers, network developers, or engineers.

We studied two particular classes of networks: (i) random associations between users and base stations which resembles the popular random Poisson point process for users distribution in wireless networks and (ii) the correlated association models based on geographical distance that are observed in real world deployments. Simulation results reveal that among all the important network parameters that influence the performance of RAT selection algorithms, the number of base stations that a user can connect to has the most significant impact. This finding provides some guidelines for the proper design of RAT selection algorithms for future 5G.

Chapter 5

Adaptive Reinforcement Learning With Forgetting Factor

THE algorithm presented in this chapter is outside the scope of most recent developments of RAT works. Existing RAT selection algorithms often assume constant physical rates, hence perform poorly in networks with high mobility. In this chapter, we propose a new distributed RAT selection solution that can effectively handle user mobility. Our algorithm is based on the reinforcement learning based regret minimization principles and has the advantage of using forgetting methods to react appropriately to the various mobility situations of mobile users. In our solution, instead of using a constant forgetting factor for all users, each user has its own time-varying forgetting factor. This enables any individual user to adaptively adjust its forgetting factor based on its current mobility profile and its own observation of the changes in the network. We prove theoretically that the proposed algorithm (1) guarantees the long-term achievable throughput for any user adopting it no worse than choosing any fixed BS, regardless of the behaviour of other users; and (2) converges almost surely to the set of correlated equilibria when all users apply it. Simulation results demonstrate that our algorithm can converge much faster when users move and greatly improves the overall network throughput compared to the non-adaptive solutions.

5.1 Introduction

Next generation 5G wireless networks typically constitute various types of radio access technologies (RATs) for devices to access the network at the edge. One of the keys to 5G is to enable mobile users to seamlessly and smoothly switch between different available RATs. Choosing the appropriate RAT a wireless devices connects to for good performance is vital but non-trivial. This is known as the RAT selection problem, and has recently received much attention from the research community due to the increasing deployment of heterogeneous wireless network (HetNets) [3] that consist of multiple wireless access technologies (such as WiFi, 3G, 4G and potential 5G technologies).

There is now an extensive body of academic research on RAT selection solutions for HetNets, e.g. [42–44, 48, 60, 74, 79, 90, 97, 98], which cover a wide ranges of solution paradigms from network-centric (wherein the RAT association is controlled by the network) to user-centric (wherein each mobile user decides its serving RAT by itself) approaches [10]. User-centric approaches have been shown to be superior than network-centric approaches in term of scalable deployment, energy efficiency and computational complexity, especially for a large-scale network [12, 82]. Most of previous user-centric solutions in this area, however, often either ignore the impact of user mobility on the algorithm performance [91] or assume basically unchanged network parameters (i.e., user PHY data rate) over the timescale of the algorithm [90]. This is often unrealistic in real-word mobile network due to the dynamic nature of the wireless environment [99]. In fact, a RAT selection algorithm can work efficiently when users are static but may lead to bad performance when users are mobile [100]. This is because user mobility occurs frequently and makes the best RAT to which a user is connected change over time. As a result, the current RAT association might not be optimum in the future when a user moves and thus reselection of the associated RAT is needed .

Since performances of the RAT selection algorithms are highly dependent on their ability to handle user mobility, the design of mechanisms considering user mobility is necessary. In this study, we address this challenge by proposing a new user-centric RAT selection algorithm that can adapt itself in response to various mobility situations of the end-users. We focus on fully distributed reinforcement learning (RL) based approaches due to the

advantages of their low overhead and superior scalability. Our algorithm follows the RL-based regret minimization principle [41] combining with the use of a variable forgetting factor. Forgetting method enables user to quickly adapt to fluctuations of its per-user throughput due to its mobility. Using our learning technique, a user can adaptively identify the change in the network condition when it moves and effectively select the appropriate serving BS at a given time in order to maximise its long-run per-user throughput.

Our main contributions are summarised as follows:

1. We address the problem of using the user-centric approach for RAT selection in a dynamic network scenario where users move. Previous proposed solutions on user-centric RAT selection cannot converge fast enough in this realistic network settings.

2. We develop a new adaptive reinforcement learning-based algorithm that leverages benefit of forgetting properties to rapidly react to the changes in the network due to user mobility. We prove theoretically that the proposed algorithm is guaranteed to converge to a stable set of correlated equilibria. Our algorithm is more efficient than previous RL-based methods because it works in a dynamic heterogeneous environment, where users could apply a number of different RAT selection procedures.

3. Using simulation, we demonstrate the adaptability and performance improvement of our adaptive learning scheme compared with non-adaptive solutions under different user mobility models, including random mobility and group mobility models.

The rest of this chapter is organised as follows. In Section 5.2, we discuss the related work. In Section 5.3, we present our system model and assumption. We formally propose our adaptive reinforcement learning with varying forgetting factor in Section 5.4. The evaluation is presented in Section 5.5. Finally, we conclude the chapter in Section 5.6.

5.2 Related Work

There is extensive literature focusing on RAT selection solutions in wireless networks. We only discuss here the major differences between our approach and the relevant works.

5.2.1 RAT Selection Algorithms

RAT selection techniques have been studied in either network-centric or user-centric contexts. In a network-centric approach [60, 74, 79], all the decisions on which RAT a user connects to are made on the network side. Although network-centric solution can find a global optimal allocation for the whole system, it requires real-time signalling between users and BSs, between different BSs that belong to different RATs, and would have a very high computational complexity particularly in a large-size dynamic network. Contrary to the network-centric approach, a user-centric approach [42–44, 48, 90, 97, 98] does not require extensive signalling and coordination among the different RATs or users. In a user-centric solution, based on local observations at the user side, the user make decision to select its serving RAT by itself. User-centric algorithms often assume global knowledge of the network [42–44], or availability of some additional information provided by the network [90, 97, 98] in order for the user to make the best RAT decision since the local view of the network observed by each individual user is not guaranteed to be accurate.

Although user mobility is an important factors that has significant impact on the performance of the RAT selection algorithms, most prior related literature, including [42–44, 48, 60, 74, 79, 90, 97, 98], neglects the effect of user mobility when evaluating algorithm performance. These works mainly ignore the change of the physical (PHY) data rates due to the movement of the users and thus are only suitable for RAT selection in such a stationary network conditions, i.e., static users and a time-invariant environment. Therefore, these schemes may not work well when a user moves during the time making its RAT selection decision.

5.2.2 Mobility Support in RAT selection

As mentioned previously, one of the main issues involved with RAT selection mechanism is the ability to support user mobility. Different mobility situations of users will need to be served whilst maintaining optimal performance under network condition changes due to users' movements. Most research studies proposed so far that aim to provide mobility support for RAT selection have mainly focused on a centralised approach [101–103]. A centralised method requires a global network controller that monitors and manages the entire network. This centralised controller decides the association of a given user to a

particular RAT with the aim to optimise the network utility such as network throughput or load balancing. The advantage of the centralised solution is its simplicity since the centralised controller can follow user movements by keeping track of the mobile signal strength data collected by the BS that the mobile user connected to. However, this approach suffers from scalability issues in a dense network environment, as a large volume of the traffic will pass through a single controller. Within the centralised approach, there have been several proposals [62, 80] that adopt the software defined networking (SDN) framework. SDN solution also uses a centralised controller to configure the data path via different RATs and thus provides the flexibility to support user mobility.

Unlike centralised solutions, by enabling a user to decide its appropriate RAT itself, distributed mechanism is more flexible and scalable, especially in dense RAT deployments [104]. However, only a number of previous works [97, 105, 106] directly support user mobility. The authors in [97] propose a distributed RAT selection algorithm that considers the impact of the time-varying network conditions due to user's mobility patterns. In [105], the authors present a distributed algorithm that can track the daily movement of the device's owner and combine with past wireless measurements in order to predict the upcoming connection quality of the network. Similarly, the work in [106] proposes a distributed scheme that uses past mobility history for the prediction of WiFi availabilities to perform data offloading from cellular to WiFi network. However, these works either assume that each user must know in advance the statistical information about the network conditions of its mobility patterns [97] or requires training [105, 106] beforehand and thus cannot work in the case where the network condition deviates from the learned pattern.

In this chapter, we propose a novel online adaptive RL-based RAT selection algorithm that supports user mobility by taking into account the effect of user's movement in the online decision making process for RAT selection. Our work differs from the above related works in the following ways:

1. We show in this chapter, using simulation with realistic network settings, that the proposed algorithm, by using a forgetting factor, can converge much faster than previous RL-based algorithms when users move.

2. We prove theoretically that any single user who uses the proposed algorithm can achieve the average per-user throughput, in the long run, no worse than choosing

any fixed BS selection, regardless the behaviour of the other users. This is an important implementation issue as this solution not only can handle user mobility but also works in a dynamic heterogeneous environment, where all users are not required to apply the same algorithms.

3. Our algorithm also works well for a moving user with time-varying mobility profiles. With our solution, a user can adaptively adjust its own forgetting factor over time depending on its current mobility situation.

5.3 System Model and Assumption

We discuss in this section the wireless network model and the assumptions made in our RAT selection algorithm.

5.3.1 Wireless Network Throughput Model

We consider a heterogeneous wireless network consisting of M base stations (BSs) and N users. We use BS to denote any network node that connects directly to users such as a base station in LTE or an access point in WiFi network. In this work, we are primarily interested in user downlink throughput and use the same throughput model as in [46]. Under this service differentiation based model, each user obtains a different user-specific throughput, which depends on its PHY rate to the BS, the number of users sharing the same BS, and the load on the associated network. The throughput of user a associated with BS k can be expressed as follows

$$U_{a,t}(k) = \frac{w_a \, \Theta^k_{a,t}}{\sum_{a'=1}^{n^k_t} w_{a'}}, \tag{5.1}$$

where $\Theta^k_{a,t}$ is the instantaneous PHY data rate of user a on BS k at time t, n^k_t is the number of users on BS k at time t, w_a is user a's weight which reflects the individual throughput demand of the user a, and $\sum_{a'=1}^{n^k_t} w_{a'}$ is the total user weight on the BS k indicating the load of the associated network at time t. This model takes into account various throughput demands of users with diverse applications and is suitable for cellular technologies such as LTE-A [46].

Most existing studies on the distributed RAT selection problem assume a perfect stable network environment, where the physical (PHY) data rate of a user to a BS remains unchanged during the iterations of RAT selection. However, in a mobile network, users' physical rates change frequently, violating the assumption of invariant physical rates in most RAT solutions. In this work, we explicitly model the change of PHY data rate of user due to mobility and develop a RAT solution for this model. In particular, the PHY data rate of a user and on a BS varies as a function of their physical distance. We describe the user mobility model that we use and how to compute the physical distance from a user to a BS using this mobility model in Section 5.3.2.

5.3.2 User Mobility Model

We use the random mobility and group mobility models [107] to model user movement (refer to [107] for a survey of different mobility models). Mobile users are assumed to move with constant velocity and fixed direction starting from a randomly chosen point inside the simulation area. The velocities and directions of mobile users (pedestrian) are assumed to be independently random variables uniformly distributed in the range of $[v_{min}, v_{max}]$ m/s and $[0, 2\pi]$ radians, respectively.

Let $\mathbf{r}_n^a$ denotes the location of a user a at time step t then the dynamic model of user a is given by

$$\mathbf{r}_{t+1}^a = \mathbf{r}_t^a + \Delta T \, \mathbf{v}_a(\theta_a),$$

where ΔT is the time interval between discrete position updates, θ_a is heading angle direction of user a, and $\mathbf{v}_a$ is its velocity vector, which is defined as follows[6]

$$\mathbf{v}_a(\theta_a) = \|\mathbf{v}_a\| \begin{bmatrix} \cos\theta_a \\ \sin\theta_a \end{bmatrix}.$$

Thus, the distance between the user a and the BS k at a given time t can be computed by

$$d_{a,t}^k = \|\mathbf{r}_t^a - \mathbf{q}^k\|,$$

where $\mathbf{q}^k$ is the location of the BS k.

[6]Where $\|(.)\|$ is the Euclidean norm

5.3.3 RAT Selection Game Model

In the following, for consistency, we use the notation developed in Chapter 3. We model the RAT selection as a repeated game where each player (mobile user) aims to maximise its long-run average payoff (per-user throughput). We consider a game with N players denoted by the set $\mathcal{N} = \{1, \ldots, N\}$ for some (finite) integer $N \geq 2$. Each player a has its set of finite actions $\mathcal{S}_a$ (set of available BSs) and we denote by $\mathcal{S} = \mathcal{S}_1 \times \cdots \times \mathcal{S}_N$, the set of joint action of all players, i.e. the Cartesian product of all players' possible actions.

We view the game from the point of view of player A (a randomly selected player among the player set). Let $\mathcal{I} = \mathcal{S}_A$ denote the set of actions of player A and $\mathcal{L} = \mathcal{S} \setminus \mathcal{S}_A$ the set of actions of all other players. Denote by X, the set of all probability mass functions (pmf) on $\mathcal{I}$ and Y the set of pmf on $\mathcal{L}$. Let Z denote the set of pmf on $\mathcal{S}$, then $X \times Y$ is a subset of Z comprised of all pmf of the form $z = (x, y)$ where $x \in X$ and $y \in Y$, i.e. all pmf of the probability of the player A's action and the actions of the others taken together.

Let $U_A : \mathcal{S} \to \mathbb{R}$ denote the payoff achieved by player A when the overall action taken by all players is $s \in \mathcal{S}$. We represent a strategy in the form $s = (i, \ell)$ where i is the action of player A and ℓ is the action of all other players. We will consider the general formulation of the game where users apply mixed strategies over the possible selection set $\mathcal{S}$. Under randomised actions with overall probability (pmf) $z \in Z$, the payoff obtained by player A is defined as

$$U_A(z) = \sum_{s \in \mathcal{S}} z(s)\, U_A(s)\ .$$

The RAT selection game then can be denoted by a 3-tuple $(\mathcal{N}, (\mathcal{S}_A)_{A \in \mathcal{N}}, (U_A)_{A \in \mathcal{N}})$. In our game model, each player A knows only its set of actions (S_A) and its stream of payoffs (U_A) received in the past. Players are not aware of other players' actions and payoffs.

In order to find the equilibrium of the RAT selection game, correlated equilibrium (CE) is used. CE is an optimality concept introduced by Aumann [23] and is proven to exist for any finite games with bounded payoffs [25]. CE models possible correlation between players' actions compared to the usual strategic equilibrium of Nash, where all players act independently. A probability distribution ψ defined on $\mathcal{S}$ is said to be a CE if for all player $A \in \mathcal{N}$, for all $\ell \in \mathcal{L}$ and for every pair of action $j, k \in \mathcal{I}$, it holds that

$$\sum_{\ell \in \mathcal{L}} \psi(j, \ell)(U_A(k, \ell) - U_A(j, \ell)) \leq 0, \tag{5.2}$$

A CE results if each player does not benefit from choosing any other probability distribution over its actions, provided that all the other players do likewise. When each player chooses their action independently of the other players, or without any implicit co-ordination mechanism, a CE is also a NE.

5.4 Reinforcement Learning With Forgetting Factor

A fully distributed algorithm that can be used to reach the CE solution is the reinforcement learning-based regret minimisation procedure in [41]. Before presenting the formulation and proposed algorithm, we first provide a brief introduction on the RL-based regret minimization in [41]. The key idea of this method is to adjust the player's play probability proportional to the "regrets" for not having played other actions. Specifically, for any two actions $j \neq k \in \mathcal{I}$ at any time t, the cumulative regret of player A up to time t for not playing action k instead of its played action j is defined as[7]

$$\bar{R}_t^A(j,k) = \frac{1}{t}\sum_{\tau \leq t} \left(U_A(k,\ell_\tau) - U_A(j,\ell_\tau)\right) \mathbb{1}_{\{i_\tau = j\}}, \tag{5.3}$$

where i_τ denotes the action taken by player A at time τ (i.e., $i_\tau = j$ means player A select BS j at time τ) and ℓ_τ denotes the actions of the others at time τ. This is the change in the average payoff that player A would have if he had played action k every time in the past that he actually chose j.

If $i_t = j$ is the action chosen by player A at time t, then the probability distribution that player A chooses an action at time $t+1$ is defined recursively as [41][8]

$$p_{t+1}(k) = \begin{cases} (1-\delta_t)\min\left\{\dfrac{\left[\bar{R}_t^A(j,k)\right]^+}{\mu}, \dfrac{1}{m}\right\} + \dfrac{\delta_t}{m} & \text{if } k \neq j, \\ 1 - \displaystyle\sum_{k' \neq j} p_{t+1}(k') & \text{if } k = j, \end{cases} \tag{5.4}$$

with the initial play probabilities at $t = 1$ uniformly distributed over the set $\mathcal{I}$; $\mu > 2mG$ is a constant with m being the cardinality of the set $\mathcal{I}$ and G being an upper bound on $|U(s)|$ for all $s \in \mathcal{S}$; $\delta_t = \delta/t^\gamma$, $0 < \delta < 1$ and $0 \leq \gamma < 1/4$.

[7] Where $\mathbb{1}(.)$ denotes the indicator function.

[8] We use the notation $x^+ := \max(x, 0)$ for a real number x throughout this chapter (e.g. $\left[\bar{R}_t^A(j,k)\right]^+ = \max(\bar{R}_t(j,k), 0)$). The definition is extended to real vectors and matrices elementwise.

It is proven in [41] that if *every* player chooses their actions according to (5.4), then the empirical distribution of joint actions s of all players until time t, which is given by

$$\bar{z}_t\big(s=(j,\ell)\big) = \frac{1}{t}\sum_{\tau\le t} \mathbb{1}_{\{s_\tau=(j,\ell)\}},$$

converges almost surely as $t\to\infty$ to the CE set of the game.

5.4.1 Recursive Formula with Forgetting Factor

In our solution, instead of computing the cumulative regret using (5.3) in each time step, player A can recursively compute its cumulative regret $\bar{R}_t^A(j,k)$ at time t using the recursive formula as follows

$$\begin{aligned}
\bar{R}_t^A(j,k) &= \frac{1}{t}\sum_{\tau\le t}\big(U_A(k,\ell_\tau)-U_A(j,\ell_\tau)\big)\mathbb{1}_{\{i_\tau=j\}}\\
&= \frac{1}{t}\left[\left(\sum_{\tau\le t-1}\big(U_A(k,\ell_\tau)-U_A(j,\ell_\tau)\big)\mathbb{1}_{\{i_\tau=j\}}\right)+\big(U_A(k,\ell_t)-U_A(j,\ell_t)\big)\mathbb{1}_{\{i_t=j\}}\right]\\
&= \frac{1}{t}\left[(t-1)\bar{R}_{t-1}^A(j,k)+\big(U_A(k,\ell_t)-U_A(j,\ell_t)\big)\mathbb{1}_{\{i_t=j\}}\right]\\
&= \left(1-\frac{1}{t}\right)\bar{R}_{t-1}^A(j,k)+\frac{1}{t}R_t^A(j,k), \qquad (5.5)
\end{aligned}$$

where we define $R_t^A(j,k) = \big(U_A(k,\ell_t)-U_A(j,\ell_t)\big)\mathbb{1}_{\{i_\tau=j\}}$ as the instantaneous regret of player A for not playing action k instead of its played action j at time t. Equation (5.5) updates the cumulative regret at each time step by adding the correction term based on the new instantaneous regret.

Note that to compute the new instantaneous regret, player A needs to know not only its own payoff $U_A(j,\ell_t)$ but also the payoff $U_A(k,\ell_t)$ of action k which is not its chosen action at time t. To overcome this problem, following [90], we assume that each BS shares network-assisted feedback in term of the number of concurrent users on the BS at each time with its connected users. User A then can use this network feedback to compute the estimated value of $U_A(k,\ell_t)$ as follows [90]

$$\tilde{U}_A(k,\ell_t) = \frac{\sum_{\tau\le t}\big(U_A(k,\ell_\tau)\times n_\tau^k\big)\mathbb{1}_{\{i_\tau=k\}}}{\sum_{\tau\le t}(n_\tau^k+1)\mathbb{1}_{\{i_\tau=k\}}},$$

where n_τ^k is the number of users on the BS k at time τ.

In a dynamic environment of wireless network where the PHY rates of users change with time and thus results in changes in their throughputs (payoffs) from time to time, the regret in the distant past becomes irrelevant. To deal with the dynamic problem, we introduce a forgetting factor in the updating formula of (5.5) as follows

$$\bar{R}_t^A(j,k) = \lambda_A \, \bar{R}_{t-1}^A(j,k) + (1-\lambda_A) \, R_t^A(j,k), \tag{5.6}$$

where $0 \leq \lambda_A \leq 1$ is a forgetting factor introduced to regulate the influence of outdated values of regret with respect to instantaneous regret, which is determined by each player based on its own observation of the environment at each time step. We discuss the derivation of λ_A shortly in Section 5.4.2.

Unlike the common approach of using either a constant or a decreasing step size as used in [108], our approach allows better adaption of the weight parameters of the old and new values of regret to dynamic changes in the environment. In our solution, each players A independently uses and adapts the value of λ_A over time depending on their individual observation of the changes in the environment. For instance, when no new information is observed, i.e., when $\lambda_A = 1$ then equation (5.6) becomes

$$\bar{R}_t^A(j,k) = \bar{R}_{t-1}^A(j,k),$$

thus the cumulative regret only depends on the past regret. On the other hand, if there is an abrupt change occurs in the environment, i.e., when $\lambda_A = 0$ then equation (5.6) becomes

$$\bar{R}_t^A(j,k) = R_t^A(j,k),$$

thus the cumulative regret counts on the instantaneous regret rather than the past regret. It is worth to mention that the previous works using a forgetting factor (such as [108]) did not address the issues of how to choose it, and how the choice of the forgetting factor is related to a specific phenomenon, which are the major factors considered in our study here for user mobility in RAT selection.

5.4.2 Updating the Forgetting Factor

We explain in the following how each user determines the forgetting factor based on its own observation of the environment. In cellular network, after selecting the associated

BS, each user monitors and keeps track of its measured Channel Quality Indicator (CQI) to its associated BS as the measure of the channel condition of the wireless link in the network. Each user A then constructs a probability mass function (pmf) of its CQI using pass observations. Figure 5.1 shows an example CQI distribution of a real-word LTE BS in North America.

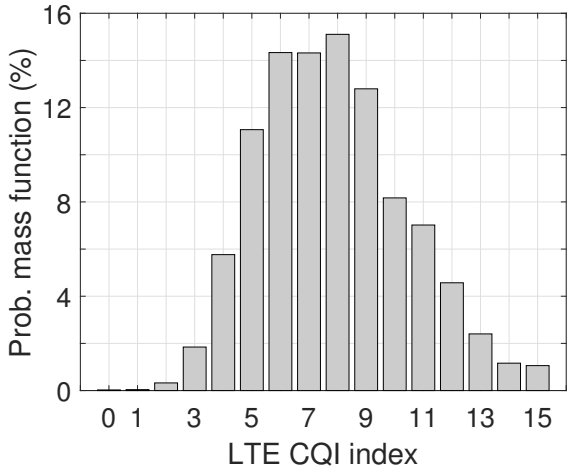

Figure 5.1. Example CQI distribution of real-world data from a Tier-1 LTE operator in North America.

From this CQI distribution, each user A can determine the forgetting factor at each time step as follows

$$\lambda_A(t) = \frac{pmf(CQI_t)}{pmf(CQI_m)}, \tag{5.7}$$

where $pmf(CQI_t)$ and $pmf(CQI_m)$ are the probabilities of the instantaneous CQI value and the median CQI value computed by the user, respectively. If $pmf(CQI_t)$ is close to the value of $pmf(CQI_m)$ then $\lambda_A \approx 1$ indicates that the environment is stable. On the other hand, if CQI_t is very different from the value of CQI_m then $\lambda_A \approx 0$ indicates that the wireless link in the environment changes abruptly.

It is worth to mention that this method can be used in a similar way for WiFi network. User connecting to WiFi BSs can replace the use of CQI by SNR (Signal-to-Noise Ratio) as a measure of the quality of signal of WiFi network in order to compute its forgetting factor at a given time. Also note that, the changing rate of the forgetting factor in the interval $[0, 1]$ is relatively slow over the timescale of the algorithm.

5.4.3 Algorithm and Convergence Analysis

Each user A independently adjusts its selection in response to the changes in the network due to its movement following our proposed procedure as described below.

Algorithm Adaptive Reinforcement Learning With Forgetting Factor (ARLFF)

1: *Initialization:* Generate random uniform probability $p_1(j)$ for all $j \in \mathcal{I}$.
2: **for** $t = 1, 2, \ldots$ **do**
3: *Action Selection:* Select action $i_t = j$ according to the probability distribution $p_t(j)$.
4: *Payoff Observation:* Obtain payoff $U_A(j, \ell_t)$ from the associated base station j.
5: *Feedback Exchange:* Receive feedback n_t^j from the corresponding base station j.
6: *Forgetting Factor Update:* Compute the forgetting factor λ due to mobility situation using (5.7).
7: *Regret Update:* Update the cumulative regret $\bar{R}_t^A(j,k)$ for all $k \neq j \in \mathcal{I}$ using (5.6).
8: *Strategy Update:* Update the probability distribution $p_{t+1}(k)$ using (5.4).
9: **end for**

We use the differential inclusion (DI) framework in [56, 109] for analysing the convergence properties of the proposed algorithm. DI is a generalization of the concept of ordinary differential equations (ODEs) that is particularly suitable to study the asymptotic trajectory of the stochastic approximation algorithms, especially the iterative process in game-theoretic learning. We show that our proposed algorithm leads to a stable system by the two following theorems.

Theorem 5.1. *If player A follows Algorithm 1, its cumulative regret is guaranteed to approach the set of non-positive values (negative orthant) almost surely irrespective of the behaviour of the other players.*

Proof. Please refer to Appendix 5.A. □

Theorem 5.2. *If all players follow Algorithm 1, the empirical distribution of joint play of all players converges almost surely as $t \to \infty$ to the set of correlated equilibria.*

Proof. Please refer to Appendix 5.B. □

Table 5.1. Simulation parameters [87].

Symbol	Description	Value
P_k	Transmission power	40 dBm (macrocell)
		20 dBm (femtocell)
N_o	Noise level	-90 dBm
α	Pathloss exponent	3

5.5 Evaluation

5.5.1 Simulation Setup

In our simulation, we consider a heterogeneous network deployment consisting of 200 users located in the partially overlapping coverage area of 1 marcocell and 4 femtocells. We assume that the marcocell is located at the centre of the network while users and femtocells are randomly and uniformly distributed over the simulation area. Simulation parameters of the networks are set according to [87]. Specifically, the PHY data rate of a user a under BS k at given time t is

$$\Theta_{a,t}^{k} = \log\left(1 + \frac{P_k}{N_o (d_{a,t}^{k})^{\alpha}}\right),$$

where $d_{a,t}^{k}$ is the distance between the user a and the BS k at time t in meter. The meaning and values of other parameters used for our simulation can be found in Table 5.1. The duration of each iteration is set to be one second in all the simulations. We assign unequal user weights $w = (1, 3, 4)$ in the resource schedule policy to a user with equal probabilities of $1/3$.

We compare the performance of the following three RL-based RAT selection algorithms:

- *Reinforcement Leaning with Network-Assisted Feedback (RLNF)* in [90]: In this algorithm, each BS shares network-assisted feedback computed at the network side in term of the number concurrent users to help its associated users in their RAT selection decisions. From the network feedback and its own observations, each users can estimate its obtainable throughput from all other available BSs and compute the

regrets (the changes in average payoff) that it would have if choosing other BSs instead of its current BS. Users then apply the RLNF procedure that assures no regret in the long run to select their BSs.

- *Combined Fully Distributed Payoff and Strategy Reinforcement Learning (CODIPAS)* in [48]: In this algorithm, users learn and select their BSs based solely on their local observation of the throughput received from past experiences, without any additional information from the network. At each iteration, using only this information, each user individually selects the best BS that maximise its own throughput.

- *Our Adaptive Reinforcement Learning with Forgetting Factor (ARLFF)*: In our solution, apart from the network-assisted feedback as used in RLNF scheme, each user adaptively uses a different forgetting factor based on the observation of its PHY rate change due to its movement. Each users then update the cumulative regret at each time step by adding the correction term based on the new observation of the instantaneous regret. Users then follow the regret-based RL principle to select their BSs.

We make comparisons in the following metrics:

- Overall network throughput (Mbps): the sum of per-user throughput of all the users in the network.

- Jain's fairness index, which is defined as

$$J = \frac{(\sum_{a=1}^{N} x_a)^2}{N \times \sum_{a=1}^{N} x_a^2},$$

where x_a is the per-user throughput of user a and N is the total number of users in the network. Notes that the highest value 1 of the Jain's fairness index means all users are obtained the same per-user throughput.

5.5.2 Random Mobility Scenario

We first report our results under the random mobility model [110]. The random mobility scenario is configured as follows. Mobile users are placed at randomly chosen points inside a small square area of 100×100 meters and are assumed not moving during the

first 1000 iterations of the simulation. We then enable the movement of all users in the network for a short duration of 50 iterations and repeat the same simulation at the 3000-th iteration. Users are assumed to move with constant velocities and fixed directions, which are independently random variables uniformly distributed in the range of $[2, 3]$ m/s and $[0, 2\pi]$ radians, respectively. Each time a user reaches the boundary of the simulation area, it immediately continues its movement in a reverse direction with the same speed.

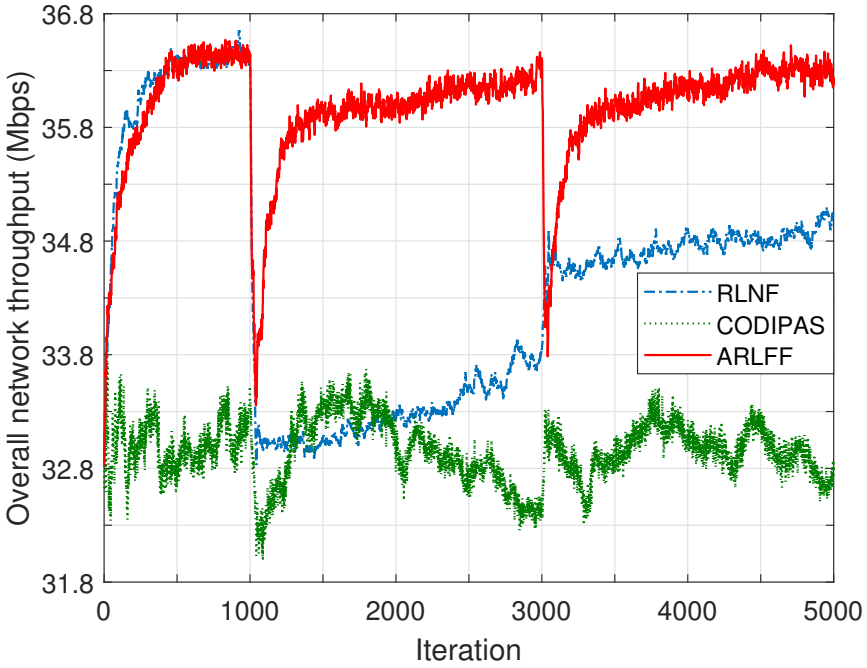

Figure 5.2. Performance comparison of the different algorithms under the random mobility scenarios, in achieving overall network throughput.

Figure 5.2 compares the overall network throughput obtained by the different algorithms. We observe that both RLNF and ARLFF quickly converge to the same performance within 1000 iteration runs. However, after the network changes due to user movement, ARLFF rapidly adapts to the changing environment and maintains good performance while the other algorithms take a long time to re-obtain a good performance. This indicates that, in our solution, the user can sense the change in the network and adjust their strategies accordingly. CODIPAS performs poorest among the three algorithms due to the lack of information on the global network conditions.

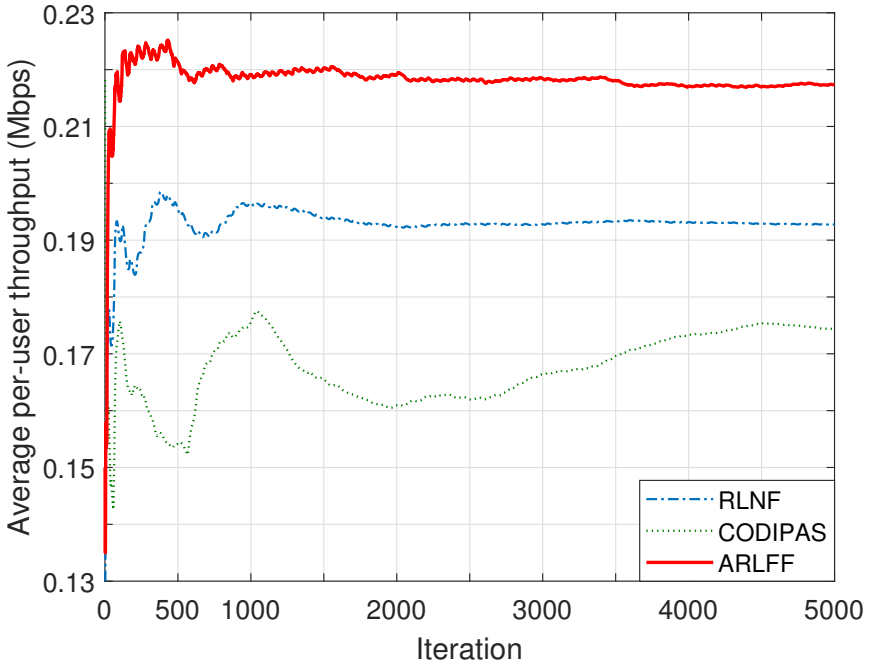

Figure 5.3. Performance comparison of the different algorithms under the random mobility scenarios, in achieving average per-user throughput.

In the simulation of Figure 5.3, only one user is allowed to move among all the users in the network. We randomly select a user and let that user applies our proposed ARLFF. We repeat the same simulation with the two other algorithms. Figure 5.3 shows the throughput performance of the user by applying different algorithms. As can be seen, the proposed algorithm outperforms the other schemes in achieving the average per-user throughput when user moves.

5.5.3 Group Mobility Scenario

We now report our simulation results under the group mobility scenario [107]. The group mobility scenario is configured as follow. The BSs are placed according to a chain topology [86] (treated as the roadside cellular network BSs), where each BS is located 200 meters away from each other. Users are randomly distributed inside a rectangular simulation area of 100×800 meters. Only part of the users can move while the rest of them are assumed to be static. The moving users are assumed to move as a group (same direction)

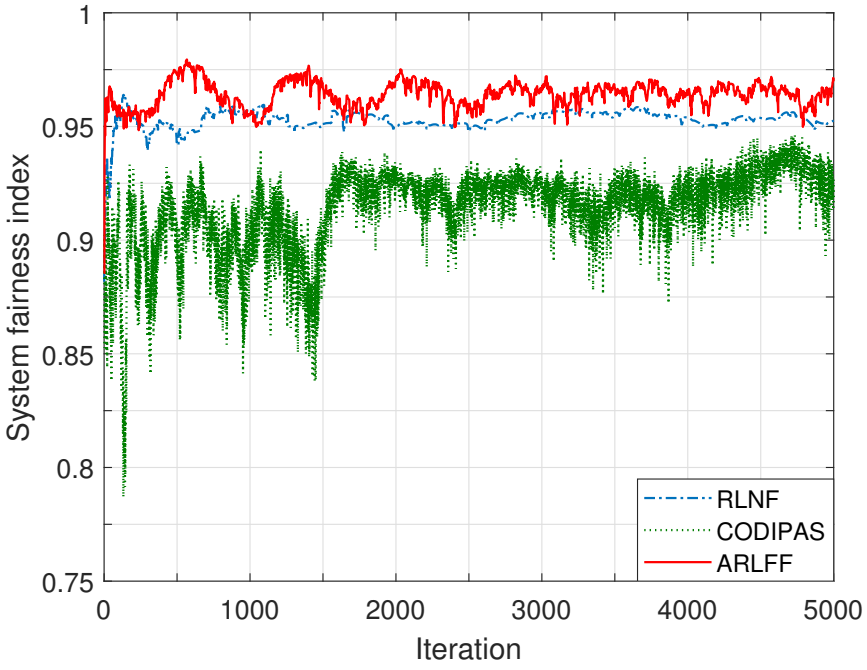

Figure 5.4. Performance comparison of different algorithms under group mobility scenarios in system fairness index (10% moving users).

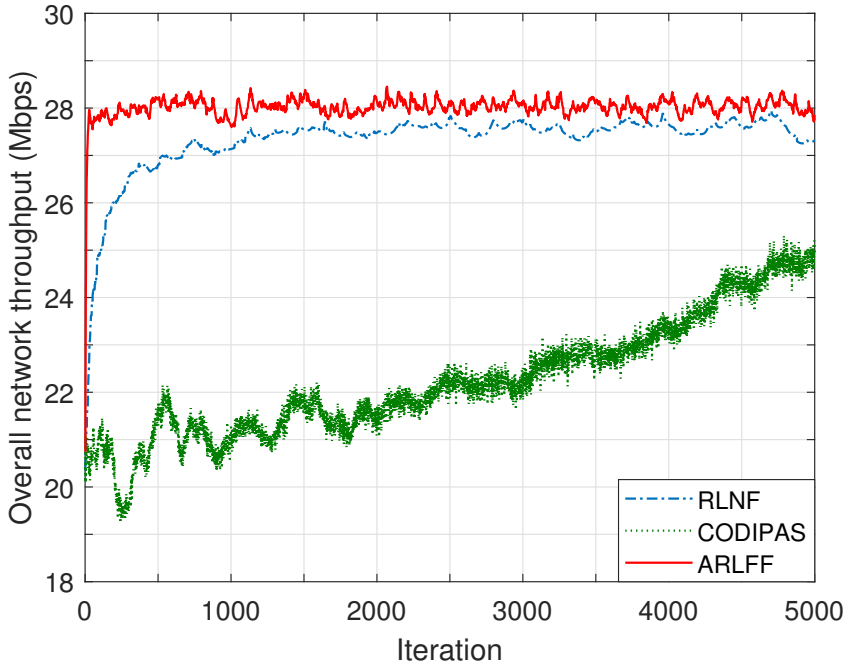

Figure 5.5. Performance comparison of different algorithms under group mobility scenarios in overall network throughput (10% moving users).

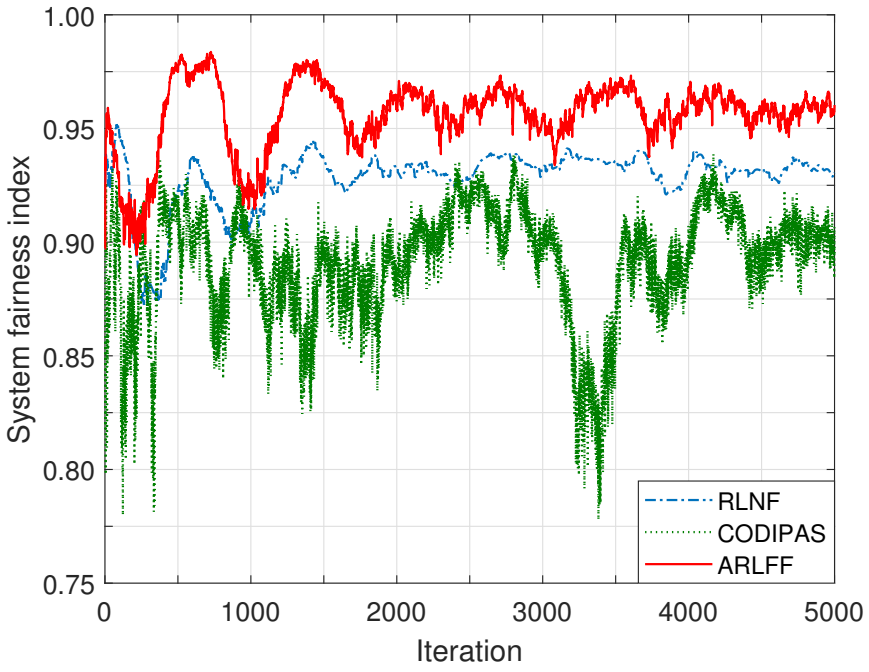

Figure 5.6. Performance comparison of different algorithms under group mobility scenarios in system fairness index (50% moving users).

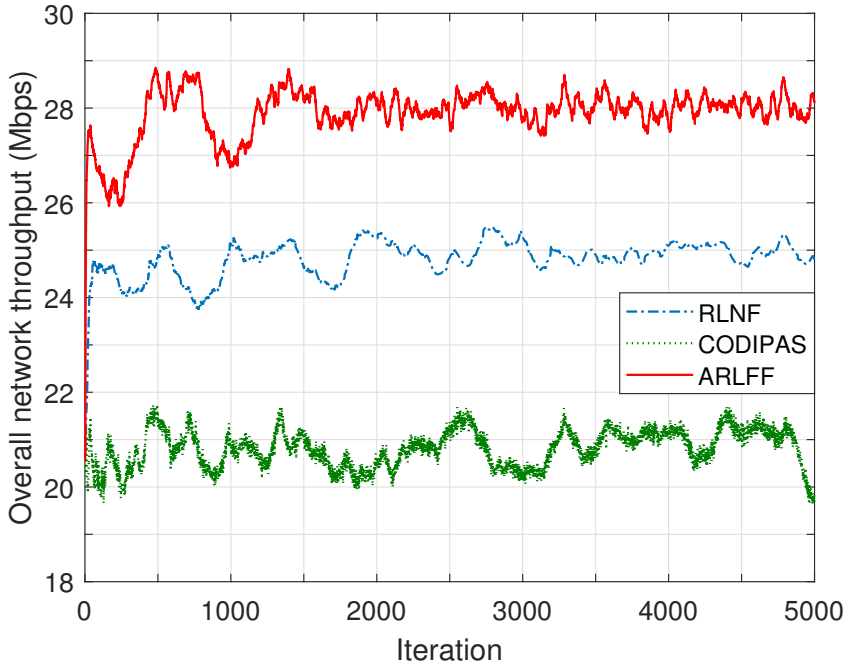

Figure 5.7. Performance comparison of different algorithms under group mobility scenarios in overall network throughput (50% moving users)

with the similar speeds, which are independently random variables uniformly distributed between $[2,3]$ m/s, along the roadside of the simulation area. Again, whenever a user reaches the boundary of the simulation area, we use a reflection techniques to ensure that all users remain within the simulation boundaries.

Figures 5.4 – 5.7 illustrate the system fairness index and overall network throughput performances of the different algorithms when 10% and 50% of the users are moving, respectively. As shown, the proposed algorithm, which is able to adapt to changing networking conditions due to the movement of users, achieves the highest performance in all cases, and is quite robust to the change of the number of moving users in the network. Increasing the total number of moving users results in reducing the performances of the remaining non-adaptive algorithms, especially the CODIPAS scheme.

5.6 Conclusion

In this chapter, we studied the problem of dynamic RAT selection games in wireless networks. We developed a novel fully distributed RAT selection algorithm, called Adaptive Reinforcement Learning with Forgetting Factor (ARLFF), that uses forgetting method to overcome the problem of slow convergence using the conventional RL-based algorithm when users move. Using simulation with realistic network settings, the ARLFF proposal has been tested and compared to relevant RL-based methods for RAT selection such as RLNF and CODIPAS. We demonstrate the superiority of the proposed adaptive learning scheme to the non-adapted solutions while retains the good theoretical convergence properties. Simulation results show the efficiency of our scheme compared to other related algorithms, as well as its ability to adapt to varying network conditions under different mobility scenarios.

For future work, we plan to apply the developed framework to a broader set of mobility models. Investigating the impact of user's speed of movement on the performance of RAT selection algorithm is another challenging problem to consider.

Appendices of Chapter 5

5.A Proof of Theorem 5.1

In the following, we view the game from the point of view of player A and thus we drop the subscript A to keep the notation simple. Define the Lyapunov function

$$P(\bar{R}) = \frac{1}{2}\left(\text{dist}[\bar{R}, \mathbb{R}^-]\right)^2 = \frac{1}{2}\sum_{j,k}\left(|\bar{R}(j,k)|^+\right)^2 , \tag{5.8}$$

where $\mathbb{R}^-$ represent the negative orthant. Taking the time-derivative of (5.8) yields

$$\frac{d}{dt}P(R) = \sum_{j,k}|\bar{R}(j,k)|^+ \times \frac{d}{dt}\bar{R}(j,k) . \tag{5.9}$$

First, we find the $d\bar{R}(j,k)/dt$ by rewriting $\bar{R}(j,k)$ from (5.6) in the form as follows

$$\begin{aligned}\bar{R}_t(j,k) &= \bar{R}_{t-1}(j,k) + (1-\lambda)\big[R_t(j,k) - \bar{R}_{t-1}(j,k)\big] \\ &= \bar{R}_{t-1}(j,k) + (1-\lambda)\big[\big(U(k,\ell_t) - U(j,\ell_t)\big) \times \mathbb{1}_{\{i_t=j\}} - \bar{R}_{t-1}(j,k)\big] .\end{aligned} \tag{5.10}$$

Let $\epsilon = 1-\lambda$, which serves as a constant step size (a small positive number). It can be seen that (5.10) has the form of a constant step size stochastic approximation algorithm $\theta_{k+1} = \theta_k + \epsilon H(\theta_k, x_k)$ and satisfies Theorem 17.1.1 of [111]. Thus, its dynamics can be characterised by an ordinary differential equation (see Chapter 17 of [111] for more details and discussion). This means that the system can be approximated by replacing x_k with its expected value. By applying Theorem 17.1.1 of [111], $\bar{R}_t(j,k)$ converges weakly (in distribution) to the averaged system corresponding to (5.10), thus

$$\begin{aligned}\frac{d}{dt}\bar{R}(j,k) &= \mathbf{E}\Big\{\big(U(k,\ell) - U(j,\ell)\big)\,\mathbb{1}_{\{i=j\}} - \bar{R}(j,k)\Big\} \\ &= p(j)\,\big(U(k,\ell) - U(j,\ell)\big) - \bar{R}(j,k) .\end{aligned} \tag{5.11}$$

Next, replacing $d\bar{R}(j,k)/dt$ from (5.11) into (5.9), we obtain

$$\frac{d}{dt}P(\bar{R}) = \sum_{j,k}|\bar{R}(j,k)|^+ \times \big(U(k,\ell) - U(j,\ell)\big)\,p(j) - \sum_{j,k}|\bar{R}(j,k)|^+ \times \bar{R}(j,k) . \tag{5.12}$$

Substitute the transition probabilities $p(j) = (1-\delta)\chi_j + \delta/m$, where χ denotes a probability vector computed from the regret according to the process in (5.4), in the first term on the r.h.s. of (5.12) yields

$$\begin{aligned}
&\sum_{j,k} |\bar{R}(j,k)|^+ \times \big(U(k,\ell) - U(j,\ell)\big)\ p(j) \\
&= (1-\delta) \sum_{j,k} |\bar{R}(j,k)|^+ \times \big(U(k,\ell) - U(j,\ell)\big)\ \chi(j) + \frac{\delta}{m} \sum_{j,k} |\bar{R}(j,k)|^+ \times \big(U(k,\ell) - U(j,\ell)\big) \\
&= (1-\delta) \sum_{j} U(j,\ell) \times \left(\sum_{k} \chi(k) |\bar{R}(k,j)|^+ - \chi(j) \sum_{k} |\bar{R}(j,k)|^+ \right) \\
&\quad + \frac{\delta}{m} \sum_{j,k} |\bar{R}(j,k)|^+ \times \big(U(k,\ell) - U(j,\ell)\big)\ . \qquad (5.13)
\end{aligned}$$

Suppose χ is such a invariant measure of the transition probabilities (5.4) that for every $j = 1, \ldots, m$ it satisfies that

$$\chi(j) \sum_{k} |\bar{R}(j,k)|^+ = \sum_{k} \chi(k) |\bar{R}(k,j)|^+$$

then the first term in (5.13) is equal to zero. Also, we assume that the payoff function $|U(.)|$ is bounded by G then

$$\sum_{j,k} |\bar{R}(j,k)|^+ \times \big(U(k,\ell) - U(j,\ell)\big)\ p(j) \leq \frac{2G\delta}{m} \sum_{j,k} |\bar{R}(j,k)|^+ \ . \qquad (5.14)$$

Then consider the last term on the r.h.s of (5.12)

$$\begin{aligned}
&\sum_{j,k} |\bar{R}(j,k)|^+ \times \bar{R}(j,k) = \sum_{j,k} \big(|\bar{R}(j,k)|^+\big)^2 = 2P(\bar{R}) \\
&\left(\text{since } \sum_{j,k} \big(|\bar{R}(j,k)|^+\big)^2 = 2P(\bar{R}) \text{ by (5.8)} \right) . \qquad (5.15)
\end{aligned}$$

Combining (5.12), (5.14), and (5.15), we obtain

$$\frac{d}{dt} P(\bar{R}) \leq \frac{2G\delta}{m} \sum_{j,k} |\bar{R}(j,k)|^+ - 2P(\bar{R}).$$

Finally, it follows that assuming $|\bar{R}(j,k)|^+ \geq \kappa > 0$, one can choose $\delta > 0$ small enough such that

$$\frac{d}{dt} P(\bar{R}) \leq -P(\bar{R}) \ .$$

Consequently,

$$P\big(\bar{R}(t)\big) \leq P(\bar{R}(0)) \exp(-t)\,.$$

This implies that $P\big(\bar{R}(t)\big)$ goes to zero at exponential rate. Therefore,

$$\lim_{t\to\infty} \operatorname{dist}\big[\bar{R}(t), \mathbb{R}^-\big] = 0\,.$$

Note that this result holds no matter what the other players do as in this proof we only require existence of such a bound on the payoff of Player A. This completes the proof.

5.B Proof of Theorem 5.2

Let ϕ_t be the global behaviour of the system up to time t, which is defined as the empirical frequency of joint action $s = (j, \ell)$ by all players, where j is the action of player A and ℓ is the actions of the others. ϕ_t can be defined using the stochastic approximation recursion as follows

$$\begin{aligned}\phi_t\big(s_t = (j,\ell)\big) &= \phi_{t-1}\big(s_{t-1} = (j,\ell)\big) + \epsilon\big[\mathbb{1}_{\{s_t=(j,\ell)\}} - \phi_{t-1}\big(s_{t-1} = (j,\ell)\big]\\ &= \epsilon \sum_{\tau\leq t} (1-\epsilon)^{t-\tau}\, \mathbb{1}_{\{s_\tau=(j,\ell)\}}\,,\end{aligned} \tag{5.16}$$

where $\mathbb{1}_{\{s_\tau=(j,\ell)\}}$ denotes the unit vector with the element corresponding to the joint action $s_\tau = (j,\ell)$ being equal to 1.

The result of Theorem 2 is immediate from the definition of the "regret". The elements of the regret matrix in (5.10) can be rewritten using the non-recursive expression as follows

$$\begin{aligned}\bar{R}_t(j,k) &= \epsilon \sum_{\tau\leq t} (1-\epsilon)^{t-\tau}\big(U(k,\ell_\tau) - U(j,\ell_\tau)\big) \times \mathbb{1}_{\{i_\tau=j\}}\\ &= \sum_{\ell\in\mathcal{L}} \epsilon \sum_{\tau\leq t} (1-\epsilon)^{t-\tau}\, \mathbb{1}_{\{i_\tau=j\}}\, y_\ell\big(U(k,\ell) - U(j,\ell)\big)\\ &= \sum_{\ell\in\mathcal{L}} \epsilon \sum_{\tau\leq t} (1-\epsilon)^{t-\tau}\, \mathbb{1}_{\{s_\tau=(j,\ell)\}}\big(U(k,\ell) - U(j,\ell)\big)\\ &= \sum_{\ell\in\mathcal{L}} \phi_t\big(s_t = (j,\ell)\big)\big(U(k,\ell) - U(j,\ell)\big)\,.\end{aligned}$$

In the last line, we substituted $\phi_t\big(s_t = (j,\ell)\big)$ from (5.16). Finally, on any convergent subsequence $\lim_{t\to\infty} \phi_t \to \psi$, we get

$$\lim_{t\to\infty} \bar{R}_t(j,k) = \sum_{\ell\in\mathcal{L}} \psi(j,\ell)\big(U(k,\ell) - U(j,\ell)\big) \leq 0\,.$$

Next, comparing with the definition of CE as in equation (5.2) completes the proof.

Chapter 6

Thesis Conclusion

THIS last chapter concludes the thesis with a summary of thesis contribution and significance, and propose potential directions for further research on the topic.

6.1 Summary

In this dissertation, we have addressed the problem of efficient and intelligent selection of radio access technologies in heterogeneous wireless networks by proposing three novel reinforcement learning frameworks, which have been theoretically proved to reach correlated equilibrium solution concept in game theory. Our key results and contribution can be summarised as follows.

In Chapter 2, we proposed a novel fully distributed reinforcement learning procedure for multi-agent non-cooperative task, which is an important area of research in repeated games. The proposed approach uses both positive and negative regrets to improve the convergence behaviour of the conventional multi-agent regret-based reinforcement learning. The research question is clear defined and the difference between positive and non-positive regret-reinforcement learning is clearly demonstrated. Careful attention has been paid to both the theoretical results and the statistical analysis of the empirical results. The simulation results presented a clear case for the superiority of the proposed algorithm. This solution has been shown to be robust to variations in the total number of learning agents in the system, and is suitable for large-scale distributed multi-agent systems. This contribution has been published in [68].

In Chapter 3, we introduced a general reinforcement learning framework for wireless network selection games where users make decision based on throughput estimation with limited network-assisted feedback from base stations. The original contribution of this study is to incorporate network observed assisted information to improve the estimation process and update the learning policy. Convergence of the proposed scheme to the set of correlated equilibria was theoretically proven. Simulation results demonstrated the outperformance of the proposed approach over other existing schemes in term of convergence speed and communication overhead, while providing competitive fairness and utility for the network users. A further contribution is that our solution guarantees, at a theoretical level, no-regret payoff in the long run for any user adopting it, irrespective the behaviours of the other users, and thus can work in a heterogeneous environment where users are freely to apply any sort of learning strategies. This is an important implementation issues as our solution can be easily implemented in software running on a end-user

device, without any modification of the current mobile network standards. These contributions have been published in [90].

In Chapter 4, we review existing RAT selection methods and different network models that were used to evaluate these solutions. The main contribution of this chapter is to propose a unified benchmark for evaluating different algorithms under the same computational environment. Using this benchmark, we provide a thorough comparative study of the impact of different aspects of the network models on the performance of various RAT selection algorithms. Our study reveals that among all the important network parameters that influence the performance of RAT selection algorithms, the number of base station that a user can connect to has the signification impact. This finding provides some guidelines for the proper designal of RAT selection algorithms for future 5G. Our unified evaluation benchmark can serve as a useful reference for future research in this area. This contribution has been published in [64]. An extension version of this conference paper for a scientific journal has been submitted for publication.

In Chapter 5, we address the problem of using user-centric approach for RAT selection in a dynamic network scenario where users move. The key contribution of this research is to propose a new distributed RAT selection algorithm that can handle mobility of users, whereas previously proposed algorithms on user-centric approach are unable to deal with. Our solution users an adaptable forgetting factor to rapidly react to the various mobility situation of mobile users. The proposed algorithm has mathematically proved its convergence properties. Experimental validation have been conducted to demonstrate the performance improvement of the new adaptive learning scheme over the relevant non-adaptive solutions under random mobility and group mobility scenarios of network users. This contribution here has been written as an article and submitted to a journal.

To conclude, I would like to highlight that although our solutions have been developed and applied for emerging 5G wireless networks, the proposed reinforcement learning frameworks described in this thesis can be used to reach the efficient and fair correlated equilibria in any large-scale decentralised multi-agent system. It is also very important to emphasize that our frameworks guarantees “no-regret” payoff in the long run for any agent running these algorithms, irrespective of the behaviour of other agents, and thus can work efficiently in a distributed heterogeneous environment, where agents may potentially apply a number of different learning strategies to maximise their own interests.

6.2 Potential Future Work

We now describe several possible recommendations for future work.

6.2.1 Satisfaction Equilibrium in Multi-agent Cooperative Games

In this thesis, we model the RAT selection problem as an non-cooperative game where mobile users act selfishly based on their own interest, without the knowledge about other users. It would be interesting to explore the benefit of cooperation versus non-cooperation in this RAT selection problem. In order to model the cooperative interactions of users, cooperative game [112], which studies situations in which players can benefit by working together, can be used. This concept provides an flexible framework for modelling collaboration in multi-agent systems to achieve mutual advantage, but also presents a number of challenges, such as dealing with uncertainty (models of incomplete information) or computing efficient solution concepts for cooperative games.

Moreover, most of existing distributed solutions for RAT selection focus on improving per-user throughput. However, from a practical point of view, the goal of wireless system is to meet users demand. In response to this concern, we can consider the situation when the user only seeks for guaranteeing a certain minimum throughput level rather than for maximising its obtainable throughput. This leads to a game solution concept of satisfaction equilibrium [113]. In a satisfaction equilibrium solution, none of the players has any reason to change their strategies since their demand payoffs are simultaneously satisfied. It is expected that cooperation and more optimal results can be achieved using this satisfaction equilibrium concept. Thus, a distributed adaptive learning procedure that can be used to quickly reach an efficient satisfaction equilibrium of the multi-agent cooperative RAT selection game is recommended for further investigation.

6.2.2 Combining Online and Offline Reinforcement Learning

This thesis focus on online reinforcement learning framework in which the agent interacts with the environment while learning. In an online learning process, the algorithm process rewards, estimate value functions, and output action. Whereas in an offline learning

process, the algorithm learns the model parameters from the training samples. In order to improve learning performance, it is desirable to use data and knowledge from similar cases. Thus, it is expected that combining both the general knowledge accumulated by an offline training with the local knowledge found online can significantly speed up and improve the learning process. Due to the efficient use of collected data, one the approximation obtained via offline training, it can be used to generate decision fast enough for use in real time. Despite a number of recent works [114–118] attempt to combine online reinforcement learing with offline training to benefit from experience replay, thorough understanding the performance and limitation of this approach under dynamic, complex and uncertain environments have not been explored in the literature. Therefore, a reinforcement learning procedure that takes into consideration both advantages of the stability of the offline training and the adaptability of the online learning should be explored further.

6.2.3 Software Defined Wireless Access Network

Future 5G systems are envisioned to have overlapping and coexisting of multiple network architectures and radio access technologies. Thus, it is important to for users to choose smartly which network to connect with in a high-dynamic environment while also considering benefits at network side (i.e., interference and congestion avoidance). A possible approach to address this challenge is to leverage the software defined networking (SDN) architecture for access network selection. In SDN solution, an SDN controller, which has global view of the network resources and traffic loads, takes optimised resource allocation decisions [62]. This enables the balance between improving the network utilisation and keeping the user's quality of service at an acceptable level. We have explored toward this direction in an early development state of our work, published in [119], where we presented our prototype policy defined networking solution for enabling automation network management and control for future wireless access networks.

Appendix A

Differential Inclusions and Approachability

THIS appendix summarises the basic ideas of differential inclusion (DI) framework and its applications to game theory, in particular the convergence analysis of regret-based algorithm in a two player game. DI is a generalization of the concept of ordinary differential equations (ODEs) that is particularly suitable to study the asymptotic trajectory of the stochastic approximation algorithm, especially the iterative process in game-theoretic learning. DI framework are used for the analysis of the convergence properties of the proposed algorithms presented in Chapter 2, Chapter 3, and Chapter 5. *For clarification, the contents presented in this Chapter were produced by Professor Langford White from The University of Adelaide and are not part of the original contributions of this thesis. We include this material as necessary background for the use of differential inclusion framework to prove our Theorems since it is not published material.*

Lyapunov Stability for ODEs

Since the approachability theory for differential inclusions (DIs) is based upon Lyapunov stability ideas, we summarise here the basic ideas of Lyapunov stability as it pertains to ordinary differential equations (ODE).

We consider here only time-invariant, homogeneous systems. Let $f : \mathbb{R}^n \to \mathbb{R}^n$ and consider the ODE

$$\dot{x} = f(x) \,, \tag{A.1}$$

where $x : [0, \infty) \to \mathbb{R}^n$, and $\dot{x}$ is the derived function. The initial condition $x(0) = x_0$ is given. Various conditions can be placed on the mapping f to ensure existence and uniqueness of solutions which we don't go into here. Let's assume that these conditions are such that there is a unique $\mathcal{C}^1$ solution given any x_0. The *equilibrium set* for (A.1) is $\Lambda = \{x \in \mathbb{R}^n : f(x) = 0\}$. It's assumed for now that Λ consists of a denumerable set of points which do not accumulate anywhere. Given any $y \in \Lambda$, we can study the behaviour of solutions to (A.1) in the vicinity of y. It's convenient to "shift" y to the origin by instead considering the behaviour of the solutions to the ODE

$$\dot{x} = f(x - y) = f_y(x) \,,$$

near the origin. So it suffices to consider the behaviour of (A.1) near the origin (by redefining f as need be).

Definition A.1. *The origin of* (A.1) *is said to be a stable attractor from* x_0 *if, given any* $\epsilon > 0$, *there is a* $\delta > 0$ *and* $T > 0$ *such that if* $\|x_0\| < \delta$, *the solution of* (A.1) *satisfies* $\|x(t)\| < \epsilon$ *for all* $t > T$.

Definition A.2. *The origin of* (A.1) *is said to be an asymptotically stable attractor from* x_0 *if there is a* $\delta > 0$ *such that if* $\|x_0\| < \delta$, *the solution of* (A.1) *satisfies* $\lim_{t\to\infty} \|x(t)\| = 0$. *More generally, we could replace the ball* $\|x_0\| < \delta$ *by an open set* $\mathcal{N}$.

The largest open set in the sense of definition A.2 is called the *domain of attraction* for $x = 0$.

Definition A.3. *The origin of* (A.1) *is said to be exponentially stable from* x_0 *if, given any* $\epsilon > 0$, *there is a* $\delta > 0$, $T > 0$ *and constants* $\alpha, \beta > 0$ *such that if* $\|x_0\| < \delta$, *the solution of* (A.1) *satisfies* $\|x(t)\| < \alpha\, e^{-\beta t}$ *for all* $t > T$.

We now come to the Lyapunov approach for guaranteeing stability for an ODE. Firstly, suppose there is an open neighbourhood $\mathcal{N}$ containing the origin, and a $\mathcal{C}^1$ function $Q : \mathbb{R}^n \to \mathbb{R}$ such that (i) $Q(x) \geq 0$ for all $x \in \mathcal{N}$; (ii) $Q(x) = 0 \Leftrightarrow x = 0$; (iii) $\nabla Q(x)^T f(x) \leq 0$ for all $x \in \mathcal{N}$. Then the origin is a stable attractor for (A.1).

If in condition (iii) we have instead that $\nabla Q(x)^T f(x) < 0$ for all $x \in \mathcal{N} \setminus \{0\}$, then the origin is asymptotically stable, and if instead, $\nabla Q(x)^T f(x) \leq -a\, Q(x)$ for all $x \in \mathcal{N} \setminus \{0\}$ and some constant $a > 0$, the origin is exponentially stable.

Comment

In the general case, finding a Lyapunov function is not straightforward, however in many cases, "physical" ideas such as energy can be applied to help select an appropriate Lyapunov function.

Invariant Sets

It is sometimes useful to consider solutions to (A.1) on the complete real line $t \in (-\infty, \infty)$ which pass through a point x_0 at $t = 0$.

Definition A.4. *A set $\Gamma \subseteq \mathbb{R}^n$ is called an invariant set for (A.1) if for each $x_0 \in \Gamma$ there is a solution to (A.1) defined on $\mathbb{R}$ with $x(0) = x_0$ and with $x(t) \in \Gamma$ for all $t \in \mathbb{R}$.*

Sometimes the equilibrium sets are more complicated than isolated points, and we would still like to say something about the limiting behaviour of (A.1). La Salle's invariance theorem provides a tool.

Theorem A.1. *Let $\lambda > 0$ be given, and let Q be a Lyapunov function satisfying $Q(x_0) \leq \lambda$. Then as $t \to \infty$, $x(t)$ converges to [9] the largest invariant set contained in the set $\{x \in \mathbb{R}^n : \dot{Q}(x) = 0, Q(x) \leq Q(x_0)\}$.*

Compact Sets of Functions

One of the fundamental ideas behind the convergence proofs for adaptive algorithms considered here is that of a set of *equicontinuous* functions. We'll consider functions f defined

[9] Convergence to a set S means that $\lim_{t\to\infty} \inf_{s\in S} \|x(t) - s\| = 0$.

on a normed space $(X, \|.\|_X)$ and taking values in a normed space $(Y, \|.\|_Y)$. A set F of such functions $f : X \to Y$ is called *pointwise bounded* if for each $x_0 \in X$, there is a $M > 0$ such that $\|f(x_0)\|_Y \leq M$ for all $f \in F$.

A set F of such functions $f : X \to Y$ is called *equicontinuous* at a point $x_0 \in X$ if given $\epsilon > 0$, there is a $\delta > 0$ such that $\|x - x_0\|_X < \delta \Rightarrow \|f(x) - f(x_0)\|_Y < \epsilon$ for all $f \in F$. The set F is said to be equicontinuous if it is equicontinuous at each $x_0 \in X$. Note that this definition doesn't require *uniform continuity* of each $f \in F$ since the δ might depend on x_0. In the case where it doesn't, the set F is called uniformly equicontinuous. The important result here is:

Theorem A.2. *(Arzela-Ascoli) A set F of functions $f : X \to Y$ for normed spaces X and Y is compact if and only if it is closed, pointwise bounded and equicontinuous.*

As a consequence, any closed, pointwise bounded and equicontinuous sequence $\{f_n\}$ has a convergent subsequence. In addition, $\{f_n\}$ is *uniformly bounded* in that the bounding constant M in the definition of pointwise boundedness can be chosen independently of $x_0 \in X$.

Projected ODEs

In many problems, we know that the trajectories of the ODE must lie in some set H, usually assumed compact. Considering the ODE (A.1), then the projected ODE [10] is given by

$$\dot{x} = f(x) + z, \quad z(t) \in -C(x(t)) \,, \tag{A.2}$$

where $C(x)$ is a set generated by the constraint set H and depends on the form of H. It always holds that $C(x) = \{0\}$ if $x \in H$. In particular, if H is a hyper-rectangle, and $x \in \partial H$, then $C(x)$ is the convex cone generated by all exterior normals at the point x. More general formulations for equality constraints and for smooth manifolds can be developed (see [120]). If (A.1) has a unique solution for each x_0, then so does (A.2) [120]. Rather than

[10] It's important to note that (A.2) is not a differential *inclusion* since the r.h.s. is not set valued, but is a (unique) $\mathbb{R}^n$-valued function (of x).

invariant sets, in the constrained case, it is often better to use limit points. The set of limit points of (A.2) is defined by

$$L_H = \lim_{t\to\infty} \bigcup_{x_0\in H} \{x(s) : s \geq t, x(0) = x_0\} \ . \tag{A.3}$$

Stochastic Approximation Algorithms

In our first consideration of stochastic approximation algorithms, we consider sequences of random variables $X_k, k \geq 0$ defined by the recursion

$$X_{k+1} = \Pi_H \left(X_k + \epsilon_k Y_k\right) \ , \tag{A.4}$$

where

(1.1) H is a bounded convex subset in $\mathbb{R}^n$, and Π_H denotes the projection onto H ;

(1.2) $\epsilon_k \to 0$ is a sequence of positive real numbers satisfying $\sum_k \epsilon_k = \infty$;

(1.3) $\sum_k \epsilon_k^2 < \infty$;

(1.4) It holds that $Y_k = f\left(X_k\right) + \delta M_k + \beta_k$ where f is a continuous measurable function, β_k are random variables and $\mathbf{E}\left\{\delta M_k | Y_0, \ldots, Y_{k-1}, X_0\right\} = 0$;

(1.5) $\sup_k \ \mathbf{E}\left\{\|Y_k\|^2\right\} < \infty$;

(1.6) $\sum_k \epsilon_k \|\beta_k\| < \infty$ with probability 1.

It will be useful to write (A.4) in the form. for $k \geq 0$,

$$X_{k+1} = X_k + \epsilon_k Y_k + \epsilon_k Z_k \ , \tag{A.5}$$

where $-Z_k \in C(X_{k+1})$ the cone generated by the set of exterior normals at $X_{k+1} \in H$. Thus, for $k \geq 0$, and $j \geq 1$

$$X_{k+j} = X_k + \sum_{i=k}^{k+j-1} \epsilon_i \left(Y_i + Z_i\right) \ . \tag{A.6}$$

For $-k \leq j \leq -1$, it holds that

$$X_{k+j} = X_k - \sum_{i=k+j}^{k-1} \epsilon_i \left(Y_i + Z_i\right) \ . \tag{A.7}$$

Now, define the sequence $\{t_k \ : k \geq 0\}$ by

$$\begin{cases} t_0 = 0 \, , \\ t_k = \sum_{i=0}^{k-1} \epsilon_i \, , \end{cases}$$

so that $t_{k+1} = t_k + \epsilon_k$ for $k \geq 0$. Also, for $t \in \mathbb{R}$ define the non-negative integer function $m(t)$ by

$$m(t) = \begin{cases} 0 & t < 0 \\ k : t_k \leq t < t_{k+1} & t \geq 0 \, . \end{cases}$$

We define the interpolated continuous-time process $X^0(t)$ corresponding to $\{X_k\}$ by the piecewise constant (random) function

$$X^0(t) = \begin{cases} X_0 & t < 0 \\ X_k & t_k \leq t < t_{k+1} \, . \end{cases}$$

We'll show that by appropriate construction, the interpolated processes can be shown to satisfy an associated ODE. The shifted process is defined by $X^k(t) = X^0(t + t_k)$ for all $t \in \mathbb{R}$. We also define the functions $Y^k(t), Z^k(t), M^k(t)$ and $B^k(t)$ for integers $k \geq 0$ and real t as follows. Firstly consider $Y^k(t)$. Suppose that $Y_k = 0$ for all $k < 0$. Define $Y^0(t) = 0$ for $t < 0$, and

$$Y^0(t) = \sum_{i=0}^{m(t)-1} \epsilon_i \, Y_i \, ,$$

for $t \geq 0$. Define $Y^k(t) = Y^0(t + t_k) - Y^0(t_k)$ for all $t \in \mathbb{R}$, then, if $t \geq 0$, noting that $m(t_k) = k$ and $m(t_k + t) \geq k$, we have

$$Y^k(t) = \sum_{i=0}^{m(t+t_k)-1} \epsilon_i \, Y_i - \sum_{i=0}^{k-1} \epsilon_i \, Y_i = \sum_{i=k}^{m(t+t_k)-1} \epsilon_i \, Y_i \, ,$$

and for $t < 0$, we have that $m(t_k + t) < k$, so that

$$Y^k(t) = - \sum_{i=m(t+t_k)}^{k-1} \epsilon_i \, Y_i \, .$$

We similarly define $Z^k(t), M^k(t)$ and $B^k(t)$, by replacing Y_k in the above development by resp. $Z_k, \delta M_k$ and β_k.

Consider $t \geq 0$, then $X^k(t) = X_{m(t+t_k)}$, so it follows from (A.6),

$$X^k(t) = X_k + \sum_{i=k}^{m(t_k+t)-1} \epsilon_i \left(Y_i + Z_i\right)$$

where the sum is given the value zero when $m(t+t_k) = k$ (i.e. when $t_k < \epsilon_k$). For $t < 0$, (A.7) yields

$$X^k(t) = X_k - \sum_{i=m(t_k+t)}^{k-1} \epsilon_i \left(Y_i + Z_i\right) \ ,$$

provided $m(t_k + t) \geq 0$. We thus obtain the common formula (for all t),

$$X^k(t) = X_k + Y^k(t) + Z^k(t) \ . \tag{A.8}$$

The ODE associated with (A.5) is (A.2).

We now come to the main convergence theorem (Theorem 2.1 on page 127 of [120]).

Theorem A.3. *Suppose conditions (1.1)-(1.6) hold for the algorithm (A.4). Then there is a set N of probability zero such that for all $\omega \notin N$, the set of functions $\left\{X^k(\omega,.), Z^k(\omega,.) : k \leq \infty\right\}$ is equicontinuous. The limit $(X(\omega,.), Z(\omega,.))$ of any convergent subsequence satisfies (A.2) and $\{X_k(\omega)\}$ converges to some limit set of (A.2) in H. In the unconstrained case, if $\{X_k\}$ is bounded with probability one, then almost surely, the limits $X(\omega,.)$ of convergent subsequences of $\left\{X^k(\omega,.)\right\}$ are trajectories of the ODE (A.1) in some bounded invariant set and $\{X_k(\omega)\}$ converges to this set.*

We give a brief outline of the proof. Essentially, it consists of the following two steps : 1. Prove that the joint sequence $\left(X^k(t), Z^k(t)\right)$ is almost surely *equicontinuous*, closed and bounded in the set of functions $f : \mathbb{R} \to \mathbb{R}^n$ with the "sup norm" defined by

$$\|f\|_\infty = \inf \ \{C \geq 0 : \|f(t)\| \leq C, \ a.e. \ t \in \mathbb{R}\} \ .$$

2. Characterising the limit of any convergent sub-sequence of $\left(X^k(t), Z^k(t)\right)$ using the associated ODE.

Now for some details : Firstly we demonstrate convergence of the martingale $M_k = \sum_{i=0}^{n-1} \epsilon_i \, \delta M_i$. Define $\delta M_k = Y_k - f(X_k) - \beta_k$, then for $t \geq 0$, and each k

$$X^k(t) = X_k + Z^k(t) + M^k(t) + B^k(t) + \sum_{i=k}^{m(t+t_k)-1} \epsilon_i \, f(X_i) \, , \tag{A.9}$$

with appropriate modifications for $t < 0$. Since M_k is a martingale sequence, it can be shown that for any $\mu > 0$,

$$\lim_{m \to \infty} \mathbf{P} \left\{ \sup_{j \geq m} \| M_j - M_m \| \geq \mu \right\} = 0 \, .$$

Thus $M^k(t) \to 0$ a.s. as $k \to \infty$ uniformly in t on any bounded interval of $\mathbb{R}$. Condition (1.6) implies that $B^k(t) \to 0$ similarly.

Now, replacing the sum in (A.9) by an integral plus an associated error term $\rho^k(t)$, we have

$$X^k(t) = X_k + \int_0^t f\left(X^k(s)\right) \, ds + z^k(t) + M^k(t) + B^k(t) + \rho^k(t) \, ,$$

where, writing m for $m(t + t_k)$,

$$\rho^k(t) = \sum_{i=k}^{m-1} \epsilon_i \, f(X_i) - \int_0^t f\left(X^k(s)\right) \, ds \; = - \left(t - t_m\right) \, f\left(X_m\right) \, .$$

To see this, consider

$$\begin{aligned} \int_0^t f\left(X^k(s)\right) \, ds &= \int_{t_k}^{t_k + t} f\left(X^0(s)\right) \, ds \\ &= \sum_{i=k}^{m(t+t_k)-1} \epsilon_i \, f\left(X_i\right) + \left(t - t_m\right) \, f\left(X_m\right) \, . \end{aligned}$$

Note that $\rho^k(t) = 0$ for each $j \geq k$ such that $t = t_j$. Now, $\rho^k(t) \to 0$ uniformly in t as $k \to \infty$, since

$$\| \rho^k(t) \| \leq \epsilon_{m(t+t_k)} \, \| f\left(X_{m(t+t_k)}\right) \| \, ,$$

and $\epsilon_m \to 0$ as $k \to \infty$ uniformly in t, X_k are a.s. bounded and f is continuous.It now remains to prove that $Z^k(t)$ are equicontinuous. This is done in [120] using a contradiction argument based on assuming a "jump" in a subsequence of $\left(X^k(t), Z^k(t)\right)$. It is then

shown that the limit $\left(X^k(t), Z^k(t)\right)$ of any convergent subsequence of $\left(X^k(t), Z^k(t)\right)$ satisfies the ODE $\dot{x} = f(x) + z$, and its limiting behaviour can be studied using Lyapunov functions.

We now consider the case where f is the negative gradient of a $\mathcal{C}^1$ function $F : \mathbb{R}^n \to \mathbb{R}$. The set of limit points is then the set of stationary points which can be written as the union of disjoint compact and connected subsets S_i. Suppose that F is constant on each S_i, then for almost all ω, $\{X_k(\omega)\}$ converges to a unique S_i.

Time Varying Systems

Often we will have the case where $Y_k = f_k(X_k) + \delta M_k + \beta_k$ where the f_k are uniformly continuous (in k). Suppose there is a continuous function f such that for each $X \in H$,

$$\lim_{k\to\infty} \left| \sum_{i=k}^{m(t_k+t)} \epsilon_i \left(f_i(X) - f(X)\right) \right| = 0 \tag{1.8}$$

for each $t > 0$. Thus f plays the role of "time-averaged" f_k. We then have the convergence result :

Theorem A.4. *Suppose we have the algorithm (A.4) and we replace conditions (1.4) and (1.6) with (1.8) together with the assumption that $\beta_k \to 0$ with probability of 1, then the conclusions of theorem A.3 hold.*

Differential Inclusions and Approachability

In the following, we apply the theoretical results on DI framework developed in [56, 109] to the class of regret-based procedures that guarantee that the correlated equilibrium set is approached. Consider the following differential inclusion (DI)

$$\dot{\mathbf{w}} \in N(\mathbf{w}) - \mathbf{w}\,, \tag{A.10}$$

where $\mathbf{w}(t) \in \mathbb{R}^m$, and N is a mapping from $\mathbb{R}^m$ into the class of all subsets of $\mathbb{R}^m$ (called a *correspondence* on $\mathbb{R}^m$) that satisfies the various conditions outlined in Hypothesis 2.1 of [56].

Let C be a given closed, convex subset of $\mathbb{R}^m$, and assume that there is a continuously differentiable non-negative function $Q : \mathbb{R}^m \to \mathbb{R}$ which is identically zero on C. This is Hypothesis 3.1 of [56].

Exponential Convergence

Suppose there is a positive constant B such that for $w \in \mathbb{R}^m \setminus C$, it holds that

$$\langle \nabla Q(w), \theta \rangle \leq B\, Q(w)\,, \tag{A.11}$$

for all $\theta \in N(w)$. Then if $\mathbf{w}(t)$ is a solution to (A.10), it holds that

$$Q\left(\mathbf{w}(t)\right) \leq Q\left(\mathbf{w}(0)\right)\ e^{-Bt}\,,$$

for all $t \geq 0$. Thus from Lyapunov theory, the set C is a global attractor for (A.10). Thus all solutions to (A.10) will *approach* C.

Discrete Stochastic Approximation

A discrete stochastic approximation (DSA) to (A.10) is a sequence of $\mathbb{R}^m$-valued random variables X_n satisfying the difference equation

$$X_{n+1} - X_n \in a_{n+1}\ \left(N(X_n) - X_n + U_{n+1}\right)\,, \tag{A.12}$$

for $n \geq 0$, where (i) $\mathbf{E}\left\{U_{n+1} | X_0, \ldots, X_n\right\} = 0$, (iii) $\sup_n\ \mathbf{E}\left\{\|U_n\|^2\right\} < \infty$, and (ii) the sequence of step sizes satisfies $\sum_n a_n^2 < \infty$.[11] To skip over technical material, these conditions basically say that the sequence X_n converges almost surely to the set of attractors of (A.10). So following the usual kind of "averaging" approach, we associate the DI (A.10) with a specific algorithm (A.12) and then try and prove convergence of the trajectories $\mathbf{w}(t)$ to some set C. Then we can say that (A.12) converges to C in a sense to be described.

Approachability in a Two Player Game

Consider a game with two players. Player one's actions are specified by the set $\mathcal{I} = \{1, \ldots, I\}$, and player two's actions are specified by the set $\mathcal{L} = \{1, \ldots, L\}$. Assume

[11] Other conditions are also given in [56].

vector payoffs $A_{i,\ell} \in \mathbb{R}^m$ when player one chooses i and player two chooses ℓ. Let $h_n = (i_1, \ell_1, i_2, \ell_2, \ldots, i_n, \ell_n)$ denote the action history of the game au until stage n. The average payoff achieved up until stage n is

$$\overline{g}_n = \frac{1}{n} \sum_{t=1}^{n} A_{i_t,\ell_t} \, .$$

Let X, Y denote the set of all probability distributions on $\mathcal{I}, \mathcal{L}$ respectively.

At stage $n+1$, each player chooses a random action from its action space (X or Y), according to a joint probability distribution

$$\mathbf{P}\{i_{n+1} = i, \ell_{n+1} = \ell | h_n\} = \sigma_i(h_n)\, \tau_\ell(h_n) \, ,$$

where $\sigma(h_n) \in X$, and $\tau(h_n) \in Y$. Then the payoff obtained is $g_{n+1} = A_{i_{n+1},\ell_{n+1}}$. There's two important implications for this : (i) at stage $n+1$, each player has access to the common history h_n, and (ii) the actions of each player are statistically independent given the past history. Let ϕ_A be a mapping from X to the class of subsets of $\mathbb{R}^m$ defined by

$$\phi_A(x) = \text{co}\left\{ \sum_{i=1}^{I} x_i \, A_{i,\ell} : l = 1, \ldots, L \right\} \, ,$$

where $\text{co}(S)$ is the set of all convex combinations of elements of the set S. Thus $\phi_A(x)$ is the set of all expected payoffs for player one using action probabilities x, which can be obtained by player one.

Let N be a correspondence on $\mathbb{R}^m$. A function $\tilde{x} : \mathbb{R}^m \to X$ is said to be N-adapted if $\phi_A(\tilde{x}(w)) \subset N(w)$ for all $w \notin C$. Here we think of $\tilde{x}(w)$ as being a probability vector for each value of $w \in \mathbb{R}^m$, so that $\phi_A(\tilde{x}(w))$ is the set of all expected payoffs that player one can obtain when it chooses its actions according to the probability vector $\tilde{x}(w)$. Thus $\tilde{x}$ is N-adapted if $N(w)$ contains all these possible payoffs. Since N is the function occurring in the DI to which we'll refer for convergence, and given we are working here in the space of average (vector) payoffs, the rule for choosing player one's actions needs to be such that all resulting average payoffs are contained in N. This is what N-adapted means.

We then have the general approachability result (Theorem 3.6 of [56]) : Let C be a given closed, convex subset of $\mathbb{R}^m$, and suppose Hypotheses 3.1 and 3.2 hold (i.e. there is a Lyapunov function Q, zero on C, and with the condition (A.11) holding). Let N be a correspondence on $\mathbb{R}^m$ satisfying Hypothesis 2.1, and let $\tilde{x}$ be N-adapted. Then any strategy

σ of player one that satisfies $\sigma(h_n) = \tilde{x}(\overline{g}_n)$ whenever $g_n \notin C$, results in $d(\overline{g}_n, C) \to 0$ as $n \to \infty$ almost surely (with respect to the probability distribution generated by all joint plays).

This firstly says that player one chooses its strategy at stage n solely based on the overall time average (vector) payoff up until stage n which depends on the complete past history of both players. Secondly, the convergence of the average payoffs to C is guaranteed irrespective of the actions of player 2, provided that the action function $\tilde{x}$ is adapted to a specified subset N of $\mathbb{R}^m$. The gradient condition (A.11) needs to hold for all $\theta = \overline{g}_n \in N$ and $w \in \mathbb{R}^m \setminus C$.

The proof of theorem 3.6 is instructive. Firstly, the algebraic identity

$$\overline{g}_{n+1} - \overline{g}_n = \frac{1}{n+1} \left(g_{n+1} - \overline{g}_n \right) , \tag{A.13}$$

holds. This result follows directly from the definition of the average $\overline{g}_n$. Let $\gamma_n = \mathbf{E}\{g_{n+1}|h_n\}$, where the expectation is over the distribution of all previous moves by both players. Then $\gamma_n \in \phi_A(\tilde{x}(\overline{g}_n))$, the set of all possible payoffs that player one can achieve by choosing its action at stage n according to the rule $\tilde{x}$ based on $\overline{g}_n$. Now since $\tilde{x}$ is N-adapted, then it follows that $\gamma_n \in N(\overline{g}_n)$ for any strategy used by player two. Let $U_{n+1} = g_{n+1} - \gamma_n$, then

$$\overline{g}_{n+1} - \overline{g}_n = \frac{1}{n+1} \left((\gamma_n - \overline{g}_n) + U_{n+1} \right) . \tag{A.14}$$

Now, $\mathbf{E}\{U_{n+1}|h_n\} = 0$ by definition of γ_n, so $\gamma_n - \overline{g}_n \in N(\overline{g}_n) - \overline{g}_n$. So $\{\overline{g}_n\}$ is a discrete stochastic approximation to (A.10). By assumption, the trajectories of the DI (A.10) approach C (exponentially) so the average rewards $\overline{g}_n$ will also approach C almost surely.

The Convex Framework

In the above, we didn't use the convexity of C. Now assume C is convex, then we can construct a Lyapunov function as follows. Since C is closed and convex, given any point $w \in \mathbb{R}^m$, there is a unique "closest" point in C to w which defined a (generally non-linear) projection operator $\Pi_C : \mathbb{R}^m \to C$. We choose as the Lyapunov function, the distance $Q(w) = \|w - \Pi_C(w)\|^2$ where $\|.\|$ is the usual Euclidean norm. Lemma 3.7 of [56] shows that Q satisfies Hypothesis 3.1, and the derivative $\nabla Q(w) = 2(w - \Pi_C(w))$.

We now have the approachability result due to Blackwell (called Proposition 3.8 in [56]). Consider the two player game above, and suppose that player one plays the strategy $\sigma(h_n) = \tilde{x}(\overline{g}_n)$, with $\overline{g}_n \notin C$, satisfying

$$\langle \overline{g}_n - \Pi_C(\overline{g}_n), \theta - \Pi_C(\overline{g}_n) \rangle \leq 0 \tag{A.15}$$

for all $\theta \in \phi_A(\tilde{x}(\overline{g}_n))$, then $d(\overline{g}_n, C) \to 0$.

The idea is to make use of the general result Theorem 3.6. We do this by letting

$$N(w) = \operatorname{co}\{A_{i,\ell} : i \in \mathcal{I}, \ell \in \mathcal{L}\} \cap \{\theta \in \mathbb{R}^m : \langle w - \Pi_C(w), \theta - \Pi_C(w) \rangle \leq 0\} \ .$$

It can be shown that N satisfies Hypothesis 2.1. In addition, (A.15) ensures that $\tilde{x}$ is N-adapted. Now, we are given the closed convex set C. The condition (A.15) can be written

$$\begin{aligned}
&\langle \overline{g}_n - \Pi_C(\overline{g}_n), \theta - \Pi_C(\overline{g}_n) \rangle \leq 0 \ \Leftrightarrow \\
&\langle \overline{g}_n - \Pi_C(\overline{g}_n), \theta - \overline{g}_n + \overline{g}_n - \Pi_C(\overline{g}_n) \rangle \leq 0 \ \Leftrightarrow \\
&\langle \overline{g}_n - \Pi_C(\overline{g}_n), \theta - \overline{g}_n \rangle + \langle \overline{g}_n - \Pi_C(\overline{g}_n), \overline{g}_n - \Pi_C(\overline{g}_n) \rangle \leq 0 \ \Leftrightarrow \\
&\langle \overline{g}_n - \Pi_C(\overline{g}_n), \theta - \overline{g}_n \rangle \leq -\|\overline{g}_n - \Pi_C(\overline{g}_n)\|^2 \ \Leftrightarrow \\
&\langle \nabla Q(\overline{g}_n), \theta - \overline{g}_n \rangle \leq -2\, Q(\overline{g}_n) \ ,
\end{aligned}$$

for all $\theta \in \phi_A(\tilde{x}(\overline{g}_n)) \supset N(\overline{g}_n)$. So by the general theorem we get approachability of the time average rewards to C.

References

[1] X. Yan, "Optimization of Vertical Handover Decision Processes for Fourth Generation Heterogeneous Wireless Networks," Ph.D. dissertation, Monash University, Australia, September 2010. [Online]. Available: https://doi.org/10.4225/03/587c033e78198.

[2] A. Gupta and R. K. Jha, "A Survey of 5G Network: Architecture and Emerging Technologies," *IEEE Access*, vol. 3, pp. 1206–1232, Jul 2015.

[3] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What Will 5G Be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, Jun 2014.

[4] "5G Radio Access: Requirements, Concept and Technologies," White Paper, NTT Docomo, July 2014. [Online]. Available: https://www.nttdocomo.co.jp/english/corporate/technology/whitepaper_5g/.

[5] "4G Americas' Recommendations on 5G Requirements and Solutions," White Paper, 4G Americas, October 2014. [Online]. Available: http://www.5gamericas.org/files/2714/1471/2645/4G_ Americas_Recommendations_on_5G_Requirements_ and_Solutions_10_14_2014−FINALx.pdf.

[6] "SK Telecom's View on 5G Vision, Architecture, Technology, and Spectrum," White Paper, SK Telecom, October 2014. [Online]. Available: http://www.sktelecom.com/img/pds/press/SKT_5G%20White%20Paper_V1.0_Eng.pdf.

[7] K. G. Van-Giang Nguyen, Anna Brunstrom and J. Taheri, "5G Mobile Networks: Requirements, Enabling Technologies, and Research Activities," in *A Comprehensive Guide to 5G Security*, T. Kumar, M. Liyanage, I. Ahmad, A. Braeken, and M. Ylianttila, Eds. John Wiley & Sons, 2018, ch. 2, pp. 31–57.

[8] S. H. Chae, J.-P. Hong, and W. Choi, "Optimal Access in OFDMA Multi-RAT Cellular Networks: Can a Single RAT Be Better?" *IEEE Transactions on Wireless Communications*, vol. 15, no. 7, pp. 4778–4789, July 2016.

[9] 3GPP, "3GPP/WLAN RAN Interworking, Release 12," 3rd Generation Partnership Project (3GPP), TS 37.834, January 2014. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/37834.htm.

[10] A. Ahmed, L. M. Boulahia, and D. Gaiti, "Enabling Vertical Handover Decisions in Heterogeneous Wireless Networks: A State-of-the-Art and A Classification," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 776–811, Jun 2014.

[11] Y. He, M. Chen, B. Ge, and M. Guizani, "On WiFi Offloading in Heterogeneous Networks: Various Incentives and Trade-off Strategies," *IEEE Communications Surveys & Tutorials*, pp. 1–1, 2016.

[12] M. Wang, J. Chen, E. Aryafar, and M. Chiang, "A Survey of Client-Controlled HetNets for 5G," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2016.

[13] "Hotspot 2.0 (Release 1) Technical Specification Package v1.0.0 ," White Paper, Wi-Fi Alliance, October 2012. [Online]. Available: https://www.wi-fi.org/file/hotspot-20-release-1-technical-specification-package-v100.

[14] "Integration of Cellular and Wi-Fi Networks," White Paper, 4G Americas, September 2013. [Online]. Available: http://www.5gamericas.org/files/3114/0622/2546/Integration_of_Cellular_and_WiFi_Networks_White_Paper_9.25.13.pdf.

[15] M. E. Helou, M. Ibrahim, S. Lahoud, K. Khawam, D. Mezher, and B. Cousin, "A Network-Assisted Approach for RAT Selection in Heterogeneous Cellular Networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 6, pp. 1055–1067, Jun 2015.

[16] D. Lopez-Perez, I. Guvenc, and X. Chu, "Theoretical Analysis of Handover Failure and Ping-Pong Rates for Heterogeneous Networks," in *Communications (ICC), 2012 IEEE International Conference on*, June 2012, pp. 6774–6779.

[17] F. Shaikh, "Intelligent Proactive Handover and QoS Management using TBVH in Heterogeneous Networks," Ph.D. dissertation, Middlesex University, United Kingdom, January 2010. [Online]. Available: https://www.mdx.ac.uk/_data/assets/pdf_file/0020/49142/fatema_shaikh_final_thesis.pdf.

[18] R. Machado and S. Tekinay, "A Survey of Game-Theoretic Approaches in Wireless Sensor Networks," *Computer Networks*, vol. 52, no. 16, pp. 3047 – 3061, 2008.

[19] B. Wang, Y. Wu, and K. R. Liu, "Game Theory for Cognitive Radio Networks: An Overview," *Computer Networks*, vol. 54, no. 14, pp. 2537 – 2561, 2010.

[20] V. Srivastava, J. Neel, A. MacKenzie, R. Menon, L. Dasilva, J. Hicks, J. Reed, and R. Gilles, "Using Game Theory to Analyze Wireless Ad Hoc Networks," *Communications Surveys Tutorials, IEEE*, vol. 7, no. 4, pp. 46–56, Oct 2005.

[21] D. E. Charilas and A. D. Panagopoulos, "A Survey on Game Theory Applications in Wireless Networks," *Computer Networks*, vol. 54, no. 18, pp. 3421 – 3430, 2010.

[22] R. Trestian, "User-Centric Power-Friendly Quality-based Network Selection Strategy for Heterogeneous Wireless Environments," Ph.D. dissertation, Dublin City University, Republic of Ireland, January 2012. [Online]. Available: http://doras.dcu.ie/16783/.

[23] R. J. Aumann, "Correlated Equilibrium as an Expression of Bayesian Rationality," *Econometrica: Journal of the Econometric Society*, vol. 55, no. 1, pp. 1–18, Jan 1987.

[24] J. Zhou, "Techniques for Quality of Service Improvement in Internetworks," Ph.D. dissertation, The University of Adelaide, Australia, 2014. [Online]. Available: http://hdl.handle.net/2440/85197.

[25] S. Hart and D. Schmeidler, "Existence of Correlated Equilibria," *Mathematics of Operations Research*, vol. 14, no. 1, pp. 18–25, Feb 1989.

[26] K. Mittal, E. M. Belding, and S. Suri, "A Game-theoretic Analysis of Wireless Access Point Selection by Mobile Users," *Computer Communication*, vol. 31, no. 10, pp. 2049–2062, June 2008.

[27] K. Fahimullah and S. Hassan, "Game-Theory Based Wireless Access Point Selection Scheme," in *Silver Jubilee International Multitopic Symposium (SIMTS), 2010 IEEE*, April 2010, pp. 1–6.

[28] K. Zhu, D. Niyato, and P. Wang, "Network Selection in Heterogeneous Wireless Networks: Evolution with Incomplete Information," in *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*, April 2010, pp. 1–6.

[29] M. Cesana, N. Gatti, and I. Malanchini, "Game Theoretic Analysis of Wireless Access Network Selection: Models, Inefficiency Bounds, and Algorithms," in *Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools*, ser. ValueTools '08, 2008, pp. 1–10.

[30] M. Khan, U. Toseef, S. Marx, and C. Goerg, "Game-Theory Based User Centric Network Selection with Media Independent Handover Services and Flow Management," in *Communication Networks and Services Research Conference (CNSR), 2010 Eighth Annual*, May 2010, pp. 248–255.

[31] ——, "Auction Based Interface Selection with Media Independent Handover Services and Flow Management," in *Wireless Conference (EW), 2010 European*, April 2010, pp. 429–436.

[32] M. Khan, F. Sivrikaya, S. Albayrak, and K. Mengaly, "Auction based interface selection in heterogeneous wireless networks," in *Wireless Days (WD), 2009 2nd IFIP*, Dec 2009, pp. 1–6.

[33] D. Charilas, A. Panagopoulos, P. Vlacheas, O. Markaki, and P. Constantinou, "Congestion Avoidance Control through Non-cooperative Games between Customers and Service Providers," in *Mobile Lightweight Wireless Systems*, 2009, vol. 13, pp. 53–62.

[34] H. Pervaiz and J. Bigham, "Game Theoretical Formulation of Network Selection in Competing Wireless Networks: An Analytic Hierarchy Process Model," in *Next Generation Mobile Applications, Services and Technologies, 2009. NGMAST '09. Third International Conference on*, Sept 2009, pp. 292–297.

[35] H. Pervaiz, "A Multi-Criteria Decision Making (MCDM) Network Selection Model Providing Enhanced QoS Differentiation to Customers," in *Multimedia Computing and Information Technology (MCIT), 2010 International Conference on*, March 2010, pp. 49–52.

[36] J. Antoniou and A. Pitsillides, "4G Converged Environment: Modeling Network Selection as a Game," in *Mobile and Wireless Communications Summit, 2007. 16th IST*, July 2007, pp. 1–5.

[37] C.-J. Chang, T.-L. Tsai, and Y.-H. Chen, "Utility and Game-Theory Based Network Selection Scheme in Heterogeneous Wireless Networks," in *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*, April 2009, pp. 1–5.

[38] Z. Han, D. Niyato, W. Saad, T. Basar, and A. Hjorungnes, *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications*. Cambridge Uni. Press, Oct 2011.

[39] R. Trestian, O. Ormond, and G.-M. Muntean, "Reputation-based Network Selection Mechanism using Game Theory," *Physical Communication*, vol. 4, no. 3, pp. 156 – 171, 2011, Recent Advances in Cooperative Communications for Wireless Systems.

[40] S. Hart and A. Mas-Colell, "A Simple Adaptive Procedure Leading to Correlated Equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, Sep 2000.

[41] ——, "A Reinforcement Procedure Leading to Correlated Equilibrium," in *Economics Essays*. Springer Berlin Heidelberg, 2001, pp. 181–200.

[42] E. Aryafar, A. Keshavarz-Haddad, M. Wang, and M. Chiang, "RAT Selection Games in HetNets," in *Proceedings IEEE INFOCOM*, Apr 2013, pp. 998–1006.

[43] E. Monsef, A. Keshavarz-Haddad, E. Aryafar, J. Saniie, and M. Chiang, "Convergence Properties of General Network Selection Games," in *Proceedings IEEE INFOCOM*, Apr 2015, pp. 1445–1453.

[44] A. Keshavarz-Haddad, E. Aryafar, M. Wang, and M. Chiang, "HetNets Selection by Clients: Convergence, Efficiency, and Practicality," *IEEE/ACM Transactions on Networking*, vol. 25, no. 1, pp. 406–419, Feb 2017.

[45] L. Chen, "A Distributed Access Point Selection Algorithm Based on No-Regret Learning for Wireless Access Networks," in *IEEE 71st Vehicular Technology Conference*, May 2010, pp. 1–5.

[46] Z. Du, Q. Wu, P. Yang, Y. Xu, J. Wang, and Y.-D. Yao, "Exploiting User Demand Diversity in Heterogeneous Wireless Networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 8, pp. 4142–4155, 2015.

[47] S. Andreev, M. Gerasimenko, O. Galinina, Y. Koucheryavy, N. Himayat, S.-P. Yeh, and S. Talwar, "Intelligent Access Network Selection in Converged Multi-Radio Heterogeneous Networks," *IEEE Wireless Communications*, vol. 21, no. 6, pp. 86–96, Dec 2014.

[48] H. Tembine, "Fully Distributed Learning for Global Optima," in *Distributed Strategic Learning for Wireless Engineers*. CRC Press, Apr 2012, pp. 317–359.

[49] S. Bhatnagar, H. Prasad, and L. Prashanth, "Reinforcement Learning," in *Stochastic Recursive Algorithms for Optimization*. London: Springer Science Business Media, 2013, pp. 187–220.

[50] T. W. Sandholm and R. H. Crites, "Multiagent Reinforcement Learning in the Iterated Prisoner's Dilemma," *Biosystems*, vol. 37, no. 1-2, pp. 147–166, Jan 1996.

[51] D. Kalathi, V. S. Borkar, and R. Jain, "Blackwell's Approachability in Stackelberg Stochastic Games: A learning Version," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 4467–4472.

[52] M. Bravo and M. Faure, "Reinforcement Learning with Restrictions on the Action Set," *SIAM Journal on Control and Optimization*, vol. 53, no. 1, pp. 287–312, Jan 2015.

[53] H. P. Borowski, J. R. Marden, and J. S. Shamma, "Learning Efficient Correlated Equilibria," in *IEEE 53rd Annual Conference on Decision and Control (CDC)*, Dec 2014, pp. 6836–6841.

[54] M. Bowling, "Convergence and No-Regret in Multiagent Learning," *Advances in neural information processing systems*, vol. 17, pp. 209–216, 2005.

[55] L. Cigler and B. Faltings, "Reaching Correlated Equilibria Through Multi-agent Learning," in *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, 2011, pp. 509–516.

[56] M. Benaïm, J. Hofbauer, and S. Sorin, "Stochastic Approximations and Differential Inclusions, Part II: Applications," *Mathematics of Operations Research*, vol. 31, no. 4, pp. 673–695, Nov 2006.

[57] K. R. Apt and E. Grädel, "A Primer on Strategic Games," in *Lectures in Game Theory for Computer Scientists*. Cambridge University Press, pp. 1–37.

[58] R. Jain, D.-M. Chiu, and W. R. Hawe, *A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System*. Eastern Research Laboratory, Digital Equipment Corporation Hudson, MA, 1984, vol. 38.

[59] Q. Chen, G. Yu, H. Shan, A. Maaref, G. Y. Li, and A. Huang, "Cellular Meets WiFi: Traffic Offloading or Resource Sharing?" *IEEE Transactions on Wireless Communications*, vol. 15, no. 5, pp. 3354–3367, May 2016.

[60] O. B. Karimi, J. Liu, and J. Rexford, "Optimal Collaborative Access Point Association in Wireless Networks," in *Proceedings IEEE INFOCOM*, Apr 2014, pp. 1141–1149.

[61] S. Singh and J. G. Andrews, "Joint Resource Partitioning and Offloading in Heterogeneous Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 2, pp. 888–901, Feb 2014.

[62] V. Sagar, R. Chandramouli, and K. P. Subbalakshmi, "Software Defined Access for HetNets," *IEEE Communications Magazine*, vol. 54, no. 1, pp. 84–89, Jan 2016.

[63] D. Niyato and E. Hossain, "Dynamics of Network Selection in Heterogeneous Wireless Networks: An Evolutionary Game Approach," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 2008–2017, May 2009.

[64] D. D. Nguyen, H. X. Nguyen, and L. B. White, "Performance of Adaptive RAT Selection Algorithms in 5G Heterogeneous Wireless Networks," in *IEEE 26th International Telecommunication Networks and Applications Conference (ITNAC)*, Dec 2016, pp. 70–75.

[65] R. Trestian, O. Ormond, and G.-M. Muntean, "Game Theory-Based Network Selection: Solutions and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 1212–1231, 2012.

[66] C. Daskalakis, R. Frongillo, C. H. Papadimitriou, G. Pierrakos, and G. Valiant, "On Learning Algorithms for Nash Equilibria," in *Algorithmic Game Theory*. Springer, 2010, pp. 114–125.

[67] V. Krishnamurthy, O. Gharehshiran, and M. Hamdi, "Interactive Sensing and Decision Making in Social Networks," *Foundations and Trends® in Signal Processing*, vol. 7, no. 1-2, pp. 1–196, 2014.

[68] D. D. Nguyen, L. B. White, and H. X. Nguyen, "Adaptive Multiagent Reinforcement Learning with Non-positive Regret," in *AI 2016: Advances in Artificial Intelligence: 29th Australasian Joint Conference, Hobart, TAS, Australia, December 5-8, 2016, Proceedings*, B. H. Kang and Q. Bai, Eds. Springer International Publishing, Dec 2016, pp. 29–41.

[69] "IEEE Standard for Architectural Building Blocks Enabling Network-Device Distributed Decision Making for Optimized Radio Resource Usage in Heterogeneous Wireless Access Networks," *IEEE Std 1900.4*, pp. 1–130, Feb 2009. [Online]. Available: http://ieeexplore.ieee.org/servlet/opac?punumber=4798286.

[70] "IEEE Standard for Information Technology-Telecommunications and information exchange between systems-Local and Metropolitan networks-specific requirements-Part II: Wireless LAN Medium Access Control and Physical Layer specifications: Amendment 9: Interworking with External Networks," *IEEE Std 802.11u*, pp. 1–208, Feb 2011. [Online]. Available: http://ieeexplore.ieee.org/servlet/opac?punumber=5721906.

[71] Nokia Siemens Network, "Mobile Broadband with HSPA and LTE - Capacity and Cost Aspects," *White Paper*, 2010. [Online]. Available: http://ec.europa.eu/information_society/newsroom/cf/dae/document.cfm_doc_id=4555.

[72] C. Phillips and E. W. Anderson, "CRAWDAD dataset cu/cu_wart (v. 2011-10-24)," Oct 2011. [Online]. Available: https://crawdad.org/cu/cu_wart/20111024/.

[73] K. Pahlavan and P. Krishnamurthy, "Wireless LANs," in *Principles of Wireless Access and Localization*. John Wiley & Sons, Sep 2013, pp. 357–404.

[74] Y. Lin, W. Bao, W. Yu, and B. Liang, "Optimizing User Association and Spectrum Allocation in HetNets: A Utility Perspective," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 6, pp. 1025–1039, Jun 2015.

[75] D. Bethanabhotla, O. Y. Bursalioglu, H. C. Papadopoulos, and G. Caire, "Optimal User-Cell Association for Massive MIMO Wireless Networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 3, pp. 1835–1850, Mar 2016.

[76] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. G. Andrews, "User Association for Load Balancing in Heterogeneous Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2706–2716, Jun 2013.

[77] S. Singh, H. S. Dhillon, and J. G. Andrews, "Offloading in Heterogeneous Networks: Modeling, Analysis, and Design Insights," *IEEE Transactions on Wireless Communications*, vol. 12, no. 5, pp. 2484–2497, May 2013.

[78] W. Li, S. Wang, Y. Cui, X. Cheng, R. Xin, M. A. Al-Rodhaan, and A. Al-Dhelaan, "AP Association for Proportional Fairness in Multirate WLANs," *IEEE/ACM Transactions on Networking*, vol. 22, no. 1, pp. 191–202, Feb 2014.

[79] M. Amer, A. Busson, and I. Guérin Lassous, "Association Optimization in Wi-Fi Networks: Use of an Access-based Fairness," in *Proc. ACM MSWiM*, Nov 2016, pp. 119–126.

[80] T. Han and N. Ansari, "A Traffic Load Balancing Framework for Software-Defined Radio Access Networks Powered by Hybrid Energy Sources," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 1038–1051, Apr 2016.

[81] S.-N. Yang, S.-W. Ho, Y.-B. Lin, and C.-H. Gan, "A Multi-RAT Bandwidth Aggregation Mechanism with Software-defined Networking," *Journal of Network and Computer Applications*, vol. 61, pp. 189–198, Feb 2016.

[82] G. Dandachi, S. Elayoubi, T. Chahed, and N. Chendeb, "Network Centric versus User Centric Multi-homing Strategies in LTE/WiFi Networks," *IEEE Transactions on Vehicular Technology*, pp. 1–1, 2016.

[83] K. Zhu, D. Niyato, and P. Wang, "Network Selection in Heterogeneous Wireless Networks: Evolution with Incomplete Information," in *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*. IEEE, 2010, pp. 1–6.

[84] P. Naghavi, S. H. Rastegar, V. Shah-Mansouri, and H. Kebriaei, "Learning RAT Selection Game in 5G Heterogeneous Networks," *IEEE Wireless Communications Letters*, vol. 5, no. 1, pp. 52–55, Feb 2016.

[85] B. Kauffmann, F. Baccelli, A. Chaintreau, V. Mhatre, K. Papagiannaki, and C. Diot, "Measurement-Based Self Organization of Interfering 802.11 Wireless Access Networks," in *INFOCOM, 2007 Proceedings IEEE*, pp. 1451–1459.

[86] Z. Du, Q. Wu, P. Yang, Y. Xu, and Y.-D. Yao, "User-Demand-Aware Wireless Network Selection: A Localized Cooperation Approach," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4492–4507, Nov 2014.

[87] W. Wang, X. Wu, L. Xie, and S. Lu, "Femto-matching: Efficient Traffic Offloading in Heterogeneous Cellular Networks," in *Proceedings IEEE INFOCOM*, Apr 2015, pp. 325–333.

[88] W. Saad, Z. Han, R. Zheng, M. Debbah, and H. V. Poor, "A College Admissions Game for Uplink User Association in Wireless Small Cell Networks," in *INFOCOM, 2014 Proceedings IEEE*, pp. 1096–1104.

[89] B. H. Jung, N.-O. Song, and D. K. Sung, "A Network-Assisted User-Centric WiFi-Offloading Model for Maximizing Per-User Throughput in a Heterogeneous Network," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 4, pp. 1940–1945, May 2014.

[90] D. D. Nguyen, H. X. Nguyen, and L. B. White, "Reinforcement Learning With Network-Assisted Feedback for Heterogeneous RAT Selection," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 6062–6076, Sept 2017.

[91] M. Wang, A. Dutta, S. Buccapatnam, and M. Chiang, "Regret-Minimizing Exploration in HetNets with mmWave," in *2016 13th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, Jun 2016.

[92] J. Choi, W.-H. Lee, Y.-H. Kim, J.-H. Lee, and S.-C. Kim, "Throughput Estimation Based Distributed Base Station Selection in Heterogeneous Networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 6137–6149, Nov 2015.

[93] O. Galinina, A. Pyattaev, S. Andreev, M. Dohler, and Y. Koucheryavy, "5G Multi-RAT LTE-WiFi Ultra-Dense Small Cells: Performance Dynamics, Architecture, and Trends," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 6, pp. 1224–1240, Jun 2015.

[94] X. Ge, S. Tu, G. Mao, C.-X. Wang, and T. Han, "5G Ultra-Dense Cellular Networks," *IEEE Wireless Communications*, vol. 23, no. 1, pp. 72–79, 2016.

[95] M. Bennis, M. Simsek, A. Czylwik, W. Saad, S. Valentin, and M. Debbah, "When Cellular Meets WiFi in Wireless Small Cell Networks," *IEEE Communications Magazine*, vol. 51, no. 6, pp. 44–50, June 2013.

[96] M. Haenggi, J. Andrews, F. Baccelli, O. Dousse, and M. Franceschetti, "Stochastic Geometry and Random Graphs for the Analysis and Design of Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 7, pp. 1029–1046, Sep 2009.

[97] M. H. Cheung, F. Hou, J. Huang, and R. Southwell, "Congestion-Aware DNS for Integrated Cellular and Wi-Fi Networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1269–1281, June 2017.

[98] Q. Wu, Z. Du, P. Yang, Y. D. Yao, and J. Wang, "Traffic-Aware Online Network Selection in Heterogeneous Wireless Networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 381–397, Jan 2016.

[99] X. Li, R. Cao, and J. Hao, "An Adaptive Learning Based Network Selection Approach for 5G Dynamic Environments," *Entropy*, vol. 20, no. 4, p. 236, 2018.

[100] N. Abbas, T. Bonald, and B. Sayrac, "How Mobility Impacts the Performance of Inter-Cell Coordination in Cellular Data Networks," in *2015 IEEE Global Communications Conference (GLOBECOM)*, Dec 2015, pp. 1–6.

[101] F. Giust, C. J. Bernardos, and A. de la Oliva, "Analytic Evaluation and Experimental Validation of a Network-Based IPv6 Distributed Mobility Management Solution," *IEEE Transactions on Mobile Computing*, vol. 13, no. 11, pp. 2484–2497, Nov 2014.

[102] L. Chen and D. B. Hoang, "Addressing Data and User Mobility Challenges in the Cloud," in *Cloud Computing (CLOUD), 2013 IEEE Sixth International Conference on*. IEEE, 2013, pp. 549–556.

[103] H. Zhang, X. Chu, W. Guo, and S. Wang, "Coexistence of Wi-Fi and Heterogeneous Small Cell Networks Sharing Unlicensed Spectrum," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 158–164, March 2015.

[104] F. Giust, L. Cominardi, and C. J. Bernardos, "Distributed Mobility Management for Future 5G Networks: Overview and Analysis of Existing Approaches," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 142–149, January 2015.

[105] A. J. Nicholson and B. D. Noble, "Breadcrumbs: Forecasting Mobile Connectivity," in *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking*. ACM, 2008, pp. 46–57.

[106] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting Mobile 3G Using WiFi," in *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*. ACM, 2010, pp. 209–222.

[107] M. Zhao and W. Wang, "A Unified Mobility Model for Analysis and Simulation of Mobile Wireless Networks," *Wireless Networks*, vol. 15, no. 3, pp. 365–389, 2009.

[108] O. N. Gharehshiran, V. Krishnamurthy, and G. Yin, "Distributed Tracking of Correlated Equilibria in Regime Switching Noncooperative Games," *IEEE Transactions on Automatic Control*, vol. 58, no. 10, pp. 2435–2450, 2013.

[109] M. Benaïm, J. Hofbauer, and S. Sorin, "Stochastic Approximations and Differential Inclusions," *SIAM Journal on Control and Optimization*, vol. 44, no. 1, pp. 328–348, 2005.

[110] C. Bettstetter, "Smooth is Better than Sharp: A Random Mobility Model for Simulation of Wireless Networks," in *Proceedings of the 4th ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. ACM, 2001, pp. 19–27.

[111] V. Krishnamurthy, *Partially Observed Markov Decision Processes From Filtering to Controlled Sensing*. Cambridge University Press, 2016.

[112] G. Chalkiadakis, E. Elkind, and M. Wooldridge, "Cooperative Game Theory: Basic Concepts and Computational Challenges," *IEEE Intelligent Systems*, vol. 27, no. 3, pp. 86–90, May 2012.

[113] S. Ross and B. Chaib-draa, "Satisfaction Equilibrium: Achieving Cooperation in Incomplete Information Games," in *Advances in Artificial Intelligence*, L. Lamontagne and M. Marchand, Eds. Springer Berlin Heidelberg, 2006, pp. 61–72.

[114] S. Gelly and D. Silver, "Combining Online and Offline Knowledge in UCT," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 273–280.

[115] J. Laumonier, "Reinforcement Using Supervised Learning for Policy Generalization," in *Proceedings of The National Conference on Artificial Intelligence*, vol. 22, no. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007, p. 1882.

[116] S. Dini and M. Serrano, "Combining Q-Learning with Artificial Neural Networks in an Adaptive Light Seeking Robot," Swarthmore College, 2012. [Online]. Available: https://www.cs.swarthmore.edu/meeden/cs81/s12/papers/MarkStevePaper.pdf.

[117] S. Lange, T. Gabel, and M. Riedmiller, "Batch reinforcement learning," in *Reinforcement learning*. Springer, 2012, pp. 45–73.

[118] T. Nishi, P. Doshi, M. R. James, and D. Prokhorov, "Actor-Critic for Linearly-Solvable Continuous MDP with Partially Known Dynamics," *arXiv preprint arXiv:1706.01077*, 2017.

[119] H. X. Nguyen, T. Pham, K. Hoang, D. D. Nguyen, and E. Parsonage, "A Prototype of Policy Defined Wireless Access Networks," in *2016 26th International Telecommunication Networks and Applications Conference (ITNAC)*, Dec 2016, pp. 101–106.

[120] H. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. Springer Science & Business Media, 2003, vol. 35.

Biography

Duong Duc Nguyen was born in Danang, Vietnam, in 1986. He received his B.Sc. degree (first class Hons) in electronic communication systems from University of Plymouth and his M.Sc. degree (merit) in mobile and personal communications from King's College London, United Kingdom, in 2008 and 2009, respectively. He started his Ph.D degree at the School of Electrical and Electronic Engineering, University of Adelaide in 2014. His research interests include machine learning and signal processing techniques for emerging 5G wireless networks, in particular the application of adaptive reinforcement learning to resource allocation.

Before coming to Australia, Mr. Duong Duc Nguyen received the Third Prize in the Vietnamese National Physics Competition for Senior High school students in 2004 and won a government scholarship from Danang City of Vietnam for study abroad in the United Kingdom. In September 2014, he received the Beacon of Enlightenment Scholarship to undertake his Doctor of Philosophy degree in Adelaide. During his postgraduate study, he received the Travel Grant of the 2016 ITNAC Conference. He has served as a reviewer for the *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, the *IEEE Communications Magazine* and the *REV Journal on Electronics and Communications*.

Duong Duc Nguyen
duong.nguyen@adelaide.edu.au