Self-Organized Coverage and Capacity Optimization for Cellular Mobile Networks

Muhammad Naseer-ul-Islam
Integrated Communication Systems
Ilmenau University of Technology

A thesis submitted for the degree of
Doktoringenieur (Dr.-Ing.)

urn:nbn:de:gbv:ilm1-2013000341

Referees:
Prof. Dr.-Ing. habil. Andreas Mitschele-Thiel, Ilmenau University of Technology
Prof. Olav Tirkkonen, Aalto University
Dr. Edgar Kuehn, Bell Labs Alcatel-Lucent

Submitted on: December 12, 2012
Defended on: April 17, 2013
Ilmenau, Germany

Abstract

The challenging task of cellular network planning and optimization will become more and more complex because of the expected heterogeneity and enormous number of cells required to meet the traffic demands of coming years. Moreover, the spatio-temporal variations in the traffic patterns of cellular networks require their coverage and capacity to be adapted dynamically. The current network planning and optimization procedures are highly manual, which makes them very time consuming and resource inefficient. For these reasons, there is a strong interest in industry and academics alike to enhance the degree of automation in network management.

Especially, the idea of Self-Organization (SO) is seen as the key to simplified and efficient cellular network management by automating most of the current manual procedures. In this thesis, we study the self-organized coverage and capacity optimization of cellular mobile networks using antenna tilt adaptations. Although, this problem is widely studied in literature but most of the present work focus on heuristic algorithms for network planning tool automation. In our study we want to minimize this reliance on these centralized tools and empower the network elements for their own optimization. This way we can avoid the single point of failure and scalability issues in the emerging heterogeneous and densely deployed networks.

In this thesis, we focus on Fuzzy Q-Learning (FQL), a machine learning technique that provides a simple learning mechanism and an effective abstraction level for continuous domain variables. We model the coverage-capacity optimization as a multi-agent learning problem where each cell is trying to learn its optimal action policy i.e. the antenna tilt adjustments. The network dynamics and the behavior of multiple learning agents makes it a highly interesting problem. We look into different aspects of this problem like the effect of selfish learning vs cooperative learning, distributed vs centralized learning as well as the effect of simultaneous parallel antenna tilt adaptations by multiple agents and its effect on the learning efficiency.

We evaluate the performance of the proposed learning schemes using a system level LTE simulator. We test our schemes in regular hexagonal cell deployment as well as in irregular cell deployment. We also compare our results to a relevant learning scheme from literature. The results show that the proposed learning schemes can effectively respond to the network and environmental dynamics in an autonomous way. The cells can quickly respond to the cell outages and deployments and can re-adjust their antenna tilts to improve the overall network performance. Additionally the proposed learning schemes can achieve up to 30 percent better performance than the available scheme from literature and these gains increases with the increasing network size.

Zusammenfassung

Die zur Erfüllung der zu erwartenden Steigerungen übertragener Datenmengen notwendige größere Heterogenität und steigende Anzahl von Zellen werden in der Zukunft zu einer deutlich höheren Komplexität bei Planung und Optimierung von Funknetzen führen. Zusätzlich erfordern räumliche und zeitliche Änderungen der Lastverteilung eine dynamische Anpassung von Funkabdeckung und -kapazität (Coverage-Capacity-Optimization, CCO). Aktuelle Planungs- und Optimierungsverfahren sind hochgradig von menschlichem Einfluss abhängig, was sie zeitaufwändig und teuer macht. Aus diesen Grnden treffen Ansätze zur besseren Automatisierung des Netzwerkmanagements sowohl in der Industrie, als auch der Forschung auf groes Interesse.

Selbstorganisationstechniken (SO) haben das Potential, viele der aktuell durch Menschen gesteuerten Abläufe zu automatisieren. Ihnen wird daher eine zentrale Rolle bei der Realisierung eines einfachen und effizienten Netzwerkmanagements zugeschrieben. Die vorliegende Arbeit befasst sich mit selbstorganisierter Optimierung von Abdeckung und Übertragungskapazität in Funkzellennetzwerken. Der Parameter der Wahl hierfür ist die Antennenneigung. Die zahlreichen vorhandenen Ansätze hierfür befassen sich mit dem Einsatz heuristischer Algorithmen in der Netzwerkplanung. Im Gegensatz dazu betrachtet diese Arbeit den verteilten Einsatz entsprechender Optimierungsverfahren in den betreffenden Netzwerkknoten. Durch diesen Ansatz können zentrale Fehlerquellen (Single Point of Failure) und Skalierbarkeitsprobleme in den kommenden heterogenen Netzwerken mit hoher Knotendichte vermieden werden.

Diese Arbeit stellt einen "Fuzzy Q-Learning (FQL)"-basierten Ansatz vor, ein einfaches Maschinenlernverfahren mit einer effektiven Abstraktion kontinuierlicher Eingabeparameter. Das CCO-Problem wird als Multi-Agenten-Lernproblem modelliert, in dem jede Zelle versucht, ihre optimale Handlungsstrategie (d.h. die optimale Anpassung der Antennenneigung) zu lernen. Die entstehende Dynamik der Interaktion mehrerer Agenten macht die Fragestellung interes-

sant. Die Arbeit betrachtet verschiedene Aspekte des Problems, wie beispielsweise den Unterschied zwischen egoistischen und kooperativen Lernverfahren, verteiltem und zentralisiertem Lernen, sowie die Auswirkungen einer gleichzeitigen Modifikation der Antennenneigung auf verschiedenen Knoten und deren Effekt auf die Lerneffizienz.

Die Leistungsfähigkeit der betrachteten Verfahren wird mittels eine LTE-Systemsimulators evaluiert. Dabei werden sowohl gleichmäßig verteilte Zellen, als auch Zellen ungleicher Größe betrachtet. Die entwickelten Ansätze werden mit bekannten Lösungen aus der Literatur verglichen. Die Ergebnisse zeigen, dass die vorgeschlagenen Lösungen effektiv auf Änderungen im Netzwerk und der Umgebung reagieren können. Zellen stellen sich selbsttätig schnell auf Ausfälle und Inbetriebnahmen benachbarter Systeme ein und passen ihre Antennenneigung geeignet an um die Gesamtleistung des Netzes zu verbessern. Die vorgestellten Lernverfahren erreichen eine bis zu 30 Prozent verbesserte Leistung als bereits bekannte Ansätze. Die Verbesserungen steigen mit der Netzwerkgröße.

Acknowledgements

The success of this work was not possible without the support of the following people. First to thank, my supervisor Prof. Andreas Mitschele-Thiel. He provided me with an excellent environment where I was never short of the guidance as well as the freedom to finish this research project. Second, I want to thank Prof. Olav Trikkonen and Dr. Edgar Kuehn for acting as the reviewers for this thesis. Third, I would like to thank Siegfried Klein from Alcatel-Lucent Bell Labs, Germany for his support with the simulator libraries. Fourth, my colleagues Nauman Zia and Stephen Mwanje with whom I also shared my office, I thank them for many hours of interesting discussions both technical and non-technical. I would also like to thank my colleagues at the Integrated Communication Systems group for their critical feedbacks during our research seminars. Finally, a big thanks to my wife Quratulann and my family for their patience and support throughout this time.

Contents

List of Figures

List of Tables

Nomenclature

Greek Symbols

α	Degree of Truth of Fuzzy Rule
β	Learning Rate
γ	Discount Factor
μ	Degree of Membership
π	Action Policy
ρ	Reward Function

Other Symbols

A	Set of All Possible Actions
a	Aggregated Action of All Activated Fuzzy Rules
AT_c	Antenna Tilt of Cell c
$E\{\ldots\}$	Expected Value
L	Fuzzy Label
q	Q-value of fuzzy rule and action pair
o	Action of Fuzzy Rule
P	State transition probability
r	Reward
s_c	Continuous Domain State of Cell c
SE_c^{center}	Center Spectral Efficiency of Cell c

SE_c^{edge} Edge Spectral Efficiency of Cell c

SQ State Quality

SQ_{avg} State Quality Average of All Cells

Q Q-value of continuous domain state-action pair

V Value Function

X Set of All Possible States

Acronyms

3GPP 3rd Generation Partnership Project

4G 4th Generation

BS Base Station

CAEDT Continuously Adjustable Electrical Downtilt

CAPEX Capital Expenditure

CCO Coverage and Capacity Optimization

CDMA Code Division Multiple Access

CPICH Common Pilot Channel

DL Downlink

E3 End-to-End Efficiency

eNB Evolved Node-B

FQL Fuzzy Q-Learning

GA Genetic Algorithms

GA Genetic Algorithms

GSM Global System for Mobile Communications

GT Game Theory

HSDPA High Speed Downlink Packet Access

HSUPA High Speed Uplink Packet Access

IRC Interference Rejection Combining

LS Local Search

LTE-Advanced Long Term Evolution Advanced

LTE Long Term Evolution

LTE Long Term Evolution

MARL Multi-Agent Reinforcement Learning

MDP Markov Decision Process

ML Machine Learning

NGMN Next Generation Partnership Project

OPEX Operational Expenditure

QL Q-Learning

QoS Quality of Service

RAT Radio Access Technology

RET Remote Electrical Tilt

RL Reinforcement Learning

SARL Single-Agent Reinforcement Learning

SA Simulated Annealing

SA Simulated Annealing

SL Supervised Learning

SOCRATES Self-Optimization and self-ConfiguRATion in wirelEss networkS

SON Self-Organizing Networks

SO Self-Organization

TS Tabu Search

UL Unsupervised Learning

UL Uplink

WCDMA Wideband Code Division Multiple Access

1 Introduction

Cellular mobile networks typically experience spatio-temporal variations in their service demands because of users' mobility and usage behavior. Therefore, traditionally these networks are designed based on average peak demand values. This allows the networks to provide reliable communication means in most of the scenarios while ensuring that the capacity upgrades and network optimizations are required only at relatively large time scales.

However, this kind of network design paradigm is highly expensive and non-optimal. Meeting these peak demand values means densification of *Base Station (BS)* deployment as more radio resources are needed, which increases the network operators' *Capital Expenditure (CAPEX)* significantly. In addition, the operation, maintenance and, optimization of these large number of BS increases the *Operational Expenditure (OPEX)*. Moreover, as the networks seldom experience these peak demands, the networks usually remain underutilized.

In addition to the financial drawbacks of the current network design and operation paradigm, complexity is also becoming an increasingly important factor for the network operators. The deployment, operation and, maintenance of cellular mobile networks is becoming increasingly complex due to a number of factors.

First, the widespread usage of mobile Internet has lead to the development of a number of *Radio Access Technologies (RATs)*. In order to remain competitive, operators have to migrate to these modern RATs. But instead of complete swap of older deployed RAT, operators normally integrate the newer RAT in the existing infrastructure. This increases network heterogeneity and complicates their management.

Second, for improved spectral efficiency and thus greater network capacity with the limited spectrum, newer RATs offer complicated algorithms for radio layer, which increases the number of operator tunable parameters enormously.

Third, to satisfy the ever increasing demand of modern data applications of indoor users, new cell deployment concepts like femtocell and home eNodeB are proposed. With the deployment of these cells, the number of network elements in an operational network does increase tremendously. Moreover, these cells are also expected to give some operational control to the users as well, which further complicates the management tasks for the operator.

Fourth, instead of some network wide optimization, more and more optimizations are now done at the cell individual level in order to better exploit the cell-specific characteristics.

To overcome this challenging task of modern cellular mobile network design and operation, currently, there is a strong interest to enhance the degree of automation in cellular mobile networks. Particularly, the concept of *Self-Organizing Networks (SON)* has got the attention of academia and industry alike.

The *Next Generation Mobile Networks (NGMN)* alliance of major network operators has identified several use cases for the application of self-configuration, self-optimization and self-healing ideas to automate different tasks in all the phases of a network lifetime like deployment, operation and maintenance [10]. The importance of *Self-Organization (SO)* in future cellular mobile networks has also been realized by *3rd Generation Partnership Project (3GPP)*. Currently, it is working to identify different concepts, requirements and solution ideas for the introduction of SO in *Long Term Evolution (LTE)* and *Long Term Evolution Advanced (LTE-Advanced)* [4] [5]. Moreover, some research projects involving both academia and industry are focusing on developing solutions for these requirements [31] [64] [75].

1.1 Coverage and Capacity Optimization

Coverage and Capacity Optimization (CCO) is one of the vital SO use cases for future cellular mobile networks. It aims to maximize the network capacity while ensuring that the targeted service areas remain covered.

The life cycle of every cellular mobile network starts with a target coverage and capacity plan. Based on these targets, network planning then identifies how many network elements need to be deployed at what location. However, as most of the planning tasks for radio networks are done by simulation tools based on network models, the actual performance after the deployment varies based on a number of reasons, as shown in Fig. 1.1. On one hand, the modeling discrepancies in the network planning tools result in variation of coverage and capacity of

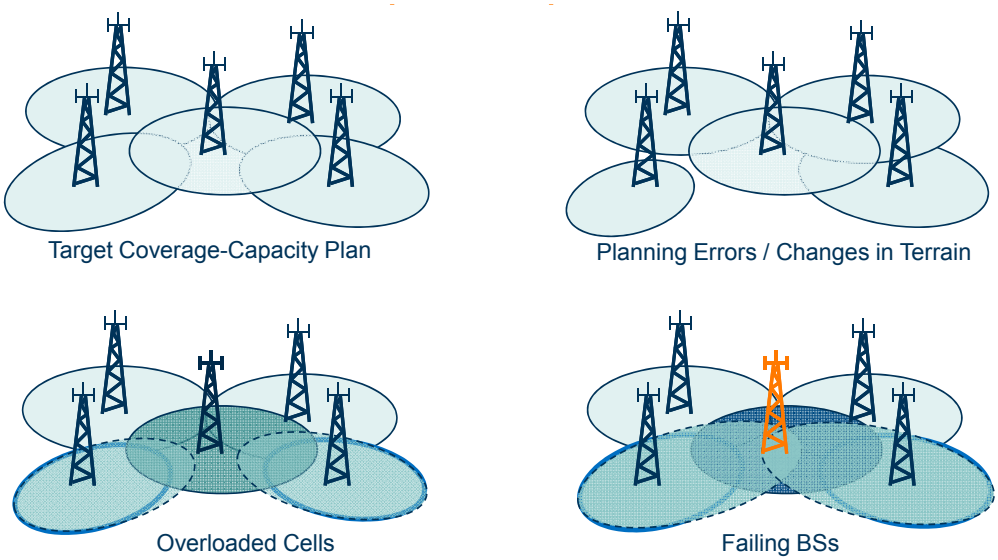

Figure 1.1: Coverage and Capacity Optimization Problem

the deployed network compared to the original targets. On the other hand, the performance also varies based on the network and environmental dynamics. As signal propagation is affected by environmental variations like changes in the terrain and seasonal changes, so the network coverage also varies over large periods of times. The spatio-temporal variations in user traffic patterns also create some congestion problems in the network and affect the Quality-of-Service (QoS) of the users. Sometimes, network elements like BSs also experience some software or hardware malfunction. In the worst case this could also lead to coverage outages in the respective BS coverage area. Currently, these degradations are rectified over a long period of time by careful drive-test mechanisms, observing different KPI logs of various network elements and the feedback from users experiencing low QoS in their desired areas.

SO Coverage and Capacity Optimization helps to empower the cellular mobile networks, so that, they can detect and overcome these performance degradations in an autonomous manner. This could significantly reduce the network response time by eliminating most of the manual time consuming tasks. In addition, the OPEX could also be reduced by minimizing the costly drive-test campaigns and site visits required for network optimization.

This autonomous network optimization requires that the cells can dynamically change their coverage either to regain their own performance targets or to help their neighbors, to improve the overall network performance as highlighted in Fig. 1.1. For this purpose, we are considering the *Vertical Antenna Tilt* as the

cell configuration parameter to change its coverage. Vertical Antenna Tilt is also known as the Antenna Downtilt or simply Antenna Tilt and we use these terms interchangeably in this thesis. As antenna tilt influences the vertical direction of the main beam of the radiation pattern of the cell so it can be used to modify the radio signal strengths in the coverage area of the cell. Higher antenna tilt values mean more concentrated transmission of radiated power in the vicinity of the BS and therefore reduced coverage and vice versa for lower antenna tilt values. But, at the same time, for mobile users it means higher achievable capacity because of better received signal strengths and less inter-cell interference. Therefore antenna tilt affects both network coverage and capacity and needs to be optimized to achieve the required tradeoff between the two.

1.2 Objective and Scope

Although a wide range of studies already exist on antenna tilt optimization for *3G UMTS (Universal Mobile Telecommunications System)* networks and now LTE, there are still some issues that remain un-addressed. For example, the available literature can be broadly divided into two major categories; offline approaches that try to automate the network planning tools with the help of some heuristic algorithms like Simulated Annealing (SA) and Genetic Algorithms (GA) in order to quickly find near optimal solutions and online approaches that propose some pre-defined rules to modify the antenna tilt depending upon the network state in an operational network.

With both of these approaches, optimization intelligence remains mainly concentrated in the design phase with the optimization engineers. The performance of the offline approaches heavily depend upon the modeling of the network as well as the environment in which it is operating. As it is quite difficult to model all the factors that influence the networks' environment, the optimization solutions implemented as a result of these approaches generally need to be fine tuned by the network optimization engineers with the help of systematic drive test campaigns. On the other hand, designing some general purpose optimization rules that can produce optimal results in all kinds of scenarios also require a deep understanding of the network and environmental dynamics by the design engineers.

Apart from this complex design phase issue, both of these approaches also do not allow the optimization algorithms to learn from their experience. Therefore, eventually it comes to the network operation and optimization engineers to fine tune either the results of these optimizations or the optimization algorithms themselves.

To overcome these challenges, in this thesis we focus on how we can shift the optimization intelligence from humans to the cellular mobile networks. Es-

pecially, we focus on *Machine Learning (ML)* techniques, so that, the networks can themselves learn their optimization control structures. This on one hand can simplify the design phase by allowing the networks to learn from their interaction with the environment instead of requiring the design engineers to come up with the optimal algorithms before the network deployment. On the other hand this could also reduce the network response time to the network and environmental dynamics by removing most of the manual tasks from the optimization phase. Moreover, as the algorithms can learn from their past experience, so they can effectively adjust to the local dynamics of each cell and can produce more and more efficient solutions with the passage of time.

In addition to this algorithmic focus on ML, from a structural point of view, we also focus on distributed solutions instead of a centralized solution like the ones which rely on network planning tools. This helps to overcome single point of failure problem as well as make it scalable even for very large network sizes.

1.3 Contributions of the Thesis

The thesis studies the effectiveness of machine learning for self-organized coverage and capacity optimization of cellular networks using antenna tilt adaptations. Especially, we focus on the interaction between different learning agents in such a multi-agent learning environment and the associated problems.

In this regard the first contribution of the thesis is a comprehensive survey of state of the art on antenna tilt optimization in chapter 2. It presents the available literature on antenna tilt adaptation for different network optimization targets like coverage-capacity optimization, load balancing, eNB deployment and energy saving. Afterwards, the algorithmic approaches used for antenna tilt adaptation are also analyzed, which shows heavy dependence on heuristic algorithms. These algorithms can be used to automate the network planning tools to simplify the network deployment and optimization tasks. In this thesis, we want to see, if we can minimize the dependence on these centralized tools and shift most of these optimization tasks to the individual cells. Therefore, we propose to use machine learning to allow the cells to learn from their interaction with their environments and build their own optimization controllers.

We propose to use *Fuzzy Q-Learning (FQL)* as the machine learning tool in this thesis because of its ease and effectiveness in learning problems. In chapter 3, we provide the complete modeling of the antenna tilt adaptation in terms of a multi-agent FQL problem where each cell acts as a learning agent. We also propose different learning strategies for this multi-agent learning problem, which takes into account different level of interaction among these learning agents. From structural point of view, we analyze the performance of distributed learning and

centralized learning in this problem. From behavioral point of view we study the effect of selfish and cooperative learning schemes. Finally, we also study the impact of various level of parallelism on the learning performance i.e. what if only one cell can take an action in each learning snapshot or all of the cells or only a cluster of cells?

All of these learning schemes are analyzed using a system level LTE simulator and the results are presented in chapter 5. We provide the performance results of the proposed learning schemes in a regular hexagonal cell deployment scenario as well as an irregular cell deployment scenario having cells of different sizes. We also compare our results to one of the related learning schemes from literature.

1.4 Thesis Organization

The rest of the thesis is organized as follows:

Chapter 2 presents an overview of antenna tilt optimization in cellular networks. The chapter starts with a brief description of different network parameters that influence the coverage and capacity of the network. Antenna tilt that can be adapted remotely and frequently is then discussed in more detail. Different mechanisms for antenna tilt adaptation are discussed followed by related studies on antenna tilt optimization in different cellular networks. After that, the optimization metrics and algorithms used for antenna tilt adaptation in literature are discussed. Finally, challenges and shortcomings of these techniques are highlighted at the end.

Chapter 3 introduces reinforcement learning and its special form Q-Learning as machine learning tools that can help automate the antenna tilt optimization as well as allow the network to learn from its experience. As antenna tilt optimization problem involves continuous variables, Fuzzy Q-learning is described as an effective combination of Fuzzy Logic and Q-Learning that can also solve learning problems with continuous variables. Different learning strategies that we propose for efficient reward characterization and learning are then described at the end of this chapter.

Chapter 4 describes the LTE network simulator we use for our simulation studies. It explains different radio network modeling concepts as well as the performance metrics that we use for our evaluations.

Chapter 5 presents the results of our evaluation studies. The learning strategies introduced in Chapter 3 are compared in a 3GPP compliant regular hexagonal cell layout. After that, the results for a bigger and irregular cell layout are also presented for more realistic evaluations.

Chapter 6 finally summarizes and presents the conclusions of the thesis. Furthermore, possible future research directions are also highlighted at the end.

2

Antenna Tilt for Coverage and Capacity Optimization: An Overview

In an operational cellular network, coverage and capacity is affected by a number of factors; environmental aspects like topographic and seasonal variations affect the wireless signal propagation, whereas, mobile users' behavior influence the service demand distribution of the network. As network operators have no control over these issues, so, it is extremely challenging for them to maintain the coverage and capacity targets. Therefore, enormous efforts are spent on network planning and optimization in order to make sure that the required network resources are available in the targeted areas of network operation. This chapter provides an overview of different network parameters that the network operator can modify to influence the coverage and capacity. After that, different mechanisms to adapt antenna tilt are also explained. Finally, a detailed discussion of the available literature on antenna tilt adaptation is provided.

2.1 Network Parameters Influencing Coverage and Capacity

Cellular network design offers a number of parameters to modify the network coverage and capacity. The deciding factors in the selection of parameters for SO coverage and capacity optimization are; the effectiveness of that parameter to overcome the problem, the ease with which it can be modified especially in an automatic manner and how quickly it can be modified. In the following section, some of these parameters are discussed along with their effectiveness for SO coverage and capacity optimization.

2.1.1 Frequency Spectrum and Frequency Reuse

Available frequency spectrum is a crucial factor in determining the capacity of the system. To reduce the co-channel interference, second generation cellular networks like *GSM (Global System for Mobile Communications)* split the available frequency spectrum among the cells to have distinct frequencies in the adjacent cells. This allows the possibility of dynamic spectrum allocation to the cells to match the traffic dynamics. However, modern cellular networks like LTE are frequency *re-use 1* systems, meaning they use the complete available spectrum in each cell, to increase the spectrum utilization [39]. Therefore in these networks, dynamic capacity enhancement in a cell by spectrum allocation is not a feasible solution.

2.1.2 Base Station Density

Today's modern cellular mobile networks are based on the concept of "Cells", which was developed by Bell Labs, USA in late 1940s. The concept allows the network coverage area to be divided into numerous small cells each with its own Base Station (BS). This allows reusing the frequency spectrum for radio access, thus enhancing the overall capacity of the network. Therefore, careful densification of BS, so, that the interference remains under a certain limit can provide significant gains in network coverage and capacity. However, BS cannot be deployed at arbitrary places. Due to legal and health obligations, they can only be deployed at some carefully selected places. Moreover, financial and timing constrains also make this option feasible to cater for the long term coverage and capacity upgrades only.

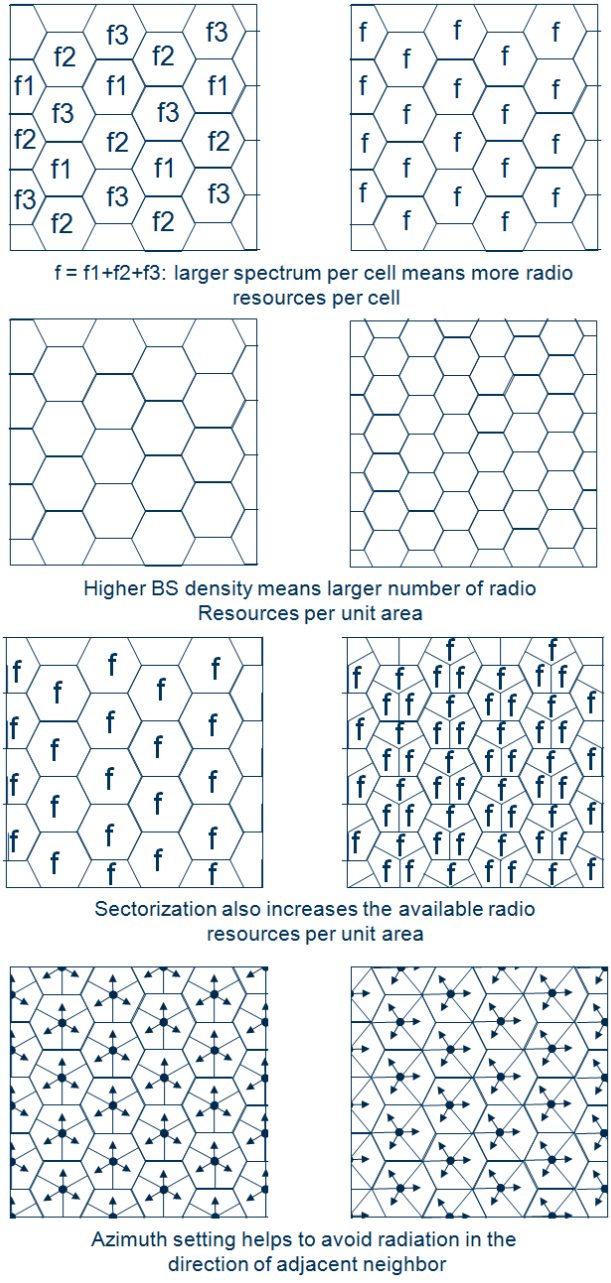

Figure 2.1: Parameters for CCO

2.1.3 Sectorization

BS coverage can be divided into multiple sectors using directional antennas. Unlike omni-directional antennas, directional antennas radiate the transmitted signals in a particular direction and therefore can increase the capacity of the network by reducing the interference in other directions. In traditional networks the number of sectors each BS has, is decided at the planning phase. As it requires site visit and hardware upgrades to change the sectorization configuration, it can only be done over large periods of time. However, with modern smart antenna systems it is also possible to dynamically change it in much shorter times [77].

2.1.4 Antenna Azimuth

Antenna azimuth is defined as the angle of main beam of a directional antenna w.r.t. the North Pole in the horizontal direction. It can be used to steer the antenna radiation pattern and to reduce the interference to the adjacent cells. If the adjacent antennas point towards each other they produce more interference compared to if they are directed away from each other as also shown in Fig. 2.1. The value of azimuth is normally influenced by the relative positions of the adjacent BS and the targeted coverage areas. Therefore, the possibility of dynamic capacity enhancements by antenna azimuth adaptation are limited.

2.1.5 Antenna Height

Antenna height of the BSs also influences the received signal strengths in its coverage area. Higher the antenna height is, further the radio signals can propagate and therefore larger is the coverage area. However, its value is fixed at the planning phase and it is extremely difficult to modify it dynamically.

2.1.6 Antenna Tilt

Antenna tilt is defined as the elevation angle of the main lobe of the antenna radiation pattern relative to the horizontal plane. If the main lobe moves towards the earth it is known as downtilt and if it moves away it is known as uptilt. Higher antenna downtilts move the main lobe closer to the BS and vice versa. Therefore, the antenna tilt value has a strong influence on the effective coverage area of the cell. Moreover, with relatively close direction of the main lobe to the BS the received signal strengths in own cell improves and the interference to neighboring cells reduces. This improves the signal to interference plus noise (SINR) ratio for the mobile terminals and the network capacity increases. Therefore, antenna tilt can be used to alter both coverage and capacity of the network at the same time.

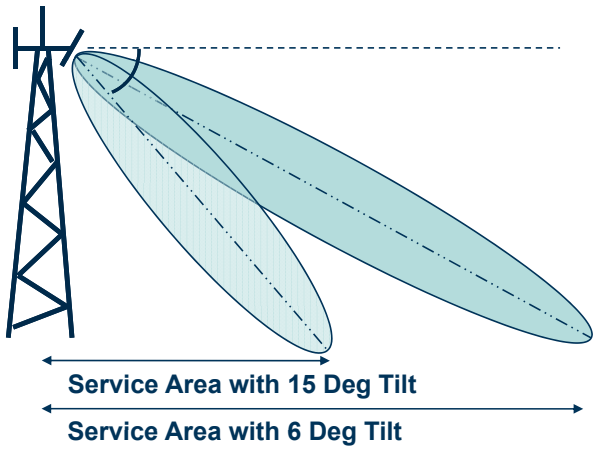

Figure 2.2: AntennaTilt

In our study, we also focus on antenna tilt adaptation for coverage and capacity optimization.

2.2 Antenna Tilt Mechanisms

Primarily antenna tilt can be modified either mechanically or electrically.

2.2.1 Mechanical Antenna Tilt

Mechanical Antenna Tilt (MAT) involves, physically changing the BS antenna so that the main lobe is directed towards the ground. The antenna radiation pattern mostly remains unchanged only a notch develops at the end of main lobe [51]. This reduces the interference in the main lobe direction. However, the effective tilt the side lobes experience varies and the rear lobe in facts experience an uptilt. Adaptation of MAT also requires a site visit, which makes it an expensive and time-consuming task.

2.2.2 Electrical Antenna Tilt

Electrical Antenna Tilt (EAT) involves, adjusting the relative phases of antenna elements of an antenna array in such a way that the radiation pattern can be tilted uniformly in all horizontal directions [80]. EAT can be performed in a number of ways like, Fixed Electrical Tilt (FET), Variable Electrical Tilt (VET),

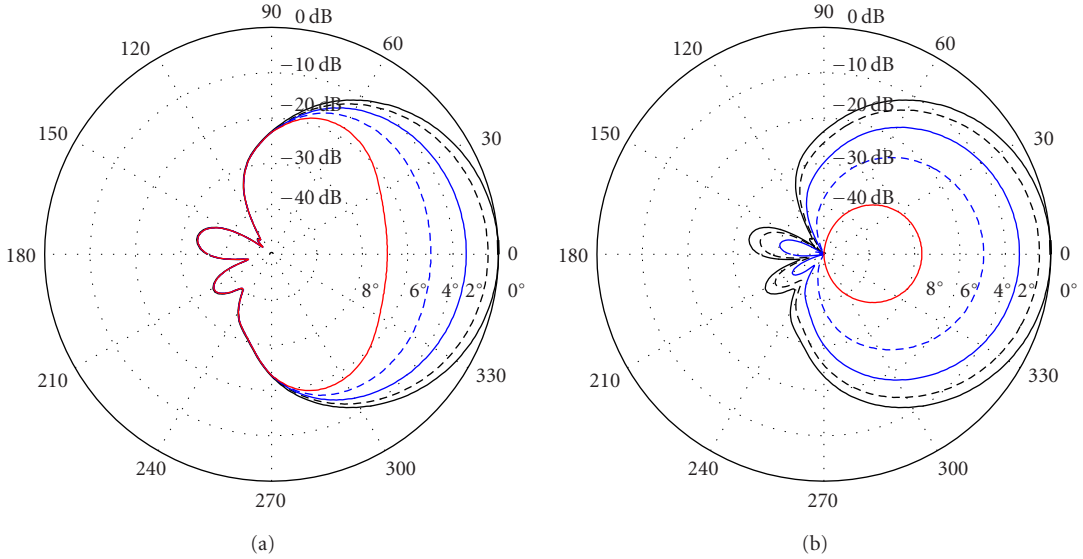

Figure 2.3: Mechanical Tilt (a) Vs Electrical Tilt (b)

Remote Electrical Tilt (RET) and Continuously Adjustable Electrical Downtilt (CAEDT). FET comes with a fixed antenna tilt and the antenna needs to be changed if the tilt needs to be modified or the tilt is adjusted mechanically. VET offer the possibility of adjustable antenna tilt within a range of values. Antenna tilts can also be modified without a site visit with the help of RET. It can adjust the antenna tilt remotely e.g. from network management centers. Hence, it can save the cost and time required for antenna tilt optimization. A further enhancement of RET mechanism is the CAEDT mechanism, which can change the antenna tilts continuously and remotely to overcome the coverage and capacity problems due to network dynamics.

2.3 Potential for Antenna Tilt Optimization

Antenna tilt is an effective parameter to control the coverage of each cell and therefore the interference it produces to the neighboring cells. For this reason, it has been widely used by cellular network engineers for network optimizations. Mechanical antenna tilt can reduce the other cell interference in the main lobe direction [51]. Therefore, it has been extensively used in GSM networks to reduce co-channel interference and achieve a tighter frequency re-use pattern. Capacity gains of up to 25% are also reported for GSM networks using MAT [46] [88]. However, achieving these performance gains is resource extensive as each MAT

requires a site visit. To make the process more efficient, a mechanism based on mobile unit reports to prioritize the cells is presented in [79]. This allows, identifying the cells where the MAT adjustment is most critical so that the human resources can be efficiently used.

Code Division Multiple Access (CDMA) based third-generation mobile communication systems like UMTS (Universal Mobile Telecommunication System) are inherently interference limited. This makes it even more critical to design these networks so that other cell interference is minimized. Capacity gains between 15% and 20% have been reported in WCDMA (Wideband CDMA) macrocellular environment with MAT [24] [58]. Similarly, MAT has also been observed to be helpful in microcell environment [27] [83]. Unlike MAT, side lobes can also be downtilt with the help of EAT, so EAT has been observed to given even better performance compared to MAT both in macrocellular [32] and microcellular [48] environment.

Apart from the type of antenna tilt mechanism, the performance gains of antenna tilt as well as the value of the optimal antenna tilt also depends on other BS configuration parameters like sectorization, antenna height, vertical beamwidth and site spacing [42]. Capacity gains of 50% for 3 sectored site and 20% for 6 sectored site has been reported for CDMA based systems [50]. Depending upon the horizontal beamwidth of the antenna radiation pattern, the gains for different sectorization schemes can also vary. Therefore it is recommended to have smaller horizontal beamwidth antennas like 65° for 3 sectored site and 33° for 4 or 6 sectored site [44]. Moreover, the value of optimal antenna tilt is observed to increase with increasing antenna height and decrease with increasing site spacing [57].

With the introduction of LTE (Long Term Evolution) systems, network operators are now planning to upgrade their networks. This transition would have been easier and quicker if the BSs could be deployed according to the configurations of the existing 3rd generation networks. However, this is not possible due to the fundamental differences in the technologies. In a field measurement campaign, throughput gains of about 26% have been reported by antenna tilt optimization of LTE BS compared to the scenario when CDMA1x network settings are applied [21].

In LTE systems, throughput gains by proper antenna tilt settings have also been observed in simulation studies [39]. However, these gains are highly scenario Dependant [86]. In 3GPP case 1, which is an interference limited scenario with an inter-site distance of 500m, higher antenna tilts with wider horizontal beamwidths are reported to provide better coverage and capacity performance. However, in 3GPP case 3, which is a noise limited scenario with an inter-site distance of 1750m, moderate antenna tilt and narrow horizontal beamwidths are observed to provide noticeable capacity improvement but little effect on the coverage of the cells. As LTE uses same frequency bands in all the cells and does not employ

macro diversity, so higher cell isolation is desirable [39]. For this reason, EAT is observed to perform better than MAT in LTE because it can tilt the side lobes as well [15] [85].

Apart from the system performance enhancement features of antenna tilt it has also been observed to simplify the network planning phase. The potential of antenna tilt optimization to overcome sub-optimal antenna heights has been reported in [8]. The results show that by optimizing the antenna tilt, optimal system performance can still be achieved even if the antennas could not be placed at the optimal height.

2.4 Antenna Tilt Optimization Objectives

It is clear from the above discussion that antenna tilt is an effective tool to influence the cellular network performance. Depending upon the network situation, it can be optimized for a number of performance metrics. In the following sections an overview of different optimization studies is presented according to the performance metric that antenna tilt adaptation can achieve.

2.4.1 Coverage and Capacity Optimization

The primary target of antenna tilt optimization is to ensure that the targeted areas remain covered by the radio network and the available capacity is sufficient to support the user demands. The simplest way is to utilize a fixed antenna tilt value across all cells. This network wide antenna tilt optimization can enhance the network performance compared to no tilt optimization [56]. However, the environment in which the cells operate changes from cell to cell and in real networks all cells can hardly be symmetrical geographically. Therefore, the radio environment changes and cell configuration parameters need to be adjusted individually [71]. This site-by-site optimization can achieve 15% capacity gains in WCDMA macrocellular environments compared to network wide tilt adjustment [76]. This capacity gain is a direct result of the inter-cell interference reduction due to better focus of the radiated power in own cell [35] [36].

In WCDMA systems the total transmission power of a BS is split between different control and traffic channels. The less power these control channels require the more power is available for the traffic channels. One of these control channels is the Common Pilot Channel (CPICH), which is a downlink broadcast channel used to announce the presence of the radio cell within its coverage area. Its power level is crucial for clear demarcation of cell boundaries. With optimized antenna tilt settings the power for CPICH can be reduced up to 60% [67] [68] [70]. Moreover, the power level for many other control channels like SCH, PICH,

CCPCH, and AICH whose power is set relative to the CPICH power can also be reduced. All of this saved power can then be used for the traffic channels to enhance the capacity. These capacity gains can also help to lower the call blocking [74] [81] and dropping [26] probabilities in a network.

For data communication, capacity is often measured in terms of achievable throughput. Antenna tilt optimization is also observed to enhance the average user throughput up to 30% in HSDPA (High Speed Downlink Packet Access) networks [7] [69]. Apart from the average throughput values another important performance metric is the cell edge throughput, especially in LTE networks, which use the same frequency spectrum in each cell. Therefore, antenna tilt optimization becomes even more critical in LTE networks to achieve better cell isolation and improved SINR distribution, generally within the cell and particularly at cell edges [73]. This improvement in the SINR distribution allows higher order modulation and coding schemes to be used and thus enhance the achievable throughputs both at cell center and at the edges [16] [17]. This performance gain becomes even more evident in irregular cell deployment scenarios where average performance is observed to increase by 10% and cell edge performance even up to 100% compared to a network wide uniform antenna tilt setting [29].

2.4.2 Load Balancing

Traffic demands in cellular networks experience spatio-temporal variations. This could lead to situations where some of the cells get overloaded. With static network configurations Quality of Service (QoS) would degrade for users in those cells. This degradation can be minimized, if the network configurations can be changed dynamically, so that, the neighboring cells which have free resources available share some of the traffic load of the overloaded cell.

As antenna tilt affects both coverage and capacity of the cell [40], it can be used to change the cell boundaries to balance the load among neighboring cells. This requires that the antenna tilts can be dynamically modified for individual cells instead of applying one tilt value across all cells [33] [55]. This dynamic adaptation of antenna tilt has been observed to produce capacity gains of about 15% in a WCDMA macrocellular environment [28] and significantly reduces the number of overloaded cells by balancing the traffic load among different neighboring cells [34]. These capacity gains highly depend on the traffic distribution and increase with the increasing imbalance among the cells [60]. This load balancing ability also increases with increasing available capacity in the neighborhood. A capacity gain of almost three fold has been observed for a hotspot scenario where the neighbors are only 20% loaded compared to the scenario where the neighbors are 80% loaded [82]. The utilization of antenna tilt for load balancing has also been reported for LTE systems [49], where capacity gains up to 40% can also be

achieved.

2.4.3 BS Deployment

Radio network planning involves the selection of the BS locations and their configuration so that the required capacity can be provided in the targeted areas. The constraints for the planning process involves a limited number of feasible locations for BS deployment, cost of the deployment, minimum area that needs to be covered and the minimum capacity that needs to be provided. Under these constraints, the objective of the network planning optimization is to reduce the overall cost of the network deployment. As BS configuration also plays a crucial role in determining the effective coverage and capacity of the cell, considering factors like transmit power and antenna parameters while optimizing the BSs location can considerably reduce the number of BS required to meet certain targets both for 2G GSM networks [9] [30] and 3G WCDMA network [6] [11].

2.4.4 Self-Healing

During operational state, cellular networks can also experience some malfunctioning cells either due to software or hardware failures. This could lead to service degradation or in the worst case a coverage outage in the service area of that cell. Presently, the detection and rectification of these outages is considerably manual and slow process [12].

Self-healing mechanisms try to automate this activity, so that, the cellular networks can themselves identify and resolve these service degradations in an autonomous manner [4]. In extreme cases, where a hardware change is required, self-healing tries to compensate for this outage by extending the coverage of the operational neighboring cells. Antenna tilt is also an effective control parameter to achieve this automatic coverage adjustments [13]. However, the potential for this coverage compensation depends on a number of parameters [14]. The higher the cell load of the compensating cell and bigger the inter-site distances are the lower is the potential for compensation. Moreover, this compensation comes at some quality degradation within the original coverage area of the compensating cell as less radio resources are available per user. Therefore, operator policy about the trade-off between outage compensation and acceptable quality degradation can significantly affect the potential of self-healing mechanisms to overcome the problem.

In isolated cells at the border of the network, by using antenna tilt, self-healing can also dynamically adapt the coverage of an active cell to overcome the outages due to planning errors [52].

2.4.5 Energy Saving

Recently, a strong interest has developed in the reduction of energy consumption of the cellular networks due to environmental and financial aspects [4]. As traffic demands vary over time for different places, one of the possible solutions is to switch off some of the cells during off-peak hours. Nevertheless, some basic coverage still needs to be provided in the service area of the switched off cells. Similar to self-healing principles, dynamic antenna tilt adaption is one of the options to provide this basic coverage for the switched-off cells [23]. The energy saving potential varies from scenario to scenario depending upon the traffic distribution and the ability of the neighboring cells to support the remaining network load. Energy saving of 5% to 13% has been reported for LTE network simulations with an inter-site distance of 500m [23] [84].

2.5 Optimization Approaches

Optimal antenna tilt value for a cell depends on the tilt values of its neighbors, so that, traffic remains balanced between the cells and desirable cell isolation is achieved. Additionally, antenna tilt can only have discrete values, so the optimization problem translates to finding the best combination of antenna tilt values across the neighborhood. However, the number of antenna tilt combinations that different cells can have is extremely large even for very small networks. This makes brute force search to select the optimal setting infeasible. Different optimization techniques have been proposed in literature to find reasonably good solutions in a time-efficient manner. Some of these optimization approaches are discussed below.

2.5.1 Meta-Heuristic Algorithms

Meta-heuristic algorithms define optimization algorithms that iteratively try to improve the solution based on some measure of quality. These algorithms can quickly search large portions of the solution space but do not guarantee to find the optimal solution. For these reasons a vast majority of the available solutions for antenna tilt optimization also focus on meta-heuristic algorithms to quickly find a reasonably good solution. These techniques rely on network planning tools to evaluate the quality of each candidate solution.

A Local Search (LS) based solution is proposed in [67]. It starts with one of the possible solutions than iteratively checks the quality of the neighboring solutions. Here, a neighbor solution is defined by changing one of the tilt values from the current candidate solution. If the neighbor solution performs better than the previous solution, it is selected as the best solution. This process repeats until a

certain quality of solution is achieved or a fixed number of iterations is completed. However, simple local search algorithm suffers from the local maxima problem. Simulated Annealing (SA) tries to solve this problem by selecting neighbor solutions even with lesser quality of measure as best solution with some probability. This probability decreases as the iterations progress to stabilize the optimization process. This technique has also been utilized for antenna tilt optimization and has proved to be quite effective [9] [69] [70] [73] [74]. The performance of these techniques depends on the parameter setting of the algorithm like the probability of bad solution selection as well as the definition of neighbor solution. Appropriate definition of neighbor solutions can enhance the quality of the final solution as well as reduce the computational complexity [34].

Another local search algorithm used for antenna tilt optimization is Tabu Search (TS) [11] [55] [68]. It tries to overcome the problem of local maxima by maintaining a list of recently tested solutions. The solutions in this list remains taboo for some time so that the optimizer does not remain stuck in a specific region of the solution space.

Genetic Algorithms (GA) are also meta-heuristic algorithms that mimic the process of natural evolution to find the best solution of the problem. Instead of making a neighboring solution by slightly modifying the current solution, GA combine two solutions from a population of solutions and tries to inherit the best qualities of the parent solutions. GA has also been proposed to automate cell planning tools for the optimization of antenna tilt and other BS parameters [28] [43]. Another population-based approach, Particle Swarm Optimization (PSO) has also been used for antenna tilt optimization [30]. In this approach, a population of candidate solutions moves in the solution space and their movement is influenced by their own best known solution and the global best known solution at any time.

Comparative studies of the above mentioned algorithms for antenna tilt optimization are also reported in [6] [7]. Under low or medium network load conditions all the algorithms provide comparable performance, but at high network load GA are reported to be more computational extensive and less efficient in terms of the quality of the solution for the scenarios under study.

Taguchi's method, a relatively unexplored technique in the domain of cellular network optimization has also been used for BS parameter optimization recently [16] [17]. This technique is based on the concept of Orthogonal Array (different from orthogonal antenna array), which select a representative set of possible parameter combinations from the full search space. The number of parameter combination defines the number of tests to be carried out and compared against a performance metric. Using the results of all the tests, a candidate solution is selected and the process is repeated until the desired criterion is matched. Compared to local search techniques, this OA offers a systematic approach to

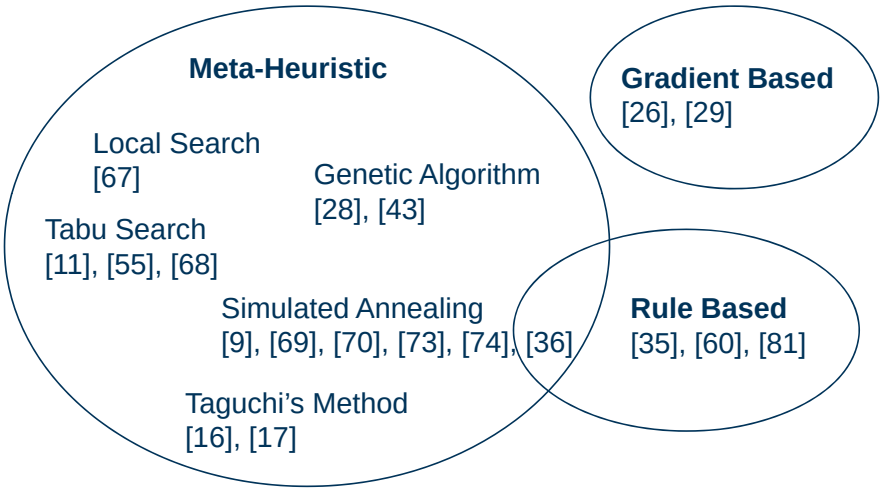

Figure 2.4: Algorithms for CCO with Antenna Tilt

explore the solution space and therefore has been reported to perform better than SA in the studied scenarios [17].

2.5.2 Rule-Based Algorithms

Some rule-based techniques have also been proposed for antenna tilt optimization. These techniques rely on traffic statistics to define the current state of the cell. This could be based on uplink (UL) load [35], downlink (DL) load [81] or the interference experienced by the cell [60]. Then rules are defined, so that, the cells with higher load increase their tilt to reduce their coverage and the cells with lower tilt decrease their tilt to increase their coverage for better load balancing among the cells.

2.5.3 Gradient-Based Algorithms

Gradient-based schemes have also been exploited for antenna tilt optimization. A utility function based on the active users and the dropped calls has been proposed in [26]. At each control update, the gradient of the utility function is used to modify the antenna tilt with a maximum limit of 2 degrees per update. Then Game Theory (GT) analysis has been done to prove that the network converges to the Nash Equilibrium (NE). In [29] a utility function based on the cell edge and cell center spectral efficiency has been defined and the tilt at each cell is iteratively adapted based on the gradient of the utility function.

2.5.4 Benefits and Challenges of Optimization Approaches

The above mentioned optimization approaches have proven to be quite effective for the antenna tilt optimization problem. Meta-heuristic algorithms are the most widely used optimization approaches as they can provide solutions with significant performance improvements in short intervals of time even for very large networks. However, they can only be run in an offline manner because of the requirement of a network planning tool to test different candidate solutions for specific quality of measure. Therefore, the quality of these solutions highly depends on the network and environmental modeling used in these tools.

In literature, the performance gains of rule-based algorithms are also proved using simulations, but principally they can also be applied in an operational network, as they define rules to iteratively adapt the antenna tilt in small steps based on some load metric of the neighboring cells. However, most of these approaches only consider synthetic hexagonal scenarios and therefore define rules based on the load difference of only two adjacent cells. In reality, the cell structures are hardly symmetrical because of the propagation conditions and the limited availability of feasible site locations. Therefore, the designed rule base needs to consider a broader neighborhood.

Another major problem with these approaches is that the optimization intelligence remains heavily concentrated in the design phase of the algorithms. The performance of these algorithms heavily depends on their parameter settings and the modeling used for the network and the environmental factors. Additionally, these algorithms are not able to learn from their experience of previous optimization steps and rely on network optimization engineers to either fine tune the results of these optimizations in the operational network or to adjust the parameters of the optimization algorithms.

To overcome these challenges, Machine Learning (ML) can help to empower the radio networks to learn from their optimization steps. This way radio networks can themselves become experienced with the passage of time and can fine tune the optimization algorithm based on their local environmental interactions. Additionally, if a distributed architecture is followed for the learning mechanism, the optimization can adjust according to the local requirements of individual cells as well.

2.6 Summary

In this chapter an overview of antenna tilt optimization is presented. The main points are summarized below:

- Coverage and capacity of cellular mobile networks is affected by numerous

BS parameters, but the ability to adapt antenna tilt remotely and frequently makes it more suitable for SO network optimization.

- The remote adaptation is possible due to the electrical antenna tilt mechanism. Moreover, with electrical antenna tilt the radiation pattern experience the tilt in all horizontal directions and therefore produces higher performance gains compared to the mechanical antenna tilt, which is only effective in the main lobe direction.

- For any cell the optimal antenna tilt value depends on different factors like traffic distribution within the cell, load difference between neighboring cells, BS sectorization, horizontal as well as vertical antenna patterns and antenna height.

- Most of the above mentioned BS parameters are fixed during the network planning phase but mobile user distribution changes from time to time. To provide the required capacity for this inhomogeneous and dynamically changing traffic distributions, dynamic antenna tilt adaptation is required. Additionally, for better load balancing, these antenna tilt adaptations need to be optimized on individual cell basis instead of same antenna tilt value across all cells.

- Numerous optimization algorithms exist for dynamic antenna tilt optimization but their optimization intelligence is limited because of fixed predefined algorithms. Therefore, network optimization engineers are still required to either fine tune the results of these optimizations or to adjust the algorithms themselves.

- Machine learning can help to fully automate this process by allowing the network to learn from its previous optimizations and adjust its behavior for future actions.

- This kind of machine learning based optimization is quite new in the domain of cellular networks and requires more detailed studies to prove its effectiveness.

3

Reinforcement Learning Based Coverage and Capacity Optimization

The SO features of cellular mobile networks require that most of the operational and maintenance tasks are automated. This minimizes the manual efforts required for network deployment and optimization. These automated procedures also benefit from machine learning to introduce some intelligence in their behavior. The objective for this intelligence capability is to allow these procedures to learn from their previous steps and amend their future actions accordingly.

In this chapter we look into Reinforcement Learning (RL) as a machine learning technique to learn the optimal control policy for antenna tilt adaptation. First, we briefly explain the basic concept of RL and the Q-Learning algorithm to solve RL problems. After that, Fuzzy Q-Learning (FQL) is described which combines Fuzzy Logic with Q-Learning to handle RL problems where the state or

action space is continuous. Next, the different components of an FQL controller for our coverage and capacity optimization problem are explained. Finally, different variants of the FQL that we developed are also presented.

3.1 Reinforcement Learning

Reinforcement learning is a branch of machine learning that iteratively tries to learn the optimal action policy through its interaction with the environment [47] [72]. In a trial-and-error manner, a learning agent[1] continuously analyzes the consequences of its actions based on a simple scalar feedback (the reinforcement) from the environment. Depending on this reinforcement signal the agent tries to develop a ranking of all the possible actions for a particular state and preferentially repeat the actions that in similar state produced better performance.

The reinforcement signal is only a qualitative feedback and tells the agent whether the selected action was good or bad in terms of some performance metric. It is less informative than the feedback in Supervised Learning (SL), in which the learning agent knows the relation between the input state and the desired output [25]. Therefore, in SL the agent always knows the error it commits based on the difference between the actual and desired outputs and thus can apply the corrective measures. However, the reinforcement feedback is better than the Unsupervised Learning (UL), where the agent would be left to discover the optimal actions on its own, without any explicit feedback from the environment [41].

This interaction between the learning agent and its environment can be modeled as a Markov Decision Process (MDP). An MDP is a 4-tuple $\langle X, A, P, \rho \rangle$, where, X is the set of all possible states, A is the set of all possible actions, $P : X \times A \times X \rightarrow [0, 1]$ is the state transition probability function and $\rho : X \times A \times X \rightarrow \Re$ is the reward function.

The complete high level process is presented in Fig. 3.1. At each time step k, the agent perceives its complete environment and determines the present state $x_k \in X$. The agent can choose one of the possible actions in this state $a_k \in A$. After the execution of this action, the environment changes from state x_k to another state $x_{k+1} \in X$, according to the state transition probability function $P(x_k, a_k, x_{k+1})$, the probability of ending in state x_{k+1} when taking an action a_k in state x_k. As a result of this transition, the agent gets a reward $r_{k+1} \in \Re$, according to the reward function $\rho : r_{k+1} = \rho(x_k, a_k, x_{k+1})$. The reward provides feedback about the immediate impact of action a_k only without any regard to its long term effect. This process repeats until the agent arrives in some terminal state.

[1] Any system that is trying to learn

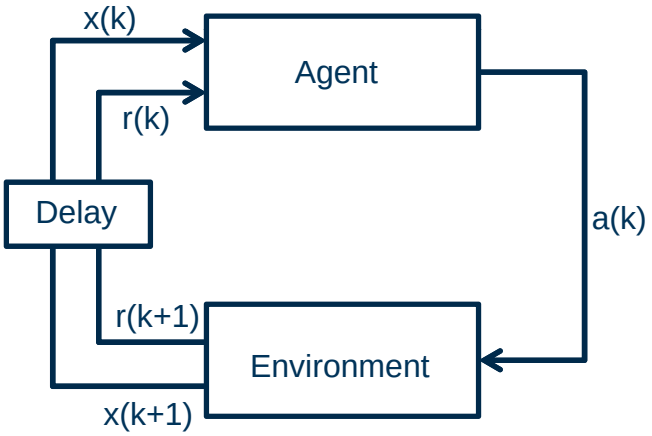

Figure 3.1: RL System

The objective of the learning agent is to maximize the accumulated future rewards as defined by the reward function $R(k)$:

$$R_k = E\left\{\sum_{i=0}^{\infty} \gamma^i r_{k+i}\right\} \tag{3.1}$$

where, $\gamma \in [0, 1)$ is the discount factor. The value of γ allows to control the period the learning agent takes the reinforcement into account. For $\gamma = 0$, the agent is "myopic" and only considers the immediate reward; the more it is closer to 1, the more the agent looks into the future. Therefore, the task of the agent is to maximize its long term performance, based on the feedback about its immediate one-step performance only.

The behavior of an agent is determined by its policy π, which specifies how the agent ought to take actions given a state. The policy can either be stochastic, $\pi : X \times A \to [0, 1]$, or deterministic, $\overline{\pi} : X \to A$. The agent, values a policy by the expectation of future reinforcements with the discounted reward. The value of a state x under the policy π is given by:

$$\begin{aligned} V^{\pi}(x_k) &= E\left\{r(x_k) + \gamma \cdot r(x_{k+1}) + \gamma^2 \cdot r(x_{k+2}) + \cdots + \gamma^k \cdot r(x_{k+m}) + \cdots\right\} \\ &= E\left\{r_k + \gamma \cdot r_{k+1} + \gamma^2 \cdot r_{k+2} + \cdots + \gamma^m \cdot r_{k+m} + \cdots\right\} \\ &= E\left\{r_k + \sum_{m=1}^{\infty} \gamma^m \cdot r_{k+m}\right\} \\ &= E\left\{r_k + \gamma \sum_{m=0}^{\infty} \gamma^m \cdot r_{k+1+m}\right\} \end{aligned} \tag{3.2}$$

which is similar to the Bellman Equation also known as the equation of Dynamic Programming.

$$V^{\pi}(x_k) = E\{r_k + \gamma \cdot V^{\pi}(x_{k+1})\} \tag{3.3}$$

which tells that the value of any state x is equal to the expected immediate reward plus the expected value achievable from the next state. This allows to estimate the value function recursively. The optimal value of a state is given as:

$$V^{\pi^*}(x_k) = V^*(x_k) = \max_{\pi}\{V^{\pi}(x)\} \tag{3.4}$$

Therefore, solving a reinforcement learning problems means, finding the optimal policy π^* that maximizes the long-term expected cumulated reward that the agent receives. This is achieved by learning the correct approximation of the state value functions.

$$\hat{V}(x_k) = V^*(x_k) - e(x_k), \quad with \quad e(x_k) \to 0 \tag{3.5}$$

Once these state value functions are correctly estimated, the optimal policy can be deduced from the optimal state values as:

$$\pi^*(x_k) = \arg\max_{x_k \in X}\{V^*(x_k)\} \tag{3.6}$$

RL problems can be classified along several different dimensions like the amount of learning agents in the system, type of learning agents, type of learning task, type of interaction among multiple agents etc. In the following sections some of the major RL problem categories are discussed.

3.2 Single-Agent Reinforcement Learning

Single-Agent Reinforcement Learning (SARL) deals with problems where we have only one agent in the system. The agent explore its solution space through its interactions with the environment and iteratively improves its behavior (action policy), hoping to reach the optimal behavior. A large number of algorithms with good convergence and stability characteristics are already available to solve these problems.

3.3 Multi-Agent Reinforcement Learning

Multi-Agent Reinforcement Learning (MARL) deals with learning problems involving multiple agents with constraints that these agents interact with each

other and their performance depends on their joint behavior. These constraints are important for the definition of MARL problem. If no interaction is required then the problem can be decomposed into completely independent tasks each solvable by a separate agent [59]. These constraints also differentiate MARL problems from SARL problems in three important aspects. First, the existence of multiple-agents expands the solution space enormously. Second, MARL may involve multiple learners, each learning and adapting in the presence of others, which makes the environment non-stationary and violates the basic assumption of numerous SARL approaches. Third, because of the interaction between multiple-agents, small changes in the learned behavior can often result in unpredictable changes in the overall behavior of the whole system.

Depending upon the number of learning agents and their behavior in the system, MARL problems can be broadly divided into two main categories, i.e. team learning and concurrent learning [59].

3.3.1 Team Learning

In Team Learning, a single learning agent is used to learn the behavior for a team of agents[1]. This simplifies the learning process compared to the scenario where each team member is learning in the presence of other co-learning team members. Consequently, team learning can utilize the better understood SARL techniques with good convergence and stability characteristics [25]. Another advantage of team learning is that it tries to improve the performance of the entire team and not just that of a single agent.

However, team learning suffers from scalability issues as the number of agents increases. The number of possible states that the team can have increases exponentially with the increasing number of agents. This explosion of state space can make it infeasible to maintain the state-value function or extremely slow-down the learning process. Another disadvantage of team learning is its centralized nature. The single learning agent needs to have information from all agents of the system and all the computational resources need to be concentrated at a single place.

3.3.1.1 Homogeneous Team Learning

Homogeneous team learning tries to learn a single agent behavior for all the agents in the team, even if the agents have different capabilities. The search space is drastically reduced for the learning process because all agents have the

[1] An agent is a system that can perform certain actions in its environment based on some information (sensors, feedback) received from the environment. Whereas, a learning agent also employs some machine learning techniques to learn from its interaction with the environment.

same behavior. This is particularly important for problems where heterogeneous solution space is infeasible to explore. Additionally, some problems do not require agent specialization for better results. Here, homogeneous team learning can perform better with less complexity.

3.3.1.2 Heterogeneous Team Learning

In heterogeneous team learning, a single learner tries to learn different behaviors for different members of the team while improving the performance of the team as a whole. This allows agent specialization within the team at the expense of increased search space.

3.3.1.3 Hybrid Team Learning

Hybrid team learning tries to combine the benefits of both homogeneous and heterogeneous team learning. It divides the team into multiple squads, with each agent belonging to only one squad. All agents in a squad have the same behavior, but different squads can have different behaviors to achieve specialized characteristics.

3.3.2 Concurrent Learning

The second major category of MARL problems is concurrent learning, where multiple learners are trying to partly solve the problem. It is especially useful in problems where some decomposition is possible and each sub-problem can be solved independently to some degree [45]. In such problems, concurrent learning can reduces the search space and computational complexity of the learning agents by projecting the large team search space onto smaller separate search spaces.

However, learning is more difficult in concurrent learning because of the presence of multiple learners. Each learner interacts with the environment and tries to learn the behavior that improve its performance. However, other learners are also co-learning and co-adapting their behavior. Each behavioral change in any agent can make the assumptions of other learning agents obsolete and ruin their learned behavior. This makes the whole environment non-stationary and the learning agents can at best try to keep track of these changes in the environment and their optimal behavior [59].

In concurrent learning, as each agent is free to learn individually, heterogeneity vs homogeneity is not considered as a design decision but an emergent behavior of the system. Many studies have investigated concurrent learning from a game-theoretic point of view and classified it into the following categories.

3.3.2.1 Cooperative Games

Fully cooperative games utilize global reward to divide the reinforcement equally among all the learning agents. All agents have the same goal to maximize the common utility.

3.3.2.2 Competitive Games

In competitive games agents compete against each other and the reward of one agent acts as a penalty for others. Therefore, the sum of rewards of all agents after each transition equals zero.

3.3.2.3 Mixed Games

Mixed games are the ones, which are neither fully cooperative nor fully competitive.

3.4 Reinforcement Learning Algorithms

Fundamentally, RL problems can be solved in three different ways: Dynamic Programming, Monte Carlo methods and Temporal Difference (TD) learning. Dynamic programming methods are very well developed mathematically but they need a complete model of the environment in terms of state transition probabilities and the expected immediate rewards. For many problems, this kind of model is not known a-priori. Monte Carlo methods do not require a model but they are not suitable for online learning, where the learning agent learns through its active interactions with the environment. Finally, TD methods also do not require any model of the environment but they can be implemented in an online manner for step-by-step learning [72].

The study of MARL has also benefited from the advancements in Game-Theory and Direct Policy Search. A number of algorithms have been developed based on the tools provided by these fields as well as their combinations [25].

In this thesis we focus on Q-learning algorithm because of its simple implementation and effective results. The main components and the design of the algorithm is explained in the following sections.

3.5 Q-Learning

Q-learning (QL) is one of the model-free learning method based on TD [78]. It solves the learning problem by estimating a Q-value function or the Q-function

for each state action pair. This Q-function defines the quality of choosing an action $a \in A$ in state $x \in X$ in terms of its long-term expected reward.

$$\begin{aligned} Q^{\pi}(x_k, a_k) &= E\left\{r(x_k, a_k) + \gamma \cdot r(x_{k+1}, a_{k+1}) + \gamma^2 \cdot r(x_{k+2}, a_{k+2}) + \cdots \right. \\ &+ \left. \gamma^k \cdot r(x_{k+m}, a_{k+m}) + \cdots\right\} \end{aligned} \tag{3.7}$$

which is similar to the value function as defined in Eq. 3.2, but, here we explicitly calculate the value of each action in every state. Similarly, the Q-function can be shown to follow the Bellman Equation as:

$$Q^{\pi}(x_k, a_k) = E\left\{r_k + \gamma \cdot Q^{\pi}(x_{k+1}, a_{k+1})\right\} \tag{3.8}$$

and the optimal Q-values, which define the expected reward of selecting an action a in state s and then following the optimal policy π^*:

$$Q^{\pi^*}(x_k, a_k) = \max_{\pi}\left\{Q^{\pi}(x_k, a_k)\right\} = \max_{a_k}\left\{Q^{\pi}(x_k, a_k)\right\} \tag{3.9}$$

and can be iteratively calculated as

$$Q^{\pi^*}(x_k, a_k) = E\left\{r(x_k, a_k) + \gamma \max_{a_{k+1}}\left\{Q^{\pi}(x_{k+1}, a_{k+1})\right\}\right\} \tag{3.10}$$

Q-Learning tries to learn the optimal Q-function in an online fashion by incrementally improving its estimate $\hat{Q}(x_k, a_k)$.

$$\hat{Q}(x_k, a_k) = Q^*(x_k, a_k) - e(x_k, a_k) \tag{3.11}$$

where $e(x_k, a_k)$ is the difference between the optimal and estimated value

$$e(x_k, a_k) = Q^*(x_k, a_k) - \hat{Q}(x_k, a_k) = \Delta\hat{Q}(x_k, a_k) \tag{3.12}$$

This difference between the optimal and estimated value can be used to incrementally improve the estimate as the learning agent becomes more and more experienced as follows:

$$\begin{aligned} \hat{Q_{k+1}}(x_k, a_k) &= \hat{Q}(x_k, a_k) + \beta_k \cdot \Delta\hat{Q}(x_k, a_k) \\ &= \hat{Q}(x_k, a_k) + \beta_k \cdot \left\{Q^*(x_k, a_k) - \hat{Q}(x_k, a_k)\right\} \\ &= \hat{Q}(x_k, a_k) + \beta_k \cdot \left\{r_k + \gamma \cdot \max_{a_{k+1} \in A}\left\{\hat{Q}_k(x_{k+1}, a_{k+1})\right\}\right. \\ &\qquad \left. - \hat{Q}_k(x_k, a_k)\right\} \end{aligned} \tag{3.13}$$

where β is the learning rate and describes how much impact the new information has on the old estimate. For $\beta = 0$ no learning is done and for $\beta = 1$ the new value completely overwrites the old estimate.

3.5.1 Q-Learning Algorithm

The general Q-Learning algorithm is an iterative process of estimating the Q-function and follows the following steps

1. Initialize the estimates of Q-values: $\hat{Q}(x, a) := 0 \forall x \in X, a \in A$
2. Observe the current state $x = x_k$
3. Select an action a_k and execute it
4. Receive the immediate reward r_k
5. Observe the new state x_{k+1}
6. Update the estimate $\hat{Q}(x_k, a_k)$ as described in 3.13
7. $x \Leftarrow x_{k+1}$
8. Repeat steps from 3 to 7 until the terminal condition is met

3.6 Fuzzy Q-Learning

Reinforcement Learning problems are generally modeled as finite state Markov Decision Process (MDP), which requires that the system can be represented as a set of finite states. As a result, it is feasible to calculate Q-value for each state-action pair. However, it becomes extremely difficult if the state or action space is continuous. For example, in our study, coverage and capacity of cellular mobile networks is usually based on the received signal strength, which are continuous in nature. State definitions based on these continuous variables would make it impossible to maintain the Q-values for each state-action pair. Moreover, using fixed thresholds for partitioning the continuous variables into discrete variables leads to abrupt transitions, which may lead to very different actions for two very closely related states.

Fuzzy logic can overcome this problem by providing the required abstraction and yet allowing a smooth transition from one state to another. Fuzzy logic uses fuzzy sets, which have elements with graded degree of membership compared to the classical set theory where the elements' membership is measured in binary terms i.e. either the element belongs to or does not belong to the set. This graded membership allows smooth transition between different sets because as the value of fuzzy variable changes, the degree of membership to a particular set can gradually decrease while its membership to another set can increase at the same time. However, designing a fuzzy logic controller requires human expertise

to define the relation between the input states and the controller's actions. These expertise may not always be available or precise enough.

Fuzzy Q-Learning (FQL) is a combination of fuzzy logic and Q-Learning and tries to overcome the problems of each while benefiting from the strong points of each. Fuzzy logic provides a flexible framework for optimization problems and q-learning can provide the learning and fine tuning mechanisms where supervised learning is not possible.

3.7 FQL Controller Components

FQL controller (FQLC) represent the control system for the optimization of a RL problem. For our studies we assume a distributed architecture where each cell has its own FQLC and tries to optimize its performance using it, as shown in Fig. 3.2. The major components of the FQLC are described in the following sections:

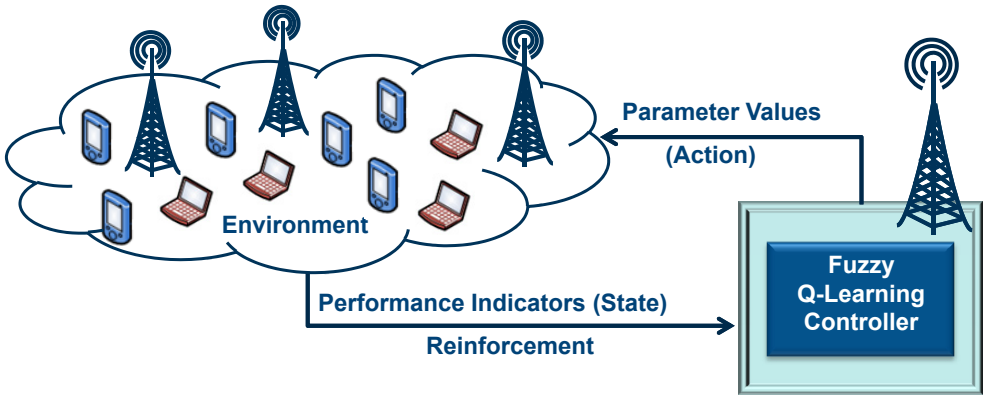

Figure 3.2: Distributed FQLC for CCO

3.7.1 States

States are used to describe different conditions of an environment. In cellular mobile networks, the performance is usually measured in terms of network KPIs (Key Performance Indicators). These KPIs represent various statistics of the BS and mobile user activity and can be used to represent the current operational state of a network.

For the CCO by antenna tilt adaptation, we consider the Spectral Efficiency (SE) statistics to measure the coverage and capacity performance of an antenna

tilt configuration. SE represents the transmitted information per unit bandwidth and therefore clearly indicate the spectrum usage efficiency. The higher the value of SE, higher the transmitted data rates achievable. In an operational network, spectral efficiency can be derived from the Signal to Interference plus Noise Ratio feedbacks from the mobile user equipments over a sufficiently large period of time so that the optimization area is adequately covered. Therefore, we define our input state vector as follows:

$$s_c = \begin{bmatrix} AT_c & SE_c^{center} & SE_c^{edge} \end{bmatrix} \tag{3.14}$$

where, AT_c is the current antenna tilt of the cell $c \in C$, the set of all cells in the network. SE_c^{center} and SE_c^{edge} are the central and edge spectral efficiencies respectively of cell c. As LTE utilizes a re-use 1 frequency allocation scheme among the cells, so the users at the cell edge experience significant inter-cell interference. Therefore, it is important to not just look at the peak SE, but its distribution within the cell. For this reason, we look at two distinct SE metrics to represent the state of a cell. SE_c^{center} is measured from the mean of the SINR distribution and SE_c^{edge} is measured from the lower 5-percentile of the SINR distribution, which represent the values at the edge of a cell. The exact procedure of converting the $SINR$ report of mobile users into SE measurements is described in the next chapter.

3.7.2 Actions and Policy

Actions are the possible steps that the FQLC can take in any state. As the optimization parameter under study is antenna tilt so we define the FQLC actions as the change to be applied to the current antenna tilt value. The optimization target for FQLC is now to learn an action policy that represent a mapping from states $s_c \in S$ to output actions $a_c \in A$, where A is the set of all possible actions for that state:

$$\pi_c : \quad s_c \rightarrow a_c \tag{3.15}$$

3.7.3 Membership Functions

Fuzzy logic is based on *Fuzzy Set Theory* introduced by Lotfi Zadeh [87]. Fuzzy Sets are sets whose elements have degrees of membership. Unlike traditional set theory, where the elements of a set have binary membership (an element either belongs or does not belong to the set), fuzzy set theory allows gradual assessment of the membership of an element to a set. This is achieved with the help of a *Membership Function* valued in the real unit interval [0,1]. A membership function represents the extent to which a statement is true generally known as

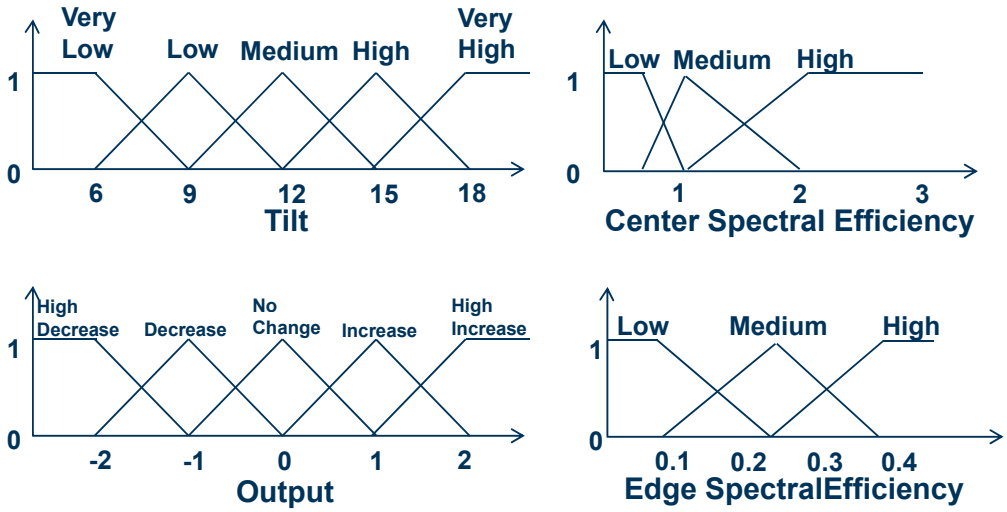

Figure 3.3: Membership Functions

the degree of truth. Therefore, a fuzzy set consists of two things [54]: a linguistic label or name of the fuzzy set, which describes some behavior of its contents and the associated membership function, which represent the degree of truth for each element.

For designing an FQLC, each variable of the input state vector s^n and output action a is discretized using a finite number of *Fuzzy Sets*. Each fuzzy set has its own label and associated membership function denoted by $\mu_{L^{n,m}}(s^n)$, where $L^{n,m}$ is the mth label defined over the nth state vector component s^n. A value of zero means state variable s^n does not belong to this label and one means it is fully a member of this label. The labels used for our state and output variables are shown in Fig.3.3. Antenna downtilt and output variables have five labels each, whereas SE^{center} and SE^{edge} have three labels each. For membership functions we use strict triangular membership functions, so that, for each value of the variable the sum of degrees of membership of all fuzzy labels is equal to 1.

$$\sum_{m=1}^{M} \mu\left(s_m^n\right) = 1 \tag{3.16}$$

where, M is the total number of labels define for each fuzzy variable. For example, for a downtilt value of 7 degrees the membership value is 0.667 for fuzzy label *Very Low* and 0.333 for fuzzy label *Low*.

3.7.4 Reinforcement Signal

In *RL* methods the environment provides the agent with the feedback about its actions in the form of *Reinforcement Signals (RS)*. These RS help the learning agent to characterize which actions produce better results in which states. As we describe the state of our CCO problem as a vector of current tilt, center SE and edge SE, we define a term *State Quality (SQ_c)* to describe the cumulative effect of both center and edge SE of a cell. SQ_c is defined as the weighted sum of center and edge *SE*. Higher SQ_c means better spectral efficiency distribution in the cell and thus higher achievable throughput.

$$SQ_c = SE_c^{center} + w \cdot SE_c^{edge} \tag{3.17}$$

where, w is a weighting factor and can be used as a policy control mechanism. Higher values of w tend to give more importance to the edge values compared to the center values. As edge values are normally significantly smaller than the center values, a value greater than 1 for w is also important to make the edge values comparable to the center values.

The RS is then calculated as the difference between the SQ_c of two consecutive states. If the FQLC transits from state $s_{c,t}$ to $s_{c,t+1}$, with state qualities $SQ_{c,t}$ and $SQ_{c,t+1}$ respectively, then the reward can be calculated as

$$r_{c,t+1} = SQ_{c,t+1} - SQ_{c,t} \tag{3.18}$$

3.7.5 Rule-Based Inference

In FQLC, the mapping between input states and output actions is defined by a rule-based *Fuzzy Inference System (FIS)* based on Fuzzy Labels. A typical FIS rule i in Fuzzy Q-Learning is initialized as [37]:

$$\begin{aligned} Rule\ i:\ IF\ s^1\ is\ L_i^{1,m}\ and\ s^{2,m}\ is\ L_i^{2,m}\ and\ \cdots\ s^n\ is\ L_i^{n,m} \\ THEN\ y = o_i^1\ with\ Q\left(L_i, o_i^1\right) = 0\ or \\ y = o_i^2\ with\ Q\left(L_i, o_i^2\right) = 0\ or \\ \cdots \\ y = o_i^k\ with\ Q\left(L_i, o_i^k\right) = 0 \end{aligned} \tag{3.19}$$

here, $L_i^{n,m}$ is a fuzzy label for a distinct fuzzy set m defined in the domain of the nth component s^n of the state vector and o_i^k is the kth output action for rule i, O being the set of K possible actions. The vector $L_i = \left[L_i^{1,m}, L_i^{2,m}, \cdots, L_i^{n,m}\right]$ is known as the *Modal Vector* of rule i and represents one state of the controller. $Q\left(L_i, o_i^k\right)$ is the q-value for state L_i and action o_i^k and is initialized to zero.

These Q-values are defined for each rule-action pair. Because of overlapping fuzzy sets, it is possible to have multiple activated rules for some states of the system. Therefore, it is important to differentiate between the Q-values for state-action pairs and rule-action pairs.

The total number of rules depends upon the dimension of the state vector as well as the number of fuzzy labels defined for each of these. Therefore, it is also a trade off factor between the accuracy of the system model and the speed of convergence of the learning process. As we have five labels for antenna downtilt and three each for SE^{center} and SE^{edge} variables, we have a total of 45 rules (states) in our FIS. The target of FQLC is now to find the best action for each rule that maximizes the overall reward. Each state has five possible actions with the exception of the states with downtilt in very low state which can only increase the downtilt or keep it constant and the states with very high downtilt which can only decrease the downtilt or keep it at the same level. Therefore, the downtilt can vary between 6 and 18 degrees only.

3.8 Fuzzy Q-Learning Algorithm

Similar to Q-learning the optimization target for FQL is also to find the optimal policy that maximizes the long term rewards for the learning agent. This is done by iteratively improving the estimated q-values for each state action pair as described in Eq. 3.13. However, the major difference is that in FQL the q-values are actually learned for the discrete states defined by the modal vector or rules of the FIS instead of the actual continuous domain state variables. Therefore, FQL algorithm is slightly different from the Q-learning algorithm.

The FQL algorithm starts with the identification of its current state based on the degree of truth of each FIS rule, defined by the product of the membership values of the corresponding fuzzy labels for each rule i:

$$\alpha_i(s) = \prod_{n=1}^{N} \mu_{L_i^{n,m}}(s^n) \tag{3.20}$$

Let P_s be the set of activated rules for state s, i.e. their degree of truth is positive. Then for each activated rule an action is chosen by the Exploration/Exploitation policy. Here we use an ϵ-greedy policy, where ϵ defines the trade off between exploration and exploitation in the algorithm. A value of 1 means no exploration only the actions with maximum q-values are always chosen and a value of 0 means totally random selection.

$$\begin{aligned} o_p &= \arg\max_{k \in K} Q\left(L_p, o_p^k\right) \quad with\ prob.\ \epsilon \\ o_p &= random_{k \in K}\left(o_p^k\right) \quad with\ prob.\ 1-\epsilon \end{aligned} \tag{3.21}$$

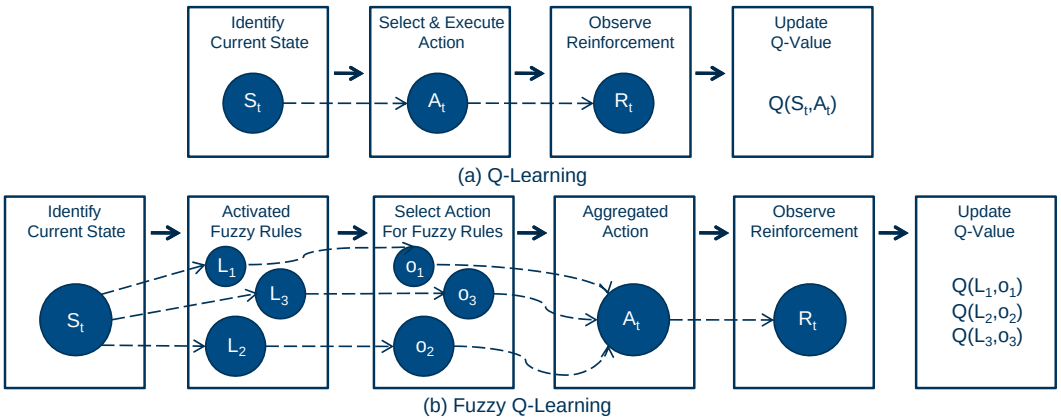

Figure 3.4: Comparison between Q-Learning and Fuzzy Q-Learning

The actual continuous domain action to be applied by the controller is the weighted average of all the proposed actions of the activated rules in state s:

$$a(s) = \frac{\sum_{p \in P_s} \alpha_p(s) \cdot o_p}{\sum_{p \in P_s} \alpha_p(s)} = \sum_{p \in P_s} \alpha_p(s) \cdot o_p \tag{3.22}$$

where, $\sum_{p \in P_s} \alpha_p(s) = 1$, because of strict triangular membership function that we use.

Q-value for the continuous state action pair $Q(x, a)$ is calculated as an interpolation of the q-values of the activate rules:

$$Q(s, a(s)) = \sum_{p \in P_s} \alpha_p(s) \cdot Q(L_p, o_p) \tag{3.23}$$

As a result of the execution of the action $a(s)$, the controller transits to another state s_{t+1} and receives a reward r. The value function of the next state, i.e. the expected reward from that state is defined as:

$$V(s_{t+1}) = \sum_{p \in P_{s_{t+1}}} \alpha_p(s_{t+1}) \max_{k \in K} Q\left(L_p, o_p^k\right) \tag{3.24}$$

This value function and the reward can be used to iteratively update the q-values for each rule of the FIS similar to Eq. 3.13:

$$Q_{t+1}(L_p, o_p) = Q_t(L_p, o_p) + \alpha_p(s_t)\, \beta \left\{r_{t+1} + \gamma V(s_{t+1}) - Q(s_t, a(s_t))\right\} \tag{3.25}$$

where β is the learning rate and α_p is the degree of truth of activated rule p, which is included here to highlight the contribution of each of the activated rule in the calculation of the actual output of the FQLC.

3.9 FQL Variants Studied for CCO

Cellular networks consist of a large number of cells, each with its own antenna tilt configuration. The optimal antenna tilt value for each cell depends on the traffic distribution within the cell as well as the antenna tilt values at the neighboring cells. Therefore, the environment for each cell consists of its own coverage area as well as the coverage area of other cells. In such a scenario, the optimal action policy of any cell not only depend on its own propagation condition but also on the actions of other cells. Therefore, antenna tilt optimization can be considered as a MARL problem, where each cell interacts with its environment by adapting its antenna tilt. In this thesis, we analyze different learning approaches for this problem based on the classification presented in section 3.3. These approaches are discussed in the following sections.

3.9.1 Concurrent vs Team Learning

In this multi-agent environment of coverage and capacity optimization, we need to optimize the antenna tilt of each and every cell in the network. To achieve this with RL, we can either have a learning agent for each cell (concurrent learning) or for the complete network (team learning). In this thesis, we analyze the performance of both of these approaches and the modeling used to realize these approaches is presented in the following sections.

3.9.1.1 Selfish Learning

The first basic approach is based on *Non-Cooperative Concurrent Learning*, which tries to optimize the performance of each learning agent (cell) without any regard to its effects on the performance of neighboring agents. Every cell in the network acts as a learning agent and has its own FQLC for antenna tilt adaptation as shown in Fig. 3.5. Each cell identifies its current state based on the input state vector as defined in Eq. 3.14. It uses three cell specific parameters: AT_c, the current antenna tilt of cell c, SE_c^{center}, the spectral efficiency experienced at the cell center and SE_c^{edge}, the spectral efficiency experienced at the cell edges calculated from the mobile measurements.

Each cell interacts with it environment by changing its antenna tilt. This change of antenna tilt modifies the *SINR* distribution in the cell and ultimately changes the *SE* values. As a result, the cell moves to another state and receives a reward r_c as defined in Eq. 3.18. The reward is based on the difference between the state qualities SQ_c of the two consecutive states, which is a weighted sum of SE_c^{center} and SE_c^{edge}. Based on this local reward, the cell independently updates its q-table. Therefore, each cell acts selfishly to optimize its own performance

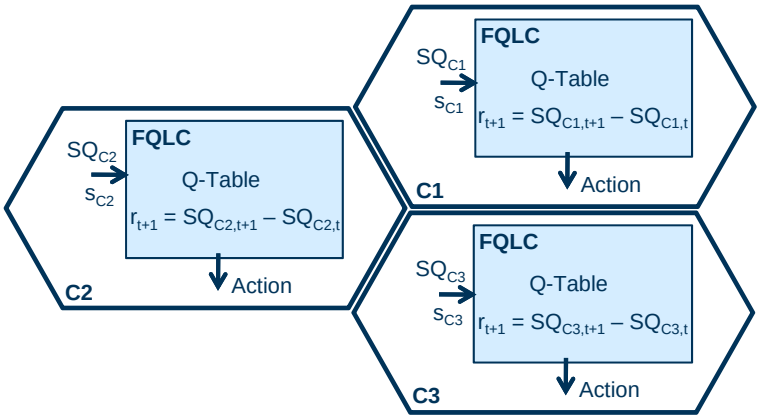

Figure 3.5: Selfish Learning

only.

The cells operate in a highly dynamic environment due to the fluctuations in the traffic demand and the propagation condition. Therefore, to allow the cells to effectively overcome the QoS degradation due to these fluctuations, the cells continuously try to learn the optimal policy as long as they remain operational. In simulations, this process is done sequentially over all cells, however, in a live network as all the cells act independently so they can learn in parallel. This makes this approach suitable for a distributed implementation even at the cell level.

3.9.1.2 Cooperative Learning

The second learning approach is based on *Cooperative Concurrent Learning*, where a learning agent is implemented for each cell but they cooperate with each other to improve the overall performance of the network and not just the performance of the cell itself. This cooperation is realized by defining a global reward and all cells use this same reward to update their individual q-tables.

Similar to the selfish learning, each cell identifies its current state by the input state vector s_c and takes an action according to its exploration-exploitation policy. This leads to change in the SQ_c values for all cells. These value are then communicated among the cells to calculate the average state quality SQ_avg of the complete network and the reward is calculated based on the difference of SQ_{avg} before and after the actions are applied by cells:

$$SQ_{avg,t} = \frac{\sum_{c=1}^{N} SQ_{c,t}}{N} \tag{3.26}$$

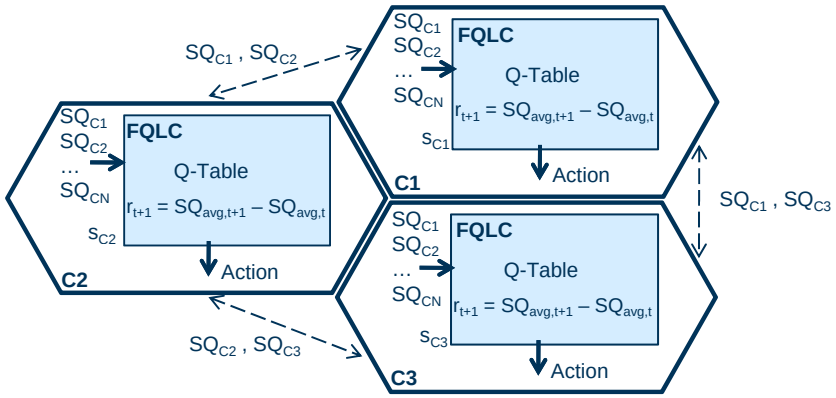

Figure 3.6: Cooperative Learning

$$r_{t+1} = SQ_{avg,t+1} - SQ_{avg,t} \tag{3.27}$$

here, N is the total number of cells in the network. As our simulation studies consider network scenarios with a maximum of two-tier neighborhood of eNBs, we can include all cells in Eq. 3.26. However, if bigger networks are considered, we can also calculate it over a subset of cells like *first* or *second-tier* neighborhood. This would ensure that the signaling effort is not increased too much.

Although, the cells share their SQ_c values but they still maintain their own q-tables of state action pairs as shown in Fig. 3.6. This helps the cells to identify the actions that lead to better performance of the network based on SQ_{avg} instead of SQ_c, but, still allows the cells to act independently based on their own q-tables for state action pairs. Additionally, this cooperation is also feasible in a distributed manner in LTE networks, as these networks support a direct interface (X2) between the neighboring eNBs.

3.9.1.3 Centralized Learning

The third learning approach is based on *Team Learning*, which uses a single learning agent to learn the optimal behavior for all cells in the network. As discussed in section 3.3.1.1 and 3.3.1.2, if we consider different cells to have different behaviors then the centralized learning agent needs to learn based on joint action of all the cells. Hence, the q-values to be maintained are of the form $Q(s, \mathbf{a})$, where $\mathbf{a}$ is a vector representing the combination of actions of all the agents $\mathbf{a} = [a_1, a_2, a_3, \cdots, a_n]^T$, n is the number of all the agents in the system and T represent the transpose of the vector. This joint action space grows rapidly and becomes extremely large even for a very small network making it infeasible to maintain such a q-table. Therefore, in this thesis we focus only on homogeneous

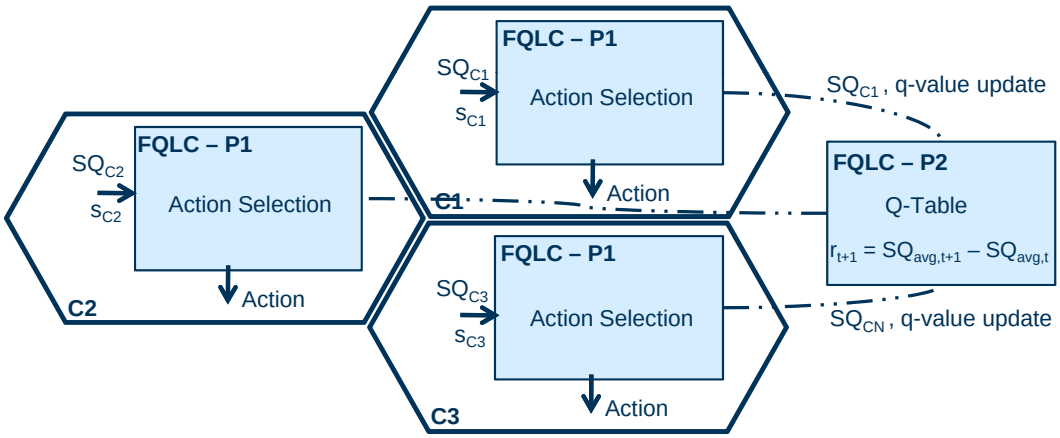

Figure 3.7: Centralized Learning

team learning, where the single learning agent is trying to learn same behavior for all cells. This means, we can maintain the q-values as $Q(s, a)$ where a is one of the possible actions, similar for all cells.

The FQLC is split into two parts as shown in Fig. 3.7. The current state identification and action selection according to the exploration-exploitation policy is still implemented at each cell as indicated by FQLC-P1 in Fig. 3.7. The cell specific definition of input state vector s_c helps to realize these procedures at each cell. However, the q-table as the representation of the learned knowledge is maintained centrally and shared by all cells in the network as indicated by FQLC-P2 in Fig. 3.7.

In this centralized learning, each cell still individually identifies its current state based on the input state vector. After that it takes an action using its exploration-exploitation policy and the shared q-table, which changes its environment and the SQ_c value. The cell then communicate the activated state-action pair as well as the updated SQ_c value to the central learning agent as shown in Fig. 3.7. Once this information is received from all cells, the learning agent can calculate the SQ_{avg} and global reward as given in Eq. 3.26 and 3.27 respectively. This global reward can then be utilized to update the q-values for all the activated state-action pairs. The global reward is easy to calculate in this scenario as the central learning agent already have an interface with all cells.

3.9.2 Learning Dynamics

Learning in a multi-agent environment is a difficult task because of the presence of co-adapting agents. The learning agents not only have to keep track of the changes

in the environment but also the changing behavior of other learning agents. This is especially important for the problem of antenna tilt optimization, where each change of antenna tilt not only modify the SE distribution within the cell but also in the neighboring cells. Therefore, for better characterization of the reward value as well as the impact of the previous action on the cell's performance, it is critical to identify how many cells update their antenna tilts simultaneously. In our studies, we consider three different learning strategies to control the amount of parallel antenna tilt updates in the network and are described in the following sections.

3.9.2.1 One Cell per Learning Snapshot

In this strategy we allow only one cell per network snapshot to execute the FQLC algorithm and thus update its antenna tilt as shown in Fig. 3.8-(a). The selection of the cell follows a uniform random distribution in order to allow all cells to update their antenna tilt. The SINR distribution within the cell depends not only on the current antenna tilt of the cell itself but also on the antenna tilt of the neighboring cells. Therefore, allowing only one cell to update its antenna tilt at each snapshot makes it easier to identify the impact of that change on the SINR distribution. This simplifies the learning process as the reinforcement signal the cell receives after its update can accurately measures the impact of the previous action. However, this strategy can severely limit the network adaptability in a dynamic environment as the network size increases.

3.9.2.2 All Cells per Learning Snapshot

In order to speed-up the learning process the second strategy allows all cells to update their antenna tilts in every snapshot as depicted in Fig. 3.8-(b). The change in the SINR distribution now depends on the actions of all cells in each snapshot. This somehow complicates the learning process as the reinforcement signal now received by a cell is the result of the collective actions of all cells and not just the action of the cell itself.

3.9.2.3 Cluster of Cells per Learning Snapshot

In order to combine the benefits of the above two strategies we propose to allow a cluster of cells to update their antenna tilts per network snapshot. The cluster should be formed such that no two direct neighbors are part of it as shown in Fig. 3.8-(c). This ensures that a relatively large number of cells can update their antenna tilts per snapshot and also the change in the environment can be easily characterized.

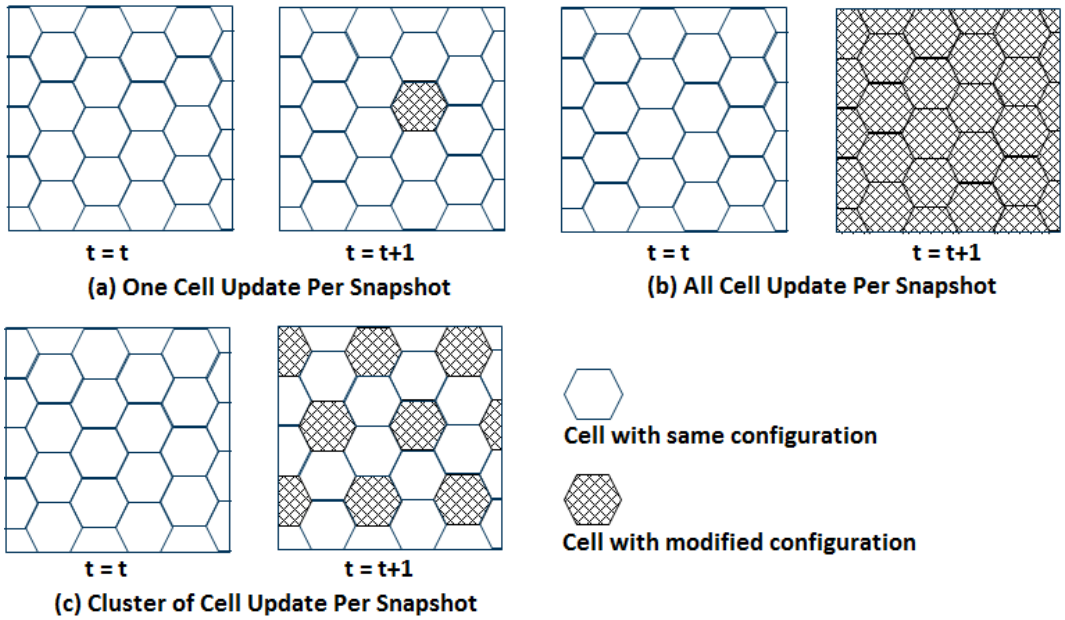

Figure 3.8: Parallel Learning Strategies Comparison

For this a simple clustering algorithm is used that starts by selecting one of the cells as the center cell as shown in Fig. 3.9. The selection follows a uniform random distribution. This center cell is than inserted into a list *To_Update_Green*. All the first tier neighbors of that center cell are than inserted into a second list *To_Ignore_Gray*. After that, starting from the center cell, we scan through all cells in ascending order of their *IDs*. If the cell is not in *To_Ignore_Gray* list it is inserted in the *To_Update_Green* list and all of its first tier neighbors are inserted in the *To_Ignore_Gray* list. Once we reach the cell with highest *ID*, we again start with the center cell and now scan the network in descending order of cell IDs and repeat the same procedure to update the *To_Update_Green* list. After we reach the cell with lowest *ID* we have all the cells in the *To_Update_Green* list, which makes our cluster of cells that can perform the FQLC algorithm simultaneously. Here, the common property of the cells in the cluster is that they are not immediate neighbors of each other. Fig. 3.9 shows examples of such a clustering when we start with cell 14 as the center cell. Cells with solid boundaries form our simulated network, whereas, the cells with dotted lines are the wraparound cells.

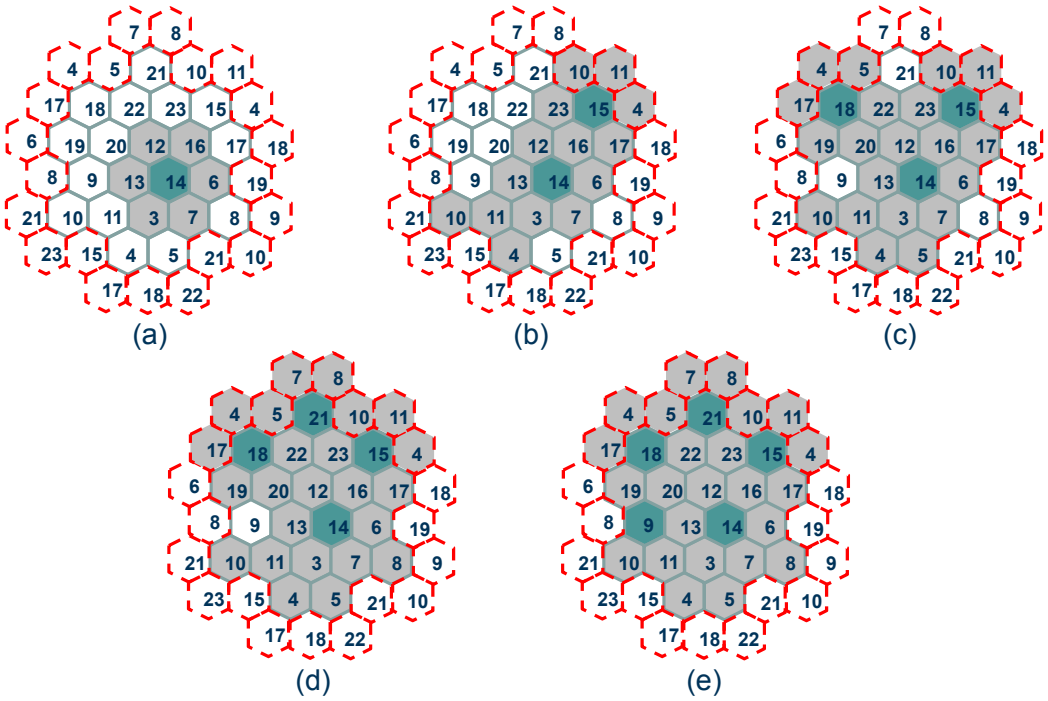

Figure 3.9: An Example of Clustering Algorithm Stages

3.10 Summary

This chapter introduces RL as the machine learning tool to optimize the cellular network operation. Instead of some pre-defined optimization rules the RL mechanisms allow cellular networks to learn from their interaction with the environment and adapt their behavior accordingly. This helps to fully automate the network operation and minimize the manual efforts required for network operation and maintenance.

Many network optimization problems like antenna tilt optimization involves continuous domain variables. This complicates the RL algorithms because they need to maintain tables of state-action pairs and with continuous variables it is unfeasible to build such a structure. Therefore, we particularly focus on FQL, which is a hybrid technique of Fuzzy Logic and Q-Learning. Fuzzy logic provides an effective abstraction level for continuous domain variables and Q-Learning provides the knowledge gathering and representation mechanism for the optimization problem.

Although, the basic FQL algorithm is already known, but our focus is on its application to cellular network optimization and the questions arising from it.

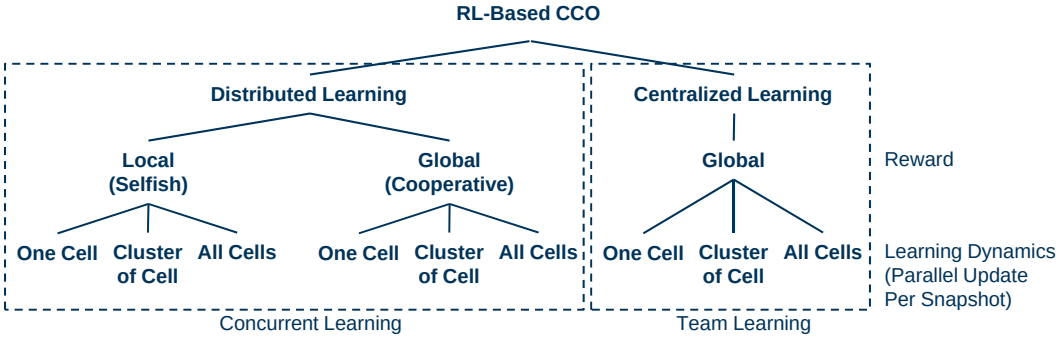

Figure 3.10: Analyzed FQL Variants

In this regard, we particularly focus on the questions of interactions among the different learning agents within the same network. We introduce several learning approaches that effectively learn the optimal action policy and try to overcome the problems of this multi-agent learning environment.

First, we use concurrent learning to develop two distributed learning approaches for CCO problem. Both of these approaches implement a learning agent for each cell in the network, which tries to learn the optimal behavior for the cell. But the approaches implement different reward functions to realize different level of cooperation among the learning agents. *Selfish Learning* uses local rewards and tries to reward the actions which produce better performance locally without any consideration to its effect on other cells. *Cooperative Learning* uses global reward and tries to learn actions that improve the overall performance of the whole network.

For comparison purposes we also develop a *Centralized Learning* approach based on team learning. This approach considers all cells to be part of one team and implements a single learning agent for all of them. All cells use and update the same central q-table to share their learned knowledge, which can help to speed-up the learning process as well.

We also look into the effect of different levels of co-adaptation in the network. In cellular networks the action of each cell like antenna tilt modification not only effect its own performance but also the performance of its neighboring cells. Therefore, we propose three different parallel update strategies i.e. only one cell update per learning snapshot, all cell update per learning snapshot and a cluster of cells update per learning snapshot. All three strategies vary in their ability to react to the network dynamics and the learning accuracy. One cell update strategy can accurately measure the affect of each update as the environment is only affected by one antenna tilt change but it is quite slow especially for large networks. All cell update strategy can quickly respond to network changes but it

experiences a complex learning scenario as the change in the environment at each learning snapshot is now the result of the combined actions of all cells. To combine the benefits of both these strategies we propose a cluster-based update strategy which tries to maximize the number of cells that can update their antenna tilt at each learning snapshot while ensuring that no two direct neighbors update their antenna tilt in same snapshot. This helps to speed-up the learning process without making the feedback procedure complicated for cells.

4

LTE Network Simulator - Radio System Aspects and Performance Metrics

This chapter introduces the network simulator used for our studies. First a brief introduction of the Long Term Evolution (LTE) is presented which is the radio access technology under study in this thesis. After that, the fundamental system model, assumptions and performance metrics are described that are used to design and evaluate our simulation studies.

4.1 Background and Objectives of LTE

Cellular mobile networks provide the flexibility of seamless communication while on the move. The first generation cellular mobile networks focused mainly on voice communication, but data communication became more and more important in the later generations. The ability to have data communication while moving led to the concept of anywhere and anytime Internet. This widespread usage of mobile Internet and other data extensive user applications led to the development of new core networks and RAT systems. Each newer generation of cellular mobile networks mainly targets to improve two factors; mobility and data rate [53] as shown in Fig. 4.1 .

Keeping in view the user traffic predictions and operators' requirements of a simple and inexpensive cellular mobile network, 3GPP started to define the requirements of LTE in 2005. These requirements can be summarized as follows [65]:

- reduced delays, in terms of both connection establishment and transmission latency
- increased user data rates

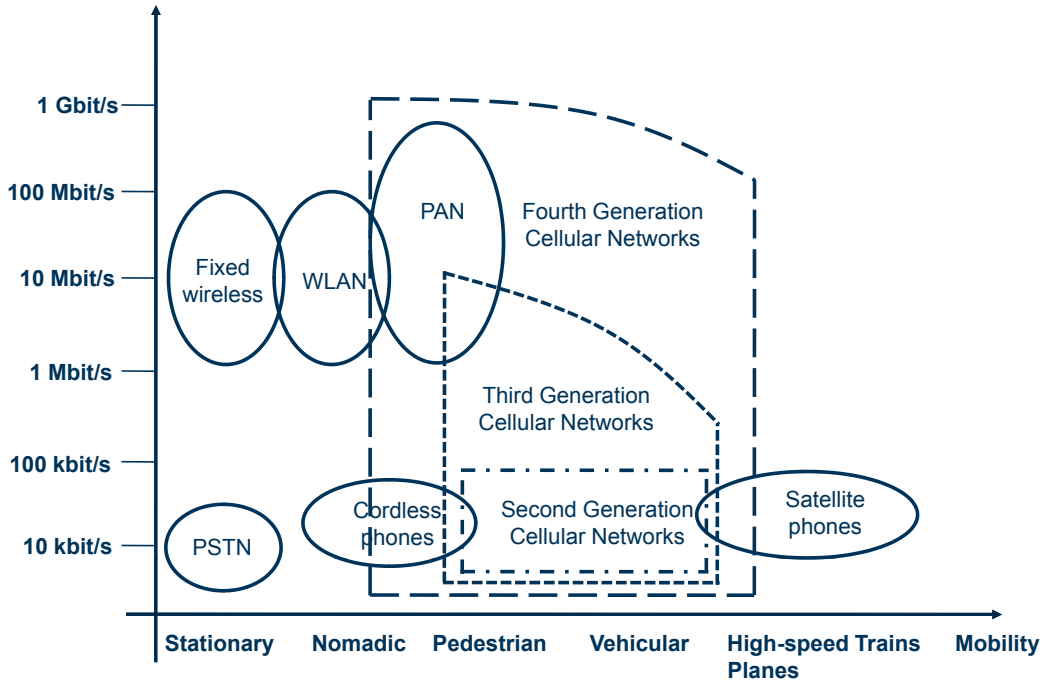

Figure 4.1: Data rate vs mobility for various networks [53]

- increased cell-edge data rates, for uniformity of service provision
- reduced cost-per-bit, implying better spectral efficiency
- greater flexibility of spectrum usage, in both new and pre-existing bands
- simplified network architectures
- seamless mobility, including between different RATs
- reasonable power consumption for the mobile terminal

LTE system design served as the last step in this evolution of cellular mobile networks towards *4th Generation (4G)* systems. It provides significant improvement in terms of system performance compared to HSDPA/HSUPA Release 6 which was the state of art at the time of LTE development. The key system performance parameters of LTE Release 8 and their comparison with Release 6 are given in Table 4.1.

4.2 LTE System Simulator

For the analysis of our proposed *FQLC* approaches we perform simulation studies of an LTE network. The LTE system simulator we use for our evaluations is based on the software libraries provided by *Alcatel-Lucent Bell Labs, Germany* and the *Institute of Communication Networks and Computer Engineering, University of Stuttgart, Germany* [66]. The libraries provides the means to simulate a 3GPP compliant LTE radio network for downlink evaluations. They were extended to allow each cell to act as an FQLC agent and to generate the required state and reinforcement values. All the different learning strategies were also implemented to observe their affect on CCO.

The basic features of the simulator for radio network modeling are explained in the following sections.

4.2.1 Cellular Deployment

The basic cellular network used for our evaluations is the 3GPP "Case-1" with 7 eNBs (LTE base station) as shown in Fig. 4.2-(a). 3GPP *Case-1* refers to a interference limited scenario with an inter-site distance of 500m [2], [3]. Each eNB operates 3 cells, where the term cell refers to the coverage area of a radio transceiver (TRx). The eNBs are deployed in a regular hexagonal structure, such that one center eNB is surrounded by one layer of eNBs.

Table 4.1: LTE System Performance Parameters

		Absolute Requirement	Release 6 (for comparison)	Comments
Downlink	Peak Transmission Rate	>100 Mbps	14.4 Mbps	LTE in 20 MHz FDD, 2x2 spatial multiplexing. Reference: HSDPA in 5 MHz FDD, single antenna transmission
	Peak Spectral Efficiency	> 5 bps/Hz	> 3 bps/Hz	
	Average Cell Spectral Efficiency	> 1.6 - 2.1 bps/Hz	0.53 bps/Hz	LTE: 2x2 spatial multiplexing, Interference Rejection Combining (IRC) receiver. Reference: HSDPA rake receiver, 2 receive antennas
	Cell edge spectral efficiency	> 0.04 - 0.06 bps/Hz/user	0.02 bps/Hz/user	As above, 10 users assumed per cell
	Broadcast spectral efficiency	> 1 bps/Hz	N/A	Dedicated carrier for broadcast mode
Uplink	Peak Transmission Rate	> 50 Mbps	> 11 Mbps	LTE in 20 MHz FDD, single antenna transmission. Reference: HSDPA in 5 MHz FDD, single antenna transmission
	Peak Spectral Efficiency	> 2.5 bps/Hz	> 2 bps/Hz	
	Average Cell Spectral Efficiency	> 0.66-1 bps/Hz/Cell	0.33 bps/Hz/Cell	LTE: single antenna transmission, IRC receiver. Reference: HSUPA rake receiver, 2 receive antennas
	Cell edge spectral efficiency	> 0.02-0.03 bps/Hz/user	0.01 bps/Hz/user	As above, 10 users assumed per cell
System	User plane latency (2 way radio delay)	< 10 ms		LTE target approx. one fifth of reference
	Connection setup latency	< 100 ms		Idle state → Active state
	Operating bandwidth	1.4-20 MHz	5 MHz	

To make our simulation scenario more realistic, a bigger and irregular cell deployment is also considered in our evaluations. This scenario consists of 19 eNBs (57 cells) such that the center eNB is surrounded by two layers of eNBs. In real environments, eNBs can hardly be deployed in a regular hexagonal structure due to limited number of feasible site locations. Therefore, the eNBs in this scenario are displaced from their location of a perfect hexagonal deployment with an inter-site distance of 500m. The displacement follows a uniform random distribution between 0 and 200 meters in a random direction. The resultant cell structure is shown in Fig. 4.2-(b).

In both scenarios, a wraparound is also implemented for better interference calculation and avoiding the border effects. This effectively replicates the simulation area six times around the original area such that the original area is in the center and the replicates form a layer around it.

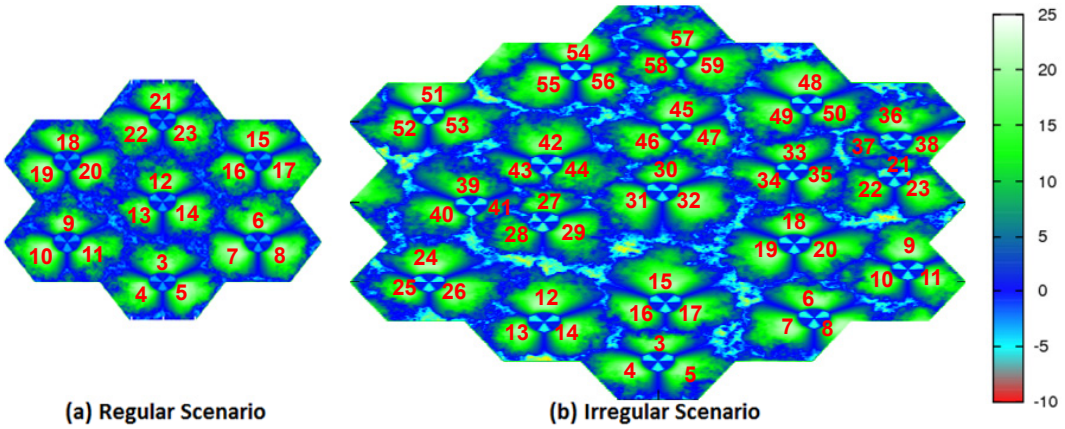

Figure 4.2: Simulation Scenario

4.2.2 Propagation Model

In wireless communications, while the transmit power is under the control of the transmitter, the received power is affected by a number of environmental factors and device characteristics. For the accuracy of the simulation studies, it is vital to have accurate propagation models for the calculation of received power as all other metrics of system performance like SINR, capacity, coverage, etc are calculated from it. Typically, the relation between transmit and received power is expressed as:

$$P_{Rx} = P_{Tx} - P_L - SF + G_{Ant} + G_{Dir} \tag{4.1}$$

where P_{Rx} is the *received power* in dBm, P_{Tx} is the *transmitted power* in dBm, P_L is the *pathloss* in dB, SF is the *slow fading* also known as *shadow fading* or *large scale fading* in dB, G_{Ant} is the *maximum antenna gain* in dBi and G_{Dir} is the *directional gain* of antenna depending upon the position of the receiver with respect to the transmitting antenna calculated in dB. In this equation *fast fading* is ignored. In the following sections, the calculation of each of the terms in Eq. 4.1 is explained.

4.2.3 Pathloss

Pathloss (PL) defines the degradation in power with respect to the distance between the transmitter and the receiver. For LTE simulation studies, 3GPP defines PL as [3]:

$$P_L = 128.1 + 37.6 \cdot x \cdot \log(d) \tag{4.2}$$

where, d is the distance between the transmitter and the receiver in meters, the carrier frequency is assumed to be 2GHz and BS height above the average rooftop is assumed to be 15 meters.

4.2.4 Shadow Fading

In wireless communications, fading is the variation in the attenuation, experienced by a signal over a propagation medium. It may vary with time, the radio frequency, or the geographical location of the transmitter and receiver and is typically modeled as a random process.

Shadow fading is caused by large obstacles such as buildings or hills that obscure the line-of-sight signal between the transmitter and the receiver. The amplitude variations caused by shadow fading is often modeled as log-normal distribution with an environment dependent standard deviation [1] [62]. Shadow fading experience a slow variation, therefore, for moving users, the successive shadow fading values are correlated. The normalized autocorrelation between two successive values is given as [38]:

$$R\left(\Delta d\right) = e^{-\frac{|\Delta d|}{d_{cor}} \ln 2} \tag{4.3}$$

where, Δd is the distance between the two positions of the mobile user and d_{cor} is the environment dependent decorrelation distance in meters.

For example, if the shadow fading value at position D_1 is SF_1. The next position is D_2, which is Δd meters away from D_1. Then, shadow fading value SF_2 at D_2 is normally distributed with mean $R\left(\Delta d\right) \cdot SF_1$ and variance $\left(1 - R\left(\Delta d\right)^2\right) \sigma^2$, where σ is the standard deviation.

4.2.5 Antenna Gain

In the simulator, directional antennas are utilized instead of omni-directional antennas. Directional antennas allow transmission of radio signals in a particular direction with higher gain and therefore, can reduce the inter-cell interference in other directions. The maximum antenna gain, is the gain of a directional antenna over the isotropic antenna and is included in Eq. 4.1 for received power calculation as G_{Ant}. The directional gain G_{Dir}, refers to the relative strength of the radiated power depending on the vertical and horizontal location of the mobile user with respect to the location of the BS antenna.

In this thesis, a 3D antenna model defined by 3GPP [2] is utilized, which includes both horizontal and vertical antenna radiation pattern. The gain at any position is calculated as the sum of both radiation patterns:

$$G_{Dir} = G\left(\varphi, \theta\right) = -\min\left\{-\left[G_H\left(\varphi\right) + G_V\left(\theta\right)\right], FBR\right\} \tag{4.4}$$

where, $G_H(\varphi)$ is the horizontal antenna gain in dB, $G_V(\theta)$ is the vertical antenna gain in dB, φ and θ are the angles between the antenna and the mobile user in horizontal and vertical direction respectively and FBR is the *front to back ratio*, which defines the ratio between the power of an antenna in the main lobe and the back lobe. It is also referred as backward attenuation.

Both horizontal and vertical radiation patterns are normalized to the maximum antenna gain i.e. the maximum value $G_H(\varphi)$ and $G_V(\theta)$ can have is 0 dB. Therefore, the directional gain decreases as the mobile moves away from the main lobe, .

4.2.5.1 Horizontal Pattern

In 3GPP specifications, the horizontal antenna pattern is models as [2]:

$$G_H(\varphi) = -\left[12\left(\frac{\varphi}{\varphi_{3dB}}\right)^2, FBR_H\right] \tag{4.5}$$

where, φ, $-180^\circ \leq \varphi \leq 180^\circ$, is the horizontal angle between the main beam direction (0° direction) and the mobile user, φ_{3dB} is the horizontal half power beamwidth, which indicates the angle between the half power points (-3dB) on the main beam of the antenna pattern and FBR_H is the horizontal front to back power ratio in dB. A plot of an example horizontal pattern with $\varphi_{3dB} = 70^\circ$ and an $FBR_H = 25dB$ is shown in Fig. 4.3.

4.2.5.2 Vertical Pattern

Similarly, vertical antenna pattern in 3GPP specification is defined as [2]:

$$G_V(\theta) = -\min\left[12\left(\frac{\theta - \theta_{etilt}}{\theta_{3dB}}\right)^2, SLA_V\right] \tag{4.6}$$

where, θ, $-90^\circ \leq \theta \leq 90^\circ$, is the elevation angle with respect to the horizontal plane ($\theta = 0^\circ$ means horizontal direction), θ_{3dB} is the half power beamwidth, θ_{etilt} is the electrical tilt angle and SLA_V is the side lobe attenuation. Sample plots of vertical pattern with $\theta_{3dB} = 10^\circ$, an $SLA_V = 20dB$ and θ_{etilt} of 0° and 15° are shown in Fig. 4.4.

4.2.6 Signal to Interference plus Noise Ratio

A mobile user not only receive signals from its serving cell but also from other cells in the network using the same frequencies. The received power at a particular channel is the combination of the signals from the serving cell and other interfering

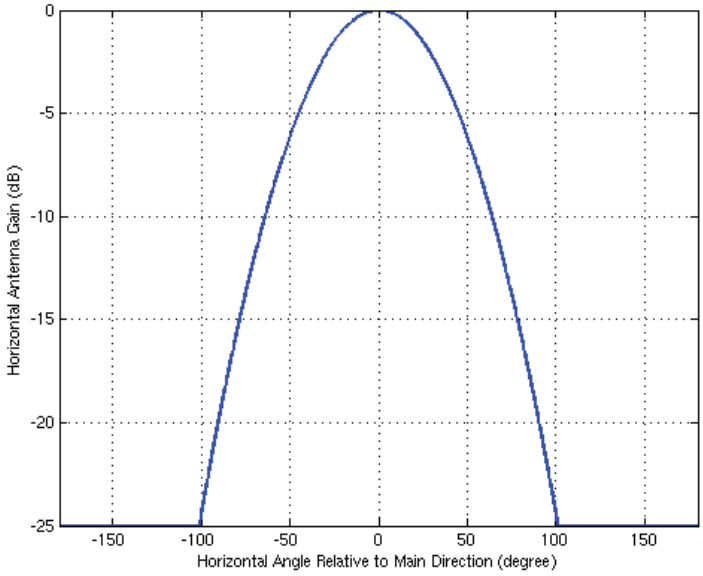

Figure 4.3: Horizontal Antenna Pattern

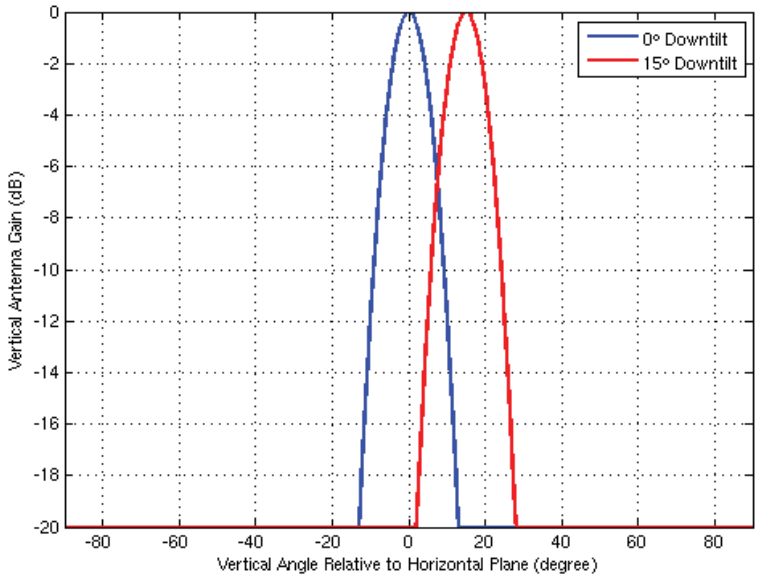

Figure 4.4: Vertical Antenna Pattern with Different Tilt Angles

cells. The quality of the received signal depends on the ratio between the power level of the desired signal and the interfering signals. This quality is generally

measured in terms of *Signal to Interference plus Noise Ratio (SINR)* as:

$$SINR\,(dB) = 10 \cdot \log_{10}\left(\frac{P_{Serving}}{\sum_{j \in AllCells\ and\ j \neq Serving} P_j + P_{noise}}\right) \tag{4.7}$$

where, $P_{Serving}$ is the received power from the serving cell in watt, P_j is the received signal power of the jth interfering cell in watt and P_{noise} is the thermal noise in watt. $P_{Serving}$ and P_j can be calculated from Eq. 4.1. Whereas, the thermal noise can be calculated in dBm as [1]:

$$P_{noise}\,(dBm) = -174 + 10 \cdot \log_{10}\,(N_{BW}) + NF_{R_x} \tag{4.8}$$

where, -174 is the thermal noise density in dBm/Hz, N_{BW} is the noise bandwidth in Hz and NF_{R_x} is the noise figure of the receiver (mobile)in dB.

4.2.7 User Throughput

User throughput defines the maximum achievable transmission date rate over a given channel. It is expressed in bits per second (bps) and according to the Shannon formula can be calculated as:

$$Th\,(bps) = BW \times \log_2\,(1 + SINR) \tag{4.9}$$

where BW is the bandwidth available for each mobile in Hz. As *resource fair scheduler* is utilized, which divides the available resources equally among the mobiles, the bandwidth per mobile can be calculated as:

$$BW_{mobile} = \frac{BW_{system}}{Mobile_{cell}} \tag{4.10}$$

where, BW_{system} is the available system bandwidth for each cell and $Mobile_{cell}$ is the number of mobile users in a cell

However, the throughput given in Eq. 4.9 is the upper bound with perfect modulation and coding schemes. Considering the realistic modulation and coding schemes used in LTE an approximate function can also be developed for realistic throughput calculations [61] as:

$$P(x) = \begin{cases} -0.0001x^3 + 0.0074x^2 + 0.1397x + 0.6218 & : \quad -7.04 \leq x \leq 20.2 \\ 0 & : \quad x < -7.04 \\ P\,(20.2) & : \quad x > 20.2 \end{cases} \tag{4.11}$$

where, x is the SINR in dB.

Eq. 4.11 gives the throughput in terms of bits/symbol. In LTE downlink with 15 kHz channel bandwidth, the symbol rate is 14000 symbols/sec. Using this symbol rate, the throughput per Hz i.e. the *Spectral Efficiency* can be calculated as

$$SE\left(SINR_{dB}\right) = P\left(SINR_{dB}\right)\frac{bits}{symbol} \times \frac{14000 symbol/sec}{15kHz} \tag{4.12}$$

The $SE\left(SINR_{dB}\right)$ gives the throughput for 1 Hz. The throughput based on the complete allocated bandwidth for the mobile user can be calculated as

$$Th\left(bps\right) = \frac{BW_{system}}{Mobile_{cell}} \times SE\left(SINR_{dB}\right) \tag{4.13}$$

Finally, as LTE can be deployed with different system bandwidths, therefore, it is better to look at throughput values normalized to system bandwidth for comparison between different network settings.

$$Th\left(bps/Hz\right) = \frac{1}{Mobile_{cell}} \times SE\left(SINR_{dB}\right) \tag{4.14}$$

4.2.8 Simulation Model

The evaluation follows a snapshot-based analysis, where each snapshot is taken after 200 msec of simulation time. The simulation starts with the deployment of our test scenario as explained in Section 4.2.1. To generate user traffic in the network and to get $SINR$ measurements, mobiles are also deployed. The deployment of mobiles follows a random distribution, such that we have 10 users per cell on average. Therefore, we have a total of 210 mobiles in the small uniform scenario and 570 mobiles in the larger non-uniform scenario.

At each snapshot, mobiles update their position depending on their mobility, which follows a random walk model at a velocity of 30 km/h. Mobiles can then measure the received power of the pilot signal from all the cells and can connect to the cell with the strongest received pilot signal at the new location. If the cell with strongest pilot signal is different from the serving cell in the previous cell, the mobile performs a handover (HO) to the cell with strongest pilot power. However, a perfect HO procedure is considered in this study without any delays or timers. Therefore, mobiles are always connected to the best server during the complete simulation.

After identifying the best serving cell, the $SINR$ on the serving cell's pilot signal is calculated and reported back to the serving cell. This way, a total of 210 $SINR$ measurements are generated and reported back to the cells at each snapshot. These measurements are gathered over a period of 200 sec to generate

the statistics of $SINR$ distribution. Therefore, a total of 210,000 measurements are generated in our small scenario and 570,000 measurements are generated in our larger scenario for one evaluation of the network state. The number of measurements that each cell receives depends upon the mobility of the mobiles.

The FQLC decision periodicity is also set to 200 sec i.e. the FQLC of each cell estimates its current state on the statistics of last 200 sec. The FQLC then calculates its action in terms of change to be applied to the antenna tilt. This change of antenna tilt modifies the environment and hence the $SINR$ distribution. If the $SINR$ distribution improves then the action is rewarded with a positive feedback otherwise it is punished with a negative feedback. This process continue for 4000 FQLC decision iterations.

The major simulation parameters are given in Table 4.2.

4.3 Performance Metrics

This section describes the *Performance Metrics* used in our simulation studies to compare the performance of different network configurations. These metrics are calculated from the *Cumulative Distribution Function (CDF)*, which describes the statistical distribution of a variable. For any given value x of a random variable X, the CDF indicates the probability that the value of X is smaller than or equal to x. Therefore, CDF can only have values between 0 and 1.

4.3.1 User Geometry

User Geometry refers to the serving cell's pilot SINR at the mobile. As explained in Section 4.2.8, each FQLC estimate the network state on the basis of statistics of 200 sec. During this duration a total of 210,000 SINR measurements are generated in the small scenario and 570,000 measurements in the large scenario. Therefore, on an average each cell receives around 10,000 SINR measurements during this duration of 200 secs. Based on these measurements each cell can then generate a CDF of the SINR in its coverage area. If CDF of two configurations are plotted, then the CDF whose plot is at right has higher SINR values than the other.

4.3.2 User Spectral Efficiency

User spectral efficiency can be directly calculated from SINR values as explained in Eq. 4.14. Therefore, similar to the *user geometry* CDF, a *user spectral efficiency* CDF can also be generated, which describes the statistical distribution of achievable data rates per unit bandwidth.

User spectral efficiency distribution tells the overall distribution of achievable data rates throughout the cell. However, it is important to clearly identify the performance at cell center and cell edge. This is particularly critical in *re-use 1* systems like *LTE*, which experience significant inter-cell interference at cell edges. Therefore, we define two metrics SE_c^{center} and SE_c^{edge} which represent the achievable spectral efficiency at cell center and cell edge respectively. These two metrics also form the input state of our FQLC along with the current tilt of the cell as explained in section 3.7.1. The calculation of these two metrics is explained in the following sections.

4.3.3 Cell Center Spectral Efficiency

Cell center defines the coverage area close to the cell's antenna, which experience less inter-cell interference and can achieve higher spectral efficiency values. Therefore, Cell Center Spectral Efficiency (SE_c^{center}) is defined as the mean of the spectral efficiency distribution.

4.3.4 Cell Edge Spectral Efficiency

Cell edges experience significant inter-cell interference, which reduces the SINR and thus the achievable spectral efficiency. Therefore, Cell Edge Spectral Efficiency (SE_c^{edge}) is defined as the lower 5% quantile of the spectral efficiency distribution. In other words, it is the spectral efficiency value where the cummulative probability is equal to 0.05.

4.4 Summary

To evaluate the performance of our proposed learning strategies we perform simulation studies with an LTE system level simulator. The basic components of this simulator are explained in this chapter. This includes the details of our cellular radio network as well as the performance metrics used to quantify the performance gains of each learning strategy.

The cellular radio network model includes our simulation scenarios, radio signal propagation modeling and our snapshot based simulation model. Especially, we focus on 3D antenna model to instead of 2D antenna model for better evaluation of antenna tilt effects.

The performance metrics include Cell Center Spectral Efficiency and Cell Edge Spectral Efficiency. LTE is a re-use 1 system, which uses same frequency resources in all cells. This could significantly increase the inter-cell interference especially at the edges. Therefore, for an efficient optimization algorithm, it is

important to also look at the cell edge performance improvement and not just the average performance gains.

The simulation studies and the performance results will be discussed in the next chapter.

Table 4.2: Simulation Parameters

Scenario	
eNBs Regular Scenario	7
eNBs Irregular Scenario	19
Cells per eNB	3
Number of mobiles	10 per cell on average
Learning Snapshot Frequency	200 sec
eNB Parameters	
Max. Tx Power	46 dBm
Inter-Site Distance (ISD)	500 m
Antenna Height	32 m
Antenna Max. Gain	15 dBi
Random Displacement (Irregular Scenario)	
Distance	0-200 meters uniform random distribution
Direction	$0° - 360°$ uniform random distribution
Horizontal Antenna Pattern	
Half Power Beamwidth	70°
Backward Attenuation	25 dB
Horizontal Gain	$G_H(\varphi) = -\left[12\left(\frac{\varphi}{\varphi_{3dB}}\right)^2, FBR_H\right]$
Vertical Antenna Pattern	
Half Power Beamwidth	10°
Backward Attenuation	20 dB
Vertical Gain	$G_V(\theta) = -\min\left[12\left(\frac{\theta-\theta_{etilt}}{\theta_{3dB}}\right)^2, SLA_V\right]$
Channel	
Pathloss	$A + B\ log_{10}(max(dkm\ ,\ 0.035))$
Pathloss A	128.1
Pathloss B	37.6
Shadow Fading Decorrelation Distance	50 m
Shadow Fading Standard Deviation	8 dB
Bandwith	10 MHz
Mobile	
Receiver Noise	8 dB
Max. Tx Power	39 dBm
Antenna Height	1.5 m
Antenna Gain	2 dBi
Speed	30 kmph
Movement	Random Walk
SINR Report Frequency	200 msec

5 Simulation Results

This chapter describes the results of our simulation studies. First, the reference systems used for benchmarking purposes are described. After that the performance results of different learning strategies in our *regular scenario* are discussed. Finally, the results for our *irregular scenario* are also described.

5.1 Reference System

To compare the performance gains of our proposed antenna tilt optimization schemes, we have also defined two reference systems. One reference system is based on a network wide static antenna tilt optimization and the other based on a similar approach of FQL for dynamic optimization of antenna tilt. Both of these reference systems are explained in the following sections.

5.1.1 Static Network Wide Optimization

For a given network setting the optimal antenna tilt value of one cell depends on the antenna tilt values of its neighbors. Therefore, antenna tilt optimization problem actually translates into a problem of finding the best combination of antenna tilts among the cells. Depending on the number of cells and the possible values the antenna tilt can have, the number of possible combinations can be extremely large making exhaustive search impossible.

The cellular deployment for our simulation studies is explained in Section 4.2.1. For our *irregular scenario* it is unfeasible to find the global optimal because of the enormous combinations of antenna tilts. But our *regular scenario* consists of ideal hexagonal cells with equal inter-site distance and uniform traffic distribution. The optimal antenna tilt angle for such a scenario can be considered to be same for all cells and can be calculated by evaluating the performance of all possible antenna tilt values. For this purpose, the performance of all integer values of antenna tilt between 0 and 18 was calculated, where the value of antenna tilt for all cells was fixed to one integer value for one simulation run.

The results for this global search for a network wide tilt are shown in Fig. 5.1. Figure 5.1-(a) shows the *State Quality (SQ)* averaged over the complete network of 21 cells for different values of antenna tilt. The optimal antenna tilt angle in this case corresponds to 15 degrees. The SQ as defined in Eq. 3.17 is the weighted sum of *edge spectral efficiency* $\left(SE^{edge}\right)$ and *center spectral efficiency* (SE^{center}), the results for these components are also presented in Fig. 5.1-(b) and Fig. 5.1-(c) respectively. The graphs show that there is a slight difference in the optimal antenna tilt value for SE^{edge} and SE^{center}, while, 14 degrees is optimal for the edge performance, 15 degrees is optimal in terms of cell center performance. As we utilize SQ to calculate the reinforcement signals, we use a network wide antenna tilt configuration of 15 degrees, which is optimal for SQ as our "reference system" in the following results for our regular scenario.

5.1.2 Dynamic Network Optimization

Apart from a static network wide optimal configuration, we also compare our results with a related scheme based on *FQL*. Parallel to our studies Razavi et al. also proposed an FQL based antenna tilt optimization [63]. They introduced the concept of *State Strength* and *Action Strength* to quantify the contribution of each fuzzy rule in calculating the final action to be applied by the FQLC. This helps to distribute the reward according to their individual contributions.

In FQL, the learning agent's (cell in our case) state is represented by fuzzy rules. These rules are based on overlapping fuzzy membership functions associated with different fuzzy labels as also shown in Fig. 3.3. Because of these

overlapping membership functions and graded membership of continuous domain values to fuzzy labels, multiple fuzzy rules can be active (having positive degree of truth) for one continuous domain state of the agent. All of these activated rules select an action according to their exploration and exploitation policy and the final output action of the *FQLC* depends on all of these selected actions. As a result of the execution of the final action the change in the environment and thus the received reward indirectly depends on the individual actions selected by each activated rule. Therefore, for proper learning and ranking of the state-action pairs it is important to quantify the contribution of each activated rule in getting that particular reward.

In traditional FQL, this contribution of each rule is highlighted by including the degree of truth of the activated rule in the q-value update equation as given

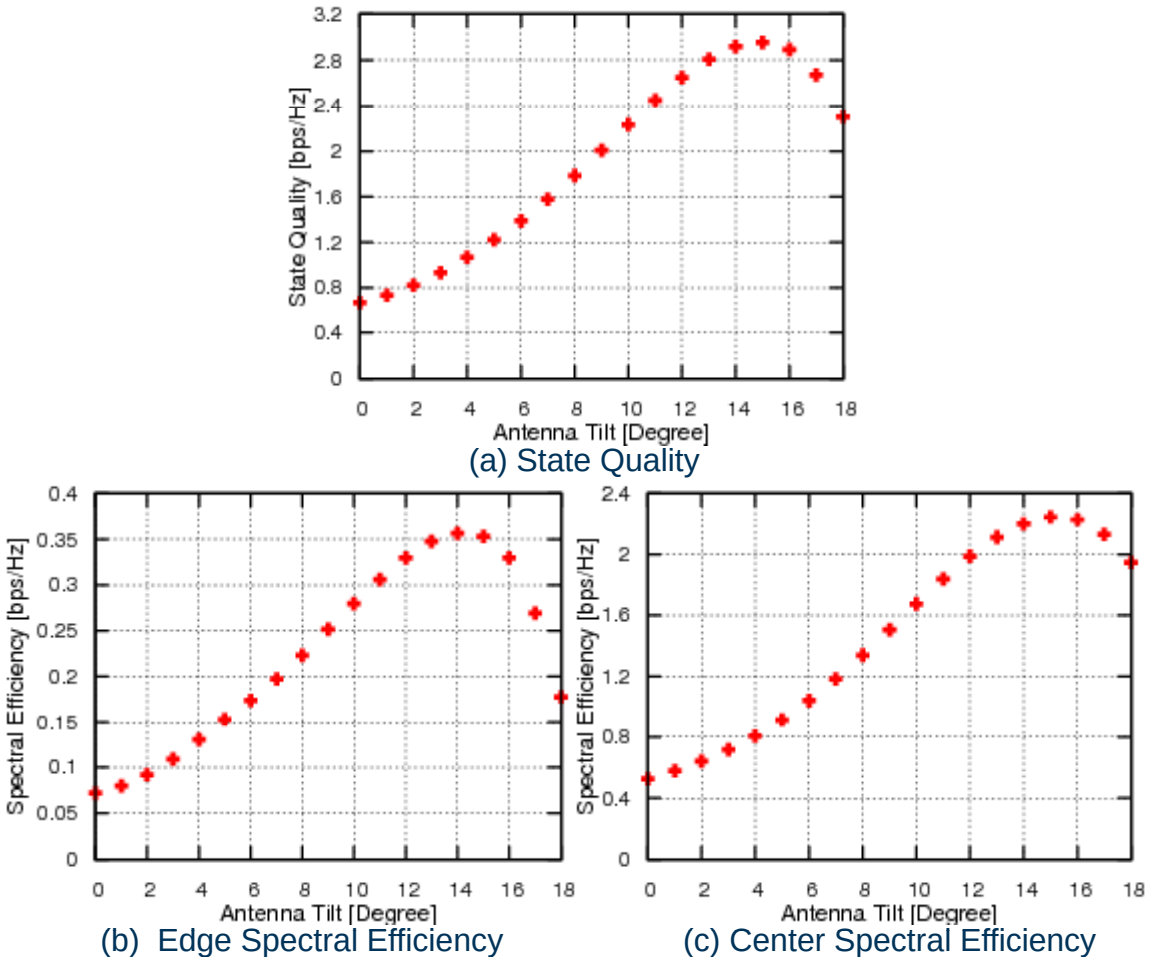

Figure 5.1: Reference System Performance

in Eq. 3.25. The equation is also given below

$$Q_{t+1}\left(L_p, o_p\right) = Q_t\left(L_p, o_p\right) + \alpha_p\left(s_t\right)\beta\left\{r_{t+1} + \gamma V\left(s_{t+1}\right) - Q\left(s_t, a\left(s_t\right)\right)\right\}$$

where $\alpha_p\left(s_t\right)$ is the degree of truth of the activated fuzzy rule p at time t, L_p is the modal vector of the activated rule p and represents the fuzzy state of the agent. As fuzzy states are represented by one and only one fuzzy rule, in the following text we use fuzzy rule to indicate the fuzzy state as well. The inclusion of degree of truth in the above equation helps to scale the update factor by the degree of truth of the activated fuzzy rule in the previous state.

Razavi et al. extended this concept by introducing the idea of *State Strength* (SS) and *Action Strength* (AS), which describe the relative strength of each activated fuzzy rule (state) among all the activated rules and the relative strength of the selected action among all the selected actions respectively. They are defined as follows:

$$\forall p \in P \quad SS\left(p\right) \equiv \frac{f_p}{\sum_{\acute{p}} f_{\acute{p}}} \tag{5.1}$$

$$\forall a \in A \quad AS\left(a\right) \equiv \frac{\sum_p f_p \times Selected\left(p, a\right)}{\sum_{\acute{p}} f_{\acute{p}}} \tag{5.2}$$

$$Selected(p, a) = \begin{cases} 1 & : \quad if\ a\ is\ selected\ for\ fuzzy\ rule\ p \\ 0 & : \quad Otherwise \end{cases} \tag{5.3}$$

where f_p is the firing strength (degree of truth) of the fuzzy rule p, SS is the state strength of the fuzzy rule, AS is the action strength of the selected action and $Selected(p, a)$ is a binary parameter indicating if an action a is selected for fuzzy rule p or not. If all the activated rules select distinct actions, then action strength is equal to the state strength of that particular rule.

Razavi et al. propose to use these *State Strength* and *Action Strength* values to modify the learning rate in the q-value update equation in order to incorporate the effect of the relative strength of each activated rule and selected action as follows:

$$\beta_p = \beta \times SS(p) \times AS(a) \times Selected(p, a) \tag{5.4}$$

$$Q_{t+1}\left(L_p, o_p\right) = Q_t\left(L_p, o_p\right) + \beta_p\left\{r_{t+1} + \gamma V\left(s_{t+1}\right) - Q\left(s_t, a\left(s_t\right)\right)\right\} \tag{5.5}$$

Although, we also use FQL for antenna tilt optimization, our studies are significantly different from their approach in a number of ways. Firstly, we only use the degree of truth in the q-value update equation as explained in Eq. 3.25. Secondly,

we also look into the question of parallel learning in a multi-agent environment like antenna tilt optimization. Unlike, Razavi et al. approach, which allows only one cell to take an action at each snapshot, we also allow all cells and a cluster of cells to take actions simultaneously. This helps to speed-up the learning process at the expense of a complicated learning environment as explained in Section 3.9.2. Thirdly, we also focus on the reinforcement signal definition and the structure of the FQLC. We study the selfish, cooperative and centralized learning strategies, which differ in terms of their reinforcement signal definition and the FQLC structural implementation. We can say, Razavi et al. focus on the question of how to provide feedback to the individual rules of one FQLC. But we focus on the interaction among the different learning agents (FQLC) implemented among different cells of the network in a multi-agent learning environment like ours.

Moreover, from simulation studies point of view, we also study the behavior of our learning strategies in irregular scenario and not just the regular hexagonal scenario.

5.2 Regular Scenario

This section describes the simulation results for our *regular scenario* as explained in Section 4.2.1. The results include the performance results of individual learning schemes and their comparisons.

5.2.1 Selfish Learning

The first learning strategy as explained in Section 3.9.1.1 is the selfish learning, where all cells act independently and try to optimize their individual performance without any consideration to its effect on the neighboring cells' performance. In this regard the first question we try to answer is, what amount of parallel learning is suitable for this antenna tilt optimization problem?

We test the three different levels of parallel update mechanisms as explained in Section 3.9.2. The first one only allows one cell to take an action and thus update its q-table in each learning snapshot. The second one allows all cells to take action and update their q-tables simultaneously in each learning snapshot. And, the third one allows only a cluster of cells to take action and update their q-tables simultaneously in each learning snapshot. The performance of these schemes is compared to our *Reference System* with 15° fixed tilt across all cells and the dynamic optimization scheme of Razavi et al.

The simulation results for all these schemes are presented in Fig. 5.2. The time variation of our performance metric *State Quality* (SQ) is presented in Fig. 5.2-(a). The x-axis represent the snapshot number T and y-axis represent the SQ in

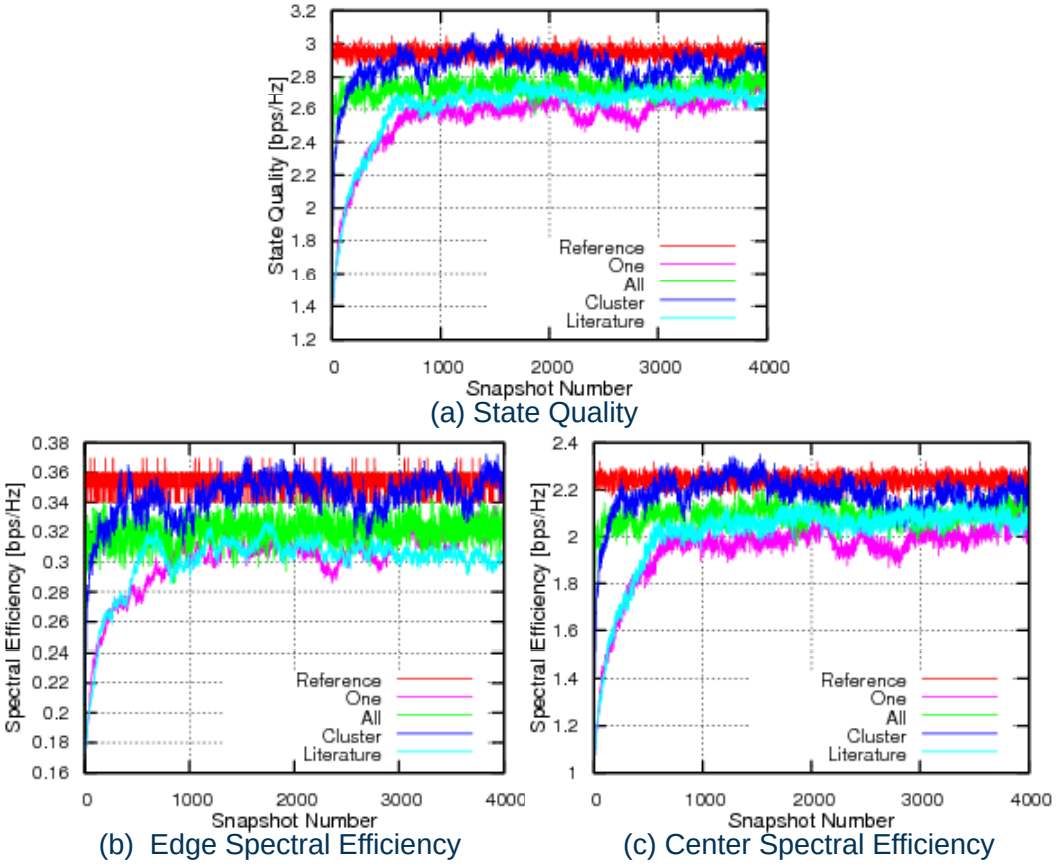

Figure 5.2: Selfish Learning Strategy Comparison

bps/Hz averaged over all cells. SQ is a weighted sum of *Edge Spectral Efficiency* (SE_c^{edge}) and *Center Spectral Efficiency* (SE_c^{center}) as defined in Eq. 3.17, the results for these individual components averaged over all cells are presented in Fig. 5.2-b and Fig. 5.2-c respectively. In all three plots, the term "Reference" refers to our reference system's performance and the term "Literature" refers to the performance of Razavi et al. scheme.

For all four dynamic optimization schemes the starting value of the tilt is set to six degrees across all antennas so that it is far away from the reference setting. In our dense eNB deployment scenario with 500 meter inter-site-distance, this small antenna tilt value produces significant inter-cell interference due to two factors: firstly, because the main beam of antenna pattern is directed further away from the eNB, this produces interference to the neighboring cells and secondly, it reduces the antenna gain in the vicinity of the eNB. This degrades the $SINR$ distribution within the cells and results in a very low value of SQ at $T = 1$ as

shown in Fig. 5.2-a.

The results show that all four strategies are able to overcome the initial bad performance as the learning snapshot progresses. However, they all vary in terms of their learning speed and convergence properties. Allowing only "one cell" to update its tilt at each snapshot significantly reduces the learning speed and the graphs show that it requires more than 1000 snapshots before it gets close to the reference system performance. This is also true for the Razavi et al. scheme as it also allows only one cell to update its tilt per snapshot.

On the other hand allowing "all cells" to update their antenna tilts at each snapshot is quite quick to overcome the starting non-optimal performance. However, after the initial performance gain it remains stuck in a near optimal region and never matches the performance of the reference system. In such a scheme of simultaneous update of antenna tilt by all cells in every snapshot, the change in the $SINR$ distribution within each cell is not only the result of its own action but of its neighboring cells as well. As a result, the reinforcement received by the cell is not completely correct because the cell is unaware of the actions of its neighbors. The ideal solution in such an environment would be to learn the best joint action of all the cells. But the joint action space grows exponentially and makes it impossible to maintain a q-table for state action pairs. Therefore, we have to rely on independent learning, where each cell maintains its own q-table of state action pairs regardless of the actions of the neighboring cells. For this reason the reinforcement signal after each action of the cell is not accurate as it also includes the effect of the actions of all the cells. Therefore, the performance of the simultaneous update of all cells strategy gets affected and never matches the optimal reference system.

To overcome this problem while maintaining a reasonable response time, we proposed a clustering mechanism in Section 3.9.2.3. At each snapshot, the clustering mechanism tries to maximize the number of cells that can take an action under the condition that no two cells in the cluster are adjacent neighbors of each other. This simplifies the reinforcement signal characterization problem as the $SINR$ distribution within the cell now only changes because of the action of the cell itself. As a result the cells can better learn their action policy and achieve higher performance. This also reflects in our results of Fig. 5.2. The "Cluster" scheme outperforms all other schemes. It is quick to change the initial performance degradation due to bad antenna tilt configuration and also reaches the performance level of our reference system. For the system to be able to respond to the network dynamics it needs to learn all the times. Therefore, the performance curves are not monotonously increasing. Sometimes, the performance also degrades a little bit because of the exploration of the solution space by the learning agents. But, they can quickly recover back without introducing too much degradation.

The graphs also show that in some snapshots, the performance of "Cluster" scheme is even better than the reference system. This shows that even for a very homogeneous hexagonal deployment of cells, the network wide fixed antenna tilt configuration is not strictly optimal in terms of SQ measures. As we rely on measurement feedbacks from the actual user equipments, the SQ varies with the user distribution. Therefore, if the network is already performing close to the reference system and in certain snapshots the user distribution is more concentrated in the cell center the SQ values can get a little higher than the "Reference" values.

In reinforcement learning problems, an important aspect is to identify the best learning parameters. For this reason all three strategies were tested with different values of the learning variables i.e. *Exploration Rate, Learning Rate and Discount Factor*. Table 5.1 shows the average SQ between snapshot 1 and 1000 over all cells to reflect the convergence speed of the different strategies. The values show that both "All" and "Cluster" schemes achieve much closer performance than "One" scheme, but the "Cluster" scheme still perform slightly better than the "All" scheme because of the ability to better characterize the reinforcement signal.

Table 5.2 represents the average SQ between snapshot 1000 and 4000 over all cells to represent the stable behavior under different learning parameter settings. Here, again the "Cluster" scheme performs better than other two schemes. But, the system keeps on learning all the times, which also leads to some non-optimal action selection in some snapshots, the performance of the "Cluster" scheme is still never equal to the "Reference" performance. Finally, Table 5.3 summarizes the above two tables and shows the overall average of SQ between snapshot 1 and 4000 over all cells.

Moreover, in all three tables there is not much difference in the performance of different parameter settings for each strategy. One reason for this is the simple homogeneous cell structure, which produces a simple performance curve w.r.t different antenna tilt settings as shown in Fig. 5.1 for reference system performance. There is only one optimal region around 15 degree tilt without any other local maxima points. The reward is also well-aligned with the overall objective of the optimization problem. Therefore, all learning parameters produce similar results.

5.2.2 Cooperative Learning

The second learning scheme we presented in Section 3.9.1.2 is the "Cooperative Learning". Compared to the selfish approaches, which try to optimize the performance of individual cells without considering its effect on the neighbors, cooperative learning tries to optimize the performance of the complete neighborhood. This is achieved by sharing the SQ values among the two tier neighbors and then calculating the reinforcement based on the average SQ over the complete neighborhood. The simulation results for the "Cooperative Learning" are

Table 5.1: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1 and 1000

Learning Parameters			State Quality [bps/Hz]			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.36	2.71	2.73
		0.5	2.95	2.35	2.70	2.81
		0.7	2.95	2.31	2.73	2.77
	0.15	0	2.95	2.37	2.68	2.70
		0.5	2.95	2.39	2.65	2.78
		0.7	2.95	2.28	2.64	2.77
0.12	0.1	0	2.95	2.31	2.69	2.76
		0.5	2.95	2.43	2.62	2.72
		0.7	2.95	2.34	2.70	2.79
	0.15	0	2.95	2.41	2.71	2.80
		0.5	2.95	2.31	2.69	2.78
		0.7	2.95	2.25	2.66	2.77

Table 5.2: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1000 and 4000

Learning Parameters			State Quality [bps/Hz]			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.71	2.72	2.86
		0.5	2.95	2.70	2.73	2.86
		0.7	2.95	2.63	2.77	2.82
	0.15	0	2.95	2.65	2.73	2.84
		0.5	2.95	2.55	2.69	2.84
		0.7	2.95	2.64	2.72	2.86
0.12	0.1	0	2.95	2.62	2.76	2.81
		0.5	2.95	2.75	2.70	2.86
		0.7	2.95	2.61	2.74	2.89
	0.15	0	2.95	2.61	2.73	2.87
		0.5	2.95	2.69	2.73	2.85
		0.7	2.95	2.60	2.75	2.86

Table 5.3: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1 and 4000

Learning Parameters			State Quality [bps/Hz]			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.62	2.72	2.83
		0.5	2.95	2.61	2.72	2.85
		0.7	2.95	2.55	2.76	2.81
	0.15	0	2.95	2.58	2.72	2.81
		0.5	2.95	2.51	2.68	2.83
		0.7	2.95	2.55	2.70	2.84
0.12	0.1	0	2.95	2.54	2.74	2.80
		0.5	2.95	2.67	2.68	2.82
		0.7	2.95	2.55	2.73	2.86
	0.15	0	2.95	2.56	2.73	2.85
		0.5	2.95	2.59	2.72	2.83
		0.7	2.95	2.51	2.72	2.84

presented in this section.

One Cell Update per Snapshot: Like selfish learning, first we analyze the performance of different parallel learning mechanisms. Figure 5.3 depicts the results for the learning mechanism where only one cell can take an action and update its q-table in each learning snapshot. In all three graphs "Selfish" means results for the selfish learning mechanism and "Cooperative" means results for the cooperative learning mechanism. The performance for both selfish and cooperative learning is quite similar in this scenario and the restriction that only one cell can take an action per learning snapshot makes both of these schemes very slow to react to the initial non-optimal configuration. Even in 4000 snapshots, non of them is able to exactly match the performance of the reference system. It takes almost 1000 snapshots to generate a performance close to the reference system. In the next 3000 snapshots, all the schemes slowly try to minimize the gap between their performance and the reference system, but cannot completely reduce it to zero.

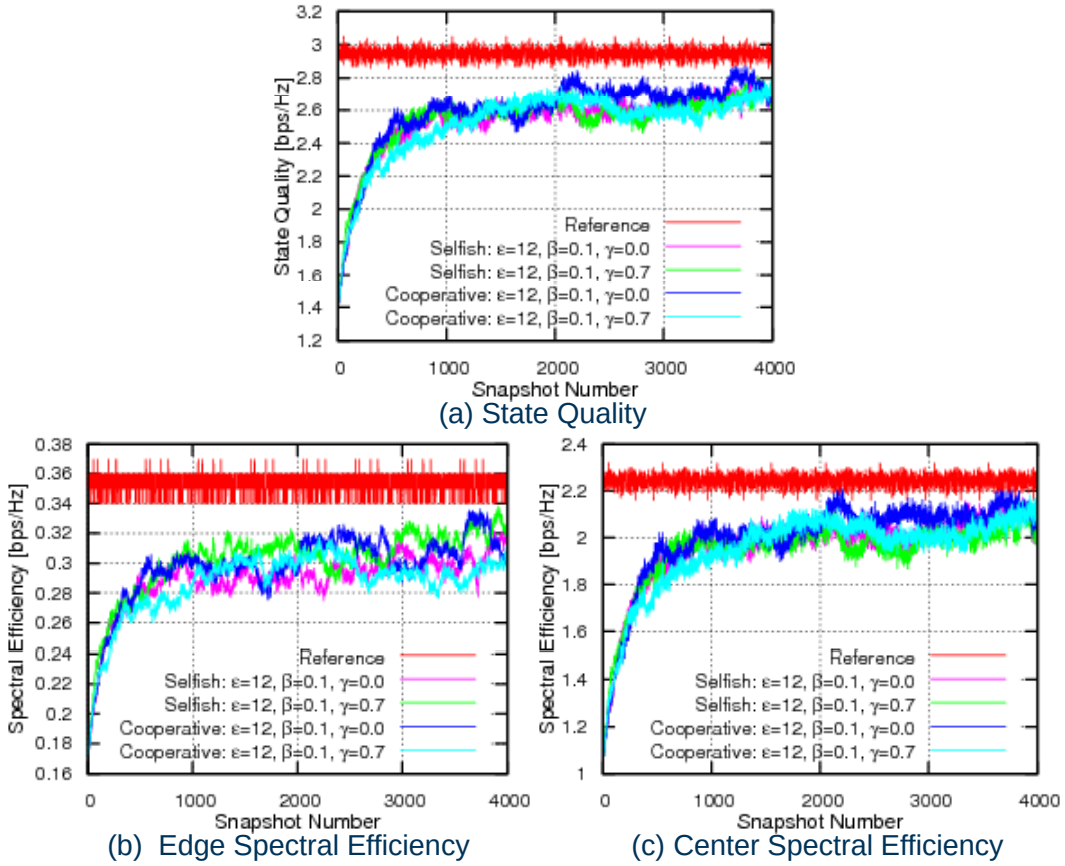

Figure 5.3: One Agent per Snapshot Cooperative vs Selfish Learning

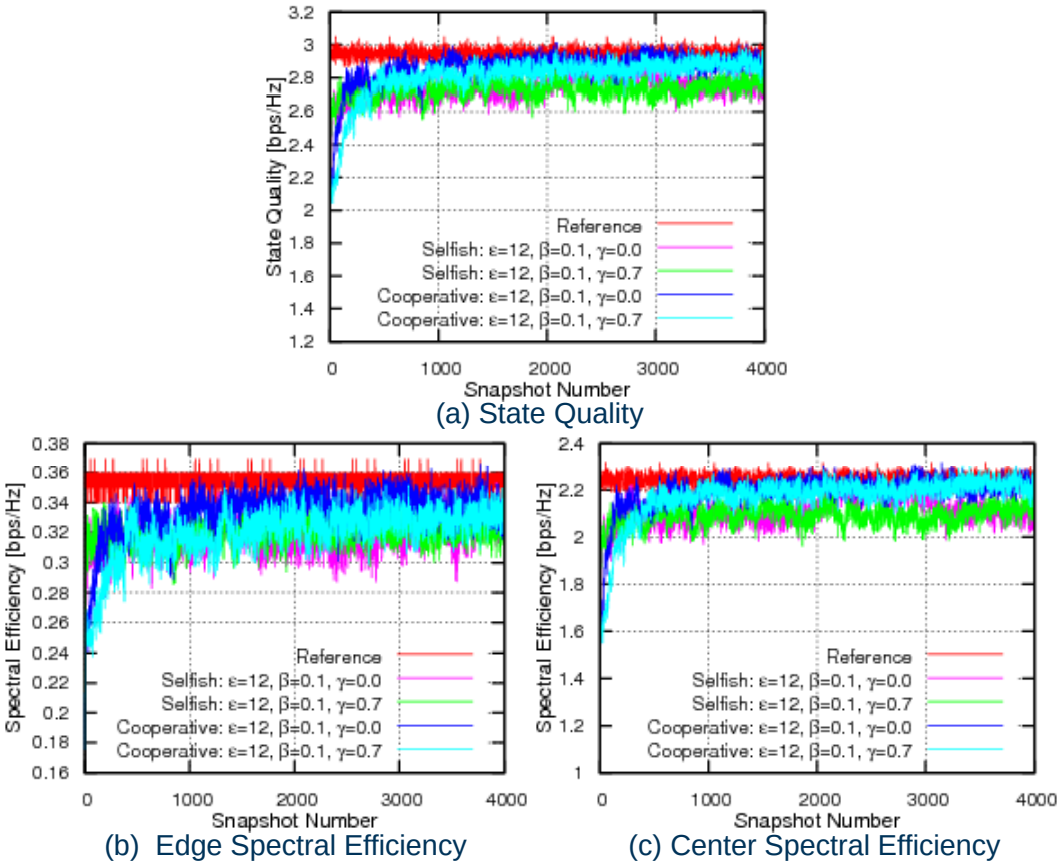

Figure 5.4: All Agents per Snapshot Cooperative vs Selfish Learning

All Cells Update per Snapshot: The results for the learning mechanism where all cells can take an action and update their q-tables in each learning snapshot are shown in Fig. 5.4. In this scenario, just like the selfish learning mechanism, the cooperative learning mechanism is also quite quick to overcome the performance degradation due to the initial non-optimal antenna tilt configuration. However, the cooperation among the neighboring cells helps to improve the convergence properties of this learning mechanism. As clear from the plots, with cooperation, the simultaneous learning strategy of all cells does not get stuck in a near optimal performance level, but can exactly match the performance of the reference system with the passage of learning snapshots. As the cells now calculate the reinforcement signal based on the SQ results of the complete neighborhood, they try to optimize to performance of the complete neighborhood instead of only their local performance. This helps to stabilize the system even if all cells update their antenna tilts in each snapshot.

Cluster of Cell Update per Snapshot: Figure 5.5 shows the results for

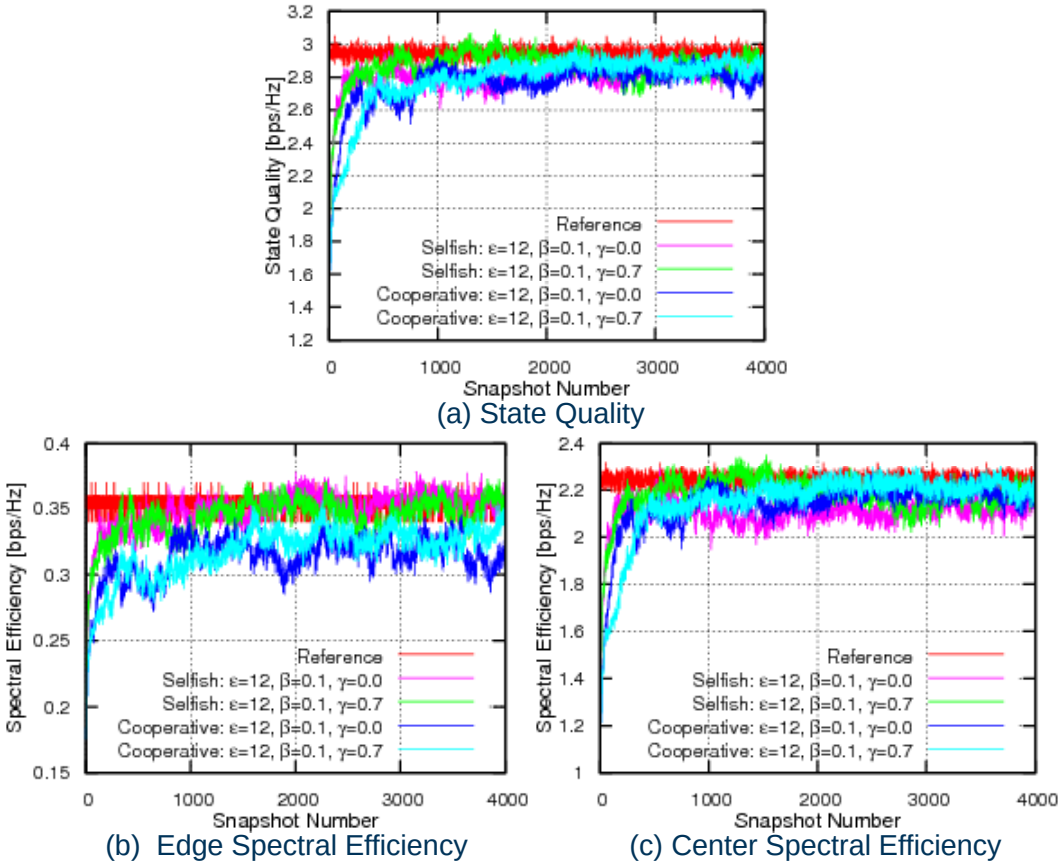

Figure 5.5: Cluster of Agents per Snapshot Cooperative vs Selfish Learning

the learning mechanism where a cluster of cells can take an action and update their q-tables in each snapshot. Here, the performance of selfish and cooperative mechanisms are quite similar. All of them can quickly overcome the initial performance degradation and can also match the reference system performance. However, the cooperative learning makes the performance curves smoother compared to the selfish learning as can be seen from the graphs of *State Quality* and *Center Spectral Efficiency*. With selfish learning, all cells try to optimize their own performance without any regard to performance of the neighboring cells. But the optimal antenna tilt depends on the antenna tilt of the neighboring cells. Therefore in selfish learning, we observe some fast oscillations in the performance curves because all cells are trying to optimize their own performance and affecting the performance of others, which try to adjust to these changes in the neighboring cells, this adjustment continues and results in some oscillating behavior of our performance metrics. However, with cooperative learning as the cells are trying

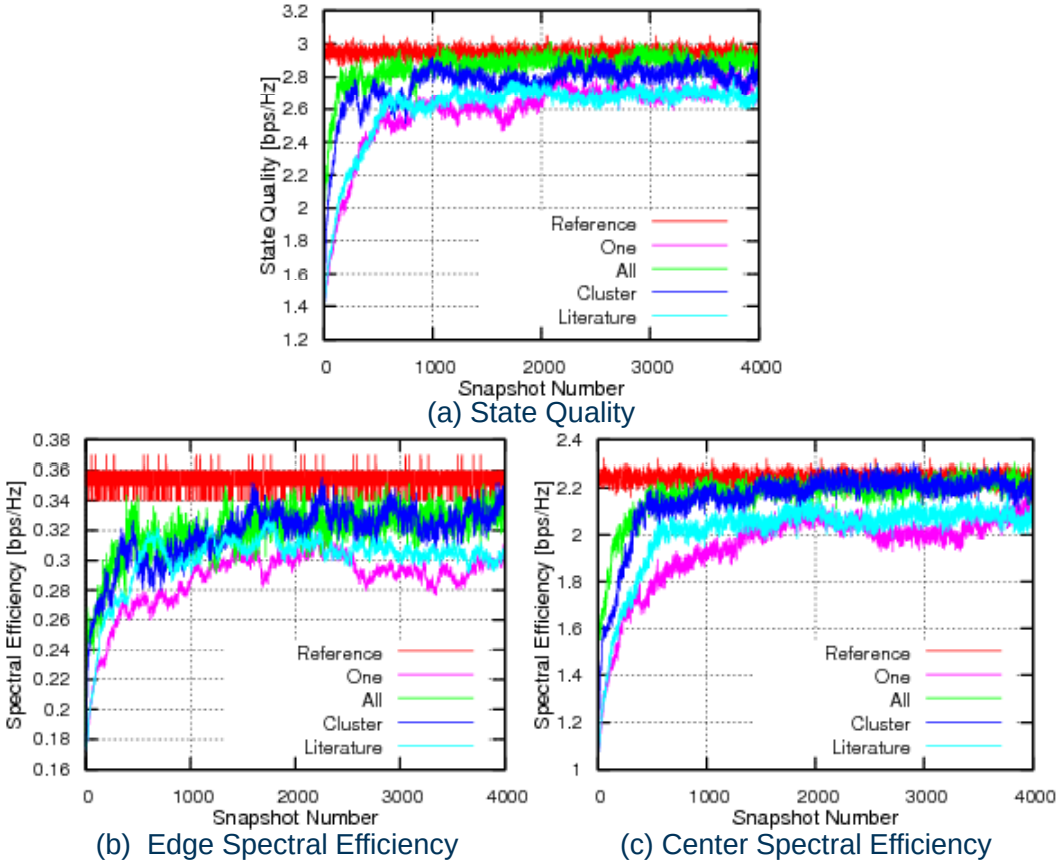

Figure 5.6: Cooperative Learning Strategy Comparison

to optimize the performance of the complete neighborhood, these oscillations can be reduced. These results are also in-line with the experiments of Balch [18], [19] and [20], where different reward functions were tested and proved that local reward can achieve faster learning rates but not necessarily better results than global reward.

Comparison of Different Levels of Parallel Updates: The comparative results for the three levels of parallel update mechanisms with cooperative learning are shown in Fig. 5.6. They are also compared to our reference system and the learning scheme of Razavi et al. shown as the "Literature" curve in the three graphs. Again, the Razavi et al. approach shows comparable performance to only one cell update per snapshot strategy as it also relies on only one cell update per snapshot. Both the "All" and "Cluster" strategies are quick to respond to the performance degradations and can also reach the reference system performance level. Moreover, compared to the selfish approach, the all cell update strategy

does not get stuck in a near optimal performance level. This helps to avoid the requirement of dividing the network into different clusters at each snapshot, as required by selfish learning to reach the reference system performance level.

The above mentioned three parallel learning mechanisms were also tested with different learning parameter settings. The results for the average SQ over complete network between snapshot number 1 and 1000 are given in Table 5.4. This serves to indicate how quickly each learning strategy can overcome the initial performance degradation. The stable state behavior is presented in Table 5.5, which shows the average SQ over the whole network between snapshot number 1000 and 4000. Finally, the overall results for the complete simulation run are described in Table 5.6. From the results in all three tables it is clear that the "All" cell update strategy performs better than the other two strategies both in terms of its initial response time and the maximum achievable performance in the stable state region.

Table 5.4: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1 and 1000 for Cooperative Learning

Learning Parameters			**State Quality [bps/Hz]**			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.34	2.70	2.59
		0.7	2.95	2.33	2.77	2.40
	0.15	0	2.95	2.26	2.80	2.54
		0.7	2.95	2.14	2.72	2.51
0.12	0.1	0	2.95	2.33	2.76	2.62
		0.7	2.95	2.30	2.67	2.57
	0.15	0	2.95	2.40	2.77	2.56
		0.7	2.95	2.32	2.79	2.59

Table 5.5: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1000 and 4000 for Cooperative Learning

Learning Parameters			**State Quality [bps/Hz]**			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.72	2.90	2.79
		0.7	2.95	2.65	2.89	2.70
	0.15	0	2.95	2.74	2.88	2.75
		0.7	2.95	2.59	2.85	2.77
0.12	0.1	0	2.95	2.68	2.89	2.81
		0.7	2.95	2.62	2.85	2.86
	0.15	0	2.95	2.75	2.91	2.82
		0.7	2.95	2.60	2.89	2.83

Table 5.6: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1 and 4000 for Cooperative Learning

Learning Parameters			State Quality [bps/Hz]			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.62	2.85	2.74
		0.7	2.95	2.56	2.86	2.62
	0.15	0	2.95	2.66	2.86	2.70
		0.7	2.95	2.52	2.82	2.70
0.12	0.1	0	2.95	2.60	2.86	2.76
		0.7	2.95	2.52	2.81	2.79
	0.15	0	2.95	2.63	2.88	2.76
		0.7	2.95	2.49	2.87	2.77

5.2.3 Centralized Learning

The third learning scheme presented in section 3.9.1.3 is the "Centralized Learning". This scheme maintains a central q-table which is updated by all the learning agents in the system. This way the learning agents can benefit from the learning experience of all other agents. Moreover, as the central q-table already has access to all the agents, the reinforcement is also calculated based on SQ_{avg}, which helps to identify the actions that improve the overall performance and not just the performance of a single cell. The results for this scheme are presented in this section.

One Cell Update per Snapshot: First the comparison of one cell update strategy is presented in Fig. 5.7. In all three graphs "Selfish" means results for the selfish learning mechanism, "Cooperative" means results for the cooperative learning mechanism and "Centralized" means results for the centralized learning mechanism. The graphs show a significant difference between the performance of the centralized approach and the distributed approaches. Unlike selfish and cooperative schemes, where each cell tries to learn the optimal action policy independently, centralized learning allows sharing of the learned knowledge among different cells. This knowledge sharing is proven to reduce tendencies for convergence to sub-optimal behaviors [22] and speed-up the learning process [25]. The simulated scenario also consists of a regular hexagonal cell deployment, which also makes it feasible to re-use the learned optimal action policy at different cells. These factors help to speed-up the overall learning process and even with one cell update per snapshot strategy the centralized learning mechanism can quickly overcome the initial non-optimal antenna tilt setting.

All Cells Update per Snapshot: The results for the centralized learning mechanism where all cells can take an action and update the central q-table in each learning snapshot are shown in Fig. 5.8. Knowledge sharing also helps in this scheme as the centralized mechanism can overcome the initial performance degradation quicker than selfish and cooperative learning mechanisms. However,

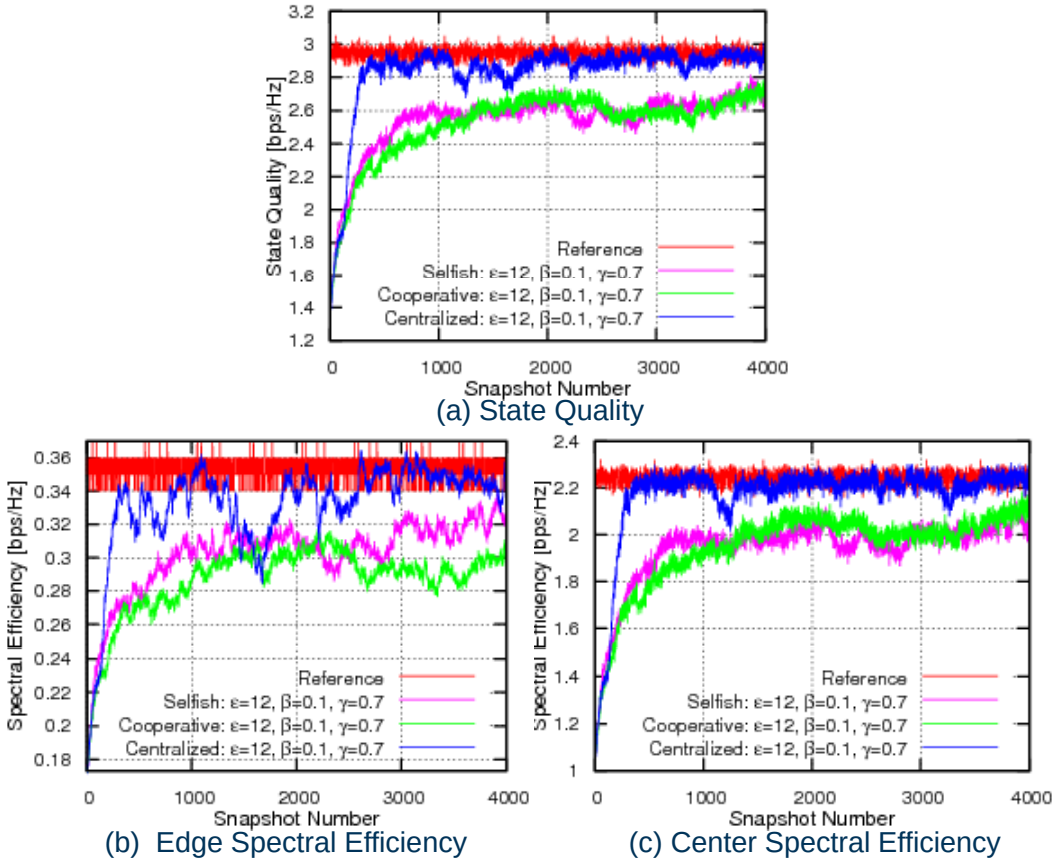

Figure 5.7: One Agent per Snapshot Centralized vs Selfish vs Cooperative Learning

the initial performance gain is not stable and performance curves fluctuate around the reference system performance. With each antenna tilt modification cells not only adapt their own performance but also effect the performance of the neighboring cells. Therefore, with all cells update at each snapshot strategy, cells face a continuously changing environment where they have to always respond to the actions of their neighbors. Moreover, to ensure continuous learning cells have to explore the solution space which involves taking random actions instead of so far best learned action in some snapshots. This could also degrade the performance if the system is already performing close to the optimal level. In such a scenario, with independent q-tables cells can converge to different action policies to improve the overall stability of the whole system as shown by the results of "selfish" and "cooperative" learning. However, in centralized learning with only one shared q-table for all cells it becomes extremely difficult for all cells to converge to one

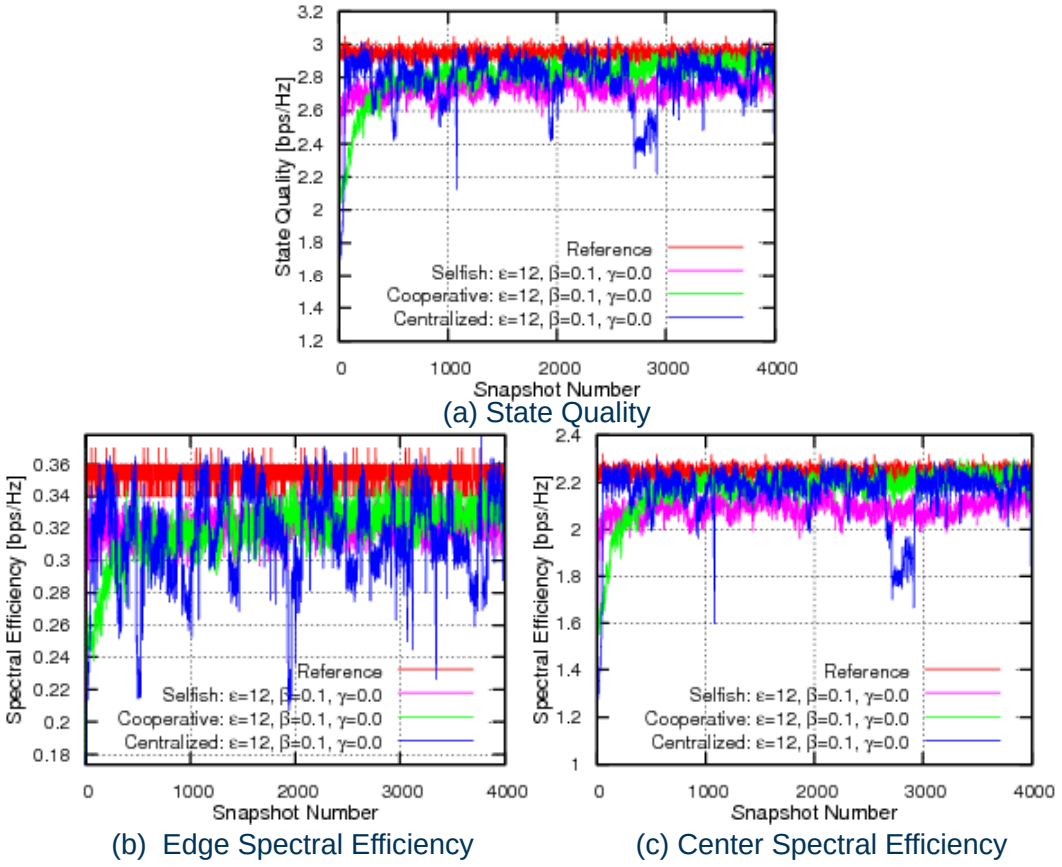

Figure 5.8: All Agents per Snapshot Centralized vs Selfish vs Cooperative Learning

stable action policy and the performance keeps on fluctuating.

Cluster of Cells Update per Snapshot: Centralized learning was also tested with the cluster of cells update per snapshot strategy and the results are given in Fig. 5.9. Here, the centralized learning mechanism is also the quickest to match the reference system performance. But the performance is not as stable as with the selfish and cooperative schemes. However, the variations are less than the all cell update per snapshot strategy because of a much better learning environment and less number of simultaneous actions in the network.

Comparison of Different Levels of Parallel Updates: The comparison between the three levels of parallel update strategies for centralized learning is shown in Fig. 5.10. Apart from the reference system, the results are also compared to the Razavi et al. approach indicated by the "Literature" curve on the graphs. All three centralized parallel update strategies perform better than the Razavi

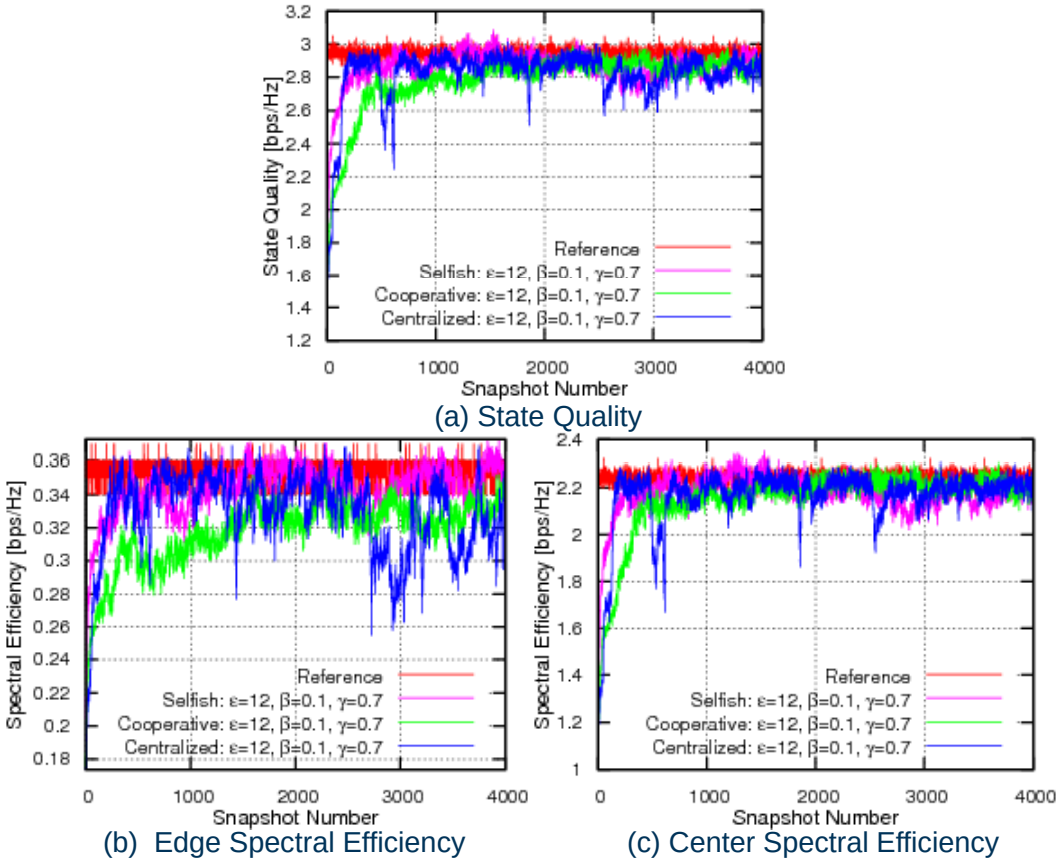

Figure 5.9: Cluster of Agents per Snapshot Centralized vs Selfish vs Cooperative Learning

et al. approach. Although their scheme also uses one cell update per snapshot but the centralized one cell update strategy performs much better because of the knowledge sharing among the different cells. It reacts much faster to the initial performance degradation and can also match the reference system performance level.

The initial very quick match to the reference system performance means that a number of states remain un-explored before reaching this performance level. Cells reaching these un-explored states afterwards need to take random actions because they are not aware of the optimal action. This random action selection can worsen the performance if the network is already performing close to the optimal level. The chances of reaching these un-explored states increases with the increasing parallelism. If more and more cells take simultaneous actions, the probability that they experience quite diversified states increases, which ulti-

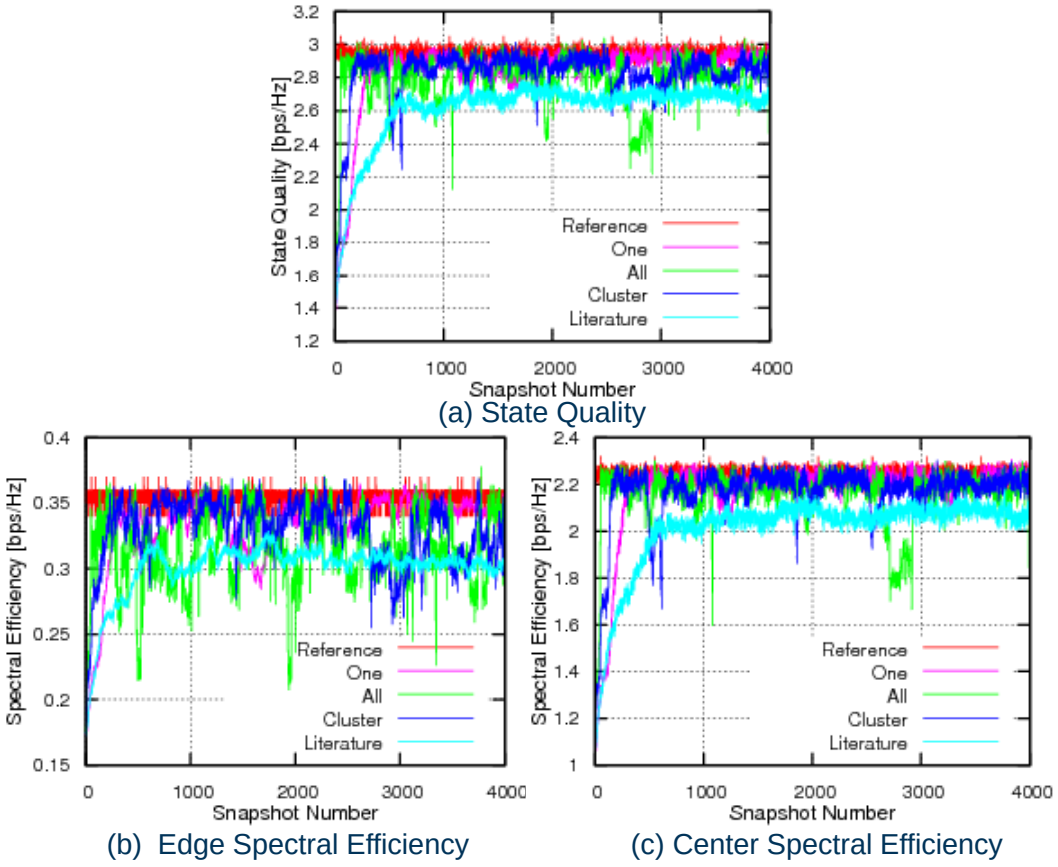

Figure 5.10: Centralized Learning Strategy Comparison

mately increases the number of random actions being taken by cells. As a result, the performance stability gets effected and shows an increasing fluctuating behavior with increasing parallel actions of different cells.

The centralized learning with three parallel learning mechanisms was also tested with different learning parameter settings. The results for the average SQ over complete network between snapshot number 1 and 1000 are given in Table 5.7. This serves to indicate how quickly each learning strategy can overcome the initial performance degradation. The stable state behavior is presented in Table 5.8, which shows the average SQ over the whole network between snapshot number 1000 and 4000. Finally, the overall results for the complete simulation run are described in Table 5.9. The one cell update strategy is relatively slower in its response to the initial non-optimal antenna tilt setting due to its limited adaptability but it is still better than the selfish and cooperative schemes using one cell update per snapshot. However, due to its better stability, it shows overall

better performance than the all cell and cluster of cell update strategy as shown in Table 5.8 and Table 5.9.

Table 5.7: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1 and 1000 for Centralized Learning

Learning Parameters			**State Quality [bps/Hz]**			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.60	2.77	2.79
		0.7	2.95	2.66	2.75	2.73
	0.15	0	2.95	2.62	2.77	2.82
		0.7	2.95	2.64	2.54	2.81
0.12	0.1	0	2.95	2.72	2.80	2.76
		0.7	2.95	2.57	2.61	2.72
	0.15	0	2.95	2.61	2.62	2.78
		0.7	2.95	2.65	2.79	2.73

Table 5.8: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1000 and 4000 for Centralized Learning

Learning Parameters			**State Quality [bps/Hz]**			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.90	2.69	2.75
		0.7	2.95	2.89	2.80	2.85
	0.15	0	2.95	2.84	2.68	2.78
		0.7	2.95	2.84	2.51	2.87
0.12	0.1	0	2.95	2.89	2.59	2.77
		0.7	2.95	2.84	2.10	2.77
	0.15	0	2.95	2.82	2.57	2.83
		0.7	2.95	2.80	2.81	2.83

Table 5.9: Regular Scenario: Average State Quality [bps/Hz] Between Snapshot 1 and 4000 for Centralized Learning

Learning Parameters			**State Quality [bps/Hz]**			
Exploration Rate	**Learning Rate**	**Discount Factor**	**Reference**	**One**	**All**	**Cluster**
0.06	0.1	0	2.95	2.83	2.71	2.76
		0.7	2.95	2.83	2.79	2.82
	0.15	0	2.95	2.78	2.71	2.79
		0.7	2.95	2.79	2.52	2.86
0.12	0.1	0	2.95	2.85	2.64	2.77
		0.7	2.95	2.77	2.23	2.75
	0.15	0	2.95	2.77	2.58	2.82
		0.7	2.95	2.76	2.81	2.81

5.2.4 Outage Recovery and eNB Deployment

Apart from the above mentioned self-optimization capability of the different learning strategies, self-healing and self-configuration performance was also analyzed. Self-healing tries to mitigate the effect of a cell outage by autonomously adjusting the coverage area of the neighboring cells. Whereas, self-configuration tries to autonomously configure the parameters of a new eNB on its deployment.

Outage and eNB Deployment Realization: For the self-healing study an outage is created by deactivating the center eNB with cells 12, 13 and 14 as shown in Fig. 4.2-a. This creates a coverage hole and the neighboring eNBs try to extend their coverage in order to compensate for this coverage loss. For self-configuration analysis the center eNB is then reactivated but with a very high downtilt value of 22°. This can be referred as a gradual deployment of an eNB as the higher value of downtilt ensures that the initial coverage of the deployed eNB is very small and after that it tries to extend its coverage in order to balance the coverage and capacity of the whole neighborhood.

SINR Distribution Throughout The Network: The SINR distribution for different phases of this outage and deployment scenario is presented in Fig. 5.11. The initial non-optimal antenna tilt setting produces very low *SINR* values across all cells as shown in Figure 5.11-a. Most of the simulated network shows *SINR* values in dark blue color i.e. around 0 dB. The initial very small antenna tilt values of 6° direct the main lobe of antenna radiation pattern further away from the eNB site, which reduces the antenna gain values in the vicinity of the eNB and produces significant interference to the neighboring cells. Therefore, initially in most of the cells even the areas very close to the cell sites are not able to have higher *SINR* values.

The situation improves as the cells learn optimal action policy and adjust their antenna tilts accordingly. Figure 5.11-b shows the *SINR* distribution in the simulated network at snapshot number 2000. With antenna tilt adjustments, inter-cell interference decreases and the antenna gain increases in the vicinity of the cells. Both of these factors improve the *SINR* distribution throughout the network and the graph shows most of the regions in higher range of *SINR* distribution with green color i.e. > 10 dB.

After this initial optimization of antenna tilt an outage is created by deactivating the center eNB with all its three cells as shown in Fig. 5.11-c. This degrades the *SINR* distribution in the center of the network and some parts experience very low *SINR* values of around -5 dB as shown by the light blue color in Fig. 5.11-c. One important thing to note here is that with the absence of center eNB after deactivation, the coverage of some of the direct neighbors gets bigger even without any antenna tilt adjustment, which provides coverage to some of the outage regions. Antenna tilt adjustments further improve the *SINR* distribution

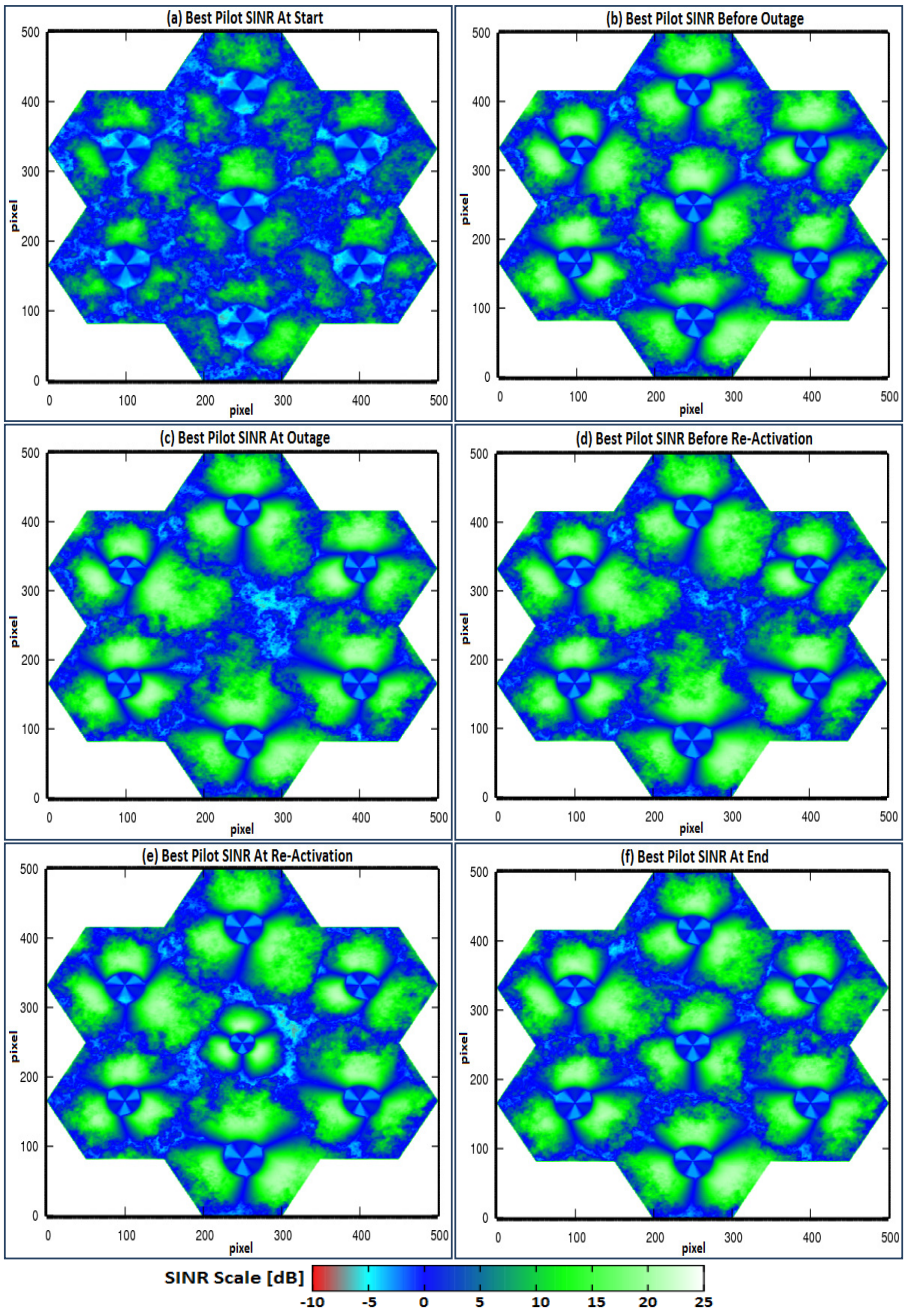

Figure 5.11: SINR Distribution for Outage Recovery and BS Deployment

and most of the light blue regions convert to dark blue and the dark blue regions to green at snapshot number 4000 as shown in Fig. 5.11-d.

Finally, to test the gradual deployment scheme, the center eNB is reactivated but with a very high antenna tilt of 22°. This deploys the center eNB with small coverage area as shown in Fig. 5.11-e. This addition of an eNB improves the *SINR* distribution near the deployed eNB but degrades it at the cell edges. This sudden appearance of a new eNB requires re-adjustment of antenna tilts by the neighboring cells to balance the coverage area with the new cells. The learning algorithm helps to identify this performance degradation and modifies the antenna tilt of the new cells and the neighboring cells to improve the overall *SINR* distribution within the network. The final *SINR* distribution is shown in Fig. 5.11-f where all cells have achieved a balance between their coverage areas and there are no light blue or red color regions indicating very low *SINR* values.

Learning Strategy Comparison: All the proposed learning strategies were tested in this dynamic scenario of eNB outage and deployment and the results are explained below. Outages are created by deactivating the center eNB at snapshot number 2000 and 6000, whereas, they are re-activated with higher antenna tilt of 22° at snapshot number 4000 and 8000. The results always represent the average over all 21 cells, even during the outages. For this reason, the values during the outage are always lower than the normal operation even for the reference network.

One Cell Update per Snapshot: First the comparison of different learning strategies using only one cell update per snapshot is presented in Fig. 5.12. The results show that using only one cell update per snapshot limits the adaptability of the network and slows down the learning process. In first 2000 snapshots selfish and cooperative learning schemes can only reach a performance level that is slightly better than the performance level of the reference system with outage. The performance further degrades with the appearance of outages. Both the learning schemes can partially recover the performance degradation due to the outage but cannot match the performance of the reference system within the 2000 snapshots of outage time. The re-activation of the central eNB improves the performance significantly at snapshot number 4000 and 8000 because of the additional radio resources being available in the area. However, the re-adjustment process of the antenna tilts is slow due to the limited number of changes in each snapshot. As a result, the performance slightly improves in the 2000 snapshots after each re-activation but still remains lower than the reference system performance.

Knowledge sharing of the centralized learning scheme helps the cells to quickly learn the optimal action policy and can match the performance level of the fully operational reference network in the first 2000 snapshots. But this performance gain reduces as the environment becomes more and more dynamic with cell outages and deployments. In such a scenario different cells experience different environments even in regular hexagonal deployments. The cells directly facing the outage area have a different environment than the cell further away from the

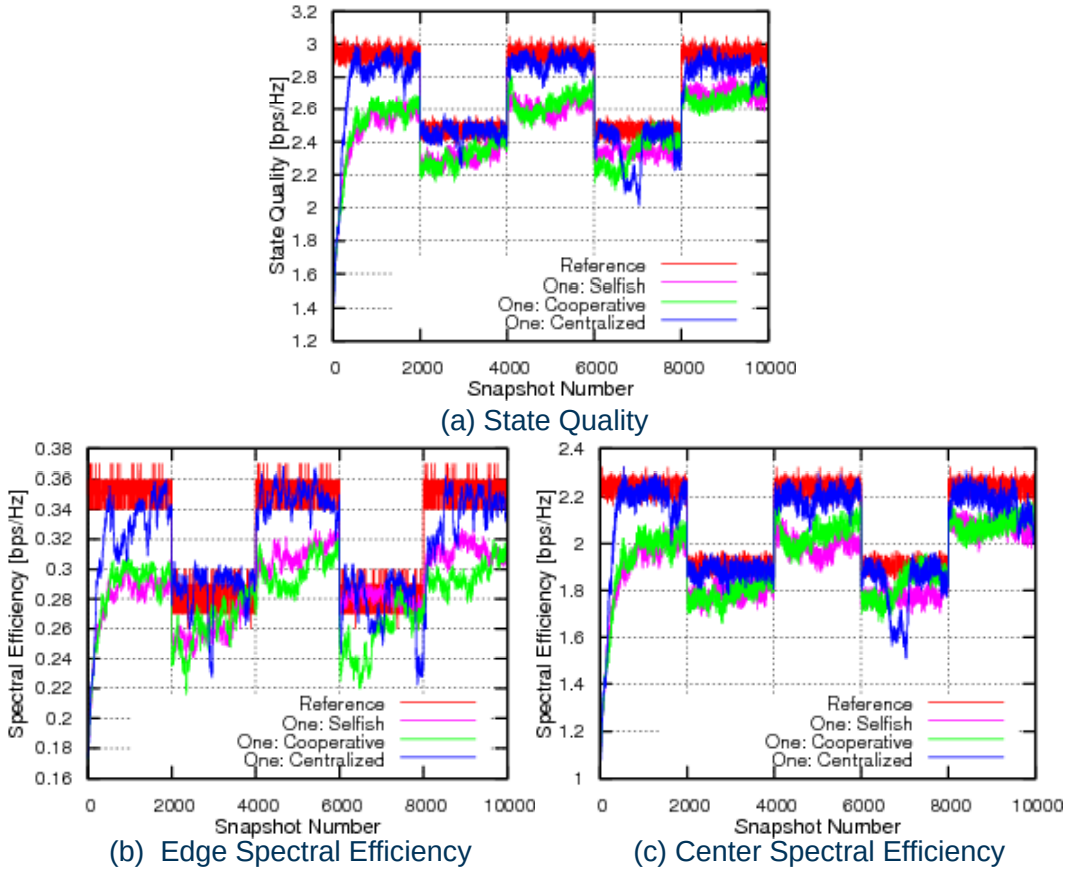

Figure 5.12: Regular Scenario Outage Comparative Results for One Agent Update per Snapshot

outage. Using the same q-tables and thus the same action policy for all the cells is not optimal in this scenario and the performance degrades. This is especially visible in the second outage and re-activation between snapshot number 6000 and 10000. The situation can be improved by having a more elaborate state definition, which clearly represent the different network situation or by having different q-tables for outage and normal scenarios. But both of these mechanisms increases the computational complexity because the state space grows exponentially with every component being added to the state vector.

All Cells Update per Snapshot: The outage and deployment of eNB scenario was also tested with learning strategies using all cells update at each snapshot and the results are presented in Fig. 5.13. In the first 2000 snapshots all three learning strategies can quickly overcome the initial performance degradation of the start-up configuration because of the increased adaptability of this

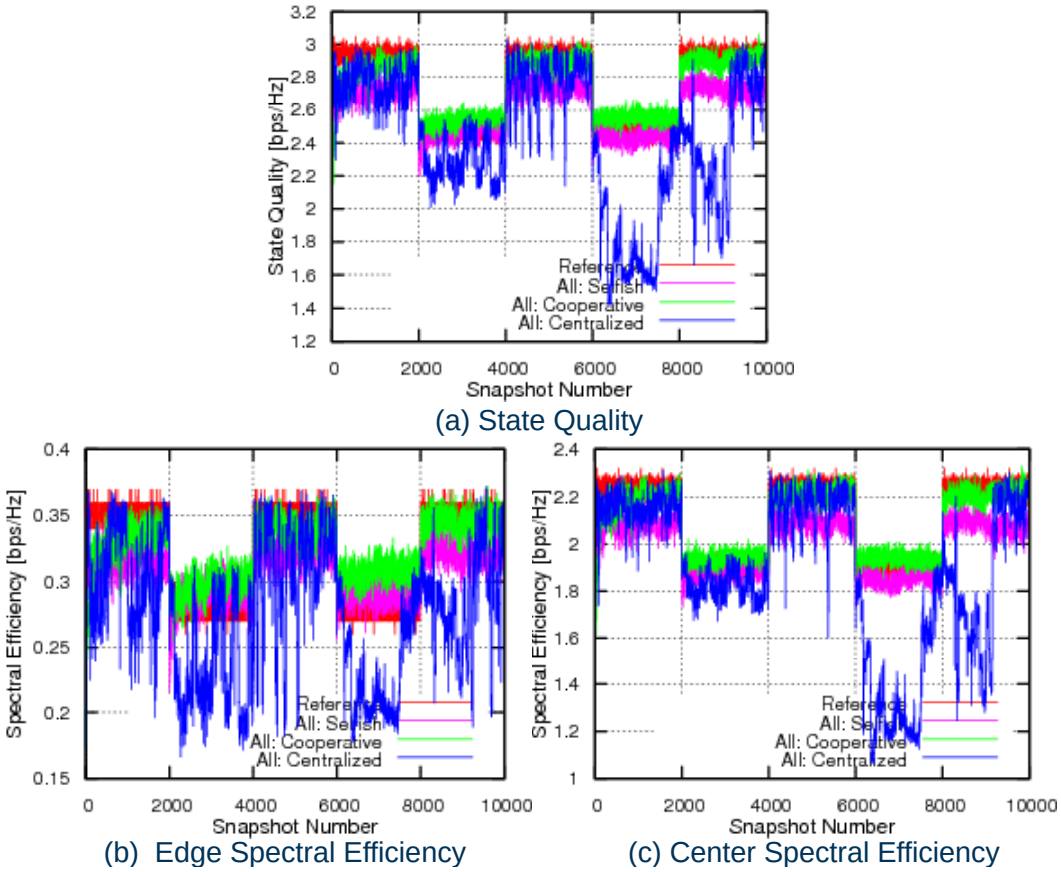

Figure 5.13: Regular Scenario Outage Comparative Results for All Agents Update per Snapshot

scheme compared to the one cell update per snapshot. However, the performance gain varies for different strategies. Selfish learning can only produce a performance close to the reference system but can not match it exactly. The centralized learning can match the reference system performance but is not stable and shows a fluctuating behavior. However, cooperative learning can not only match the reference system performance but can also maintain it over large number of snapshots.

The performance gap among the different learning schemes widens as the network experience the eNB outages and deployments. Especially, the performance of centralized learning is extremely effected with this all cell update per snapshot strategy. Compared to one cell per snapshot strategy, now the central shared q-table is updated more often by cells having different local conditions. Therefore, the q-table is not able to converge to a single best action policy for all the cells

and negatively effects the overall performance of the network.

The cooperative learning scheme shows much better performance even with the outages and deployment of eNBs. In fact it gives slightly better results than the reference system during outages. Much of this gain comes from the improvement in the cell edge performance as shown in Fig. 5.13-b. This is achieved when the direct neighbors of the cells in outage extend their coverage areas by lowering their antenna tilts. In cooperative learning the cells have independent q-tables, which allows them to learn individual action policies according to their local environment. Additionally, the reinforcement signal is based on the overall performance of the network and not just the cell, therefore the cells independently learn actions which are good for the whole neighborhood and improve the overall performance.

Another important thing to note is that the gain in the cell edge performance does not come at the expense of cell center performance. The simulated network is an interference limited scenario with inter-site-distance (ISD) of only 500 m. As also shown by the SINR distribution in Fig. 5.11 at such small ISD much of the outage area is already covered by the neighboring cells without any antenna tilt modification. Therefore, only small antenna tilt adjustments are made by the learning algorithms, which does not adversely effect the cell center performance but still improve the cell edge performance.

Cluster of Cells Update per Snapshot: Finally, this regular hexagonal network with eNB outages and deployments was also tested with different learning strategies using a cluster of cell update in each snapshot and the results are presented in Fig. 5.14. The results show that all three learning schemes can quickly respond to the changing conditions in the network. However, again the centralized scheme is not able to maintain the initial performance gain and its performance deteriorates with the passage of time as more and more cells update and use the central q-table. In this case the performance fluctuations are less severe than the all cell update strategy because of less changes per snapshot in the network.

Both distributed learning schemes perform better than the centralized scheme. They can quickly adjust the antenna tilts to improve the initial performance degradation of the start-up configuration and can also maintain that performance. Also during the outages they can slightly improve the cell edge performance compared to the reference system but the gain is less than the all cell update strategy. Our clustering mechanism simply tries to avoid neighboring cell to update in the same snapshot and picks the center cell randomly without any consideration to the special needs of any cell. Therefore, it reduces the antenna tilt update ability of the neighboring cells to the outage area compared to all cells update strategy and limits the performance gains in this scenario.

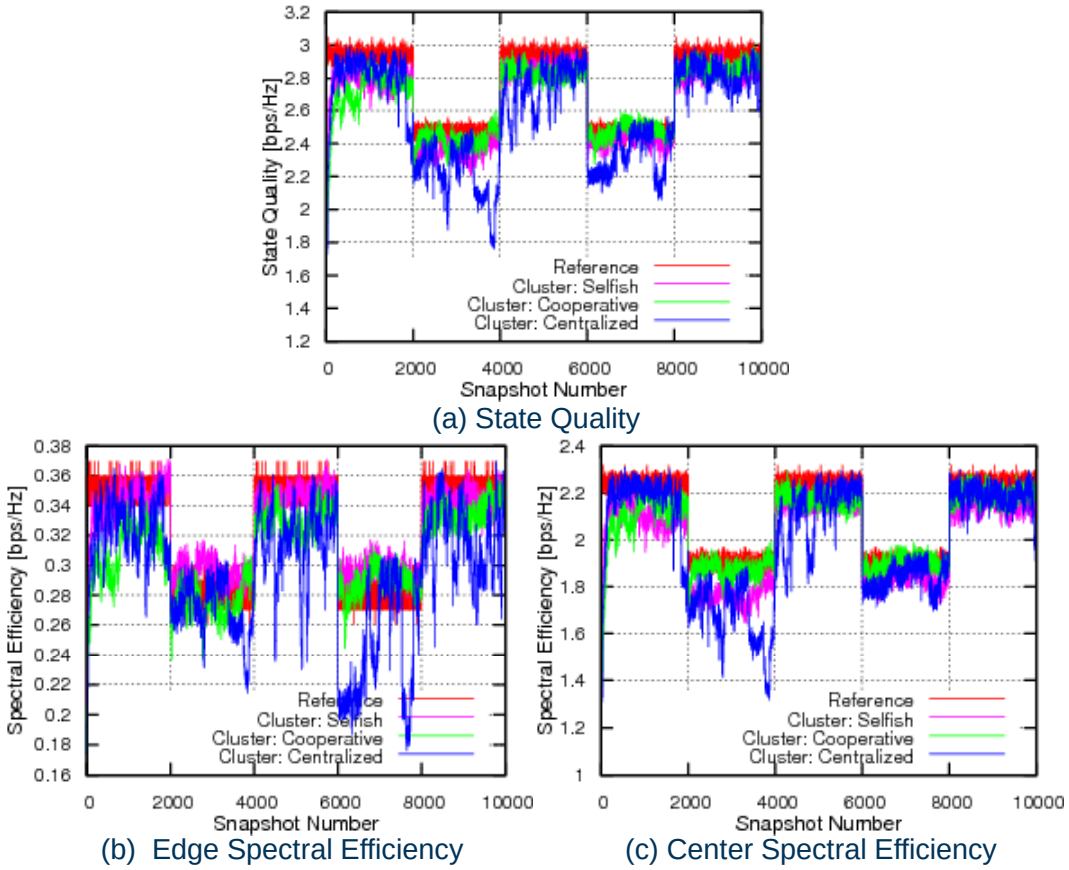

Figure 5.14: Regular Scenario Outage Comparative Results for Cluster of Agents Update per Snapshot

5.3 Irregular Scenario

Apart from the regular hexagonal scenario we also tested our learning strategies in an irregular scenario as described in Section 4.2.1. The performance results of different learning strategies in this scenario are described in the following sections.

5.3.1 Learning Strategy Comparison

This section presents the temporal results of different learning strategies in the irregular scenario. Because of long simulation times required for this larger scenario, each simulation run was restricted to only 4000 snapshots with only one outage at snapshot number 2000 instead of 10,000 snapshots and two cycle of outage and eNB deployment for the regular scenario. The initial antenna tilt

across all cells in this irregular scenario is also set to 11° instead of 6° as in the regular scenario.

For the regular scenario, the reference system was created by testing different values of uniform antenna tilt across all cells because of the regular hexagonal deployment. However, for this irregular deployment with cells of different sizes, it would be sub-optimal to have the same antenna tilt across all cells. Multitude of antenna tilt combinations even for this small network also makes an exhaustive search infeasible. Therefore, for this irregular scenario we do not have a reference system and only compare our results with the results of Razavi et al. approach.

One Cell Update per Snapshot: First, the results of different learning strategies using only one cell update per snapshot are presented in Fig. 5.15. Due to similar update strategy the selfish and cooperative learning schemes perform quite similar to the Razavi et al. approach shown as the "Literature" curve. However, the performance gains are limited because of the limited adaptability of these strategies in this bigger scenario. In the first 2000 snapshots of normal operation, the SQ_{avg} over the complete network improves from around 2.1 bps/Hz to around 2.3 bps/Hz only.

The centralized learning approach is much quicker to overcome the initial non-optimal antenna tilt configuration. It can improve the SQ_{avg} from 2.1 bps/Hz to 2.5 bps/Hz in the first 700 snapshots. But, unlike the regular scenario the performance is not stable even for this one cell update per snapshot strategy in this irregular scenario. As more and more cells with different cell sizes update the central shared q-table and use it for their action selection, the performance starts degrading and shows a fluctuating behavior. Especially, the edge performance is extremely effected compared to the cell center performance, which shows a more stable behavior. Although, we have cells with different cell sizes but the difference is not too big to have significantly different optimal antenna tilt values for these cells. Therefore, centralized learning using the same action policy at all cells can still achieve an antenna tilt configuration that produces good cell center performance but is not optimal because of the fine tuning required for the cell edge performance and further improvement in the cell center performance.

The performance of all three distributed schemes is also quite similar in the post outage scenario i.e. between snapshot 2000 and 4000. At snapshot number 2000 the graphs show a sudden decrease of performance because of the de-activation of the center eNB with all its three cells. The cells respond to this situation autonomously and adjust their antenna tilts accordingly. As a result, the performance improves and becomes nearly equivalent to the pre-outage level.

Before the outage is realized, the achieved performance level for the centralized learning is higher than other schemes, therefore, the de-activation of central eNB degrades its performance also to a lesser extent. In fact, the post-outage performance of centralized learning is equal to the pre-outage performance of the

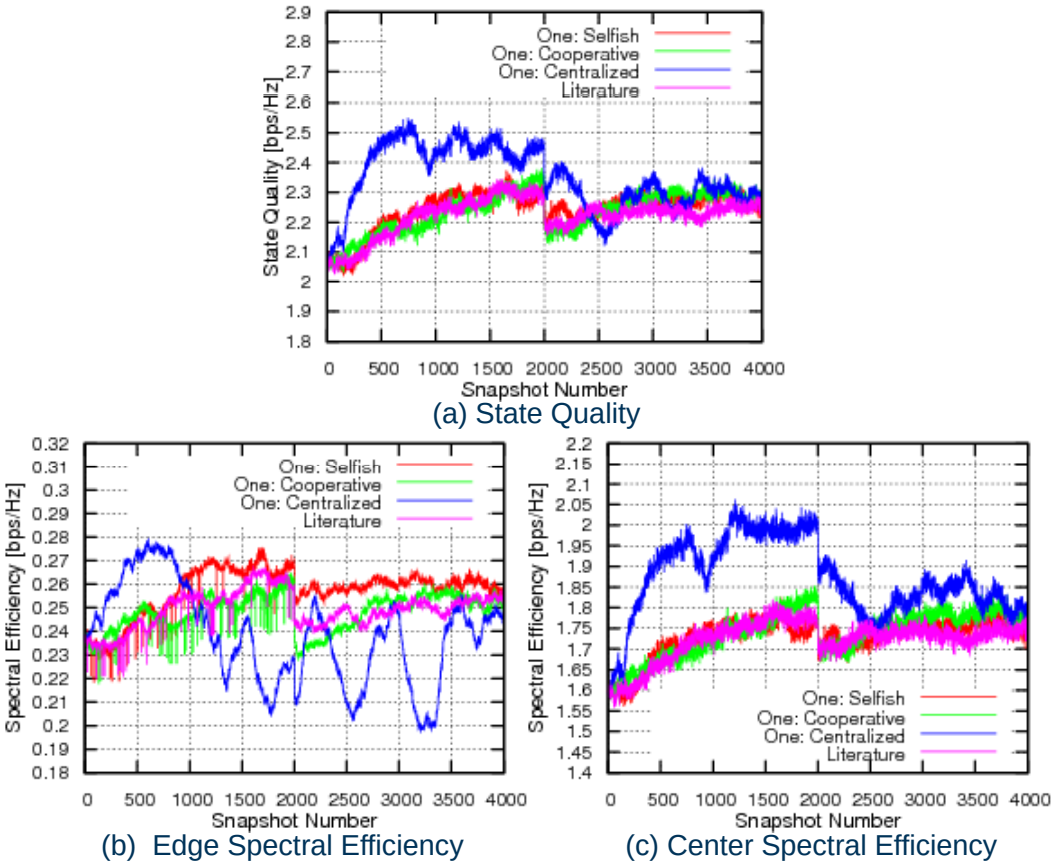

Figure 5.15: Irregular Scenario Comparative Results for One Agent Update per Snapshot

distributed schemes. However, it is not able to recover from this performance degradation and at the end of the simulation the performance of all schemes is nearly at same level. In this outage scenario, even the cell center performance is not stable for the centralized learning. Because with this outage some of the cells need to extend their coverage areas while others need to maintain their coverage areas at the same time. The central shared q-table is not optimal in this scenario and the performance gets effected as shown by the graphs.

All Cells Update per Snapshot: The results for the learning strategies with all cells update per snapshot are presented in Fig. 5.16. The ability to update the antenna tilt at all cells in all snapshots makes this scheme very quick to respond to the initial non-optimal configurations and changes in the network. However, the performance gains varies depending upon the learning strategy being used to maintain and update the q-tables.

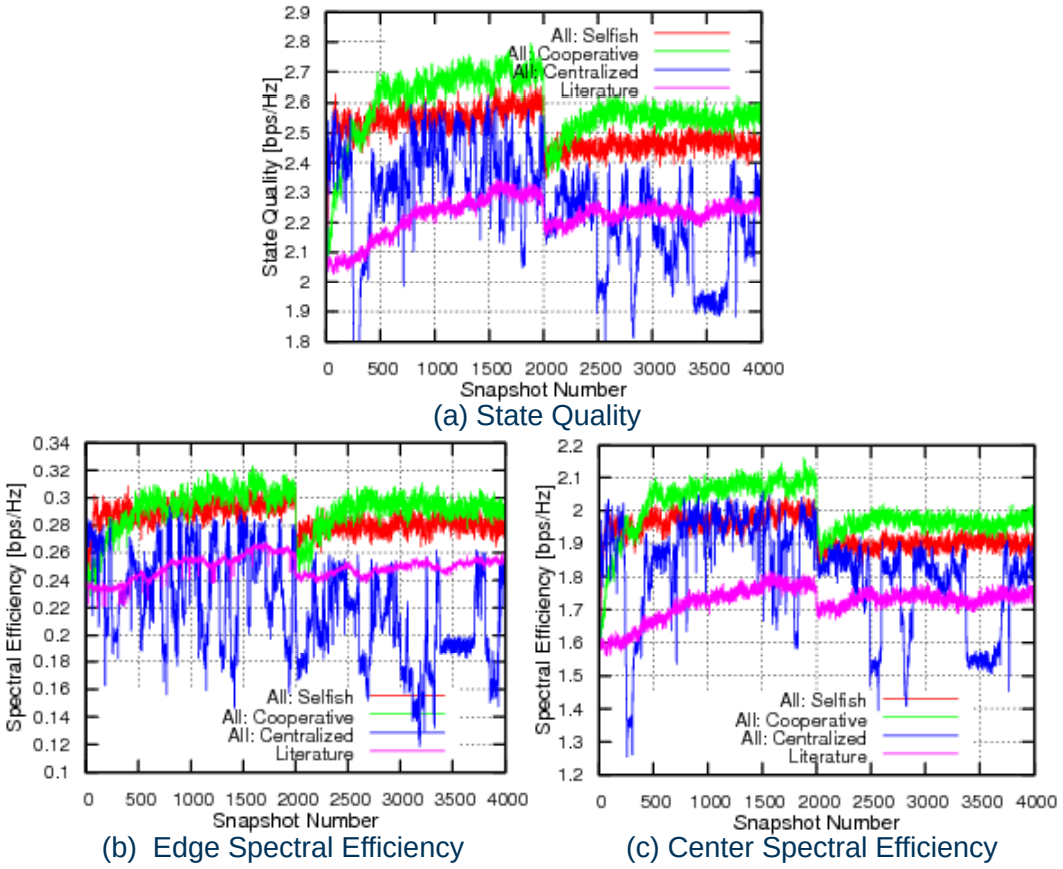

Figure 5.16: Irregular Scenario Comparative Results for All Agents Update per Snapshot

The performance gain of the selfish learning scheme saturates after the initial improvement in the first few snapshots. This is also true for its performance during the outage in the network where it is unable to significantly recover the performance degradation due to the central eNB de-activation. Because all cells take simultaneous actions and only look into their own performance improvement, a very dynamic and difficult situation arises for the learning mechanism to learn the optimal action policy in this always changing environment. Therefore, cells can although very quickly produce significant gain in the network performance but after a certain level their performance remains at same level even for very large number of snapshots.

The cooperative learning overcomes this problem by learning the actions that improve the overall performance and not just the local performance of the cell. Indirectly this schemes encourages the combinations of actions of all cells that

improve the performance of the complete network and discourage the actions of cells that improve the performance locally but not globally. The results show that this scheme can achieve performance level higher than the selfish scheme even in the presence of simultaneous actions by all cells because of that cooperation. Its performance does not saturate and steadily builds on the initial very quick performance gain achieved in the first few snapshots. Also during the center eNB outage its performance is better than the other schemes. It can recover the performance degradation due to this outage to a great extend and can even achieve performance level close to the pre-outage performance of the selfish scheme. Especially, the cell edge performance can be recovered almost completely. The cell center performance also shows some improvement but it is not equal to its own pre-outage performance. The reason is that some of the cells directly facing the outage area have to lower their antenna tilt to increase their coverage areas, which reduces the antenna gain in the vicinity of the eNB and the average is also calculated over all 57 cells even if three of them are in outage. Therefore, the values are always lower than the pre-outage values.

However, centralized learning with all cells update per snapshot does not achieve stable performance. Initially, it reacts very quickly and show comparable performance to the selfish and cooperative learning schemes but then it keeps on fluctuating. The fluctuation are even worse than the one cell update per snapshot strategy both for the cell edge and center performance because of larger amount of simultaneous actions in the network and thus the update of q-table at much greater frequency.

Cluster of Cells Update per Snapshot: Finally, the results for the learning strategies with cluster of cells update per snapshot are described in Fig. 5.17. Compared to the smaller regular hexagonal scenario this schemes is a little slower in this bigger irregular scenario. Our clustering mechanism tries to maximize the number of cells that can take an action in a snapshot while ensuring that no two adjacent neighbors take an action in same snapshot. This means for each selected cell for an update in a particular snapshot its six adjacent neighbors cannot take an action in that particular snapshot. Therefore, the number of cells which cannot take an action in a snapshot grows with the increasing network size. This slows down this strategy and the performance gains reduce. This is also visible in the graphs as the achieved performance of both selfish and cooperative schemes is slightly lower than the performance of these schemes with all cell update strategy. However, the selfish learning does not stuck at a certain level like the all cell update per snapshot strategy.

During the outage as well this strategy can only slightly recover the performance degradation because of the central eNB de-activation. The clustering mechanism does not give any priority to the cells close to the outage area but selects them randomly, this limits the ability of these cells to respond to this

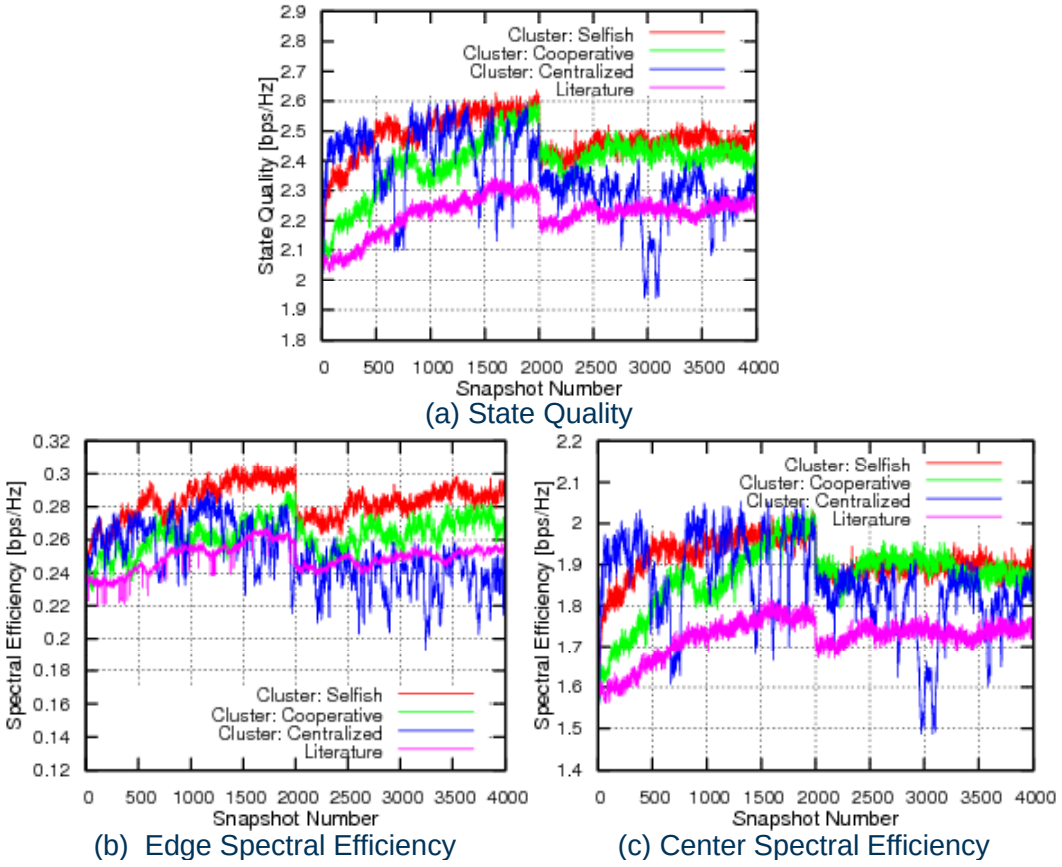

Figure 5.17: Irregular Scenario Comparative Results for Cluster of Agents Update per Snapshot

situation compared to the all cell update per snapshot strategy. Therefore, the recovery is slower and not higher enough.

The centralized scheme here also gets effected by multiple simultaneous actions and update of q-table by cells of different environments. Only the scale and frequency of the fluctuations is a little smaller than the all cell update per snapshot strategy. But they are still worse than the one cell update strategy because of more actions being taken in the network.

All the above mentioned learning strategies were also tested with different learning parameter settings and the results are given below. First the SQ_{avg} over the complete network between snapshot number 1 and 500 is presented in Table 5.10. This shows how quickly each learning strategy can overcome the initial non-optimal configuration and the achievable performance level. The more stable state behavior is given in Table 5.11, which shows the achievable SQ_{avg}

between snapshot 1500 and 2000 for this irregular scenario. Finally, Table 5.12 describes the achievable SQ_{avg} for the complete simulation run of the normal network operation between snapshot 1 and 2000.

The results show that the amount of simultaneous actions in each snapshot greatly influence the ability of the learning mechanism to respond to the initial non-optimal configuration. The one cell update strategy achieves the least gain and the all cell update strategy the largest gain for both the selfish and cooperative schemes. However, the gains for the cooperative schemes are always a little lower than the corresponding selfish scheme. The cooperative schemes need not just to identify the actions of one cell that improve its own performance but the combination of all cell's actions that improve the overall performance. Therefore, it always need more learning snapshots to achieve equivalent performance levels. The centralized learning scheme on the other hand gives quite comparable initial performance for all three learning strategies because the cells update the same shared q-table and can benefit from the learning experience of each other.

In the stable state analysis, the amount of simultaneous actions still have the same effect on the achieved performance as in the initial state. However, unlike the regular scenario here the cluster based learning scheme is not able to outperform the all cell update strategy. The clustering scheme in a way limits the adaptability of the network so the performance gain also reduces with the increasing network size. Moreover, centralized learning with all three different update strategies does not show significant improvement over the initial state results. The temporal results shown in the previous graphs in fact show a fluctuating behavior due to the single action policy being applied to cells with different local conditions that are not fully represented in the state definition.

Table 5.10: Irregular Scenario: Average State Quality [bps/Hz] Between Snapshot 1 and 500

Learning Parameters			**Selfish Learning**			**Cooperative** Learning			**Centralized Learning**		
Exploration Rate	**Learning Rate**	**Discount Factor**	**One**	**All**	**Cluster**	**One**	**All**	**Cluster**	**One**	**All**	**Cluster**
0.06	0.1	0	2.09	2.57	2.42	2.08	2.43	2.21	2.28	2.24	2.44
		0.7	2.14	2.57	2.39	2.07	2.43	2.14	2.28	2.34	2.36
	0.15	0	2.12	2.52	2.38	2.02	2.45	2.19	2.29	2.43	2.44
		0.7	2.16	2.51	2.39	2.05	2.33	2.20	2.33	2.37	2.47
0.12	0.1	0	2.03	2.50	2.37	2.03	2.44	2.20	2.33	2.39	2.47
		0.7	2.10	2.51	2.37	2.03	2.41	2.23	2.33	2.39	2.45
	0.15	0	2.08	2.53	2.38	2.02	2.46	2.34	2.24	2.38	2.42
		0.7	2.08	2.54	2.43	2.02	2.42	2.30	2.45	2.23	2.38

Table 5.11: Irregular Scenario: Average State Quality [bps/Hz] Between Snapshot 1500 and 2000

Learning Parameters			Selfish Learning			Cooperative Learning			Centralized Learning		
Exploration Rate	**Learning Rate**	**Discount Factor**	**One**	**All**	**Cluster**	**One**	**All**	**Cluster**	**One**	**All**	**Cluster**
0.06	0.1	0	2.33	2.59	2.55	2.21	2.72	2.38	2.44	2.37	2.44
		0.7	2.54	2.62	2.55	2.32	2.66	2.38	2.51	2.33	2.46
	0.15	0	2.44	2.63	2.52	2.23	2.68	2.47	2.42	2.43	2.44
		0.7	2.46	2.61	2.53	2.16	2.64	2.38	2.43	2.42	2.23
0.12	0.1	0	2.40	2.59	2.57	2.19	2.70	2.53	2.52	2.42	2.47
		0.7	2.51	2.58	2.52	2.21	2.65	2.45	2.47	2.34	2.44
	0.15	0	2.39	2.61	2.56	2.16	2.69	2.48	2.18	2.38	2.09
		0.7	2.32	2.60	2.56	2.23	2.67	2.46	2.49	2.12	2.42

Table 5.12: Irregular Scenario: Average State Quality [bps/Hz] Between Snapshot 1 and 2000

Learning Parameters			Selfish Learning			Cooperative Learning			Centralized Learning		
Exploration Rate	**Learning Rate**	**Discount Factor**	**One**	**All**	**Cluster**	**One**	**All**	**Cluster**	**One**	**All**	**Cluster**
0.06	0.1	0	2.24	2.58	2.51	2.19	2.63	2.31	2.41	2.35	2.43
		0.7	2.36	2.60	2.49	2.18	2.56	2.30	2.44	2.36	2.46
	0.15	0	2.30	2.59	2.47	2.14	2.59	2.39	2.41	2.39	2.39
		0.7	2.35	2.57	2.48	2.10	2.51	2.36	2.41	2.35	2.41
0.12	0.1	0	2.21	2.55	2.50	2.11	2.62	2.37	2.45	2.40	2.41
		0.7	2.36	2.56	2.47	2.13	2.58	2.39	2.44	2.40	2.42
	0.15	0	2.30	2.58	2.48	2.08	2.62	2.41	2.33	2.38	2.32
		0.7	2.21	2.58	2.51	2.13	2.59	2.40	2.46	2.20	2.39

5.3.2 SINR Distribution

The SINR distribution for different phases of this irregular scenario simulation with an outage is presented in Fig. 5.18. The results represent the performance of cooperative learning with all cell update strategy per snapshot, which shows the best performance among all strategies.

The initial non-optimal antenna tilt configuration produces very low *SINR* values in large portion of the simulated network as shown in Figure 5.18-a. Most of the simulated network shows *SINR* values in dark and light blue colors i.e. from 0 to -5 dB. The initial uniform antenna tilt values of 11◦ is not optimal for this irregular scenario with cells of different sizes. But this relatively large value of antenna tilt still ensures that most of the cells have a good cell center performance also at the start. There are only a small number of cells that do not have good *SINR* distribution close to the eNB deployment. The edge performance is however more severely degraded with most of the cells having low *SINR* values in large parts of the border regions.

The situation improves as cells learn their optimal action policy and adjust their antenna tilts accordingly during the first 2000 snapshots. As a result the cell

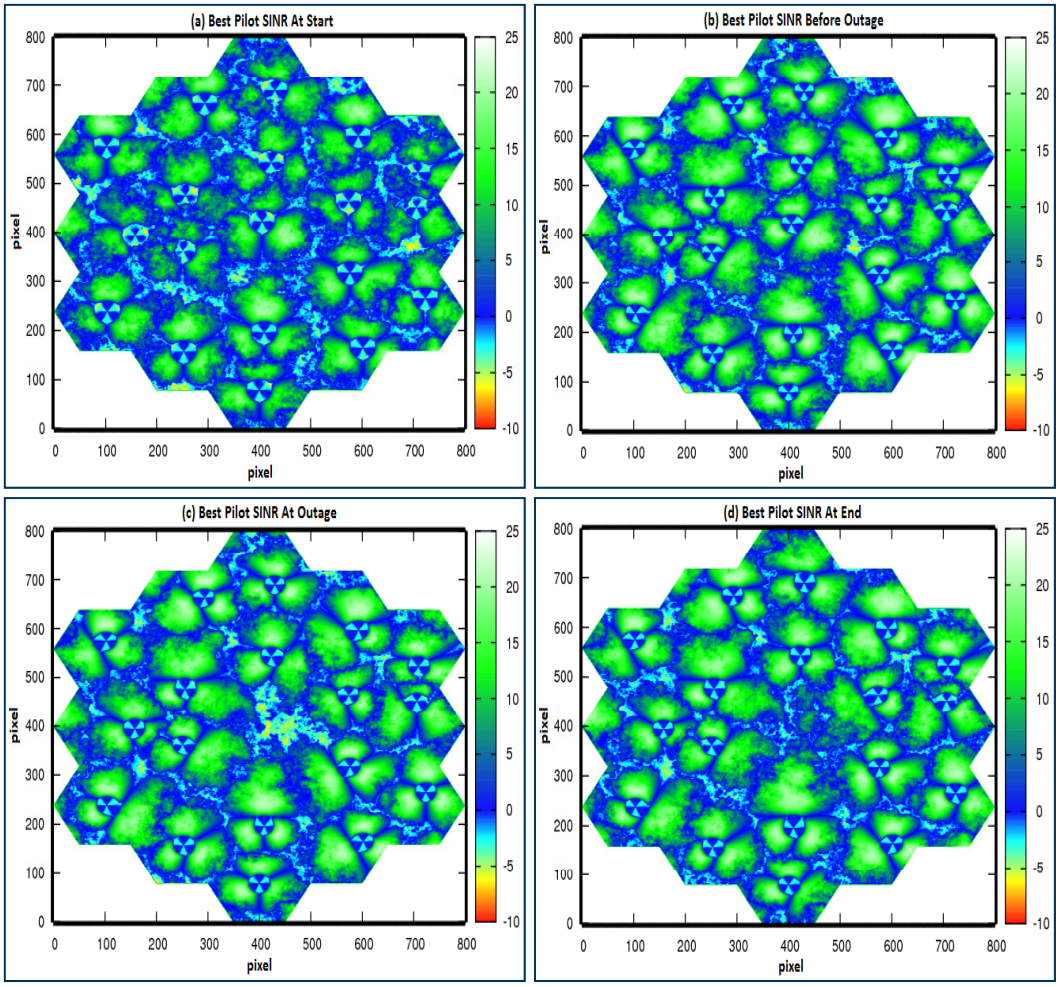

Figure 5.18: Pilot SINR During Outage Recovery in Irregular Scenario

edge performance improves and the blue regions in Fig. 5.18-b are significantly smaller at snapshot number 2000. At this point the outage is realized by de-activating the center eNB with its three cells. This degrades the performance in this region to a great extent with *SINR* values reaching below -5 dB as indicated by the yellow regions in Fig. 5.18-c.

With our learning strategies, cells can autonomously respond to this network topological change and re-adjust their antenna tilts to improve the overall coverage availability. Figure 5.18-d, shows the *SINR* distribution at snapshot number 4000. Most of the outage regions now have positive *SINR* values as indicated by dark blue and green colors in that region.

5.3.3 Antenna Tilt Variation

The individual antenna tilt variations for all cells of this scenario are presented in this section. First, the antenna tilt values of all cells at the start of the simulation i.e. $T = 1$ are presented in Fig. 5.19. At the start all the cells have a similar antenna tilt value of 11°. This is not a very good configuration for this irregular scenario and the system tries to learn a better configuration through its interaction with the environment. This can be observed in Fig. 5.20, which shows the antenna tilt values of all cells at $T = 2000$ i.e. just before the outage is introduced into the network. Most of the cells adjust their antenna tilt within 14° and 18° based on their cell size and their neighborhood configuration. The randomized deployment does not change the cell size drastically for most of the cells therefore this range is quite reasonable. However, some of the cells like 7, 15, 19, 22, 26, 29, 32, 42, 49 and 52, which experience a large ISD either maintain their initial antenna tilt values or decrease it even further to increase their coverage areas.

Finally, the antenna tilt values at the end of simulation are shown in Fig. 5.21,

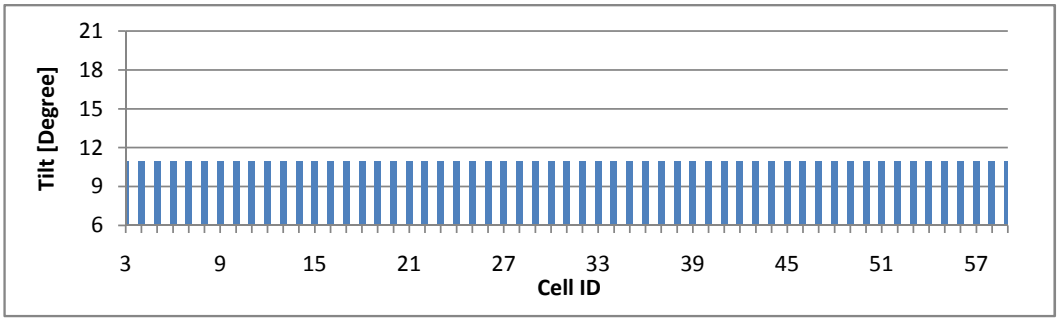

Figure 5.19: Antenna Tilt for All Cells at T=1

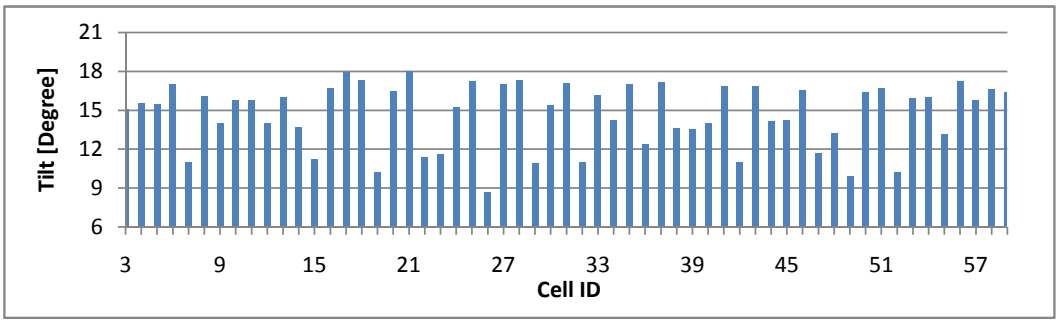

Figure 5.20: Antenna Tilt for All Cells at T=2000

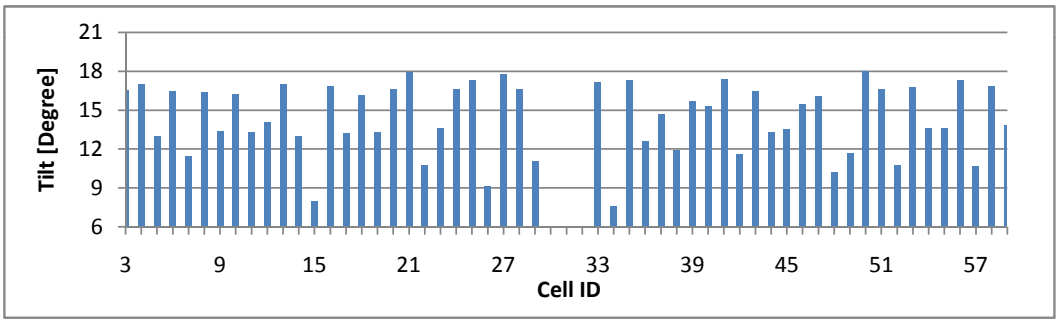

Figure 5.21: Antenna Tilt for All Cells at T=4000

which represent the response of the network to the center eNB outage. Cells 30, 31 and 32 are in outage and their values are not shown in this figure. Most of the cells like 9, 13, 21, etc. maintain their pre-outage antenna tilts values or experience only a slight change like cell 3, 6, 50 etc. But the cells adjacent to the outage areas like 15, 26 and 34 lower their antenna tilts to around 8° to increase their coverage areas and compensate for the coverage outage in their neighboring areas.

Figure 5.22 depicts the CDF plots of *SINR* distribution for four different cells at different network stages of the simulation. Two of the cells i.e. 3 and 25 are far away from the outage area and the other two cells 15 and 34 are the direct neighbors of the cells in outage. The graphs show the CDF at four distinct simulation stages i.e. at $T = 1$, the start of the simulation, at $T = 2000$, just before the outage, at $T = 2001$, right after the outage and $T = 4000$, at the end of the simulation. Ideally, we would like to have higher spectral efficiency values in most of the cell coverage area, which means the more the CDF plot is inclined to the right the better is the performance.

The results show that the cells further away from the outage area experience little of no degradation in their performance at the time of outage and they can maintain or further improve their achieved pre-outage performance level. For example, cell 3 and 25 are two such cells that are not the adjacent neighbors of the cells in outage. Because of their small coverage areas and relatively small antenna tilt values their performance is not very good initially as indicated by the red dotted line. The CDF improves as both of these cells increase their antenna tilt to around 15° and 17° respectively in the first 2000 snapshots as shown in Fig. **??**. At the outage both of these cells do not experience noticeable change in their performance as shown by the blue and green dotted lines. However, cell 3 increases its antenna tilt further up to 17° until snapshot number 4000, which brings additional improvement in the spectral efficiency distribution especially in

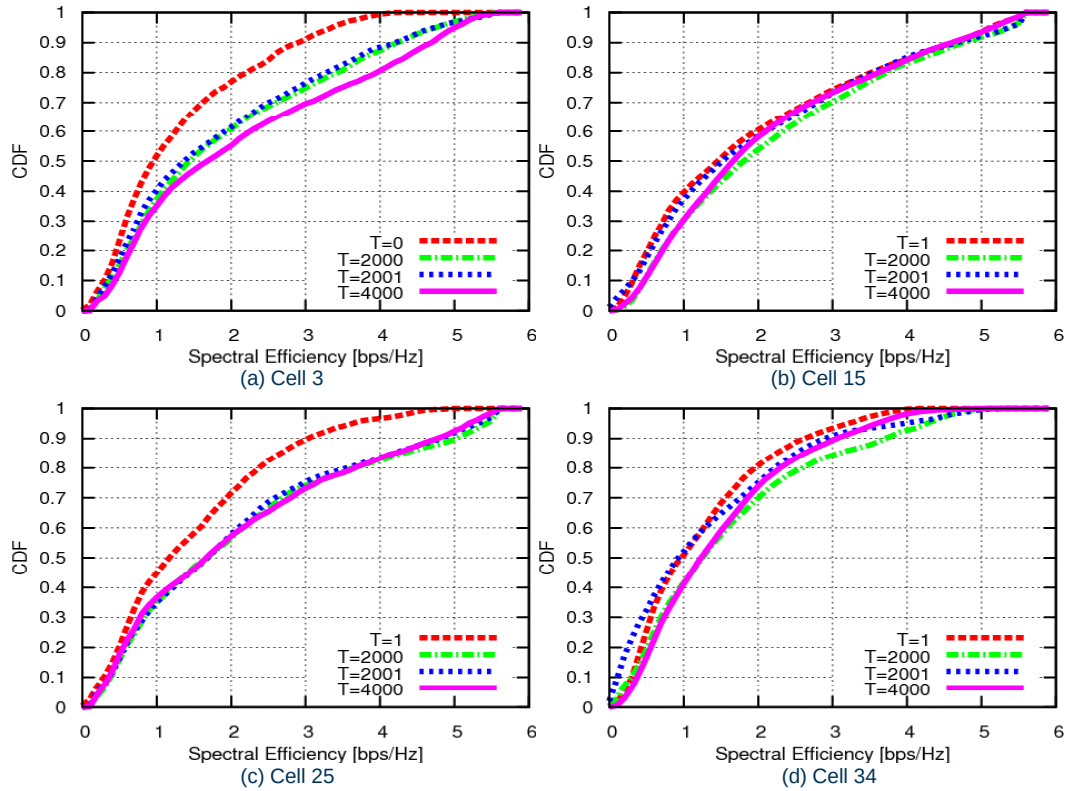

Figure 5.22: Individual TRx's SINR Distribution Variation for Irregular Scenario

the middle range.

Cells 15 and 34 are the example of cells located adjacent to the cells in outage. Because of the large coverage area of cell 15, initial 11° antenna tilt is not bad and it maintains it in the first 2000 snapshots as shown in Fig. **??**. Therefore, the CDF plots in Fig. 5.22 also do not show much variations from $T = 0$ to $T = 2000$. However, cell 34 has a much smaller coverage area and it increases its antenna tilt to around 14° until $T = 2000$, which also improves its spectral efficiency CDF. The outage effect the performance of both of these cells and the CDF degrades as indicated by the green and blue dotted lines for $T = 2000$ and $T = 2001$ respectively. But both of these cells autonomously respond to it and adjust their antenna tilts to around 8° by $T = 4000$. This increases the coverage areas of these cells and provide the required network coverage in the outage areas. As a result the CDF plots of both cells show some improvement, especially in lower and middle ranges. This is quite reasonable, because lowering of antenna tilt increases the coverage area and the antenna gains at cell edge but at the expense of lower antenna gains in the areas close to the eNB. Therefore, the higher CDF

range shows little or no further improvement.

5.4 Summary

This chapter describes the simulation results and performance comparison of the proposed learning strategies. The simulations are performed using both the regular hexagonal and irregular scenarios. For comparison purposes two reference systems are also studied, one with static network wide antenna tilt optimization for the regular hexagonal scenario and the second using a similar learning approach from the literature for both the regular and irregular scenarios.

The simulation results show that the learning speed and convergence is greatly influenced by different settings of the learning algorithm. For example the learning speed heavily depends on how many agents can take an action in each learning snapshot. Allowing only one cell to take an action in each snapshot is good for the learning as the environment is effected by only one agent and it can provide good feedback to that agent about its action. However, it extremely slows down the whole process as the network size grows.

Allowing all cells to take an action in each snapshot speed-up the learning process but at the expense of a complicated learning environment. The change in the environment from one snapshot to other is now the result of the combined actions of all the agents, which makes it difficult to evaluate the impact of each individual action. This is especially true for our problem where the actions of one agent not only modify their own environment but also impact the environment of their neighboring learning agents.

The impact of simultaneous actions of all learning agents becomes more evident in the case of selfish learning, where each agent is selfishly trying to maximize its own performance only without any consideration to its effect on the neighboring agents. In this case, the agents can stuck into a sub-optimal policy, where they are always responding to the actions of their neighbors in an oscillating behavior.

In such a highly dynamic environment, a little cooperation among the learning agents can help to achieve better performance as indicated by the results of our "Cooperative Learning" scheme. The proposed cooperative scheme involves only the sharing of local performance metrics among the learning agents and still allows them to maintain their own learning tables. This helps the individual learning agents to get a feeling of the overall performance of the neighborhood while maintaining the distributed structure. The results show that this cooperation can improve the learning performance even when all agents are taking simultaneous actions.

Because of the performance gains of cooperation among the agents, a second

level of higher cooperation among the agents was also tested in the form of proposed "Centralized Leaning". This scheme maintains a central shared q-table to share the learning experience among all agents. The results show that this knowledge sharing can help to significantly speed-up the learning process but the differences among the local conditions of individual agents need to be taken care of for better convergence. Especially, in non-homogeneous environments using the learning experience of one agent for another agent with different local conditions can make the whole system completely un-stable.

In situations where cooperation among the agents is not possible, another method to improve the learning performance can be to select the agents for an update in each learning snapshot in a way that they are far apart in the environment to have little effect on each other's actions. This was analyzed in the proposed "Clustering" scheme and the results show that it can simplify the learning environment compared to all agents taking an action in each snapshot and can achieve better performance without any cooperation among the agents. However, the gains start to decrease with the increasing network size because the amount of parallelism decreases, meaning the system requires more learning snapshots to achieve equivalent performance level.

6
Conclusions and Future Research

The increasing complexity of cellular network management and inhomogeneous traffic patterns demand an enhanced level of automation in most of the network deployment and operational phases. It can not only simplify the complex network management tasks but also improve the user quality of experience by efficient resource utilization and minimizing the network response time to the network and environmental changes.

In this thesis, we study the self-organized coverage and capacity optimization of cellular mobile networks using antenna tilt adaptations. We propose to use machine learning for this problem in order to empower the individual cells to learn from their interaction with the local environments. This helps the cells to get experienced with the passage of time and improve the overall network performance.

We model this optimization task as a multi-agent learning problem using Fuzzy Q-Learning, which is a combination of Fuzzy Logic and Reinforcement Learning-based Q-Learning. Fuzzy logic simplifies the modeling of continuous domain variables and Q-learning provides a simple yet efficient learning mechanism. We study different structural and behavioral aspect of this multi-agent learning environment in this thesis and propose several enhancements for the basic FQL algorithm for this particular optimization tasks. Especially, we look into the effect of parallel antenna tilt updates by multiple agents on the learning speed and convergence, the effect of selfish and cooperative agent behavior and the effect of distributed and centralized learning in this thesis.

For performance evaluation of our learning methods we perform system level

simulations of an LTE network. We perform the evaluations for both regular hexagonal cell deployment scenario as well as an irregular scenario. We also test our learning schemes in a dynamic network where cells are switched-off to create an outage and then switched-on to simulate the deployment of a new eNB in the network.

The simulations are performed for a worst case situation from learning point of view as no a-priori information is provided to the learning agents about the state-action pairs. However, in reality a vast experience is available in the domain of network management and can be included to further enhance the performance of these learning schemes. Here, FQL is also quite effective as this a-priori knowledge can be easily integrated by modifying the rules set or setting the q-values at the design phase.

The performance of our proposed learning mechanisms is also compared to a relevant learning method from literature for both the regular and irregular scenarios and to a static network wide antenna tilt optimization for the regular scenario. The main results from these studies are presented below.

Amount of Parallel Actions: Increasing the possibility of simultaneous actions by multiple learning agents greatly enhance the learning speed of the system. We tested three different levels of parallel actions i.e. only one agent per learning snapshot, all agents per learning snapshot and a restricted group of agents per learning snapshot. The results show that one agent per snapshot is extremely slow to react to network dynamics and the performance decreases with the increasing network size and the changes in the network topology.

Simultaneous actions of all agents makes it very quick to respond to the network and environmental changes but it can stuck into an oscillating behavior where agents are continuously responding to the actions of their neighboring agents. This effects the achievable performance gains especially if the actions of one agent not only modifies its own performance but also the performance of its neighboring agents.

In such a scenario allowing only a restricted group of agents to take parallel actions can help to simplify the learning problem and enhance the achieved performance. But the gains are limited to only small network sizes. With the increasing network sizes the restricted parallel actions limit the network adaptability and more time is required to achieve same performance gains.

Selfish vs Cooperative Learning: The learning agent's behavior also has a strong impact on the performance of overall system. With selfish learning behavior agents try to optimize their own performance without and consideration to its effect on the performance of their neighboring agents. This becomes highly relevant in problems like antenna tilt optimization where antenna tilt update at each cell not only modifies its own coverage-capacity performance but also the performance of its neighboring cells. The simulation results show that this kind

of selfish behavior only achieves a sub-optimal performance level.

The situation improves if the agents cooperate with each other and try to optimize the overall performance of the neighborhood and not just their own performance. The results show that the proposed cooperative learning scheme improves the convergence property of the learning system even in the presence of simultaneous actions of all agents in all snapshots.

Distributed vs Centralized: Distributed learning means all the agents try to learn their own optimal action policy. This can be done either selfishly or cooperatively as long as each agent maintains its own learning tables and its own action policy. This kind of learning is highly effective in non-homogeneous environments where different agents experience different local conditions. In such a scenario all agents can better exploit the local conditions and modify their parameters accordingly.

With centralized learning all agents share the same central learning table. Each action is based on this shared table and all agents update it based on their local environmental interactions. The results show that this knowledge sharing among the different learning agents greatly enhance the learning speed. Even with only one agent update per learning snapshot this technique can overcome the initial non-optimal configurations very quickly. However, it does not produce stable results. As more and more agents with diversified environments try to build a common action policy, the performance of this centralized learning deteriorates. The performance degradation becomes even worse with high degree of parallel actions by multiple agents. The performance can be improved by having more elaborate state definitions in the learning table that accurately captures all the local variations or by maintaining different tables for different types of cells like the group of cells in normal operation and the group of cells facing an outage in the neighborhood. However, both of these solutions increases the computational complexity because the state space grows exponentially with each new element being introduced.

From the overall results we can conclude that the cooperative learning scheme with all cells update per snapshot performs best among all the proposed schemes. All cell update per snapshot makes it highly responsive to the environmental and network changes and cooperation among the cells helps improve the convergence behavior. As a result this scheme can achieve up to 30 percent better performance than the learning scheme taken from literature.

Future Research: The work in this thesis can be further extended to study some more aspects of this multi-agent learning problem. Some of the ideas that would be interesting to investigate are mentioned below.

The basic principle of all learning schemes is to explore the solution space and to identify the best action for each possible state of the system. This exploration also means sometimes taking random actions, which may not be optimal accord-

ing to the present knowledge. Therefore, for the stability of the whole system it is important to look into when to explore and when to only exploit the available knowledge.

Another area of interest is to look into learning for control of multiple parameters like antenna tilt and transmit power of cells for coverage-capacity optimization. An important question in this regard is should it be a combined learning problem or should it be divided into multiple problems and the results are coordinated to stabilize the performance?

Network heterogeneity is also increasing because of the deployment of multiple radio access technologies as well as different cell structures like macro/micro/femto cells. Therefore, the multi-agent learning concepts presented in this thesis can also be extended to study different aspects of these heterogeneous networks. This would especially be interesting for the under-lay small cells like the micro- and femto-cells because the overlay macro-cells can already ensure some basic connectivity requirements and make it flexible for the small cells to explore and learn through their interaction with the environment.

Finally, coverage and capacity optimization is just one aspect of the network optimization. Cellular networks also need to be optimized for other targets like handover performance, load balancing etc. Most of these optimizations are not fully independent i.e. they either modify the same configuration parameters or try to improve the same performance metric. Therefore, one optimization task can effect the performance of others as well. In this domain the tools to study multi-agent systems and multi-agent reinforcement learning can also be beneficial to analyze the interaction among different optimization procedures and its effect on the overall performance of the network.

Bibliography

[1] 3rd Generation Partnership Project. Selection procedures for the choice of radio transmission technologies of umts (release 1999), Apr. 1998. 52, 55

[2] 3rd Generation Partnership Project. Physical layer aspect for evolved universal terrestrial radio access (utra) tr 25.814, Oct. 2006. 49, 52, 53

[3] 3rd Generation Partnership Project. Further advancements for e-utra physical layer aspects tr 36.814, Mar. 2010. 49, 51

[4] 3rd Generation Partnership Project. Self-configuring and self-optimizing network (son) use cases and solutions tr 36.902, Mar. 2011. 3, 17, 18

[5] 3rd Generation Partnership Project. Overall description ts 36.300, Mar 2012. 3

[6] R. Abou-Jaoude and C. Hartmann. Radio parameters frequent remote adaptation: Antenna tilt and pilot power for dedicated channels. In *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, pages 1 –5, sept. 2009. 17, 19

[7] R. Abou-Jaoude, N. Ulhaq, and C. Hartmann. Hsdpa throughput optimization with antenna tilt and pilot power in a moving hot-spot scenario. In *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, pages 1 –5, sept. 2009. 16, 19

[8] H. Al Hakim, H. Eckhardt, and S. Valentin. Decoupling antenna height and tilt adaptation in large cellular networks. In *Wireless Commu-*

nication Systems (ISWCS), 2011 8th International Symposium on, pages 11 –15, nov. 2011. 15

[9] S.M. Allen, S. Hurley, and R.M. Whitaker. Automated decision technology for network design in cellular communication systems. In *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, page 8 pp., jan. 2002. 17, 19

[10] NGMN. Alliance. Use cases related to self organising network: Overall description, 2007. 3

[11] E. Amaldi, A. Capone, and F. Malucelli. Radio planning and coverage optimization of 3g cellular networks. *Wireless Networks*, **14**[4]:435–447, 2008. 17, 19

[12] M. Amirijoo, L. Jorguseski, T. Kurner, R. Litjens, M. Neuland, L.C. Schmelz, and U. Turke. Cell outage management in lte networks. In *Wireless Communication Systems, 2009. ISWCS 2009. 6th International Symposium on*, pages 600 –604, sept. 2009. 17

[13] M. Amirijoo, L. Jorguseski, R. Litjens, and R. Nascimento. Effectiveness of cell outage compensation in lte networks. In *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*, pages 642 –647, jan. 2011. 17

[14] M. Amirijoo, L. Jorguseski, R. Litjens, and L.C. Schmelz. Cell outage compensation in lte networks: Algorithms and performance assessment. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1 –5, may 2011. 17

[15] F. Athley and M.N. Johansson. Impact of electrical and mechanical antenna tilt on lte downlink system performance. In *Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st*, pages 1 –5, may 2010. 15

[16] A. Awada, B. Wegmann, I. Viering, and A. Klein. A joint optimization of antenna parameters in a cellular network using taguchi's method. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1 –5, may 2011. 16, 19

[17] A. Awada, B. Wegmann, I. Viering, and A. Klein. Optimizing the radio network parameters of the long term evolution system using taguchi's method. *Vehicular Technology, IEEE Transactions on*, **60**[8]:3825 –3839, oct. 2011. 16, 19, 20

[18] T. Balch. *Behavioral diversity in learning robot teams*. PhD thesis, Georgia Institute of Technology, 1998. 73

[19] T. Balch et al. Learning roles: Behavioral diversity in robot teams. *College of Computing Technical Report GIT-CC-97-12, Georgia Institute of Technology, Atlanta, Georgia*, 1997. 73

[20] Tucker Balch. Reward and diversity in multirobot foraging. In *In IJCAI-99 Workshop on Agents Learning About, From and With other Agents*, 1999. 73

[21] D. Barker and K. Radousky. The path to lte overlay optimization. *Quintel Technology White Paper*, 2010. 14

[22] H.R. Berenji and D. Vengerov. Advantages of cooperation between reinforcement learning agents in difficult stochastic problems. In *Fuzzy Systems, 2000. FUZZ IEEE 2000. The Ninth IEEE International Conference on*, **2**, pages 871–876. IEEE, 2000. 75

[23] O. Blume, H. Eckhardt, S. Klein, E. Kuehn, and W.M. Wajda. Energy savings in mobile networks based on adaptation to traffic statistics. *Bell Labs Technical Journal*, **15**[2]:77–94, 2010. 18

[24] S.C. Bundy. Antenna downtilt effects on cdma cell-site capacity. In *Radio and Wireless Conference, 1999. RAWCON 99. 1999 IEEE*, pages 99 –102, 1999. 14

[25] L. Busoniu, R. Babuska, and B. De Schutter. A comprehensive survey of multiagent reinforcement learning. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, **38**[2]:156–172, 2008. 24, 27, 29, 75

[26] G. Calcev and M. Dillon. Antenna tilt control in cdma networks. In *Proceedings of the 2nd annual international workshop on Wireless internet*, page 25. ACM, 2006. 16, 20

[27] Ho-Shin Cho, Young-Il Kim, and Dan Keun Sung. Protection against cochannel interference from neighboring cells using down-tilting of antenna beams. In *Vehicular Technology Conference, 2001. VTC 2001 Spring. IEEE VTS 53rd*, **3**, pages 1553 –1557 vol.3, 2001. 14

[28] L. Du, J. Bigham, L. Cuthbert, C. Parini, and P. Nahi. Using dynamic sector antenna tilting control for load balancing in cellular mobile communications. In *International Conference on Telecommunications, ICT2002, Beijing*, **2**, pages 344–348. Citeseer, 2002. 16, 19

[29] H. Eckhardt, S. Klein, and M. Gruber. Vertical antenna tilt optimization for lte base stations. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1 –5, may 2011. 16, 20

[30] H.M. Elkamchouchi, H.M. Elragal, and M.A. Makar. Cellular radio network planning using particle swarm optimization. In *Radio Science Conference, 2007. NRSC 2007. National*, pages 1–8. IEEE, 2007. 17, 19

[31] FP7 E3. End-to End Efficiency. http://ict-e3.eu, 2008-2009. 3

[32] I. Forkel, A. Kemper, R. Pabst, and R. Hermans. The effect of electrical and mechanical antenna down-tilting in umts networks. In *3G Mobile Communication Technologies, 2002. Third International Conference on (Conf. Publ. No. 489)*, pages 86 – 90, may 2002. 14

[33] M. Garcia-Lozano and S. Ruiz. Effects of downtilting on rrm parameters. In *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on*, **3**, pages 2166 – 2170 Vol.3, sept. 2004. 16

[34] M. Garcia-Lozano, S. Ruiz, and J.J. Olmos. Umts optimum cell load balancing for inhomogeneous traffic patterns. In *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, **2**, pages 909 – 913 Vol. 2, sept. 2004. 16, 19

[35] A. Gerdenitsch, S. Jakl, Y.Y. Chong, and M. Toeltsch. A rule-based algorithm for common pilot channel and antenna tilt optimization in umts fdd networks. *ETRI journal*, **26**[5]:437–442, 2004. 15, 20

[36] A. Gerdenitsch, S. Jakl, M. Toeltsch, and T. Neubauer. Intelligent algorithms for system capacity optimization of umts fdd networks. In *3G Mobile Communication Technologies, 2003. 3G 2003. 4th International Conference on (Conf. Publ. No. 494)*, pages 222 – 226, june 2003. 15

[37] P.Y. Glorennec. Reinforcement learning: An overview. In *Proceedings European Symposium on Intelligent Techniques (ESIT-00), Aachen, Germany*, pages 14–15. Citeseer, 2000. 35

[38] M. Gudmundson. Correlation model for shadow fading in mobile radio systems. *Electronics letters*, **27**[23]:2145–2146, 1991. 52

[39] F. Gunnarsson, M.N. Johansson, A. Furuskar, M. Lundevall, A. Simonsson, C. Tidestav, and M. Blomgren. Downtilted base station antennas - a simulation model proposal and impact on hspa and lte

performance. In *Vehicular Technology Conference, 2008. VTC 2008-Fall. IEEE 68th*, pages 1 –5, sept. 2008. 9, 14, 15

[40] G. Hampel, K.L. Clarkson, J.D. Hobby, and P.A. Polakos. The tradeoff between coverage and capacity in dynamic optimization of 3g cellular networks. In *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th*, **2**, pages 927 – 932 Vol.2, oct. 2003. 16

[41] G. Hinton and T.J. Sejnowski. *Unsupervised learning: foundations of neural computation*. MIT press, 1999. 24

[42] T. Isotalo, J. Niemelä, and J. Lempiäinen. Electrical antenna downtilt in umts network. In *5th European Wireless Conference*, page 265271, 2004. 14

[43] S.B. Jamaa, Z. Altman, J.M. Picard, and B. Fourestie. Combined coverage and capacity optimisation for umts networks. In *Telecommunications Network Strategy and Planning Symposium. NETWORKS 2004, 11th International*, pages 175 – 178, june 2004. 19

[44] T. Janevski and V. Nikolic. Radio optimization of high speed mobile networks. In *Telecommunications (AICT), 2010 Sixth Advanced International Conference on*, pages 503 –508, may 2010. 14

[45] T. Jansen and R. Wiegand. Exploring the explorative advantage of the cooperative coevolutionary (1+ 1) ea. In *Genetic and Evolutionary ComputationGECCO 2003*, pages 197–197. Springer, 2003. 28

[46] Wu Jianhui and Yuan Dongfeng. Antenna downtilt performance in urban environments. In *Military Communications Conference, 1996. MILCOM '96, Conference Proceedings, IEEE*, **3**, pages 739 –744 vol.3, oct 1996. 13

[47] L.P. Kaelbling, M.L. Littman, and A.W. Moore. Reinforcement learning: A survey. *Arxiv preprint cs/9605103*, 1996. 24

[48] Dong Hee Kim, Dong Do Lee, Ho Joon Kim, and Keum Chan Whang. Capacity analysis of macro/microcellular cdma with power ratio control and tilted antenna. *Vehicular Technology, IEEE Transactions on*, **49**[1]:34 –42, jan 2000. 14

[49] S. Klein, I. Karla, and E. Kuehn. Potential of intra-lte, intra-frequency load balancing. In *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, pages 1 –5, may 2011. 16

[50] J. Laiho-Steffens, A. Wacker, and P. Aikio. The impact of the radio network planning and site configuration on the wcdma network capacity and quality of service. In *Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st*, **2**, pages 1006 –1010 vol.2, 2000. 14

[51] D.J.Y. Lee and Ce Xu. Mechanical antenna downtilt and its impact on system design. In *Vehicular Technology Conference, 1997, IEEE 47th*, **2**, pages 447 –451 vol.2, may 1997. 12, 13

[52] I. Luketic, D. Simunic, and T. Blajic. Optimization of coverage and capacity of self-organizing network in lte. In *MIPRO, 2011 Proceedings of the 34th International Convention*, pages 612 –617, may 2011. 17

[53] A.F. Molisch. *Wireless communications*. Wiley, 2011. xii, 48

[54] M. Mukaidono. *Fuzzy logic for beginners*. World Scientific, 2001. 34

[55] M. Naseer ul Islam, R. Abou-Jaoude, C. Hartmann, and A. Mitschele-Thiel. Self-optimization of antenna tilt and pilot power for dedicated channels. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2010 Proceedings of the 8th International Symposium on*, pages 196 –203, 31 2010-june 4 2010. 16, 19

[56] J. Niemela, T. Isotalo, J. Borkowski, and J. Lempiainen. Sensitivity of optimum downtilt angle for geographical traffic load distribution in wcdma. In *Vehicular Technology Conference, 2005. VTC-2005-Fall. 2005 IEEE 62nd*, **2**, pages 1202 – 1206, sept., 2005. 15

[57] J. Niemelä, T. Isotalo, and J. Lempiäinen. Optimum antenna downtilt angles for macrocellular wcdma network. *EURASIP Journal on Wireless Communications and Networking*, **2005**[5]:816–827, 2005. 14

[58] J. Niemela and J. Lempiainen. Impact of mechanical antenna downtilt on performance of wcdma cellular network. In *Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th*, **4**, pages 2091 – 2095 Vol.4, may 2004. 14

[59] L. Panait and S. Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, **11**[3]:387–434, 2005. 27, 28

[60] M. Pettersen, L.E. Braten, and A.G. Spilling. Automatic antenna tilt control for capacity enhancement in umts fdd. In *Vehicular Technology*

Conference, 2004. VTC2004-Fall. 2004 IEEE 60th, **1**, pages 280 – 284 Vol. 1, sept. 2004. 16, 20

[61] A.M. Rao, A. Weber, S. Gollamudi, and R. Soni. Lte and hspa+: Revolutionary and evolutionary solutions for global mobile broadband. *Bell Labs Technical Journal*, **13**[4]:7–34, 2009. 55

[62] T.S. Rappaport. *Wireless communications: principles and practice*. Prentice Hall, 2002. 52

[63] R. Razavi, S. Klein, and H. Claussen. A fuzzy reinforcement learning approach for self-optimization of coverage in lte networks. *Bell Labs Technical Journal*, **15**[3]:153–175, 2010. 62

[64] FP7 Socrates. Self-Optimisation & Self-Configuration in Wireless Networks. http://www.fp7-socrates.org/, 2008-2010. 3

[65] S. Sesia, I. Toufik, and M. Baker. *LTE-the UMTS long term evolution: from theory to practice*. Wiley, 2011. 48

[66] IKR. Simulation Library. http://www.ikr.uni-stuttgart.de. 49

[67] I. Siomina. P-cpich power and antenna tilt optimization in umts networks. In *Telecommunications, 2005. advanced industrial conference on telecommunications/service assurance with partial and intermittent resources conference/e-learning on telecommunications workshop. aict/sapir/elete 2005. proceedings*, pages 268 – 273, july 2005. 15, 18

[68] I. Siomina, P. Varbrand, and Di Yuan. An effective optimization algorithm for configuring radio base station antennas in umts networks. In *Vehicular Technology Conference, 2006. VTC-2006 Fall. 2006 IEEE 64th*, pages 1 –5, sept. 2006. 15, 19

[69] I. Siomina and Di Yuan. Enhancing hsdpa performance via automated and large-scale optimization of radio base station antenna configuration. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2061 –2065, may 2008. 16, 19

[70] Iana Siomina, Peter Varbrand, and Di Yuan. Automated optimization of service coverage and base station antenna configuration in umts networks. *Wireless Communications, IEEE*, **13**[6]:16 –25, dec. 2006. 15, 19

[71] H. Sugahara, Y. Matsuda, K. Kobayashi, Y. Watanabe, Y. Matsunaga, Kin Ming Chan, and K. Ko. Individually targeted radio network optimization. In *Network Operations and Management Symposium (NOMS), 2010 IEEE*, pages 535 –542, april 2010. 15

[72] R.S. Sutton and A.G. Barto. *Reinforcement learning: An introduction*, **1**. Cambridge Univ Press, 1998. 24, 29

[73] A. Temesvary. Self-configuration of antenna tilt and power for plug & play deployed cellular networks. In *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*, pages 1 –6, april 2009. 16, 19

[74] U. Turke and M. Koonert. Advanced site configuration techniques for automatic umts radio network design. In *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, **3**, pages 1960 – 1964 Vol. 3, may-1 june 2005. 16, 19

[75] FP7. UniverSelf Project. http://www.univerself-project.eu/, 2010. 3

[76] A. Wacker, K. Sipila, and A. Kuurne. Automated and remotely optimization of antenna subsystem based on radio network performance. In *Wireless Personal Multimedia Communications, 2002. The 5th International Symposium on*, **2**, pages 752 – 756 vol.2, oct. 2002. 15

[77] A. Wang and V. Krishnamurthy. Mobility enhanced smart antenna adaptive sectoring for uplink capacity maximization in cdma cellular network. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, **4**, page IV, may 2006. 11

[78] C.J.C.H. Watkins and P. Dayan. Q-learning. *Machine learning*, **8**[3]:279–292, 1992. 29

[79] V. Wille, M. Toril, and R. Barco. Impact of antenna downtilting on network performance in geran systems. *Communications Letters, IEEE*, **9**[7]:598 – 600, july 2005. 14

[80] G. Wilson. Electrical downtilt through beam-steering versus mechanical downtilt [base station antennas]. In *Vehicular Technology Conference, 1992, IEEE 42nd*, pages 1 –4 vol.1, may 1992. 12

[81] Jiayi Wu, J. Bigham, Peng Jiang, and J.P. Neophytou. Tilting and beam-shaping for traffic load balancing in wcdma network. In *Wireless Technology, 2006. The 9th European Conference on*, pages 63 –66, sept. 2006. 16, 20

[82] Jung-Shyr Wu, Jen-Kung Chung, and Chang-Chung Wen. Performance study of traffic balancing via antenna-tilting in cdma cellular systems. In *Vehicular Technology Conference, 1996. 'Mobile Technology for the Human Race'., IEEE 46th*, **2**, pages 1081 –1085 vol.2, apr-1 may 1996. 16

[83] Jung-Shyr Wu, Jen-Kung Chung, and Chang-Chung Wen. Hotspot traffic relief with a tilted antenna in cdma cellular networks. *Vehicular Technology, IEEE Transactions on*, **47**[1]:1 –9, feb 1998. 14

[84] Ruixiao Wu, Zhigang Wen, Chunxiao Fan, Jie Liu, and Zhenjun Ma. Self-optimization of antenna configuration in lte-advance networks for energy saving. In *Broadband Network and Multimedia Technology (ICBNMT), 2010 3rd IEEE International Conference on*, pages 529 –534, oct. 2010. 18

[85] O. Yilmaz, S. Hamalainen, and J. Hamalainen. Comparison of remote electrical and mechanical antenna downtilt performance for 3gpp lte. In *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, pages 1 –5, sept. 2009. 15

[86] O.N.C. Yilmaz, S. Hamalainen, and J. Hamalainen. Analysis of antenna parameter optimization space for 3gpp lte. In *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, pages 1 –5, sept. 2009. 14

[87] L.A. Zadeh. Fuzzy sets. *Information and control*, **8**[3]:338–353, 1965. 33

[88] L. Zordan, N. Rutazihana, and N. Engelhart. Capacity enhancement of cellular mobile network using a dynamic electrical down-tilting antenna system. In *Vehicular Technology Conference, 1999. VTC 1999 - Fall. IEEE VTS 50th*, **3**, pages 1915 –1918 vol.3, 1999. 13