UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Self Organisation for 4G/5G Networks

by Jessica Moysen Cortés

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Departament de Teoria del Senyal i Comunicacions

CTTC
Centre Tecnològic de Telecomunicacions de Catalunya

Self Organisation for 4G/5G Networks

PhD Thesis Dissertation

By

Jessica Moysen

Submitted to the Universitat Politècnica de Catalunya (UPC)
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Barcelona, July 2016

Supervised by Dr. Lorenza Giupponi
PhD program on Signal Theory and Communications

It is not a move, even the best move,

that you must seek, but a realisable plan.

Znosko Borovsky

The research leading to these results has received funding from the Spanish Ministry of Economy and Competitiveness FPI research programme (BES-2011-047309) under the grant TEC2010-21100.

Abstract

Nowadays, the rapid growth of mobile communications is changing the world towards a fully connected society. Current 4G networks account for almost half of total mobile traffic, and in the forthcoming years the overall mobile data traffic is expected to dramatically increase. To manage this increase in data traffic, operators adopt network topologies such as Heterogeneous Networks (HetNets). Thus, operators can deploy hundreds of small cells for each macro cell, allowing them to reduce coverage holes and/or lack of capacity. The advent of this technology, is expected to tremendously increase the number of nodes in this new ecosystem, so that traditional network management activities based on, e.g., classic manual and field trial design approaches are just not be viable anymore. As a consequence, the academic literature has dedicated a significant amount of effort to Self-Organising Network (SON) algorithms. These solutions aim to bring intelligence and autonomous adaptability into cellular networks, thereby reducing capital and operation expenditures (CAPEX/OPEX). Another aspect to take into account is that, these type of networks generate a large amount of data during their normal operation in the form of control, management and data measurements. This data is expected to increase in 5G due to different aspects, such as densification, heterogeneity in layers and technologies, additional control and management complexity in Network Functions Virtualisation (NFV) and Software Defined Network (SDN), and the advent of the Internet of Things (IoT), among others. In this context, operators face the challenge of designing efficient technologies, while introducing new services, reaching challenges in terms of customer satisfaction, and where the global objective of an operator is to build networks, which are self-aware, self-adaptive, and intelligent.

This dissertation provides a contribution to the design, analysis, and evaluation of SON solutions to improve network operator performance, expenses, and users' experience, by making the network more self-adaptive and intelligent. It also provides a contribution to the

design of a self-aware network planning tool, which allows to predict the Quality of Service (QoS) offered to end-users, based on data already available in the network.

The main thesis contributions are divided into two parts. The first part presents a novel functional architecture based on an automatic and self-organised Reinforcement Learning (RL) based approach to model SON functionalities, in which the main task is the self-coordination of different actions taken by different SON functions to be automatically executed in a self-organised realistic Long Term Evolution (LTE) network. The proposed approach introduces a new paradigm to deal with the conflicts generated by the concurrent execution of multiple SON functions, revealing that the proposed approach is general enough to model all the SON functions and their derived conflicts. The second part of the thesis is dedicated to the problem of QoS prediction. In particular, we aim at finding patterns of knowledge from physical layer data acquired from heterogeneous LTE networks. We propose an approach that not only is able to verify the QoS level experienced by the users, through physical layer measurements of the UEs, but it is also able to predict it based on measurements collected at different time, and from different regions of the heterogeneous network. We propose then to make predictions independently of the physical location, in order to exploit the experience gained in other sectors of the network, to properly dimension and deploy heterogeneous nodes. In this context, we use Machine Learning (ML) as a tool to allow the network to learn from experience, improving performances, and big data analytics to drive the network from reactive to predictive.

During the elaboration of this thesis, two general key conclusions have been extracted. First, we highlight the importance of designing efficient SON algorithms to effectively deal with several challenges, such as, the most appropriate location of SON functions and algorithms to solve properly the distributed vs. centralized SON implementation issue, or the solution of conflicts among SON functions executed in different nodes, or networks. Second, in terms of network planning tools, different tools can be found covering a wide range of platform systems and applications oriented to the industry as well as for research purposes. In this context, research solutions are continuously undergoing major changes, where one of the major drivers is to present more cost-effective solutions.

Resumen

Hoy en día, el rápido crecimiento de las comunicaciones móviles están cambiando el mundo hacia una sociedad completamente conectada. Las redes 4G actuales representan casi la mitad del tráfico móvil total, y en los próximos años se espera que el tráfico total de los dispositivos móviles aumente drásticamente. Para gestionar este incremento de tráfico de datos, los operadores adoptan tecnologías de redes como las redes heterogéneas (HetNet). De esta manera, los operadores pueden desplegar centenares de pequeñas celdas por cada macro celda, permitiendo reducir zonas sin cobertura y/o falta de capacidad. Con la introducción de esta tecnología, se espera que incremente de manera sustancial el número de nodos en el nuevo ecosistema, de manera que las actividades de gestión de las redes tradicionales, basadas en, por ejemplo, el diseño manual sean inviables. Como consecuencia, la literatura académica ha dedicado un esfuerzo significativo al diseño de algorítmos de redes auto-organizadas (SON). Estas soluciones tienen como objetivo introducir inteligencia y capacidad autónoma a las redes móviles, reduciendo la capacidad y costes operativos (CAPES/OPEX). Otro aspecto a tener en cuenta es que este tipo de redes generan una gran cantidad de datos durante su funcionamiento habitual, en forma de medidas de control y gestión de datos. Se espera que estos datos incrementen con la tecnología 5G, debido a diferentes aspectos como los son la densificación de redes heterogéneas, la complejidad adicional en el control y la gestión de la virtualización de las funciones de redes (NFV) y las redes definidas por software (SDN), así como la llegada del internet de las cosas (IoT), entre otros. En este contexto, los operadores se enfrentan al reto de diseñar tecnologías eficientes, mientras introducen nuevos servicios, consiguiendo objetivos en términos de satisfacción del cliente, en donde el objetivo global del operador es la construcción de redes auto-conscientes, auto-adaptables e inteligentes.

Esta tesis ofrece una contribución al diseño y evaluación de soluciones SON para mejorar

el rendimiento de las redes, los costes y la experiencia de los usuarios, consiguiendo que la red sea auto-adaptable e inteligente. Así mismo, proporciona una contribución al diseño de una herramienta de planificación de red auto-consciente, que permita predecir la calidad de servicio brindada a los usuarios finales, basada en la explotación de datos disponibles en la red.

Las contribuciones principales de esta tesis se dividen en dos partes. La primera parte presenta una nueva arquitectura funcional basada en el aprendizaje reforzado y auto-organizado, enfocado en modelar funcionalidades SON, donde la tarea principal es la auto-coordinación de las diferentes acciones llevadas a cabo por las diferentes funciones SON ejecutadas de forma automática en una red auto-organizada LTE. El esquema propuesto introduce un nuevo paradigma para afrontar los conflictos generados por la ejecución simulatanéa de múltiples funciones SON y los conflictos derivados. La segunda parte de la tesis está enfocada al problema de la predicción de la calidad de servicio. En particular, nuestro objetivo es encontrar patrones de comportamiento a partir de datos en la capa física de redes LTE heterogéneas. Proponemos un enfoque que no solo es capaz de verificar el nivel de calidad de servicio experimentado por los usuarios, a través de las medidas de la capa física de los usuarios, sino que también es capaz de predecir basado en las medidas adquiridas en diferentes instantes, y diferentes regiones de la red heterogénea. Proponemos por lo tanto hacer predicciones independiente de la ubicacíon física, aprovechando la experiencia adquirida en otros sectores de la red, para dimensionar y desplegar redes heterogéneas de manera correcta. En este contexto, utilizamos el aprendizaje máquina (ML) como una herramienta que permite que la red aprenda de la experiencia pasada, mejorando el rendimiento, y el análisis de grandes volumenes de datos para conducir la red de reactiva a predictiva.

Durante la elaboración de esta tesis, se han extraído dos principales conclusiones. En primer lugar, destacamos la importancia de diseñar algorítmos SON eficientes para afrontar de manera eficaz diversos retos, como lo son la ubicación más adequada de funciones SON y algorítmos para resolver de manera adecuado el problema de implementación distributiva o centralizada, o la solución de conflictos entre funciones SON executadas en diferentes nodos o redes. En segundo lugar, en términos de planificación de redes, se pueden encontrar diferente herramientas cubriendo una amplia gama de sistemas y aplicaciones orientadas a la industria, así como para fines de investigación. En este contexto, las soluciones investigadas están sometidas continuamente a cambios importantes, donde uno de los principales retos es presentar soluciones más rentables.

Resum

Avui en dia, el ràpid creixement de les comunicacions mòbils està canviant el món cap a una societat completament connectada. Les xarxes 4G actuals representan casi la m trànsit mòbil total, i en els propers anys s'espera que el trànsit total de dades mòbils augmenti dràsticament. Per gestionar aquest increment de trànsit de dades, els operadors adopten topologies de xarxa com ara les xarxes heterogènies (HetNets). D'aquesta manera, els operadors poden desplegar centenars de cel·les petites per a cada cel·la macro, permetent reduir forats en la cobertura i/o la manca de capacitat. Amb l'arribada d'aquesta tecnologia, s'espera que incrementi enormement el nombre de nodes en el nou ecosistema, de manera que les activitats de gestió de xarxa tradicionals, basades en, per exemple, el disseny manual i els assaigs de camp esdevenen simplement inviables. Com a conseqüència, la literatura acadèmica ha dedicat una quantitat significativa d'esforç als algorismes de xarxa auto-organitzada (SON). Aquestes solucions tenen com a objectiu portar la intel·ligència i capacitat d'adaptació autònoma a les xarxes mòbils, reduint el capital i les despeses operatives (CAPES /OPEX). Un altre aspecte a tenir en compte és que aquest tipus de xarxes generen una gran quantitat de dades durant el seu funcionament habitual, en forma de mesuraments de control, gestió i dades. S'espera que aquestes dades incrementin amb la tecnologia 5G, degut a diferents aspectes com ara la densificació, l'heterogeneïtat en capes i tecnologies, la complexitat addicional en el control i la gestió de la virtualització de les funcions de xarxa (NFV) i xarxes definides per software (SDN), i l'adveniment de l'internet de les coses (IoT), entre d'altres. En aquest context, els operadors s'enfronten al repte de dissenyar tecnologies eficients, mentre introdueixen nous serveis, aconseguint objectius en termes de satisfacció del client, i on l'objectiu global d'un operador és la construcció de xarxes que són auto-conscients, auto-adaptables i intel·ligents. Aquesta tesis ofereix una contribució al disseny, l'anàlisi i l'avaluació de les solucions SON per millorar el rendiment de l'operador de xarxa, les

despeses i l'experiència dels usuaris, fent que la xarxa sigui més auto-adaptable i intel·ligent. També proporciona una contribució al disseny d'una eina de planificació de xarxa auto-conscient, el que permet predir la qualitat de servei (QoS) oferta als usuaris finals, basada en dades ja disponibles a la xarxa. Les contribucions principals d'aquesta tesis es divideixen en dues parts. La primera part presenta una nova arquitectura funcional basada en un aprenentatge per reforç (RL) automàtic i auto-organitzat, enfocat en modelar funcionalitats SON, on la tasca principal és l'auto-coordinació de les diferents accions dutes a terme per les diferents funcions SON a ser executades de forma automàtica en una xarxa Long Term Evolution (LTE) auto-organitzada. L'enfocament proposat introdueix un nou paradigma per fer front als conflictes generats per l'execució simultània de múltiples funcions SON, revelant que l'enfocament proposat és prou general per modelar totes les funcions SON i els seus conflictes derivats. La segona part de la tesis està dedicada al problema de la predicció de la qualitat de servei. En particular, el nostre objectiu és trobar patrons de coneixement a partir de dades de la capa física adquirides de xarxes LTE heterogènies. Proposem un enfocament que no només és capaç de verificar el nivell de QoS experimentat pels usuaris, a través de mesuraments de la capa física dels UEs, sinó que també és capaç de predir-ho basant-se en mesuraments adquirits en diferents instants, i de diferents regions de la xarxa heterogènia. Proposem per tant fer prediccions amb independència de la ubicació física, aprofitant l'experiència adquirida en altres sectors de la xarxa, per dimensionar i desplegar nodes heterogenis correctament. En aquest context, utilitzem l'aprenentatge automàtic (ML) com a eina per permetre que la xarxa aprengui de l'experiència, millorant el rendiment, i l'anàlisi de grans volums de dades per a conduir la xarxa de reactiva a predictiva. Durant l'elaboració d'aquesta tesis, s'han extret dues conclusions principals clau. En primer lloc, destaquem la importància de dissenyar algorismes SON eficients per fer front eficaçment a diversos reptes, com ara la ubicació més adequada de funcions SON i algorismes per resoldre adequadament el problema d'implementació distribuïda o centralitzada, o la solució de conflictes entre funcions SON executades a diferents nodes o xarxes. En segon lloc, en termes d'eines de planificació de xarxes, es poden trobar diferents eines cobrint una àmplia gamma de sistemes i aplicacions orientades a la indústria, així com per a fins d'investigació. En aquest context, les solucions investigades són sotmeses contínuament a canvis importants, on un del principals impulsors és presentar solucions més rentables.

Acknowledgements

Though only my name appears on the cover of this dissertation, a great many people have contributed to its production. In the first place, I am grateful to the CTTC for giving me the opportunity to complete my doctoral studies. I would also like to sincerely thank my supervisor, Dr. Lorenza Giupponi, for her support and guidance all these years.

I want to express my appreciation to all the people I have worked with during this time. In particular Ana Maria, Pol, Moises, Angelous, and Giussepe. I also want to owe my deepest gratitude to Laia, Maria, Laura, Miquel, and Onur my friends and colleagues who helped my final effort despite the difficulties we were facing together.

Certainly, "thank you" is not enough to express my gratitude to God, for the family He provided, and the friends He has sent my way. I would like to thank to my parents, Holger and Amanda, who have always supported and believed in me, and for provide me with all the means to get where I am today, and my brothers, Holger Ludwig and Cheryl for their unconditional love.

Finally, I would like to dedicate this thesis to my sister Amanda, who stood by me all this time, always being supportive and profound understanding, especially in the difficult moments. This thesis is also yours.

Jessica Moysen
Barcelona, Spain
July, 2016

List of Publications

Journals

[J1] **Jessica Moysen**, Lorenza Giupponi, Josep Mangues-Bafalluy, "A Mobile Network Planning Tool based on Data Analytics", *to be submitted to Mobile Information Systems, Special Issue "Design, Dimensioning, and Optimization of 4G/5G Wireless Communication Networks"*, 2016.

[J2] **Jessica Moysen** and Lorenza Giupponi, "Machine Learning Enabled SONs: a Survey", *to be submitted to the IEEE Communications Surveys & Tutorials*, 2016.

[J3] **Jessica Moysen** and Lorenza Giupponi, "Self-coordination of parameter conflicts in D-SON architectures: a Markov Decision Process framework", *EURASIP Journal on Wireless Communications and Networking*, 2015.

[J4] O. Onireti, A. Zoha, **J. Moysen**, A. Imran, L. Giupponi, M. Ali Imran, A. Abu-Dayya, "A Cell Outage Management Framework for Dense Heterogeneous Networks" , *IEEE Transactions on Vehicular Technology*, Vol. 64, No. 4, pp. 2097-2113, April 2015.

International Conferences

[C1] **Jessica Moysen**, Lorenza Giupponi, Josep Mangues-Bafalluy, "A Machine Learning enabled network Planning tool", *in proc. of the 27th IEEE Personal, Indoor and Mobile Radio Communications (PIMRC)*, Valencia, Spain, 2016.

[C2] **Jessica Moysen**, Lorenza Giupponi, Josep Mangues-Bafalluy, "On the Potential of Ensemble Regression Techniques for Future Mobile Network Planning", *in proc. of the 21th IEEE Symposium on Computers and Communications (ISCC), Messina, Italy*, 2016.

- [C3] **Jessica Moysen**, Lorenza Giupponi, Nicola Baldo, Josep Mangues-Bafalluy, "Predicting QoS in LTE HetNets based on location-independent UE measurement", *in proc. of the 20th IEEE Computer Aided Modelling and Design of Communication Links and Networks (CAMAD), Guildford, UK*, 2015.
- [C4] **Jessica Moysen** and Lorenza Giupponi, "Self Coordination among SON functions in LTE Heterogeneous Networks", *in proc. of the 81th IEEE Vehicular Technology Conference (VTC Spring), Glasgow, Scotland*, 2015.
- [C5] **Jessica Moysen** and Lorenza Giupponi, "A Reinforcement Learning based solution for Self-Healing in LTE networks", *in proc. of the 80th IEEE Vehicular Technology Conference (VTC Fall)*, Vancouver, Canada, 2014.
- [C6] **Jessica Moysen** and Lorenza Giupponi, "A Functional Architecture for Self-Coordination in LTE networks", *in proc. of the 20th IEEE European Wireless (EW)*, Barcelona, Spain, 2014.

Contents

List of Figures

List of Tables

Notation

$\mathbf{x}$	Vector
$\mathbf{x}^T$	Transpose vector
$\hat{y}$	Predicted output
$\mathcal{S}$	Set of possible states of the environment
$\mathcal{A}$	Set of possible actions
$\mathcal{R}$	Reward function
$\mathbb{E}[\cdot]$	Expected value
π	Policy
π^*	Optimal policy
$\sum$	Summation
τ	Softmax action selection policy temperature
γ	Discount factor
β	Learning rate

Chapter 1

Introduction

The secret of getting ahead is getting started. **Mark Twain**

Contents

1.1 Motivation

In the coming years, future mobile networks will be infinitely more complex than nowadays. Future next generation networks will consist of numerous coexisting Radio Access Technologies (RAT)s, and multiple cell infrastructures, ranging from classical macrocells, to micro, pico and femtocells. This heterogeneous network structure is in charge of providing efficient and ubiquitous broadband wireless access to a wide variety of advanced multimedia applications. In this context, network operators face the challenge to design efficient technologies, while introducing new services and achieving goals in terms of customer satisfaction. The network becomes more difficult to be managed, since it consists of heterogeneous segments where radio and network resources cannot be easily managed from a centralized and distributed perspective, so network designers find themselves thinking how to face the limitations of current networks.

The objectives when designing this kind of networks can then be classified at high level as short-term e.g., by maximizing the revenues, minimizing the Capital Expenditure (CAPEX) and the Operational Expenditure (OPEX), reducing the energy consumption, optimising the spectral efficiency, and as medium-term e.g., by decreasing the churning rate, or building new infrastructures. It would be particularly interesting if these approaches could be achieved considering minimal human intervention, since this would definitely ease the network management and reduce the corresponding costs.

In this context, Self-Organising Network (SON) is a common term used to refer to mobile network automation and minimization of human intervention in the cellular/wireless network management. This concept has been introduced by 3GPP in Release 8 and it has been expanding across subsequent releases. 3rd Generation Partnership Project (3GPP) work is inspired by a set of requirements defined by the operators' Alliance Next Generation Mobile Networks (NGMN). The main objective of SON can be roughly classified into three main points: 1) to bring intelligence and autonomous adaptability into cellular networks, 2) to reduce capital and operation expenditures (CAPEX/OPEX), 3) to enhance network performances in terms of network capacity, coverage, offered service/experience, etc. SON is considered today as a driving technology that aims at improving spectral efficiency, simplifying management, and reducing the operation costs of the next generation Radio Access Networks (RANs). The overall complex SON problem has been decomposed in a set of useful use cases, which have been identified by 3GPP, NGMN, 5G Infrastructure Public Private Partnership (5GPPP) and different EU projects [1–6]. The academic literature has dedicated significant effort to SON algorithms in the context of the above mentioned use cases, providing smart solutions to optimise network operator performance, expenses and users' experience. Many of these works are already reviewed here [7]. The market also offers already complete sets of SON solutions, (e.g. [8–11], among others) many products have been advertised and presented in Mobile World Congress (MWC) 2016 [12, 13]. For example, AirHop's eSON from Jio & AirHop communications [11], which employs a multi-vendor, multi-technology, real-time SON solution based on scalable and virtualised software platform has recently been awarded for the 2016 Small Cell Forum HetNet management software and service award [14].

However, to the best of our knowledge, the SON solutions available in the market are mainly based on heuristics, the automated information processing is usually limited to low complexity solutions like triggering, many operations are still done manually (e.g. network faults are usually fixed directly by engineers), SON solutions do not really capitalize on the huge amount of information that is available in mobile networks to build next generation network management solutions, and several open challenges are still unsolved, like the problem of coordination of SON functions [15, 16], or the proper solution of the trade-off between centralized and distributed SON implementations [17].

Another aspect that should be considered is that, as we observed in [18] a huge amount of data is currently already generated in 4G networks during normal operations by control and management functions, and more is expected to come in 5G networks due to the densification process [19], heterogeneity in layers and technologies, the additional control and management complexity in NFV and SDN architectures, the advent of Machine to Machine (M2M) and IoT paradigms, the increasing variety of application and services, each with distinct traffic patterns and QoS/Quality of Experience (QoE) requirements, etc. 5G network management is expected to provide a whole new set of challenges due to 1) the need to manage future network complexity, due to ultra dense deployments, heterogeneous nodes, networks, applications, RANs coexisting in the same setting, 2) the need to mange very dynamic networks where part of the nodes is controlled directly by the users (e.g. femtocells), energy saving policies are in place generating a fluctuating number of nodes, active antennas are a reality, etc. 3) the need to support 1000x traffic, and 10x users, and improve energy efficiency, 4) the need to improve the experience of the users by enabling Gbps speeds, and highly reduced latency, 5) the need to manage new virtualised architectures and heterogeneous spectrum Access privileges through novel LTE-Unlicensed (LTE-U) and Licensed Assisted Access (LAA) paradigms.

In this challenging context, we believe that the use of SON and of smart network management policies is crucial and inevitable for operators running multi-RAT, multi-vendor, multi-layer networks, where an overwhelming number of parameters needs to be configured and optimised. The high level objective is to make the networks 1) more self-aware, by exploiting the information already available in the network to gain experience in the network management, 2) more self-adaptive, by exploiting intelligent control decisions procedures which allow to automatize the decision processes based on the experience.

For this reason, Big Data analytics are currently receiving big attention in research and in the market, due to their capability of providing insightful information from the analysis of data already available to operators. In this context, a lot of research is being carried out and interesting results have been drawn in areas of; (1) *Mobile data to network management*, where the measurements available to operators can be used to optimise the network performance, develop fault detection, identify coverage holes, sleeping cells or cell in outage, among others. Data also can be exploited to find patterns of users, and take better decisions in terms of handovers, resource allocation, switching on/off base stations, etc., (2) *Network Aware business Intelligent*, where the objective is that a proactive business customer service takes more informed business decisions. That is, the service can be aware of the problem before the user calls to complain, but also, it can make better decisions, e.g., the operator could find out that insufficient network infrastructure is the source of the problem. Therefore, by investing in infrastructure, it can add more value to the customers, and (3) *Open mobile network data*, where the mobile network data, like, social media feeds and data coming from sensors and applications is exported to third parties for objectives like smart cities application, optimization of public transport network, but also could be exported to third parties to provide location, services, context aware content, among others [18]. Among the three of them, we focus on the study and analysis of mobile data to network management.

1.2 Contribution

In this thesis we focus on addressing all the open aspects described in previous section. First, we address open issues in SON. While self-optimisation and self-configuration are well addresses, we pay much attention to self-healing and to the open issue of self-coordination. We resort on machine learning approaches, to build solutions that learn from experience, and of providing decisions adapting to the evolution of the scenario.

The general wireless framework is a Heterogeneous Network (HetNet), where traditional macrocell networks coexist with small cells networks. In particular, we rely on RL theory to take advantage of Temporal Difference (TD) learning to implement autonomous SON functions in wireless dynamic environments that we cannot model theoretically. We then propose a functional architecture and a theoretical framework based on the theory of Markov Decision Process (MDP)s for the self-coordination of different actions taken by different SON functions.

In addition to that, we study the benefit that learning from data already available in the network may have. Then, we focus on designing a tool for QoS prediction, able to assist network operators in smartly planning future dense deployments. The objective is to predict QoS in a given area, given a certain deployment where the interference patterns are extremely variable due to the very high frequency reuse. 3GPP already provides an interesting data base to collect useful data for the purpose of QoS estimation. The Minimization of Drive Tests (MDT) feature has been introduced by 3GPP since Release 10. Among the targets, there are the standardization of solutions for coverage optimization, mobility, capacity optimization, parametrization of common channels, and QoS verification [20]. Since operators are also interested in estimating QoS performance, in Release 11, the MDT functionality has been enhanced to properly dimension and plan the network by collecting measurements indicating throughput and connectivity issues [21]. We exploit these data and present a Big Data approach to make the networks more self-aware, by exploiting the information already available in the network to properly dimension and deploy heterogeneous nodes.

In this challenging context, we believe that machine learning can be effectively used to allow the network to learn from experience, while improving performance. We focus on supervised learning and compare results from different regression techniques, for different amounts and kinds of input features selected by applying dimensionality reduction techniques. Additionally, we build ensemble methods that combine multiple learners to enhance the performance of each regression analysis in a reduced space. We then propose a smart network planning tool based on Big Data analysis through regression techniques in a reduce space.

The main contributions and the structure of this dissertation will be discussed in detail in the following section.

1.3 PhD Thesis Overview

In this section, we present the topic of each chapter and outline the contributions that are included in the present dissertation.

In **Chapter 2**, the background information which is required for the better understanding of the thesis is provided. At the beginning of the chapter, the basic concepts and use cases of SON are briefly presented, and the evolution of SON in 3GPP is included. Moreover, the chapter gives an overview of the key concepts in ML, but it also describes different approaches, and it elaborates the general objective of applying ML techniques in SON. Finally, it reviews SON's recent work in the area of ML. It provides a description of different ML approaches applied as a current solutions for different SON uses cases.
The publication contained in this chapter is listed here:

- Jessica Moysen and Lorenza Giupponi, "Machine Learning Enabled SONs: a Survey", *to be submitted to the IEEE Communications Surveys & Tutorials*, 2016.

In **Chapter 3**, an automatic and self-organised RL based approach for COC is presented. It provides the detail of the scenario, and it introduces technical details of the learning approach to successfully recover users from outage. Finally, an evaluation of the proposed approach on the ns3 LTE-EPC Network Simulator (LENA) platform based on 3GPP component LTE is provided.
The publications contained in this chapter are listed here:

- O. Onireti, A. Zoha, J. Moysen, A. Imran, L. Giupponi, M. Ali Imran, A. Abu-Dayya, "A

Cell Outage Management Framework for Dense Heterogeneous Networks", *IEEE Transactions on Vehicular Technology*, Vol. 64, No. 4, pp. 2097-2113, April 2015.

- Jessica Moysen and Lorenza Giupponi, "A Reinforcement Learning based solution for Self-Healing in LTE networks", *in proc. of the 80th IEEE Vehicular Technology Conference (VTC Fall)*, Vancouver, Canada, 2014.

In **Chapter 4**, a Distributed SON (D-SON) architecture for the solution of SON conflicts is presented. A theoretical framework based on the theory of MDPs for the self-coordination of different actions taken by different SON functions is proposed. Moreover, an example about how the proposed functional architecture can be used to deal with the conflicts generated by the concurrent execution of multiple SON functions is provided. As a particular case, in this chapter, the coordination of two specific SON functions is analysed. Finally, the details of the simulation platform and scenarios, as well as meaningful simulation results are described.
The publications contained in this chapter are listed here:

- Jessica Moysen and Lorenza Giupponi, "Self-coordination of parameter conflicts in D-SON architectures: a Markov Decision Process framework", *EURASIP Journal on Wireless Communications and Networking*, 2015.
- Jessica Moysen and Lorenza Giupponi, "Self Coordination among SON functions in LTE Heterogeneous Networks", *in proc. of the 81th IEEE Vehicular Technology Conference (VTC Spring)*, Glasgow, Scotland, 2015.
- Jessica Moysen and Lorenza Giupponi, "A Functional Architecture for Self-Coordination in LTE networks", *in proc. of the 20th IEEE European Wireless (EW)*, Barcelona, Spain, 2014.

In **Chapter 5**, a network planning tool based on ML and Genetic Algorithms (GAs) is presented. In particular, an approach which allows to predict QoS offered to end-users, based on data collected by the MDT function is proposed. In order to evaluate the performance of the proposed scheme, two cases of study are considered. Finally, a performance analysis is provided by means of simulations results.
The publications contained in this chapter are listed here:

- Jessica Moysen, Lorenza Giupponi, Josep Mangues-Bafalluy, "A Mobile Network Planning Tool based on Data Analytics", *to be submitted to the Mobile Information Systems,* *Special Issue Design, Dimensioning, and Optimization of 4G/5G Wireless Communication Networks*, December 2016.

- Jessica Moysen, Lorenza Giupponi, Josep Mangues-Bafalluy, "A Machine Learning enabled network Planning tool", *in proc. of the 27th IEEE Personal, Indoor and Mobile Radio Communications (PIMRC)*, Valencia, Spain, 2016.
- Jessica Moysen, Lorenza Giupponi, Josep Mangues-Bafalluy, "On the Potential of Ensemble Regression Techniques for Future Mobile Network Planning", *in proc. of the 21th IEEE Symposium on Computers and Communications (ISCC)*, Messina, Italy, 2016.
- Jessica Moysen, Lorenza Giupponi, Nicola Baldo, Josep Mangues-Bafalluy, "Predicting QoS in LTE HetNets based on location-independent UE measurement", *in proc. of the 20th IEEE Computer Aided Modelling and Design of Communication Links and Networks (CAMAD)*, Guildford, UK, 2015.

Finally, in **Chapter 6**, the final conclusions of the current thesis and future work are discussed

Bibliography

[1] Gandalf. *Monitoring and self-tuning of RRM parameters in a multi-system network*. URL: http://www.celtic-initiative.org/.

[2] SOCRATES. *Self-Optimisation and self-ConfiguRATion in wirelEss networkS*. URL: {http://www.fp7-socrates.org/}.

[3] SEMAFOUR. *Self-Management for Unified Heterogeneous Radio Access Networks*. URL: http://fp7-semafour.eu/.

[4] SESAME. *Small Cells Coordination for Multi-tenancy and Edge services*. URL: https://5g-ppp.eu/sesame/.

[5] SELFNET. *Framework for Self-Organised Network Management in Virtualised and Software Defined Networks*. URL: http://www.cognet.5g-ppp.eu/cognet-in-5gpp/.

[6] COGNET. *Cognitive Networks*. URL: https://5g-ppp.eu/.

[7] Aliu, O. and Imran, A. and Imran, M. and Evans, B. “A Survey of Self Organisation in Future Cellular Networks”. *Communications Surveys Tutorials, IEEE* 99 (2012), pp. 1–26.

[8] Sistelbanda. *SN4G SON*. URL: http://sistelbanda.es/.

[9] Qualcomm. *Self managing and enabling seamless roaming*. URL: https://www.qualcomm.com/videos/qualcomm-wi-fi-son.

[10] Huawei. *Huawei's Innovative Single SON Solution*. URL: http://www1.huawei.com/en/solutions/broader-smarter/hw-104610-son-multi-mode-o.htm.

[11] Airhop communications. *Powering 4G networks*. URL: http://www.airhopcomm.com.

[12] Cellwize. *Driving value through SON*. URL: http://www.cellwize.com.

[13] AVIAT. *Wireless products for small cell applications*. URL: https://startupgenome.co/aviat-networks.

[14] S. C. Forum. *Small cell industry award*. 2016. URL: http://www.smallcellforum.org/events/awards.

[15] 3GPP. *Technical Specification Group Services and System Aspects; Telecommunication Management; Self-Organizing Networks (SON); Concepts and requirements (Release 13)*. Tech. rep. TS 32.500, version 13.0.0. 2016.

[16] Ali Imran, Ahmed Zoha, and Adnan Abu-Dayya. "Challenges in 5G: How to Empower SON with Big Data for Enabling 5G". *IEEE Network* (2014).

[17] NOKIA. "Intelligent Self-Organizing Networks: Increased automation for higher network performance with lower costs". *White paper* (2013).

[18] Nicola Baldo, Lorenza Giupponi, Josep Mangues-Bafalluy. "Big Data Empowered Self Organized Networks". *20th European Wireless (EW) Conference; Barcelona, Spain* (2014).

[19] Bashar Romanous, Naim Bitar, Ali Imran, Hazem Refai. "Network Densification: Challenges and Opportunities in enabling 5G". *Special Session: Dense and Elastic Big Data Empowered SON* (2015).

[20] Seppo Hämäläinen,Henning Sanneck,Cinzia Sartori. *LTE Self-Organising Networks (SON): Network Management Automation for Operational Efficiency*. John Wiley and Sons, 2012.

[21] Johansson, J., Hapsari, W.A., Kelley, S., Bodog, G. "Minimization of drive tests in 3GPP (Release 11)". *IEEE Communications Magazine* (2012).

Chapter 2

State of the Art

Start where you are. Use what you have. Do what you can. **Arthur Ashe**

Contents

This chapter provides background information which is required for the better understanding of the thesis. The chapter is devoted to SON as a feature of 3GPP LTE systems. Section 2.1 briefly presents the basic concepts, and gives a high level introduction of the structure of 3GPP SON use cases. Section 2.1.1 summarizes the evolution of SON in 3GPP. Section

2.2, provides the key concepts of ML algorithms. In Section 2.3, we elaborate the general objective of applying machine learning techniques in SON. Finally, Section 2.4 concludes the chapter.

2.1 Self Organizing Network (SON)

A promising approach, which is receiving significant interest from industrial and research communities, to maximize total performance in cellular networks is to bring into them intelligence and autonomous adaptability. This is traditionally referred to as self-organization, a concept introduced by Ashby in [1–3]. Self-organization as applied to cellular networks is referred to SON and it is a key driver for improving Operations, Administration, and Maintenance (OAM) activities [4]. SON aims at reducing the cost of installation and management by simplifying operational tasks through the capability to configure, optimize and heal itself. The main objective of SON is to reduce the costs associated with network operations, i.e., CAPEX and OPEX, by diminishing human involvement, while enhancing network performance, in terms of network capacity, coverage and service quality. The main motivation behind the increasing interest in the introduction of SON from operators, standardization bodies and projects is twofold. On the one hand, from the technical perspective, the complexity and large scale of future radio access technologies imposes significant operational challenges due to the multitude of tuneable parameters and the intricate dependencies among them. In addition, the advent of new heterogeneous kind of nodes like femto, pico, relays, etc., is expected to make tremendously increase the number of nodes in this new ecosystem, so that traditional network management activities based on e.g., classic manual and field trial design approaches are just not viable anymore. Similarly, manual optimization processes or fault diagnosis and cure, performed by experts are no longer efficient and need to be automatized, as this causes time intensive experiments with limited operational scope, or delayed, manual and poor handling of cell/sites failures. Key operational tasks, such as radio network planning and optimization are largely separated nowadays and this causes intrinsic shortcomings like the abstraction of access technologies for network planning purposes, or the consideration of performance indicators that are of limited relevance to the end user's service perception. These problems have been approached through SON by European projects such as SOCRATES, which aims at the development of self-organization methods to enhance the operations of wireless access networks [5], and Gandalf, which focuses on automating Radio Resource Management (RRM) tasks in heterogeneous radio access networks (Global System for Mobile Communications (GSM), General Packet Radio Service (GPRS), Universal Mobile Telecommunications System (UMTS) and Wireless Local Area Network (WLAN)), including advanced RRM, auto-tuning, and automated diagnosis in troubleshooting [6]. Also FP7 and 5GPPP EU projects have been dealing with SON. In particular FP7 SEMAFOUR [7], which develops a unified self-management to operate complex HetNets. Among 5GPPP projects, we highlight SESAME [8], which proposes the Cloud-Enabled Small Cell (CESC) concept, i.e., a new multi-operator enabled Small Cell by deploying Network Functions Virtualisation (NFV), supporting powerful self-management inside the access network infrastructure. In terms of self-organised network management in SDN and NFV, SELFNET project aims at enabling the use of this kind of tools to achieve real time autonomous 5G network management [9], and for the demand prediction to resize the network using virtualisation, the COGNET project [10] aims at developing several use cases by applying ML algorithms. On the other hand, from the market perspective, the ever increasing demand for a diversity of offered services, and the need to reduce the time to market of innovative services, further add to the pressure to remain competitive by effectuating cost reductions. The overall idea of SON is to integrate network planning, configuration, and optimization into a single mostly automated process requiring minimal human intervention.

SON has been introduced by 3GPP as a key component of LTE network starting from the first release of this technology in Release 8, and expanding to subsequent releases. In SOCRATES project [5] and in 3GPP [11] meaningful SON use cases have been defined, which can be classified according to the phases of the life cycle of a cellular systems (planning, deployment, optimization and maintenance) into: self-configuration, self-optimization and self-healing.

In this chapter, first we give an overview of the evolution of SON in 3GPP. We discuss about the different architectures that have been considered for these SON functionalities. Then, we go through self-configuration, optimization and healing functionalities, introducing the use cases that have been defined for each one of them, mainly talking about how they have been addressed in 3GPP. We do not focus on the general literature, as it has already been reviewed in other works [12]. We discuss about the self-coordination problem, to handle the potential conflicts that may exist between the parallel execution of multiple SON functions. This is a very challenging issue to keep in mind when designing SON functions. We present the MDT functionality, which is introduced to collect useful data for analysis from UE measurements, in order to improve coverage and capacity issues, verify QoS etc. We introduce the core-networks related 3GPP SON use cases. Finally, we discuss the role of virtualised and software defined networks in the context of 5G SON.

2.1.1 SON evolution in 3GPP

Release 8 includes SON functionalities regarding initial equipment installation and integration. The SON functions developed in Release 9 are designed to optimise deployed LTE networks. Release 10 introduces SON functions to enhance interoperability between small cells and macrocells and includes NGMNs recommendations, i.e., new functionalities such as CCO, enhanced ICIC, COD and COC, self-healing functions, MDT and ES, are introduced. Release 11 SON functions are related to the automated management of heterogeneous networks. It includes mobility robustness optimization enhancements and inter-radio access technology HO optimization. Release 12 has studied how the NM centralized CCO function should be enhanced. Release 13 studies the enhancements of OAM aspects of distributed MLB [13–15], as well as enhanced NM or centralized CCO [16, 17]. Finally, Release 14 focuses on meeting the 5G requirements in terms of latency reduction, use of unlicensed spectrum in a fair manner, support for carrier aggregation, energy efficiency at OAM level, SON for active antennas, etc. i.e. all aspects which received a lot of attention in Release 13 [18]. Table 2.1 summarizes the evolution of SON in 3GPP.

Table 2.1: Evolution of SON in 3GPP

Release	WI	Feature	TS or TR
Rel.8	SA5-SON concepts and requirements	SON concepts and requirements	[19]
Rel.8	SA5-Self establishment of eNBs	Self configuration	[20–25]
Rel.8	SA5-SON Automatic Neighbour Relation (ANR) list management	ANR, PCI	[26–29]
Rel.9	SA5: Study of SON related OAM Interfaces for HeNBs	SON related OAM Interfaces for HeNBs	[30]
Rel.9	SA5: Study of self-healing of SON	Self-healing management	[31]
Rel.9	SA5:SON OAM aspects: Automatic radio network	Automatic radio network configuration data preparation	[20–22]
Rel.9	SA5:SON OAM aspects self-organisation management	Self-optimisation (MRO, MLB, ICIC)	[32]
Rel.9	RAN3: Self-organising networks	CCO, MRO, MLB, RACH opt.	[33–36]
Rel.10	SA5: SON self-optimization management continuation	Self-coordination, self-optimization (MRO, MLB, ICIC, RACH)	[27, 32, 37–39]
Rel.10	SA5: Self-healing management	CCO, COC	[40]
Rel.10	SA5: OAM aspects of ES in radio networks	ES	[27–29, 32, 41–43]
Rel.10	RAN2-3: LTE SON enhancements	CCO, ES, MLB, MRO enhancements	[34–36, 44]
Rel.11	SA5: ULTRAN SON management	SON management	[19, 37, 38, 45–48]
Rel.11	SA5: LTE SON coordination management	SON coordination [49]	[19, 27, 32, 37, 38, 47]
Rel.11	SA5: Inter RAT ES management	OAM aspects of ES management	[37, 38, 43, 45, 48, 50]
Rel.11	RAN3: Further SON enhancements	MRO, MDT enhancements	[33–36, 44, 51–54]
Rel.12	SA5: Enhanced NM centralized CCO	Enhanced NM centralized CCO	[32, 55–60]
Rel.12	SA5: Multi-vendor plug and play eNB connection to the network	Multi-vendor plug and play eNB connection to the network	[20, 61, 62]
Rel.12	SA5: Enhancements on OAM aspects of distributed MLB	OAM aspects of distributed MLB	[63]
Rel.12	SA5: Energy efficiency related performance measurements	Energy efficiency related performance measurements	[32]
Rel.12	SA5: HetNets management/OAM aspects of network sharing	HetNets/network sharing	[64, 65]
Rel.12	RAN2-3: Next generation SON for ULTRAN/EUTRAN	SON per UE type, active antennas, small cells	[15]
Rel.12	RAN2-3: ES enhancements for EUTRAN	ES	[66]
Rel.13	RAN2-3: Enhanced Network Management centralized CCO	CCO	[16]
Rel.13	SA5: Study on Enhancements of OAM aspects of Distributed Mobility Load Balancing SON function	MLB	[17]
Rel.14	RAN: OAM (SON for Active Antenna Systems (AAS)-based deployments)	Energy efficiency	[18, 67]

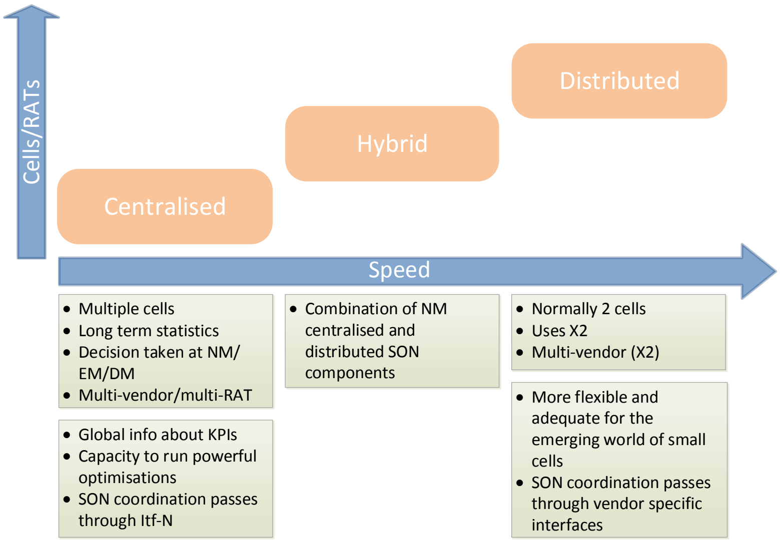

Figure 2.1: SON implementations.

2.1.2 SON implementations

In this section, we discuss about the different architectures that have been considered for SON, Centralized SON (C-SON), D-SON and Hybrid SON, ranging from a C-SON functionalities, where the self-organising algorithms reside in the network management system, in the Operation and Maintenance Center (OMC) or in the Network Management Systems (NMS), to a D-SON solution, where the SON functions are distributed, in the control plane, across the edges of the network, typically in the Enhanced Node Base stations (eNBs). On the one hand, C-SON can take into consideration data from all nodes in the network to identify and address network-wide issues. However, centralized systems may respond too slowly in the emerging world of small cells that experience very transitory traffic loads. On the other hand, D-SON functionalities are designed for near real-time response in seconds or milliseconds, which makes the SON functions highly dynamic and enables the network to adapt to local changes more rapidly. The main challenge in a D-SON implementation is that, it is more vulnerable than C-SON against network instabilities caused by the concurrent operation of SON functions with conflicting objectives. The SON implementations are depicted in Figure 2.1.

2.1.3 Self-Configuration

Self-configuration is the process of bringing a new network element into service with minimal human operator intervention [20]. This covers the cellular system life cycle phase related to planning and deployment. Self-configuring algorithms take care of all configuration aspects of the eNB. When the eNB is powered on, it detects the transport link and establishes a connection with the core network elements. After this, the eNB is ready to establish OAM, S1 and X2 links and finally sets itself in operational mode. After the eNB is configured, it performs a self-test to deliver a status report to the network management node. Since Release 8 ANR and automated configuration of Physical Cell ID (PCI) use cases have been considered [68, 69].

1. ANR. This function is one of the first SON functions to be standardized in 3GPP, it relies on the eNB, and it aims at automatic setting of neighbour relations [70]. According to the standards, the UE measures and reports the following types of cells: (1) the serving cell, (2) listed cells (i.e., cells that are indicated by the Evolved Universal Terrestrial Radio Access Network (E-UTRAN) as part of the list of neighbouring cells), (3) detected cells (i.e., cells that are not indicated by the E-UTRAN but detected by the UE). However, E-UTRAN does indicate the carrier frequency. So the detected cell can be a LTE cell within the same frequency or a LTE cell with a different frequency or even a cell belonging to another RAT. To detect inter-frequency cells or inter-RAT cells, the eNB needs to instruct the UE to do the measurement on that frequency [46]. Authors in [71] propose a method to generate the neighbouring cell list based on geographical conditions, antenna patterns and power transmission. The performance of the ANR use case also has been evaluated in HetNets [72], where the authors use the SINR from adjacent cells to generate the neighbour list.

   Prior to Release 11, focus was on E-UTRAN ANR. Release 11 ANR focuses on the management aspects of ANR for UMTS Universal Terrestrial Radio Access Network (UTRAN) and Inter-Radio Access Technology (IRAT) ANR, specifically defining the following SON use cases in the context of UTRAN ANR.

2. Automatic PCI assignment. The automated configuration of PCI use case aims to configure automatically the neighbour cell list, where, each cell has a unique cell ID and needs to create a neighbour cell list, e.g., the list update is required when a new cell is deployed or is temporarily out of service in the network [13].

Recommended practices for both use cases can be found in [73]. Some contributions in Self-configuration in LTE networks include [74–76].

2.1.4 Self-Optimization

Self-optimization consists of the set of mechanisms which optimize parameters during operation, based on measurements from the network. Here, one of the most important thing is the need for the network to continuously self optimise its operation. This approach has been studied in a number of papers [13, 77], and we can find a survey of the literature in [4], where the authors focus on Load Balancing (LB), ICIC and CCO use cases in order to classify them with respect to the state of the art. 3GPP approved several use cases since Release 9 [13]. These use cases provide a description of how Self-optimisation functionalities can be applied. These use cases cover many different aspects for self-organization in LTE networks, and can be found in [47]. Specific use cases are described as follows,

1. Mobility Load Balancing (MLB). The MLB is a SON function where cells with congestion can transfer load to other cells. The main objective is to improve end-user experience and achieve higher system capacity by distributing user traffic across system radio resources. The implementation of this function is generally distributed and supported by the load estimation and resource status exchange procedure. The messages containing useful information for this SON function (resource status request, response, failure and update) are transmitted over the X2 interface [36]. MLB can be implemented by tuning the Cell Individual Offset (CIO) parameter. The CIO contains the offsets of the serving and the neighbour cells that all UEs in this cell must apply in order to satisfy the A3 handover condition [78].

2. Mobility Robustness Optimisation (MRO). The MRO is a SON function designed to guarantee proper mobility, i.e., proper handover in connected mode and cell re-selection in idle mode. Among the specific goals of this function we have the minimization of call drops, the reduction of Radio Link Failures (RLFs), the minimization of unnecessary handovers, ping pongs, due to poor handover parameters settings, the minimization of idle problems. Its implementation is commonly distributed. The messages containing useful information are: the S1AP handover request or X2AP handover request, the handover report, the RLF indication/report. The Release 11 improvements to the handover optimization are included [79]. MRO operates over connected mode and idle mode parameters. In connected mode, it tunes meaningful handover trigger parameters, such as the event A3 offset (when referring to intra-RAT, intra-carrier handovers), the Time to Trigger (TTT), or the Layer 1 and Layer 3 filter coefficients. In idle mode, it tunes the offset values, such as the Qoffset for the intra-RAT, intra-carrier case.

3. Inter-Cell Interference Coordination (ICIC). ICIC is a SON function, which aims to minimize interference among cells using the same spectrum. It involves the coordination of physical resources between neighbouring cells to reduce interference from one cell to another. ICIC can be done in both uplink and downlink for the data channels Physical Downlink Shared Channel (PDSCH), and Physical Uplink Shared Channel (PUSCH), or uplink control channel Physical Downlink Control Channel (PDCCH). ICIC can be static, semi-static or dynamic. Dynamic ICIC relies on frequent adjustments of parameters, supported by signalling among cells over X2 interface. To support proactive coordination among cells the High Interference Indicator (HII) and the Relative Narrowband Transmit Power (RNTP) indicators have been defined, while to support reactive coordination, the Overload Indicator (OI) has been introduced [36].

4. Random Access Channel (RACH). RACH optimization aims at optimising the random access channels in the cells based on UE feedback and knowledge of its neighbouring eNBs RACH configuration. RACH optimization can be done by adjusting the Power control (Pc) parameter or change the preamble format to reach the set target access delay [80].

In Release 10, 3GPP defines new use cases as follows:

1. Coverage and Capacity Optimisation (CCO) is a SON function, which aims to provide capacity and coverage optimization. The targets that can be optimised may be vendor dependent and include coverage, cell throughout, edge cell throughput, or a weighted combination of the above.

2. Energy Saving (ES) aims at providing the quality of experience to end users with minimal impact on the environment. The objective is to optimize the energy consumption, by designing Network Elements (NEs) with lower power consumption and temporarily shutting down unused capacity when not needed [41].

Release 11 provides enhancements to MLB optimization, HO optimization, CCO, and ES. In Release 12 a study on enhancements of OAM aspects of distributed MLB SON function has been done, and can be found in [81].

2.1.5 Self-healing

Self-healing focuses on the maintenance phase of a cellular network. Wireless cellular systems are prone to faults and failures, and the most critical domain for fault management is the RAN.

Every eNB is responsible for serving an area, with little or none redundancy. If a NE is not able to fulfil its responsibilities, it results in a period of degradation of performances, during which users are not receiving a proper service. This results in severe revenue loss for the operator. 3GPP has technical specifications [40] that describe the concepts and requirements related to self-healing. Also we can find documents specifying the Information Service (IS) [27, 37] and Solution Sets (SS) [38, 39] for SON, Network Resource Model (NRM) Integration Reference Point (IRP). The Self-Healing methods are triggered by the failure of a cell or site. These methods aim to resolve the loss of coverage/capacity induced by such events [82]. By automatic adjustment of network parameters and algorithms in surrounding cells the problem is solved. Once the actual failure has been repaired, all parameters are restored to their original settings. A study on Self-healing was carried out since Release 9 [31], and in Release 10, features for detection, and adjustment of parameters have been specified [83]. These specifications have been updated in Release 11 [40], and are describe as follows,

1. Self-recovery of NE Software. If the NE software failed due to load earlier software version and/or configuration, the most important thing is to ensure that the NE runs normally by removing the fault software, and restoring the configuration.

2. Self Healing of board Faults. This use case aims to solve hardware failures in the NE [84].

3. Cell Outage Management. This use case is split in two main functions: 1) Cell Outage Detection. The main objective here is to detect a cell outage through the monitor performance indicators, which are compared against thresholds and profiles, and 2) Cell Outage Compensation. This use case aims at alleviating the outage caused by the loss of a cell from service [40]. It refers to the automatic mitigation of the degradation effect of the outage by appropriately adjusting suitable radio parameters, such as the pilot power and the antenna parameters of the surrounding cells. For this use case an adequate reaction is vital for the continuity of the service, so vendor specific Cell Outage Detection (COD) schemes have to be designed [85, 86].

Significant work was carried out by the GANDALF consortium in the area of diagnosis and detection in troubleshooting in the context of heterogeneous RANs, like GSM, GPRS, UMTS [87].

2.1.6 Self-Coordination

SON functionalities are often designed as stand-alone functionalities, by means of control loops. When they are executed concurrently in the same or different network elements, the impact of their interactions is not easy to be predicted, and unwanted effects may even occur among instances of the same SON function, when implemented in neighbouring cells. The risk of unacceptable oscillations of configuration parameters or undesirable performance results increase with the number of SON functions, so that it is considered necessary to define and implement a self-coordination framework [5, 88]. 3GPP has proposed different architectures for SON implementation, ranging from centralized C-SON to distributed D-SON, and the choice of the architecture has a strong impact on the efficiency of the self-coordination framework. If C-SON is used, SON functions are implemented in the OMC or in the NMS, as part of the Operation and Support System (OSS). This implementation benefits from global information about metrics and Key Performance Indicator (KPI)s, as well as computational capacity to run powerful optimization algorithms involving multiple variables or cells. However, it suffers from long time scales. In order to avoid oscillations of decision parameters, 3GPP requires [49] that each SON function asks for permission before changing any configuration parameter. This means that a request must be sent from the SON function to the SON coordinator and a response has to be returned. In C-SON all these requests must pass through the Interface-N, which is not suitable for real-time communication, so that there is no possibility to give priority to SON coordination messages over other OAM messages. If in turn, distributed coordination is used, the interaction between the SON function and the local SON coordinator will be over internal vendor-specific interfaces, with much lower latency characteristics. This makes the D-SON architecture much more flexible and adequate for small cell networks, which experience very transitory traffic loads, thus requiring high reactivity to propagation and traffic conditions. Figures 2.2 and 2.3 depict the Centralized vs. Distributed implementation describe before.

Market implementations of C-SON are offered by vendors like Celcite (acquired by AMDOCS), Ingenia Telecom and Intucell (acquired by Cisco), while D-SON solutions have traditionally been more challenging to implement and vendor specific, not allowing for easy interaction of products from different vendors, so that a supervisory layer is commonly still needed to coordinate the different instances of D-SON across a much broader scope and scale. Only recently, vendors like Qualcomm or Airhop have started proposing D-SON as a SON mainstream, as small cells and HetNets require the millisecond response times of D-SON.

The topic of conflicts resolution and coordination has been receiving growing interest

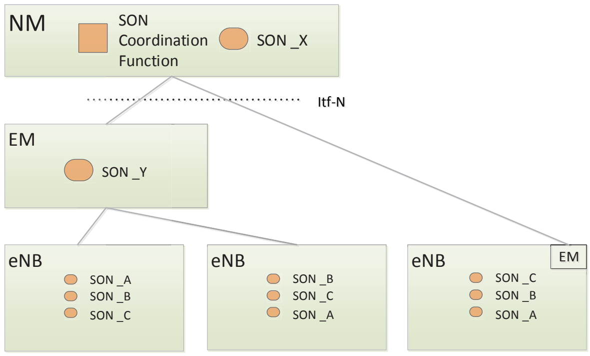

Figure 2.2: C-SON self-coordinator implementation

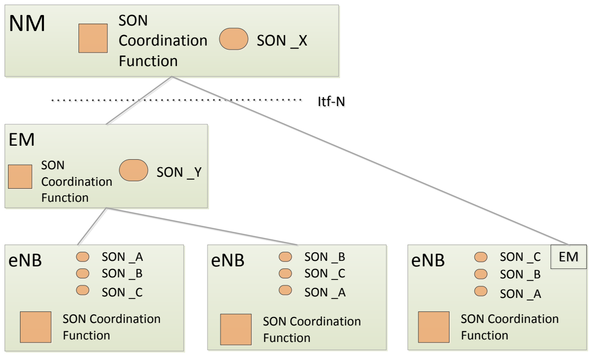

Figure 2.3: D-SON self-coordinator implementation

also from academic community. In [84, 89] the authors focus on the classification of potential SON conflicts and on discussing the valid tools and procedures to implement a solid self-coordination framework. Examples of centralized and distributed implementations of SON coordination are offered in [90] and [91], respectively. A preventive coordination mechanism that uses policy based decision making has been proposed in [91]. Guard functions have been proposed in [88] to detect undesirable network behaviours and trigger countermeasures. Decision trees have been proposed in [92] for properly adjusting Remote Electrical Tilt (RET) and transmission power.

2.1.7 Minimisation of Drive Tests

MDT enables operators to collect User Equipments (UEs) measurements together with location information, if available, with the purpose of optimising network management while reducing operational effects and maintenance costs. This feature has been studied by 3GPP since Release 9 [93], among the targets there are the standardization of solutions for coverage optimization, mobility, capacity optimization, parametrisation of common channels, and QoS verification [84]. Since operators are also interested in estimating QoS performance, in Release 11, MDT functionality has been enhanced through QoS performance to properly dimension and plan the network by collecting measurements indicating throughput and connectivity issues [94]. Some use cases are as follows:

1. Coverage optimisation. This function covers multiple objectives, such as coverage mapping, coverage hole detection, identification of weak coverage, detection of excessive interference, uplink coverage verification, etc.

2. Mobility optimisation. This function aims to optimize the mobility performance by moni-

toring the cells with a high handover rate and the provided coverage.

3. Capacity optimisation. This function aims to improve strategies of network planning and network capacity verification/optimization

4. QoS verification. This function is aimed to verify the service level that can be offered to the users in different parts of the network. Some contributions on these use cases can be found in [95–98].

These MDT functions have been further elaborated in Release 11, while Release 12 has included specific enhancements in terms of correlation of information, which can be found in the study on enhanced network management centralized CCO. These improvements and extensions of SON enhancements introduced until Release 13 can be found in [16].

2.1.8 Core networks

The core network is that part of the network that connects the different parts of the access networks and provides a gateway to other networks. The core network operations can be enabled with SON functionalities. The benefits also in this case come from the reduced human intervention and so from reduced operational costs. SON for core network allows to self-adapt traffic loads and prevent bottlenecks. It also helps operators route calls to a point of interconnection that improve capital and operational expenditures. In addition, SON for Core enables the core network to handle signalling more efficiently. In this regard, Nokia [99] already automates core networks operations based on SON technology. The objective is to automatically and rapidly allocate core network resources to meet unpredictable behaviours and demands in terms of broadband. Notice that SON use cases for core networks are not limited to LTE networks, but many of them can be taken into account also for other kinds of networks, like 2G/3G.

2.1.9 Virtualised and Software defined networks

In order to deal with the open challenges related with the autonomic 5G networks management new deployment approaches based on NFV and SDN architectures, are being introduced to make mobile network deployments more cost-effective. The main idea behind these novel architectures is to provide a framework capable of assisting network operators to solve management problems, such as, cyber attacks, network failures, optimisation to improve network performance, and QoE of the users, among others. In this context, SON can be useful to achieve real time autonomous network management. This research line is extremely novel and no much work can be found so far. However, we highlight the work that is underdevelopment in the context of the SELFNET project [9], which consider valuable. SELFNET SON will operate in virtualised and software defined network and thus can be compatible with non-3GPP systems, but also is expected to be complementary with those in 3GPP networks.

2.2 Key concepts of ML

Learning is the process of gaining knowledge by instruction or study, and the discovery of new facts by experience. Computer modelling of learning processes that are able to introduce such capabilities in computers is the main challenge of ML. The term of ML was originally introduced in 1959 by Arthur Samuel [100]. He defined ML as a field study that gives computers the ability to learn without being explicitly programmed. According to [101], a computer is said to learn from experience, with respect to some task, and some performance metric, if its performance of the task, improves with experience. ML studies computer algorithms for learning to complete a task, or to make predictions based on observations, i.e., it is about learning to do better in the future based on what was experienced in the past. The target is to discover learning algorithms that do the learning without human intervention [102]. The primary goal of ML is to develop efficient algorithms, where time and the amount of data are one of the most important things. Learning algorithms should also be as general as possible, i.e., the learning algorithm has to be able to make a prediction that is as accurate as possible. But also, those predictions have to be easily understandable by experts. Machine Learning is in our everyday life, and its applications are huge and cross-disciplinary, e.g., web search, spam filters, recommender systems, ad placement, fraud detection, stock trading, etc. A recent report from McKinsey Global Institute asserts that machine learning will be the driver of a next big wave of innovation [103].

The objective of ML is to improve performance of a particular sets of tasks by creating a model that helps find patterns through learning algorithms. ML taxonomy is traditionally organised onto: 1. Supervised Learning (SL), 2. Unsupervised Learning (UL), 3. RL, and 4. New trends: Deep Learning (see Table 2.2). The four techniques are described as follows.

2.2.1 Supervised Learning

SL is a Machine learning technique which takes training data (organised into an input vector ($\mathbf{x}$) and a desired output value (y)) to develop a predictive model, by inferring a function $f(\mathbf{x})$, returning the predicted output $\hat{y}$. For that, the construction of a dataset is needed. The

Table 2.2: Summary of machine learning techniques

Supervised Learning	
Classification	K-Nearest Neighbours
	Generalized Linear Model
	Support Vector Machines
	Naive Bayes
	Neural Networks
Regression	K-Nearest Neighbours
	Generalized Linear Regression
	Support Vector Regression
	Neural Networks
	Decision Trees
Unsupervised Learning	
Clustering	Non-overlapping clustering
	Hierarchical clustering
	Overlapping clustering
Dimensionality Reduction	Feature Extraction
	Feature Selection
Anomaly Detection	Pruning techniques
	Rule-based systems
Reinforcement Learning	
Model-based	Dynamic Programming
	Monte Carlo
Model-free	Temporal Difference
New trends	
Deep Learning	Deep networks for supervised learning
	Deep networks for unsupervised learning
	Hybrid deep network

dataset contains training samples (rows), and features (columns), and is usually divided into 2 sets. The training set, used to train the model, and the test set, used to make sure that the predictions are correct. The goal of the training model is to minimize the error between the predictions and the actual values. Hence, by applying ML, we aim to estimate how well a learning algorithm generalizes beyond the samples in the training set. The input space is represented by a n-dimensional input vector $\mathbf{x} = (x^{(1)}, \ldots, x^{(n)})^T \in \mathbf{R}^n$. Each dimension is an input variable. In addition a training set involves m training samples $((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m))$. Each sample consists of an input vector $\mathbf{x}_i$, and a corresponding output y_i. Hence $x_i^{(j)}$ is the value of the input variable $x^{(j)}$ in training sample i, and the error is usually computed via $|\hat{y}_i - y_i|$. The SL technique has two main applications, classification and regression. On the one hand, classification is applied when y, the output value we try to predict is discrete, e.g., we want to predict if a cancer is benign or malign, based on a dataset constructed based on medical records, and collecting many features, e.g., tumour size, age, uniformity of cell size, uniformity of cell shape. On the other hand, a regression problem is applied when y is a real number, i.e., we try to predict continuous valued outputs, e.g., we want to predict housing prices, based on a dataset of examples of prices per size of houses.

A huge amount of SL algorithms for classification can be found in the literature, and a study to evaluate the performance of some of them can be found in [104]. At this point the key factor in ML is how to evaluate ML models. A ML model is a function that represents the relationship between different aspects of data. For example, consider a linear model $y = \mathbf{\Phi}^T\mathbf{x}$, this model uses a line to represent the relationship between the vector $\mathbf{x}$ (representing the features of the data) and a scalar y (representing the value we want to learn to predict). The variable $\mathbf{\Phi}$ is a weight vector that specifies the slope of the line. This is referred as a *model parameter* and it is learned during the training phase. The goal is to determine the best model parameter that fits the data. Other parameters that we need to consider are the *nuisance parameters* [105]. These parameters control the model's capacity, which gives us information about the flexibility of the model, i.e., how to control a low error on the training set (under-fitting), and how to control a big gap between the training error and the test error (over-fitting). These concepts are fundamental to characterize notions of:

1. How to measure the error from erroneous assumptions in the learning algorithm (bias). High bias can cause under-fitting.

2. How to quantify how much the response varies from one learning sample to another (variance). High variance can cause over-fitting.

Some parameters that influence these two concepts are:

1. The complexity of the model.
2. How much y varies from the model (noise).
3. The learning sample size.
4. The learning algorithm, among others.

Hence the capacity of the model is one of the main factors we need to take into account when implementing a ML algorithm. Many SL algorithms can be reduced to some variant of those described as follows:

1. k-Nearest Neighbours (k-NN) can be used for classification and regression. k-NN is a non-linear method where the input consists of the k closest training samples in the input space. The predicted output is the average of the values of its k nearest neighbours. A commonly used distance metric for continuous variables is the Euclidean distance. The k-NN method has the advantage of being easy to interpret, fast in training, and the amount of parameter tuning is minimal. However, the accuracy of the prediction is generally limited.

2. Generalized Linear Models (GLM). The linear model describes a linear relationship between the output and one or more input variables, and where the approximation function maps from x_i to y_i as follows,

   $$\hat{y}_i = \Phi_0 + \Phi_1 x_i^{(1)} + \ldots + \Phi_n x_i^{(n)} \tag{2.1}$$

   where Φ_i are the unknown parameter. The idea is to choose Φ_i so that $\hat{y}_i$ minimizes the loss function. Typically we make the assumption that the samples in each dataset are independent from each other, and that the training set and testing set are identically distributed. Note that if the relation is not linear, the model should be generalized, in an attempt to capture this relationship [106].

3. Naive Bayes (NB). The method is used for classification and is based on Bayes theorem, i.e., calculating probabilities based on the prior probability. The main task is to classify new data points as they arrive. A NB classifier assumes that all attributes are conditionally independent, and is recommended when the dimensionality of the input is high [107]. Since NB assumes independent variables, it only requires a small amount of training data to estimate the means and variances of the variables.

4. Support Vector Machine (SVM) can be used for classification and regression. SVM are inspired by statistical learning theory, which is a powerful tool for estimating multidimensional functions [108, 109]. This method can be formulated as a mathematical optimization problem, which can be solved by known techniques. For this problem, given m training samples $((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m))$, the goal is to learn the parameters of a function which best fits the data. It samples hyperplanes. Thus, the hyperplane with the main minimum distance from the sample points is maintained. The sample points that form margin are called support vectors and establish the final model. This method in general shows high accuracy in the prediction, and it can also behave very well with non-linear problems when using appropriate kernel methods. Also, when we cannot find a good linear separator, kernel techniques are used to project data points into a higher dimensional space where they can become linearly separable. Hence the correct choice of kernel parameters is crucial for obtaining good results. In practice, this means that an exhaustive search must be conducted on the parameter space, thus complicating the task [110].

5. Artificial Neural Networks (ANN) is a statistical learning model inspired by the structure of a human brain, where the interconnected nodes represent the neurons producing appropriate responses. ANN supports both classification and regression algorithms. The basic idea is to efficiently train and validate a neural network. Then, the trained network is used to make a prediction on the test set. In this method the weights are the parameters in charge to manipulate the data in the calculations. Here, the interconnection pattern between the different layers of neurons, the learning process for updating the weights of the interconnections, and the activation function that converts a neuron's weighted input to its output activation are the most important parameters to be considered [111]. ANNs methods require parameters or distribution models derived from the data set, and in general they are also susceptible to over-fitting.

6. Decision Tree (DT) is a flow-chart model in which each internal node represents a test on an attribute. Each leaf node represents a response, and the branch represents the outcome of the test [112]. DTs can be used for classification and regression, and they have nuisance parameters, such as the desired depth and number of leaves in the tree [113]. Also, they do not require any prior knowledge of the data, are robust (i.e., do not suffer the curse of dimensionality as they focus on the salient attributes) and work well on noisy data. However, DTs are dependent on the coverage of the training data as with many classifiers. Moreover, they are also susceptible to over-fitting.

2.2.2 Ensemble Methods

Ensemble methods combine the predictions of multiple learning algorithms to produce a final prediction. This technique has been investigated in a huge variety of works [114, 115]. A general method is sub-sampling the training examples, where the most useful techniques are bagging and boosting techniques [116]. Bagging manipulated the training examples to generate multiple hypothesis. It runs the learning algorithm several times, each one with different subset of training samples. Random Forest (RF) is a modification of bagging that build a collection of trees, and then average them [117]. Every tree in the ensemble is grown on an independently drawn sample of input data. It takes an average of predictions from individual trees (for regression) or takes votes from individual trees (for classification). Like Bagging, AdaBoost manipulates the training examples to generate multiple hypotheses. It maintains a probability distribution over the training samples. AdaBoost maintains a set of weights over the original training set, and adjusts these weights by increasing the weight of examples that are misclassified, and decrease the weight of examples that are correctly classified [118, 119]. In literature, the authors in [102] show that AdaBoost is one of the best methods for constructing ensembles of decision trees. However in [120] the authors have shown that in domains with noisy training data AdaBoost can perform badly. It places high weight on incorrectly labelled training examples, and, consequently constructs bad classifiers. In [121], the authors have showed that combining bagging with error correcting output coding improved the performance of both methods, which suggest that combinations of other ensemble methods should be explored. However, ensembles method requires large amount of memory to store and large amount of computation to apply. An important line of research is to find ways of converting these ensembles into less redundant representations. A difficulty is that an ensemble provides little insight into how it makes its decisions, e.g., a single DT can often be interpreted by human users, but an ensemble of hundreds of DTs is much more difficult to understand. In [122], the authors propose to take advantage of ensembles of DTs. Although the accuracy of each of the individual DT is less than the accuracy of a single tree grown with all the data, the accuracy of the ensemble is better than the accuracy of one DT.

2.2.3 Unsupervised Learning

UL is a ML technique, which receives unlabelled input patterns. In this case, we let the computer learn by itself, without providing the correct answer to the problem we want to solve. The goal is to construct representation of inputs that can be used for predicting future inputs without giving the algorithm the right answer, i.e., it tries to find patterns in the data without unstructured noise [123]. The three most important families of algorithms are clustering, dimensionality reduction and anomaly detection techniques. Nowadays we can find real UL applications, e.g., news.google.com, understanding genomics, organize computer clusters, social network analysis, astronomical data analysis, market segmentation, etc.

1. Clustering. This technique aims at identifying groups of data to build representation of the input. The most common methods to create clusters by grouping the data are: non-overlapping, hierarchical and overlapping clustering methods. K-means [124] and Self Organising Maps (SOMs) [125] methods belong to non-overlapping clustering techniques. When the clusters at one level are joined as clusters at the next level (cluster-tree), this is refereed in literature as a hierarchical clustering method [105]. In case that an observation can exist in more than one cluster simultaneously, it is known as overlapping or fuzzy clustering. Fuzzy C-means and Gaussian mixture models belong to this kind of technique [124, 126]. This kind of algorithms have been proposed in a wide range of fields, such as, robotics, wireless systems, and routing algorithms for mobile ad hoc networks, among others.

2. Dimensionality Reduction. High-dimensional datasets present many challenges. One of the problems is that, in many cases, not all the measured variables are necessary to understand the problem of interest. In the state of the art we can find a huge amount of algorithms to predict models with good performance from high-dimensional data. However it is of interest for many problems to reduce the dimension of the original data. For example, in [127], the authors face the problem of the huge amount of potential features the system have as input, and they suggest that the regression analysis has a better performance in a reduced space. In this context, the most common methods are: Feature Extraction (FE) and Feature Selection (FS) [128]. Both methods seek to reduce the number of features in the dataset. FE methods do so by creating new combinations of features (e.g., Principal Component Analysis (PCA)), which project the data onto a lower dimensional subspace by identifying correlated features in the data distribution. They retain the Principal Components (PCs) with the greatest variance and discard all others to preserve maximum information and retain minimal redundancy [129]. Correlation based FS methods include and exclude features present in the data without changing them. For example, Sparse Principal Component Analysis (SPCA), which extends the classic method of PCA for the reduction of dimensionality of data by adding sparsity constraint

on the input features.

3. Anomaly Detection. The idea of this technique consists in identifying events, which do not correspond to an expected pattern. By modelling the most common behaviours the machine selects the set of unusual events [130]. The two most common techniques are:

   - Rule based systems are very similar to DTs, but they are more flexible than DTs as new rules may be added, without creating a conflict with the existing ones.
   - Pruning techniques aim at identifying outliers, where there are errors in any combination of variables.

   Identifying irregular behaviour in the data is crucial in many applications such as medical imaging, in mobile networks specify the case of fault detection, and network security.

2.2.4 Reinforcement Learning

Differently from the case of SL, RL aims to learn from interactions how to achieve a certain goal. In many real applications and in particular, in sequential decision and control problems, it is not possible to provide an explicit supervision to the training (i.e., the right answer to the problem). In these cases, we can only provide a reward/cost functions, which indicates to the algorithm when it is doing well and when it is doing poorly. RL has already been proven effective in many real world applications, such as autonomous helicopters, network routing, robot legged automation, etc. [131–133]. The learner or decision maker is called *agent*, and it interacts continuously with the so-called *environment*. The agent selects actions and the environment responds to those actions and evolves into new situations. In particular, the environment responds to the actions through *rewards*, i.e., numerical values that the agent tries to maximize over time.

The agent has to exploit what it already knows in order to obtain a positive reward, but it also has to explore in order to take better actions in the future. The problem is then defined by means of a Markov decision process $\{\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}\}$, where $\mathcal{S}$, is the set of possible states of the environment $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$, $\mathcal{A}$ is the set of possible actions $\mathcal{A} = \{a_1, a_2, \ldots, a_q\}$ that each decision maker may choose, $\mathcal{T}$ is the probability of moving to state $s+1$ when action a is taken in state s, and $\mathcal{R}$ is a reward function, which specifies the expected immediate return $\mathfrak{R}$, for each state s, action a, and next state $s+1$. The interactions between the multi-agent system and the environment at each time instant t consist of the following sequence.

- Agent i senses the state $s_t^i = s \in \mathcal{S}$.

- Based on s, agent i selects an action $a_t^i = a \in \mathcal{A}$.
- As a result, the environment makes a transition to the new state $s_{t+1}^i = v \in S$.
- The transition to the state v generates a reward $r_t^i = r \in \mathfrak{R}$.
- The reward r is fed back to the agent and the process is repeated.

The solution to a MDP is based on the RL framework [134]. At each time step, the agent implements a mapping from states to probabilities of selecting each possible action. This mapping is the agent's *policy*. The objective of each learning process is to find an optimal policy $\pi^*(s) \in \mathcal{A}$ for each s, to maximize some cumulative measure of the reward r received over time. Almost all RL algorithms are based on estimating a so called *value function*, which is a function of the states estimating how good it is for an agent to be in a given state. The objective is to choose a policy π that minimizes the expected discounted sum cost over a potentially infinite horizon.

$$V^*(s) = \min_{\pi} \mathbb{E}\left(\sum_{t=0}^{\infty} \gamma^t c_t\right) \tag{2.2}$$

where $\mathbb{E}$ stands for the expectation operator, t is any time step, and $0 \leq \gamma \leq 1$ is a discount factor. The principle of Bellman's optimality proves that there is an optimal policy.

Single vs Multi agent

Learning can be centralized in a single agent or distributed across a multiple agents. In single agent systems, the machine learning approaches are capable of finding optimal decision policies in dynamic scenarios with only one decision maker. In multi agent systems, the distributed decisions are made by multiple intelligent decision makers, and the optimal solutions or equilibria are not always guaranteed [135]. Some characteristics of this kind of system are:

- Each agent has incomplete information and capabilities to solve the decision making problem.
- No central entity providing global control.
- Spatially distributed sources of information.
- Decisions of each agent strongly depend on the decisions made by the other agents.

RL approaches

RL literature offers two approaches to solve MDPs. These two approaches are: model-based and model-free.

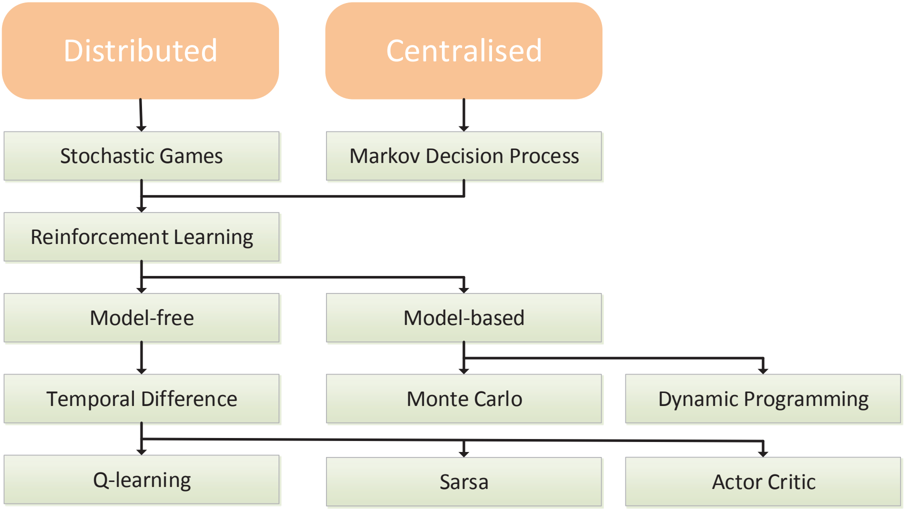

Figure 2.4: Useful taxonomy for RL.

1. Model-based. Dynamic Programming (DP) and Monte Carlo (MC) methods belong to model-based approach. DP, relies on the knowledge of the state transition probability between two states after executing a certain action. The main drawback of DP methods are that they require to fully known the MDP model. DP algorithms are based on update rules derived from the Bellman equation. The basic building blocks are:

   - Policy evaluation and Policy improvement. Combining these two processes we obtain the two most popular solving algorithms, which are Policy and Value iteration. Although their application on practical cases is limited, they provide foundation for other RL methods.

   MC method requires only experience, i.e., sample sequences of states, actions and rewards. The estimations are only updated after the episodes conclude. Although their application on practical cases is limited, they provide foundation for other RL methods.

2. Model-free. TD methods are model free approaches to solve RL problems. TD learning is a combination of MC and DP ideas. It uses the current estimate V_t^π instead of the true V^π, like DP. If $\mathcal{T}$ is known, we can solve the MDP through DP, otherwise we need TD methods. Some examples of TD methods are: Sarsa, Q-learning and AC [134].

(a) Q-learning is an off-policy TD control. In Q-learning the learned action-value function, Q, directly approximates Q, independent of the policy followed. The policy guides the algorithm selecting which state-action pairs are visited and updated.

(b) Sarsa is an on-policy control method, it is necessary to estimate $Q^{\pi}(s,a)$ instead of $V^{\pi}(s)$. The convergence properties depend on the policy's dependence on Q. In MC control methods, if a policy that causes the agent to stay in the same state is found, the episode never ends. In step-by-step methods like sarsa this can not happen. The sequence $(s_t, a_t, r_{t+1}, s_{t+1}, a_{t+1})$ gives rise to the name Sarsa.

(c) AC method. As already mentioned, we rely on RL theory to take advantage of TD learning to implement autonomous SON functions in wireless dynamic environments that we cannot model theoretically. Among the literature of TD learning schemes, in this dissertation the AC approach has been selected to solve independently each SON function (i.e., to learn the optimal policy), as it is one of the most representative TD schemes and it is not computationally complex. Therefore, for this particular method we have provided a more in-depth description of the algorithm.

AC methods are TD methods that have a separate memory structure to represent the policy independently of the value function. The policy structure is known as the *actor*, since it is used to select the actions, while the estimated value function is known as the *critic*. The critic learns and critiques whatever policy is currently being followed by the actor and takes the form of a TD error δ, which is used to determine if a_t was a good action or not. δ is a scalar signal, which is the output of the critic and drives the learning procedure. After each action selection, the critic evaluates the new state to determine whether things have gone better or worse than expected, as it is defined by the TD error:

$$\delta_t = r_t + \gamma V_t(s_{t+1}) - V_t(s_t) \tag{2.3}$$

where V is the current value function implemented by the critic, to evaluate the action a_t taken in s_t. If the TD error is positive, it suggests that the tendency to select a_t should be strengthened for the future, whereas if the TD error is negative, it suggests the tendency should be weakened. We identify this tendency with a preference function $P(s_t, a_t)$, which indicates the tendency or preference to select a certain action in a certain state. Then the strengthening or weakening described above can be implemented by increasing or decreasing $P(s_t, a_t)$ by

$$P(s_t, a_t) \leftarrow P(s_t, a_t) + \beta\ \delta_t \tag{2.4}$$

where β is a positive learning parameter. This is the most simple implementation of a AC algorithm. The variation that we consider for implementation, is to add different weights to different actions, for example based on the probability of selecting action a_t in state s_t, i.e., $\pi(s_t, a_t)$, which results in the following update rule:

$$P(s_t, a_t) \leftarrow P(s_t, a_t) + \beta\ \delta_t(1 - \pi(s_t, a_t)) \tag{2.5}$$

In this implementation, AC directly implements the Boltzmann exploration method to select actions as follows:

$$\pi(s_t, a_t) = \frac{e^{P(s_t,a_t)}}{\sum_{a_t \in \mathcal{A}} e^{P(s_t,a_t)/\tau}} \tag{2.6}$$

This means the probability to select an action a in state s at time t depends on the temperature parameter τ, and on the preference values $P(s_t, a_t)$ at time t. In this kind of exploration, actions that seem more promising, because of higher preference values, have a higher probability of being selected. Figure 2.4 depicts the taxonomy of RL.

2.2.5 New trends: Deep Learning Networks

Large ML problems are beginning to arise in database applications, where there can be millions of measurements every day, and it is desirable to have ML algorithms that can analyse such large data sets. As a result, people who are interested in collecting such data look inside ML to find useful insights and predictions. However, ML techniques seem to be limited in their ability to process the data in the original form. In this context ML is rapidly expanding by inventing new formalizations of ML problems. The idea is to be able to fed a machine with row data and discover the representations needed for detection and/or classification/regression. Deep learning allows this [136]. Deep learning is a new area of ML research that attempts to learn in multiple levels. It allows to find patterns from data. This is done by applying computational models that are composed of multiple levels of representation and abstraction that help to make sense of data. This method is making major advances in solving problems, such that, language, vision, speech recognition, object and audio detection, and many other fields. Deepening on the application, this technique can broadly categorize into three major classes [137] described as follows:

1. Deep networks for supervised learning. They try to find patterns when visible information about labels data is available. Some of examples of this kind of deep networks are;

Restricted Boltzmann Machines (RBMs) [138], Deep Belief Networks (DBNs) [139], Deep Boltzmann Machines (DBMs) [140].

2. Deep networks for unsupervised learning. They try to find patterns when no visible information about labels data is available. Examples of these architectures are, Hidden Markov Modelss (HMMs) and Conditional Random Fieldss (CRFs) [141] .

3. Hybrid deep networks. This type of neural networks make use of both, supervised and unsupervised components. The goal is to use supervised learning to estimate some parameters in any of the deep unsupervised category. An example of this is a DBN [139]. This type of neural network is trained on a set of examples in an unsupervised way, once this kind of neural network has learned how to reconstruct its inputs, it can be trained in a supervised way.

2.3 Machine Learning Enabled SONs

The exploitation of the huge amount of data that we referred to in the introduction can be analysed with proper tools used as a part of advanced analytics, e.g., predictive analytics, where the primary goal is to make more informed decisions by analysing large data sets. Here, ML is a great opportunity due to its capability of providing insightful information from the analysis of data already available to operators, which can be used to make improvements or changes.

This section presents some relevant information in the area of big data and ML for SONs. It first presents sources of information relevant for mobile networks, and then, it provides an overview of SON's related work, where ML techniques have been adopted.

2.3.1 Big data within mobile cellular networks

As we observed in [142] a huge amount of data is currently already generated in mobile networks during normal operations by control and management functions. This kind of data can be exploited to find patterns of users, and take better decisions to optimise the network performance, in terms of handovers, mitigate/eliminate session drops, resource allocation, among others. Some examples of different sources of information available within mobile networks, but also the kind of the usage made by operators are depicted in Table 5.1. Notice, that the kind of sources provided in this table are the most relevant 3GPP technical specifications for SON solutions, and are described as follows.

Table 2.3: Information elements relevant for ML enabled SONs

Source	Data	Usage	TS
Charging Data Records (CDR)	Includes statistics at the service, bearer and IP Multimedia Subsystem (IMS) levels.	Coarse time granularity	TS 32.298 [143]
Performance Management (PM)	Call/session setup, release, maintenance QoS, RRC, idle and connects mode mobility	Discarded after usage	TS32.401 [144], TS32.425 [32]
Minimization of Drive Tests (MDT)	Radio measurements for coverage, capacity, mobility optimization, QoS optimisation/verification	Heuristic algorithms typically discard info after	TS37.320 [145]
E-UTRA Control plane protocols and interfaces	Lots of info (coverage, UE connectivity, mobility idle/connected, inter-cell interference, resource management, load balancing)	Typically thrown away after usage	TS36.331 [44], TS36.413 [35], TS36.423 [36]

1. Charging Data Records (CDR). It refers to data, which most of the time is not be able for networks managements purposes. It can be very useful to find mobility patterns of users to take better decision in terms of handovers, switch off eNBs, among others. Already are used for Big Data.

2. Performance Management (PM). It provides a significant amount of data on network performance aspects such as, the performance of the radio access network.

3. Minimization of Drive Tests (MDT). It refers to the data provided by the users, which covers aspects such as, power measurements, radio link failure events among others.

4. E-UTRA Control plane protocols and interfaces. This kind of data covers aspects of the Radio Resource Control (RRC), S1-AP, and X2-AP protocols.

In this context, the literature already offers different works, where these kind of information is exploited for network management purposes, such as [127, 146–149]. But also, information regarding the benefits of big data in 5G networks, for example, the work of [150], where the authors identify different source of data, which can be exploited to develop fault detection, identify coverage holes, sleeping cells, or cell in outage, but also they list source of data that can act as an input to the SON entity to perform load balancing and prediction operations.

2.3.2 Overview of SON's relevant work in the area of ML

This section reviews SON's recent work in the area of ML. Table 2.4 summarizes the main works in this area and classifies them per 3GPP use case, technique and specific algorithm adopted by the authors.

1. Use case: Indicates the 3GPP use case achieved.
2. Reference: Indicates the related work.
3. Technique: Indicates the ML method applied (Supervised Learning, Unsupervised Learning, Reinforcement Learning).
4. Problem: Indicates the main problem to solve.
5. Algorithms: Indicates the algorithm applied to the data (see Table2.4).

Mobility Load Balancing

The literature offers some examples of application of ML techniques to the MLB use case. The majority of applications falls in the area of RL, as the main problem to solve is a sequential decision problem about how to set configuration parameters, which optimise network performance and user experience. As a consequence, several works on this use case were carried out by different authors, and a couple of them take advantage of ML techniques to reach the target of this use case. An example of application of RL for MLB is [155], where the authors present a distributed Q-learning approach that learns for each load state the best MLB action to take, while also minimizing the degradation in HO metrics, another option to take advantage also of fuzzy logic capabilities of dealing with heterogeneous sources of information is provided in [154], where fuzzy logic is combined with Q-learning in order to target the load balancing problem. For similar reasons, fuzzy logic is also proposes in [152] to enhance the network performance by tuning HO parameters at the adjacent cells. Alternatively, a centralized solution is approached in [156], where a central server in the cellular network determines all HO margins among cells by means of DP approach. Besides RL also clustering schemes have been proposed in this area, to group cells with similar characteristics and provide for them similar configuration parameters [153].

Mobility Robustness Optimization

Also for the case of MRO, we find in literature different solutions based on RL to solve a control decision problem. In [157, 160, 161], the authors focus on the optimization of the users' experience and of the HO performance. The three of them target this problem by means of a Q-learning approach. [157] focuses on the enhancement for handover robustness in a wireless

Table 2.4: Related work

	Reference	ML technique	Problem	Algorthim
Self-configuration				
ANR	[151]	RL	Sequential decision	Q-learning
Self-optimization				
MLB	[152]	RL	Sequential decision	Fuzzy Q-learning
	[153]	UL	Finding large classes of time series	K-means clustering
	[154]	RL	Solving congestion	Q-learning
	[155]	RL	Decision making	Q-learning
	[156]	RL	Decision making	Dynamic Programming
MRO	[157]	RL	Sequential decision	Fuzzy Q-learning
	[158, 159]	UL	Detecting clusters within data	SOM
	[160]	RL	Decision making	Q-learning
	[161]	RL	Sequential decision	Q-learning
	[162]	SL	Extract patterns from data	Multiple linear regression
CCO	[163]	RL	Decision making/control	Fuzzy Q-learning
	[164]	UL, RL	Sequential decision/control	Fuzzy NN/Q-learning
	[165]	RL	Decision making/control	Fuzzy Q-learning
ICIC	[166–168]	RL	Control optimisation	Q-learning
	[169]	UL	Control optimisation	Neural Networks
	[170]	SL	Prediction/estimation	SVM for regression
	[171]	RL	Control optimisation	Fuzzy Q-learning
ES	[172]	RL	Decision making	Q-Learning
	[173]	UL	Decision making	Fuzzy logic
Self-healing				
COC	[174]	SL	Control optimisation	Fuzzy logic
	[175]	RL	Decision making	Actor Critic
	[176]	UL	Anomaly detection	Dimensionality Reduction
	[177]	UL	Information overflow/curse of dimensionality	Clustering, Diffusion Maps
	[178]	RL	Detection making	Actor-critic
COD	[179]	SL	Anomaly detection	SVM, Ensemble methods
	[180]	SL	Diagnosis	Fuzzy logic
	[181]	UL	Anomaly detection	Diffusion Maps
	[178]	SL/UL	Anomaly detection	KNN, local-outlier-factor
	[87]	UL	Diagnosis	Bayesian networks
Self-coordination				
	[182]	RL	Decision making	Actor Critic
	[183]	SL	Sequential decisions	Classification (DTs)
	[184]	RL	Decision making	Actor Critic
	[185]	RL	Decision making	Q-learning
Minimization Drive Tests				
	[146, 186]	SL	Verification/estimation	Linear correlation
	[127]	SL	Prediction	Regression models
	[147]	SL/UL	Prediction/curse of dimensionality	Regression models/Dimensionality reduction
	[149]	UL	Curse of dimensionality/detection making	Dimensionality Reduction
	[148]	SL	Prediction	Bagged-SVM/Dimensionality reduction
Core Networks				
	[187]	SL	Prediction	Adaboost, SVM

network optimisation, whereas in [160] and [161] focus in a LTE SON network optimisation. The solution enables self-tuning of HO parameters to learn optimal parameters' adaptation policies. In [160] the authors take advantage of the Q-learning approach to effectively reduce the call drop rates, whereas in [161], unlike other solutions that assume a general constant mobility, the authors adjust the HO settings in response to the mobility changes in the network by means of a distributive cooperative Q-learning. Different from [160, 161], in [188] and [157], the authors take advantage also of fuzzy logic capabilities. Additionally, in terms of solutions based on unsupervised learning, we can find in the literature the works of [158] and [159]. In these works the authors propose an approach to HO management based on UL and SOM analysis. The idea is to exploit the experience gained from analysing data of the network based on the angle of arrival and the received signal strength of the user to learn specific locations where HO have occurred and decide whether to allow or forbid certain handovers.

Coverage and Capacity Optimization

A number of papers that relate to CCO solutions based on ML methods, are also find in the current literature. For this approach, a RL technique is quite common to interact with the environment to optimise the antenna tilt and transmission power. In [163] and [165] a fuzzy Q-learning approach to optimise the complex wireless network by learning the optimal antenna tilt control policy has been proposed, and a similar approach is followed also in [189] and [164]. In both works, the authors apply Q-Learning technique in a decentralized manner for the joint optimization of coverage and capacity task. But they also combine the fuzzy rule with the Q-Learning to deal with the realistic problem whose input and output variables are continuous. In addition, in [189], a central control mechanism which is responsible for the initialization and the termination of the optimization process of every learning agent deployed in every eNB is introduced, whereas in [164], the authors optimise the antenna tilt and transmission power considering the traffic load distribution of the cells in a specific region. The solution introduces a mechanism to facilitate cooperative learning among different SON entities.

Inter-cell Interference Coordination

ML has been proposed in the literature of ICIC use case as a valid solution, where RL has the greatest amount of works compared to other areas. In addition, several work target the problem to minimize de interference among cells by means of Q-learning. [166–168, 171]. In [166] the authors proposed a distributed Q-learning algorithm to control the interference at each resource block by integrating multi-user scheduling in the operation of the macro cell network, whereas in [171] a fuzzy Q-learning approach has been proposed to solve the ICIC in OFDMA cellular network, and a decentralised Q-learning framework for interference management in small cells is proposed in [168]. Different from previous works, an example of channel assignment based on geographical location using neural networks is offered in [169]. Regarding the prediction/estimation to ICIC, in [162, 170], the authors focus on mobility management using regression to find correlations patterns between different KPIs and the RRM parameters.

Cell Outage Compensation

The literature already offers different works targeting the problem of COC. For this use case RL has been proved as a valid solution since it is a continuous decision making/control problem. In this context a contribution in the area of self-healing has been presented in [175, 178], where the authors present a complete solution for the automatic mitigation of the degradation effect of the outage by appropriately adjusting suitable radio parameters of the surrounding cells. The solution that they provide is based on optimising the coverage and capacity of the identified outage zone, by adjusting the gain of the antenna due to the electrical tilt and the downlink transmission power of the surrounding eNBs. To implement this approach, they propose a RL scheme to take advantage of its capability of making online decisions at each eNB, and of providing decisions adapting to the evolution of the scenario in terms of mobility of users, shadowing, etc., and of the decisions made by the surrounding nodes to solve the same problem. A COC contribution also based on ML is targeted in [174], where fuzzy logic and RL is proposed as the driving technique to fill a coverage gap. The authors show performance gains by using different parameters, such as, the power transmission, the antenna tilt, and a hybrid of both schemes. Alternatively, in [176, 177] the authors proposed self-healing and self optimization algorithms. They introduce a data mining framework to detect coverage holes, and sleeping cells. They take advantage of different KPIs provided by the users to guarantee the network performance by means of regression models.

Cell Outage Detection

As we already mentioned, COD aims to autonomously detect cells that are not operating properly due to possible failures. For this kind of problem, anomaly detection algorithms offer an interesting solution which allows to identify outliers measurements which can be highlighting a hidden problem in the network. As a consequence, approaches on this use case were carried out by different authors, such the one in [181], where a contribution for the detection of sleeping cell problem is presented. The method applied to detect anomalous network behaviour is diffusion maps by means of clustering. Additionally, a solution using fuzzy logic, for the automated the diagnosis stage of a troubleshooting system is presented in [180]. In order to determine if there is a failure, the authors propose a controller, which receives as an inputs a set of representative KPIs. A similar approach is presented by [87], where the authors present an automated diagnosis model for UMTS networks based on naive bayesian classifier, and where the model uses both network simulator and real UMTS network measurements. For similar reasons, the use of different KPIs is also exploited in [179], where the authors focus on anomaly detection problem. The paper addresses both the outage case and the case where the cell can provide a certain level of service, but its performance has degraded UEs experience. They use ensemble method to train the KPIs extracted by human operators to make informed decisions. Also, for control COD, in [178], the authors have utilized the data gathering of MDT reports to develop models by applying anomaly detection techniques. They compared the performance of two anomaly detecting algorithms.

Energy Savings

Energy savings schemes for wireless cellular systems have been proposed in the past, enabling cells to go into a sleep state, in which consume a reduced amount of energy. In order to reduce the energy consumption of the eNBs, we can found several works related to ML techniques. This domain has attracted a lot of attention to research communities since it allows energy savings by learning a policy by the iterations with the environment taking into account every change over time, such as the daily solar irradiation. An example of that can be found in [172], where the authors take advantage of RL to propose a decentralized Q-learning approach to enhance this case of study. Also, in [173], the authors switch off some underutilized cells during off peak hours. The proposed approach optimises the number of base stations in dense LTE pico cell deployments in order to maximize the energy saving, they use a combination of Fuzzy Logic, Grey Relational Analysis and Analytic Hierarchy Process tools to trigger the switch off actions, and jointly consider multiple decision inputs for each cell. Also for HetNets, we find several works, such as, [190, 191]. where the authors take advantage of the channel quality information available in the network for the construction of different kind of databases. In [190] the authors take advantage of such a a database to analyse the potential gains that can be achieved in clustered small cell deployments.

SON Conflicts Coordination

As the deployment of stand-alone SON functions is increasing, the number of conflicts and dependencies between them also increases. Hence, an entity has been proposed for the coordination of this kind of conflicts. In this context, current literature includes several works based on ML. In [183] the authors focus on the classification of potential SON conflicts and on discussing the valid tools and procedures to implement a solid self-coordination framework. Q-Learning, as a RL method, has been proposed in [184] to take advantage of experience gained in past decisions, in order to reduce the uncertainty associated with the impact of the SON coordinator decisions when picking an action over another to resolve conflicts. Consequently, in [185], the authors use Q-learning to deal with the conflict resolution between two SON instances. Additionally, in [182] the authors provide a functional architecture that can be used to deal with the conflicts generated by the concurrent execution of multiple SON functions. They show that the proposed approach is general enough to model all the SON functions and their derived conflicts. They focus on the self-optimization functionality and on all the associated SON functions, as defined in [84]. First they introduce these SON functions in the context of the general SON architecture, together with high-level examples of how they may interfere. Second, they define the state and action spaces of the global MDP that models the self-optimization procedure of the overall RAN segment. Finally, they show that the global self-optimization problem can be decomposed onto as many Markov decision sub-processess (sub-MDPs) as SON functions.

Minimization of Drive Tests

Based on the works that the literature already offers, we observe that most of the approaches use the supervised and unsupervised learning techniques to provide different solutions for this use case. An example of that can be observed in [146, 186], where the authors address the QoS estimation by selecting different KPIs and correlating them with common nodes measurements, to establish whether a UE is satisfied with the received QoS. We observe that the most of the works available in literature cover only traditional macrocell scenarios and do not focus on more complex multi layer heterogeneous networks. Additionally, in [127], the authors focus on multi layer heterogeneous networks. They present an approach, based on regression models, which allows to predict QoS in heterogeneous networks for UEs, independently of the physical location of the UE. This work is extended in [147] by taking into account the most promising regression models, but also analysing dimensional reduction techniques. By doing

PCA/SPCA on the input features, and promoting solutions in which only a small number of input features capture most of the variance, the number of random variables under consideration is reduced. Based on previous results, in [148] the same authors defined a methodology to built a tool for smart and efficient network planning, based on QoS prediction derived by proper data analysis of UE measurements in the network. Finally, in [149], the authors consider large data sets to identify anomaly behaving base station. They proposed an algorithm consisting of preprocessing, detection and analysis phases. The results show that by using dimensionality reduction and anomaly detection techniques irregularly behaving base stations can be detected in a self-organised manner.
In addition, the previous works cover a wide range of active measurements, like, time, location, serving-cell info, Reference Symbol Received Power (RSRP), Reference Symbol Received Quality (RSRQ), strongest intra-LTE neighbours, best quality intra-LTE neighbours, Wide-band Channel Quality Indicator (WCQI), Power Headroom Report (PHR). However determining the correct KPIs deserve our attention.

Core Networks

As we already mentioned in section 2.1, the operational aspects of core networks elements can be enhanced through, for example, the automatic configuration of neighbour cell relations function. In this regards, the idea of applying data mining and ML to this function is not new. The works in [192, 193] focus on the ANR function to perform handovers without dropping the connection. The authors have developed an algorithm, which solves a part of this problem by automatically creating and updating neighbour cell relation lists, based on measured network data. Moreover, in [187] the authors study the benefits of using ML to root-cause analysis of session drops, as well as, drop prediction for individual sessions. They present an offline Adaboost and SVM method to create a predictor, which is in charge of eliminate/mitigate the session drops by using real LTE data.

Virtualised and Software Define Networks

Despite the use of SON for virtualised architectures and ML has still not been exploited, there are some ongoing European projects, which aim to target these domains. The COGNET project is an example of that. It aims at developing several use cases by applying ML research to build intelligent network management systems to fulfil the 5G vision [10].

2.4 Conclusions

In this chapter, we have review the field of SON by giving an overview of the functionalities, improvements and enhancements. We presented different machine learning methods, and how these learning methods can be used to target different SON applications. In this context, machine learning was proved as useful tool for current solutions of SON. We can see in these contributions that the level of performance increase with SON implementations.

Bibliography

[1] W.Ross Ashby. "Principles of the self-organizing dynamic system". *Journal of General Psychology* 37 (1947), pp. 125–128.

[2] W.Ross Ashby. *Design for a Brain: The Origin of Adaptive Behaviour. Front Cover. William Ross Ashby. Chapman and Hall, 1960 - Brain - 286 pages.* Cameron. Wiley.

[3] W. Ashby. *Principles of the Self-Organizing System*. Pergamon Press, 1962.

[4] Osianoh Glenn Aliu, Ali Imran, Muhammad Ali Imran, and Barry Evans. "A survey of Self Organisation in Future Cellular Networks". *IEEE Communications Surveys & Tutorials* 15 (2013), pp. 336–361.

[5] SOCRATES. *Self-Optimisation and self-ConfiguRATion in wirelEss networkS*. URL: `{http://www.fp7-socrates.org/}`.

[6] Gandalf. *Monitoring and self-tuning of RRM parameters in a multi-system network*. URL: `http://www.celtic-initiative.org/`.

[7] SEMAFOUR. *Self-Management for Unified Heterogeneous Radio Access Networks*. URL: `http://fp7-semafour.eu/`.

[8] SESAME. *Small Cells Coordination for Multi-tenancy and Edge services*. URL: `https://5g-ppp.eu/sesame/`.

[9] SELFNET. *Framework for Self-Organised Network Management in Virtualised and Software Defined Networks*. URL: `http://www.cognet.5g-ppp.eu/cognet-in-5gpp/`.

[10] COGNET. *Cognitive Networks*. URL: `https://5g-ppp.eu/`.

[11] 3GPP. *Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Self configuring and self optimizing network uses case and solutions (Release 9)*. Tech. rep. TR 36.902, version 9.2.0.

[12] Aliu, O. and Imran, A. and Imran, M. and Evans, B. "A Survey of Self Organisation in Future Cellular Networks". *Communications Surveys Tutorials, IEEE* 99 (2012), pp. 1–26.

[13] 3GPP. *Technical Specification Group Radio Access Network, Evolved Universal Terrestrial Radio Access Network (E-UTRAN), Self configuring and Self optimizing Network (SON) Uses Case and Solutions (Release 9)*. Tech. rep. TR 36.902, version 9.3.1. 2011.

[14] 3GPP. *Study on Network Management (NM) centralized Coverage and Capacity Optimization (CCO) Self-Organizing Networks (SON) function*. Tech. rep. TR 32.836.

[15] 3GPP. *Technical Specification Group Radio Access Network; study on next generation SON for UTRAN and E-UTRAN*. Tech. rep. TR 37.8224, version 12.1.0 Release 12.

[16] 3GPP. *Technical Specification Group Services and System Aspects; Telecommunication Management; Self-Organizing Networks (SON); Concepts and requirements (Release 13)*. Tech. rep. TS 32.500, version 13.0.0. 2016.

[17] 3GPP. *3GPP work items on Self-Organising Networks (Release 13)*. Tech. rep. version 0.1.3., 2014.

[18] 3GPP. *Overview of 3GPP Release 14*. URL: `http://www.3gpp.org/release-14`.

[19] 3GPP. *Technical Specification Group Services and System Aspects; Telecommunication management; Self-Organizing Networks (SON); Concepts and requirements (Release 11)*. Tech. rep. TS 32.500, version 11.1.0. 2011.

[20] 3GPP. *Technical Specification Group Services and System Aspects; Telecommunication management; Self Configuration of Network Elements; Concepts and Requirements (Release 9)*. Tech. rep. TS 32.501. 2010.

[21] 3GPP. *Technical Specification Group Services and System Aspects; Telecommunication management; Self-configuration of network elements Integration Reference Point (IRP); Information Service (IS)(Release 10)*. Tech. rep. TS 32.502, version 10.1.0. 2010.

[22] 3GPP. *Self-Configuration of Network Elements Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA) Solution Set (SS) (Release 8)*. Tech. rep. TR 32.503. 2008.

[23] 3GPP. *Software Management (SWM); Software and Integration Reference Point (IRP) Requirements (Release 10)*. Tech. rep. TR 32.531. 2010.

[24] 3GPP. *Software management Integration Reference Point (IRP); Information Service (IS) (Release 8)*. Tech. rep. TR 32.532. 2009.

[25] 3GPP. *Software Management (SWM) Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS) (Release 9)*. Tech. rep. TR 32.533. 2009.

[26] 3GPP. *Technical Specification Group Services and System Aspects; Telecommunications management; Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP); Requirements (Release 9)*. Tech. rep. TS 32.761. 2009.

[27] 3GPP. *Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP); Information Service(IS), 3rd Generation Parthnership Project (3GPP), (Release 10)*. Tech. rep. TS 32.762. 2010.

[28] 3GPP. *Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP): CORBA solution set (Release 9)*. Tech. rep. TR 32.763. 2009.

[29] 3GPP. *Evolved Universal Terrestrial Radio Access Netwrok (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP): eXtensible Markup Language (XML) definitions (Release 9)*. Tech. rep. TR 32.765, version 9.6.0. 2004.

[30] 3GPP. *Study of Self-Organizing Networks (SON) related Operations, Administration and Maintenance (OAM) for Home Node B (HNB) (Release 9)*. Tech. rep. TR 32.821, version 9.0.0. 2004.

[31] 3GPP. *Study on self-healing (Release 9)*. Tech. rep. TR 32.823. 2010.

[32] 3GPP. *Performance measurements Evolved Universal Terrestrial Radio Access Network (Release 10)*. Tech. rep. TS 32.425, version 10.6.0. 2012.

[33] 3GPP. *Universal Mobile Telecommunications System (UMTS); UTRAN Iu Interface RANAP Signalling*. Tech. rep. TS 25.413, version 3.4.0. 1999.

[34] 3GPP. *Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2 (Release 12)*. Tech. rep. TS 36.300. 2015.

[35] 3GPP. *LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 Application Protocol (S1AP)*. Tech. rep. TS 36.413, version 9.4.0 Release 9. 2010.

[36] 3GPP. *Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access Network (EUTRAN); X2 Application Protocol (X2AP) (Release 11)*. Tech. rep. TS 36.423, version 10.7.0. 2013.

[37] 3GPP. *Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Information Service(IS), 3rd Generation Parthnership Project (3GPP), (Release 10)*. Tech. rep. TS 32.522. 2011.

[38] 3GPP. *Self-Organizing Networks (SON): Policy Network Resource Model (NRM) Integration Reference Point (IRP); Information Service(IS); Solution Set (SS) definitions, (Release 10)*. Tech. rep. TS 32.526. 2011.

[39] 3GPP. *Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Network Resource Model (NRM) Integration Reference Point (IRP); Solution Set(SS) definitions, (Release 10)*. Tech. rep. TS 32.766. 2011.

[40] 3GPP. *Digital cellular telecommunications system; Universal Mobile Telecommunications System (UMTS);LTE; Telecommunication management; Self-Organizing Networks (SON); Self-healing concepts and requirements*. Tech. rep. TS 32.541, version 12.0.0 Release 12.

[41] 3GPP. *Study on Energy Savings Management (ESM) (Release 10)*. Tech. rep. TR 32.826. 2010.

[42] 3GPP. *Technical Specification Group Services and System Aspects; Study on Operations, Administration and Maintenance (OAM) aspects of inter-Radio-Access-Technology (RAT) energy saving (Release 11)*. Tech. rep. TR 32.834. 2012.

[43] 3GPP. *Universal Mobile Telecommunications System (UMTS); LTE; Telecommunication management; Energy Saving Management (ESM); Concepts and requirements*. Tech. rep. version 10.1.0 Release 10. 2011.

[44] 3GPP. *Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol Specification (Release 10)*. Tech. rep. TS 36.331, version 10.7.0. 2010.

[45] 3GPP. *Telecommunication management; Performance Management (PM); Performance Measurements UTRAN (Release 9)*. Tech. rep. TS 32.405. 2009.

[46] 3GPP. *Telecommunication management; Automatic Neighbor Relation (ANR) management, Concepts and Requirements, (Release 11)*. Tech. rep. TR 32.511. 2011.

[47] 3GPP. *Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Requirements (Release 10)*. Tech. rep. TR 32.521, version 10.2.0. 2012.

[48] 3GPP. *Configuration Management (CM); UTRAN network resources Integration Referecne Point (IRP); Network Resource Model (NRM)(Release 9)*. Tech. rep. TS 322.642. 2009.

[49] 3GPP TSG SA WG5 (Telecom Management) Meeting 85. *Study of implementation alternative for SON coordination*. Tech. rep. S5-122330.

[50] 3GPP. *Configuration Management (CM); UTRAN network resources Integration Referecne Point (IRP); Solution Set (Release 10)*. Tech. rep. TS 32.646, version 10.3.0. 2011.

[51] 3GPP. *Technical Specification Group (TSG) RAN; Working Group 2 (WG2); Radio Resource Control (RRC); Protocol specification (Release 10)*. Tech. rep. TS 25.331, version 10.6.0. 2011.

[52] 3GPP. *UTRAN overall description (Release 10)*. Tech. rep. TS 25.401. 2011.

[53] 3GPP. *UTRAN Iu Interface: General Aspects and Principles (Release 10)*. Tech. rep. TS 25.410. 2011.

[54] 3GPP. *UTRAN Iur interface Radio Network Subsystem Application Part (RNSAP) signalling (Release 9)*. Tech. rep. TS 25.423. 2010.

[55] 3GPP. *Study on Network Management (NM) centralized Coverage and Capacity Optimization (CCO) Self-Organizing Networks (SON) function*. Tech. rep. TS 32.836. 2013.

[56] 3GPP. *Telecommunication management; Integration Reference Point (IRP) overview and usage guide (Release 10)*. Tech. rep. TS 32.103, version 10.0.0. 2011.

[57] 3GPP. *Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Requirements (Release 11)*. Tech. rep. TS 28.627, version 11.1.0. 2012.

[58] 3GPP. *Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS) (Release 11)*. Tech. rep. TS 28.628, version 11.4.0. 2014.

[59] 3GPP. *Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS) (Release 11)*. Tech. rep. TS 28.658, version 11.0.0. 2012.

[60] 3GPP. *Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP); Solution Set (SS) definitions (Release 11)*. Tech. rep. TS 28.659. 2012.

[61] 3GPP. *Gap analysis between 3GPP SA5 specifications and NGMN Top Operational Efficiency (OPE) Recommendations (Release 12)*. Tech. rep. TS 32.508. 2013.

[62] 3GPP. *Data formats for multi-vendor plug and play eNode B connection to the network (Release 12)*. Tech. rep. TS 32.509. 2013.

[63] 3GPP. *Gap analysis between 3GPP SA5 specifications and NGMN Top Operational Efficiency (OPE) Recommendations (Release 12)*. Tech. rep. TR 32.838. 2013.

[64] 3GPP. *Study of heterogeneous networks management (Release 12)*. Tech. rep. TR 32.835, version 12.0.0. 2014.

[65] 3GPP. *Study on Operations, Administration and Maintenance (OAM) aspects of Network Sharing (Release 12)*. Tech. rep. TR 32.851, version 12.1.0. 2013.

[66] 3GPP. *Study on Energy Saving Enhancement for E-UTRAN (Release 12)*. Tech. rep. TR 36.887. 2013.

[67] 3GPP. *RAN Evolution of LTE in Release 14*. 2016. URL: {http://www.3gpp.org/news-events/3gpp-news/1768-ran_rel14}.

[68] 3GPP. *Technical Specification Group Services and System Aspects; Telecommunication management; Project scheduling and open issues for SA5 (Release 8)*. Tech. rep. TR 30.818, version 8.0.0. 2010.

[69] t. . M. Amirijoo, P. Frenger, F. Gunnarsson ().

[70] Flavio Parodi, Mikko Kylvaja, Gordon Alford, Juan Li and Jose Pradas. "An Automatic Procedure for Neighbor Cell List Definition in Cellular Networks". *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)* (2007), pp. 1–6.

[71] 3GPP. *From large lists of potential neighbour cells to self-optimised neighbour cell lists*. Tech. rep. R3-071239. Mitsubishi Electric, 2010.

[72] D. Kim et al. "Self-Configuration of Neighbor Cell List Utilizing E-UTRAN NodeB Scanning in LTE Systems". *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE* (2010), pp. 1–5.

[73] Flavio Parodi, Mikko Kylvaja, Gordon Alford, Juan Li and Jose Pradas. "Recommended Practices for Multi-vendor SON Deployment". *D2- NGMN Aliance V1.0* (2014).

[74] A. Temesvary. "Self-configuration of antenna tilt and power for plug and play deployed cellular networks". *IEEE Wireless Communications and Networking Conference (WCNC)* (2009), pp. 1–6.

[75] C. Huan and X. Minghai. "Self-configuration model of HeNB within LTE". *in proc. of 12th IEEE Int Communication Technology (ICCT)* (2010), pp. 84–97.

[76] A. Eisenblatter, U. Turk e, and C. Schmelz. "Self-Configuration in LTE Radio Networks: Automatic Generation of Enodeb Parameters". *IEEE 73rd Vehicular Technology Conference (VTC Spring)* (2011), pp. 1–3.

[77] Nawrocki M.J., Dohler M. "Understanding UMTS Radio Network Modelling, Planing and Automated Optimsation in Theory and Practice". *Wiley and Sons* (2008).

[78] 3GPP. *Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures (Release 10)*. Tech. rep. TS 36.213, version 10.2.0. 2011.

[79] NGMN. "Recommended Practices for Multi-vendor SON Deployment" (2014).

[80] 3GPP. *Envolved Universal Terrestrial Radio Access(E-UTRA)and Evolved Universal Terrestrial Radio AccessE-UTRAN; Overall description; Stage 2*. Tech. rep. TS 36.300.

[81] 3GPP. *Study on Enhancements of OAM aspects of distributed Self-Organizing Networks (SON) functions (Release 12)*. Tech. rep. TR 32.860. 2014.

[82] C.M. Muller, M. Kaschub, C. Blankenhorn, S. Wanke. “A cell outage detection algorithm using neighbour cell list reports”. *International Workshop on Self Organizing System* (2008), pp. 218–229.

[83] 3GPP. *WG S5-460036 SON Self-healing management*. URL: `http://www.3gpp.org/DynaReport/GanttChart-Level-2.htm#bm460036`.

[84] Seppo Hämäläinen,Henning Sanneck,Cinzia Sartori. *LTE Self-Organising Networks (SON): Network Management Automation for Operational Efficiency*. John Wiley and Sons, 2012.

[85] M. Amirijoo, L. Jorguseski, R. Litjens and R. Nascimento. “Effectiveness of Cell Outage Compensation in LTE Networks”. *The 8th Annual IEEE Consumer Communications and Networking Conference - Wireless Consumer Communication and Networking* (2011).

[86] M. Amirijoo, L. Jorguseski, T. Kurner, R. Litjens, M. Neuland, L.C. Schmelz, U. Turke. “Cell Outage Management in LTE Networks”. *6th International Symposium on Wireless Communication Systems (ISWCS)* (2009), pp. 600–604.

[87] Rana M. Khanafer, Beatriz Solana, Jordi Triola, Raquel Barco, Lars Moltsen, Zwi Altman, and Pedro Lazaro. “Automated Diagnosis for UMTS Networks Using Bayesian Network Approach”. *IEEE Transactions on Vehicular Technology* 57 (2008).

[88] Lars Christoph Schmelz, Mehdi Amirijoo, Andreas Eisenblaetter, Remco Litjens, Michaela Neuland, John Turk. “A Coordination Framework for Self-Organisation in LTE Networks”. *IEEE International Symposium on Integrated Network Management* (2011).

[89] T. Jansen, M. Amirijoo, U. Turke, L. Jorguseski, K. Zetterberg, R. Nascimiento, L.C. Schelz, J.Turk and I. Balan. “Embedding multiple self organisation functionalities in future radio access networks”. *Information of Vehicular Technology Conference (VTC Spring)* (2009), pp. 1–5.

[90] J. Chen, H. Zhuang, B. Andrian, Y. Li. “Difference-based joint parameter configuration for MRO and MLB”. *IEEE Vehicular Technology Conference VTC2012-Spring* ().

[91] T. Bandh, H. Sanneck, and R. Romeikat. “An Experimental System for SON Function Coordination”. *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd* (2011), pp. 1–2.

[92] Raphael Romeikat, Bernhard Bauer, Tobias Bandh, Georg Carle, Henning Sanneck, Lars-Christoph Schmelz. "Policy-driven workflows for mobile network management automation". *IWCMC* (2010), pp. 1111–1115.

[93] 3GPP. *Technical Specification Group Radio Access Network, Evovled Universal Terrestrial Radio Access (E-UTRA); Study on Minimisation of Drive Tests in Next Generation Networks*. Tech. rep. TR 36.805, version 9.0.0 Release 9.

[94] Johansson, J., Hapsari, W.A., Kelley, S., Bodog, G. "Minimization of drive tests in 3GPP (Release 11)". *IEEE Communications Magazine* (2012).

[95] W. Hapsari, A. Umesh, M. Iwamura, M. Tomala, B. Gyula, and B. Sebire. "Minimization of drive tests solution in 3GPP". *Communications Magazine* (2012), pp. 28–36.

[96] ERICSSON. "How to verify an LTE deployment-Transparent network-performance verification for LTE rollouts". *White paper* (2012).

[97] C. Rattaro and P. Belzarena. "Throughput prediction in wireless networks using statistical learning". *Latin-American Workshop on Dynamic Networks. Buenos Aires* (2010).

[98] Mirza, M. and Springborn, K. and Banerjee, S. and Barford, P. and Blodgett, M. and Xiaojin Zhu. "On The Accuracy of TCP Throughput Prediction for Opportunistic Wireless Networks". *6th Annual IEEE Communications Society; Sensor, Mesh and Ad Hoc Communications and Networks* (2009), pp. 1–9.

[99] N. siemens networks. *Automating Core Network Management, Mobile World Congress*. 2012. URL: `http://networks.nokia.com/news-events/press-room/press-releases/world-s-first-intelligent-automation-for-core-networks-mwc12`.

[100] E. A. Weiss. "Arthur Lee Samuel (1901-90)". *IEEE Annals of the History of Computing 14* (1992), pp. 55–69.

[101] Thomas M. Mitchell. "Machine Learning". *McGraw Hill, Inc. New York, NY, USA* (1997).

[102] Schapire, R.E. "Using Output Codes to Boost Multiclass Learning Problems". *14th International Conference on Machine Learning* (1996), pp. 313–321.

[103] McKinsey Global Institute. "Big data: The next frontier for innovation, competition, and productivity" (2011).

[104] Caruana, R., Niculescu-Mizil. "An empirical comparison of supervised learning algorithms". *in proc. of 23rd International Conference on Machine Learning* (2006).

[105] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome. *The Elements of Statistical Learning (2nd ed.)* 2009, pp. 520–528.

[106] McCullagh, Peter, Nelder, John. "Generalized Linear Models, Second Edition". *Boca Raton: Chapman and Hall* (1989).

[107] Zhang, Harry. "The Optimality of Naive Bayes" (2004).

[108] V. Vapnik. "An Overview of Statistical Learning Theory". *IEEE Transactions on Neural Networks* 10.5 (1999).

[109] A. J. Smola and B. Scholkopf. "A tutorial on support vector regression". *Statistics and Computing* (2004), pp. 199–222.

[110] Andreas Müller. "Kernel Approximations for Efficient SVMs (and other feature extraction methods)" (2012).

[111] Bishop M.C. "Pattern recognition and machine learning". *Busines Dia, Llc. Springer Science* (2006).

[112] Quinlan, J. R. "Induction of Decision Trees. Machine Learning ". *Kluwer Academic Publishers* (1986), pp. 81–106.

[113] Rokach, Lior; Maimon, O. "Data mining with decision trees: theory and applications". *World Scientific Pub Co Inc.* (2008).

[114] Opitz, D., Maclin, R. "Popular Ensemble Methods: An Empirical Study". *Journal of Artificial Intelligence Research 11* (1999), pp. 169–198.

[115] Rokach, L. "Ensemble-based classifiers". *Artificial Intelligence* (2010), pp. 1–39.

[116] Dietterich, T.G. "An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting and Randomization". *Machine Learning* 40 (2000), pp. 139–157.

[117] Trevor Hastie, Robert Tibshirani, Jerome Friedman. *The Elements of Statistical Learning Data Mining, Inference, and Prediction*. Springer series in statistic, 2009.

[118] Thomas G. Dietterich. "Machine-Learning Research: Four Current Directions". *American Association for Artificial Intelligence* (1997).

[119] Freund, Y., and Schapire, R.E. "A Decision-Theoretic Generalization of On-Line Learning and Application to Boosting". *2nd European COnference on Machine Learning* (1995), pp. 148–156.

[120] Quinlan J.R. "Bagging, Boosting and c4.5". *13th National Conference on Artificial Intelligence* (1996), pp. 725–730.

[121] Kong, E.B., and Dietterich T.G. "Error-Correcting Output Coding Correct Bias and Variance". *12th International Conference on Artificial Intelligence* (1995), pp. 313–321.

[122] Chan, P.K., and Stolfo, S.J. "Learning Arbiter and Combiner Trees from Partittioned Data for Scaling Machine Learning". *1st International Conference on Knowledge Discovery and Data Mining* (1995), pp. 39–44.

[123] Zoubin Ghahramani. "Unsupervised Learning". *Gatsby Computational Neuroscience Unit University College London, UK* (2004).

[124] Hartigan, J., and Wang, M. "A K-means clustering algorithm". *Applied Statistics* (1979), pp. 100–108.

[125] Ultsch, Alfred, Siemon, H. Peter. "Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis". *International Neural Network Conference* (1990).

[126] Nock, R. and Nielsen, F. "On Weighting Clustering". *IEEE Trans on Pattern Analysis and Machine Intelligence* 28 (2006), pp. 1–13.

[127] Jessica Moysen, Lorenza Giupponi, Nicola Baldo, Josep Mangues-Bafalluy. "Predicting QoS in LTE HetNets based on location-independent UE measurement". *in proc. of 20th IEEE Computer Aided Modelling and Design of Communication Links and Networks, Guildford, UK* (2015).

[128] Roweis, S. T., Saul, L. K. "Nonlinear Dimensionality Reduction by Locally Linear Embedding". *Science* (2000), pp. 2323–2326.

[129] S. T. Roweis. "EM algorithms for PCA and SPCA". *Advances in Neural Information Processing Systems, The MIT Press* (1998).

[130] Hodge, V. J., Austin, J. "A Survey of Outlier Detection Methodologies". *Artificial Intelligence Review* 22 (2004).

[131] S. Schaal and Christopher Atkeson. "Robot juggling: An implementation of memory-based learning". *Control Systems Magazine* (1994).

[132] Sebastian Thrun. "Learning to play the game of chess". *Advances in Neural Information Processing Systems 7, Cambridge, MA, 1995. The MIT Press.* (1995).

[133] Michael L. Littman. "Markov games as a framework for multi-agent reinforcement learning". *the Eleventh International Conference on Machine Learning* (1994), pp. 157–163.

[134] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[135] L. Panait, S. Luke. “Cooperative Multi-Agent Learning: The State of the Art, Autonomous Agents and Multi-agent systems” (2005).

[136] Y. Bengio, I. J. Goodfellow, and A. Courville. “Deep Learning”. Book in preparation for MIT Press. 2015. URL: http://www.iro.umontreal.ca/~bengioy/dlbook.

[137] Li Deng and Dong Yu. “Deep Learning, Methods and Applications”. *Foundations and Trends in Signal Processing* (2014).

[138] Geoffrey Hinton. “A Practical Guide to Training Restricted Boltzmann Machines” (2010).

[139] D. B. Networks. *Online*. URL: http://deeplearning.net/tutorial/DBN.html.

[140] Ruslan Slakhutdinov, Geoffrey Hinton. “Deep Boltzmann Machines”. *in proc. of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS)* (2009).

[141] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Dalong Du, Chang Huang, and Philip H. S. Torr. “Conditional Random Fields as Recurrent Neural Networks”. *International Conference on Computer Vision (ICCV)* (2015).

[142] Nicola Baldo, Lorenza Giupponi, Josep Mangues-Bafalluy. “Big Data Empowered Self Organized Networks”. *20th European Wireless (EW) Conference; Barcelona, Spain* (2014).

[143] 3GPP. *Technical Specification Group Services and Systems Aspects; Charging management; Charging Data Record (CDR) parameter description*. Tech. rep. TS 32.298, version 13.3.0 Release 13. 2016.

[144] 3GPP. *Telecommunication management; Performance Management (PM); Concept and requirements*. Tech. rep. TS 32.401, version 10.2.0, Release 10.

[145] 3GPP. *Technical Specification Group TSG RAN; Radio measuremnt collection for Minimization of Drive Tests (MDT)*. Tech. rep. TS 37.320, version 2.0.0 Release 10. 2010.

[146] Fedor Chernogorov and Jani Puttonen. “User satisfaction classification for Minimization of Drive Tests QoS verification”. *24th IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications, PIMRC 2013, London, United Kingdom* (2013), pp. 2165–2169.

[147] Jessica Moysen, Lorenza Giupponi, Josep Mangues-Bafalluy. “On the Potential of Ensemble Regression Techniques for Future Mobile Network Planning”. *in proc. of The Twenty-First IEEE Symposium on Computers and Communications (ISCC), Messina, Italy* (2016).

[148] Jessica Moysen, Lorenza Giupponi, Josep Mangues-Bafalluy. “A Machine Learning enabled network Planning tool”. *in proc. of The Twenty-Seven IEEE Personal Indoor and Mobile Radio Communications (PIMRC), Valencia, Spain* (2016).

[149] Jussi Turkka, Gil David. “Anomaly Detection Framework for Tracing Problems in Radio Networks”. *ICN: The Tenth International Conference on Networks* (2011).

[150] Ali Imran, Ahmed Zoha, and Adnan Abu-Dayya. “Challenges in 5G: How to Empower SON with Big Data for Enabling 5G”. *IEEE Network* (2014).

[151] P. Hoen and K. Tuyls. “Analyzing Multi-agent Reinforcement Learning using Evolutionary Dynamics”. *15th European Conference on Machine Learning (ECML)* (2004).

[152] P. Muñoz, R. Barco, I. de la Bandera. “Load Balancing and Handover joint optimization in LTE networks using Fuzzy logic Reinforcement Learning”. *Computer Networks Elsevier* (2014), pp. 113–125.

[153] E. Bergner. “Unsupervised Learning of Traffic Patterns in Self-Optimizing 4th Generation Mobile Networks”. *Master of Science Thesis, KTH Computer Sicence and Communications, Stockolm, Sweden* (2012).

[154] P. Muñoz, R. Barco, I. de la Bandera. “Optimization of load balancing using fuzzy Q-Learning for next generation wireless networks”. *Expert Systems with Applpications* (2013), pp. 984–994.

[155] Stephen S. Mwanje, Andreas Mitschele-Thiel. “Minimizing Handover Performance Degradation Due to LTE Self Organized Mobility Load Balancing”. *77th IEEE Conference on Transactions on Vehicular Technology (VTC Spring)* (2013), pp. 1–5.

[156] Junichi Suga, Yuji Kojima, Masato Okuda. “Centralized Mobility Load Balancing Scheme in LTE Systems”. *8th International Symposium on Wireless Communication Systems, Aachen* (2011).

[157] Andreas Klein, Nandish P. Kuruvatti, Jorg Schneider, Hans D. Schotten. “Fuzzy Q-Learning for Mobility Robustness Optimization in Wireless Networks” (2013).

[158] Neil Sinclair, David Harle, Ian A. Glover, James Irvine and Robert C. Atkinson. “Parameter Optimization for LTE Handover using an Advanced SOM Algorithm”. *VTC* (2013).

[159] N. Sinclair, D. Harle, I. Glover, J. Irvine, and R. Atkinson. “An advanced SOM algorithm applied to handover management within LTE” (2013), pp. 183–1894.

[160] W. Qin, Y. Teng, M. Song, Y. Zhang, and X. Wang. “A Q-learning approach for mobility robustness optimization in LTE-SON”. *15th IEEE International Conference on, Communication Technology (ICCT)* (2013), pp. 818–822.

[161] Stephen S. Mwanje, Andreas Mitschele-Thiel. “Distributed cooperative Q-learning for mobility-sensitive handover optimization in LTE SON”. *IEEE Symposium the Computers and Communication (ISCC)* (2014), pp. 1–6.

[162] B. S. Moazzam Islam Tiwana and Z. Altman. “Statistical learning for automated RRM: Application to eUTRAN mobility” (2009).

[163] R. Razavi, S. Klein and H. Claussen. “A fuzzy reinforcement learning approach for self-optimization of coverage in LTE networks”. *Bell Labs Tachnical Journal* 15 (2010), pp. 153–175.

[164] H. T. Shaoshuai Fan and C. Sengul. “Self-optimization of coverage and capacity based on a fuzzy neural network with cooperative reinforcement learning”. *Journal on Wireless Communications and Networking* (2014).

[165] M. Naseer ul Islam, Mitschele-Thiel. “A: Cooperative fuzzy Q-learning for self-organized coverage and capacity optimization”. *IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)* (2012).

[166] Galindo-Serrano, A. and Giupponi, L. and Auer, G. “Distributed Learning in Multiuser OFDMA Femtocell Networks”. *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd* (2011), pp. 1–6.

[167] Meryem Simsek, Andreas Czylwik, Ana Galindo-Serrano, Lorenza Giupponi. “Improved decentralized Q-learning algorithm for interference reduction in LTE-femtocells”. *5th International Workshop on Self-Organizing Networks IWSON, Glasgow* (2015).

[168] Mehdi Bennis, Samir M. Perlaza, Pol Blasco. “Self-Organization in Small Cell Networks: A Reinforcement Learning Approach”. *IEEE Transactions on Wireless Communications* 12.7 (2013).

[169] Osianoh Glenn, Aliu Muhammad, Ali Imran, Barry Evans. “Self organised cellular networks as the future of wireless communication” (2010).

[170] Tiwana, M.I., Aitman, Z., and Sayrac, B. “Automated RRM optimization of LTE networks using statistical learning”. *8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)* (2010), pp. 564–572.

[171] M. Dirani and Z. Altman. “A cooperative reinforcement learning approach for inter-cell interference coordination in OFDMA cellular networks”. *8th Int Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)* (2010), pp. 170–176.

[172] M.Miozzo, L. Giupponi, M. Rossi, P. Dini. “Distributed Q-Learning for Energy Harvesting Heterogeneous Networks”. *IEEE ICC 2015 Workshop on Green Communications and Networks with Energy Harvesting, Smart Grids, and Renewable Energies* (2015).

[173] Anna Dudnikova, Paolo Dini, Lorenza Giupponi, Daniela Panno. “Multiple Criteria plus Fuzzy Logic Switch Off Method for Dense Heterogeneous Networks”. *in proc. of the 20th IEEE Computer Aided Modelling and Design of Communication Links and Networks, Guildford (UK)* (2015).

[174] Saeed, A. and Aliu, O.G. and Imran, M.A. *IEEE Wireless Communications and Networking Conference (WCNC), April, Paris, France,* ().

[175] J. Moysen and L. Giupponi. “A Reinforcement Learning based solution for Self-Healing in LTE networks”. *in proc. of the 80th IEEE Vehicular Technology Conference (VTC Fall), Vancouver, Canada* (2014).

[176] Jussi Turkka, Fedor Chernogorov, Kimmo Brigatti, Tapani Ristaniemi, and Jukka Lempiainen. “An Approach for Network Outage Detection from Drive-Testing Databases”. *Journal of Computer Networks and Communications* (2012).

[177] J. Turkka. “Aspects of Knowledge Mining on Minimizing Drive Tests in Self-organizing Cellular Networks”. *Doctoral dissertation, Tampereen teknillinen yliopisto - Tampere University of Technology* (2014).

[178] O. Onireti, A. Zoha, J. Moysen, A. Imran, L. Giupponi, M. Ali Imran, A. Abu-Dayya. “A Cell Outage Management Framework for Dense Heterogeneous Networks”. *IEEE Transactions on Vehicular Technology* 64 (2015), pp. 2097–2113.

[179] Gabriela F. Ciocarlie, Szabolcs NovHenning Sanneckaczki. “Detecting Anomalies in Cellular Networks Using an Ensemble Method”. *9th CNSM and Workshops* (2013).

[180] E. J. Khatib, R. Barco, A. Gomez-Andrades, I. Serrano. “Diagnosis based on genetic fuzzy algorithms for LTE Self-Healing”. *IEEE Transactions on Vehicular Technology* (2015).

[181] Chernogorov, F. and Turkka, J. and Ristaniemi, T. and Averbuch, A. “Detection of Sleeping Cells in LTE Networks Using Diffusion Maps”. *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd* (2011), pp. 1 –5.

[182] J. Moysen and L. Giupponi. “Self-coordination of parameter conflicts in D-SON architectures: a Markov decision process framework”. *EURASIP Journal on Wireless Communications and Networking* (2015).

[183] Hafiz Yasar Lateef, Ali Imran, Adnan Abu-dayya. “A Framework for Classification of Self-Organising Network Conflicts and Coordination Algorithms”. *In Proceedings of IEEE Personal Indoor and Mobile Radio Communications PIMRC* (2013).

[184] j. Iacoboaeia, O. and Sayrac, B. and Ben Jemaa, S. and Bianchi, P. ().

[185] Pablo Muñoz, Raquel Barco, and Sergio Flores. "Conflict Resolution Between Load Balancing and Handover Optimization in LTE Networks". *IEEE Communications Letters* 18.10 (2014), pp. 1795–1798.

[186] Fedor Chernogorov and Timo Nihtilä. "QoS Verification for Minimization of Drive Tests in LTE Networks". *Proceedings of the 75th IEEE Vehicular Technology Conference, (VTC) Spring, Yokohama, Japan* (2012), pp. 6–9.

[187] Bálint Daróczy, András Benczúr, Péter Vaderna. "Machine Learning based session drop prediction in LTE networks and its SON aspects". *5th International Workshop on Self-Organizing Networks IWSON, Glasgow* (2015).

[188] Pablo Muñoz, Raquel Barco, and Isabel de la Bandera. "On the Potential of Handover Parameter Optimization for Self-Organizing Networks". *IEEE Transactions on Vehicular Technology* (2013).

[189] X. S. W. L. Jingyu Li Jie Zeng and J. Wang. "Self-Optimization of Coverage and Capacity in LTE Networks Based on Central Control and Decentralized Fuzzy Q-Learning". *International Journal of Distributed Sensor Networks, HINDAWI* (2012).

[190] Emmanuel Ternon, Patrick Agyapong, Liang Hu, Armin Dekorsy. "Energy Savings in Heterogeneous Networks with Clustered Small Cell Deployments". *in proc. of The 11th IEEE International Symposium on Wireless Communications Systems (ISWCS)* (2014).

[191] E. Ternon, P. Agyapong, L. Hu, and A. Dekorsy. "Database-aided Energy Savings in Next Generation Dual Connectivity Heterogeneous Networks". *IEEE WCNC'14 Track 3: Mobile and Wireless Networks* (2014).

[192] Anders Dahlén, Arne Johansson, TeliaSonera, Fredrik Gunnarsson, Johan Moe, Thomas Rimhagen, Harald Kallin. "Evaluations of LTE Automatic Neighbor Relations". *Ericsson* (2015).

[193] Ho Van Quan, Tobias Astrom,Mikael Jern, Johan Moe, Fredrik Gunnarsson, Harald Kallin. "Visualization of Self-Organizing Networks Operated by the ANR Algorithm". *IEEE International Conference on Computing and Communication Technologies* (2009).

Chapter 3

RL based solution for self-healing

Strategy requires thought, tactics require observation. **Max Euwe**

Contents

In this Chapter we present a contribution in the area of Self Healing (SH). We provide a solution for COC based on the TD learning tools described in previous chapters. As we mentioned in chapter 2, COC is applied to alleviate the outage caused by the loss of service from the faulty sector. For this use case, an adequate reaction is vital for the continuity of the service. As a result, vendor specific COD schemes have also to be designed [1]. We consider that the network has already been capable of detecting the degradation of the service and identifying the outage. We propose that a COC module is implemented in a distributed manner in the eNBs in the scenario and intervenes when a fault is detected and so the associated outage, i.e., during a certain period of time a sector is not able to provide service to its users.

The solution that we provide is based on optimising the coverage and capacity of the identified outage zone, by adjusting the gain of the antenna due to the electrical tilt and the downlink transmission power of the surrounding eNBs. With the objective of controlling the inter-cell interference generated at the borders of the extended cells, a modified Fractional Frequency Reuse (FFR) scheme is proposed for scheduling. Among the RL methods, we select a TD learning approach, the AC, for its capability of continuously interacting with the complex wireless cellular scenario and learning from experience. Results, validated on the ns3 LENA platform based on 3GPP component LTE, demonstrate that our approach outperforms state of the art resource allocation schemes in terms of number of users recovered from outage.

The outline of the chapter is organised as follows. Section 3.1 provides the details of the COC proposed approach. Section 3.2 introduces the details of our learning approach. Section 3.3 presents the considered simulation platform. Section 3.4 discusses relevant simulation results. Finally, section 3.5 concludes the chapter.

3.1 System model

We consider a LTE cellular network composed of a set of $\mathcal{M}$ eNBs. The $M{=}|\mathcal{M}|$ eNBs form a regular hexagonal network layout with inter-site distance D, and provide coverage over the entire network. We denote by $\mathbf{p}^m = (p_1^m, \ldots, p_R^m)$ the transmission power vector of eNB $m{\in}\mathcal{M}$, with p_r^m denoting the downlink transmission power of Resource Block (RB) r. The multi user resource assignment to the R RBs is carried out by a FFR like scheduler, described later in this section. Each of the N users is assigned a Channel Quality Indicator (CQI) value $CQI = (CQI_1, ..., CQI_N)$. The maximum transmission power of each eNB is $P_{\max}^{\mathrm{M}}$, such that $\sum_{r=1}^{R} p_r^m \leq P_{\max}^{\mathrm{M}}$. We consider U UEs moving around the scenario and provided with service by the $|\mathcal{M}|$ eNBs. We assume that a sector in the scenario is down, in particular, sector 4 depicted in Figure 3.1. This generates a deep outage zone affecting the population of users active in the geographical area. We assume that this fault is already diagnosed and that the solution provided by the self-organised network is to activate the COC module implemented in the eNBs.

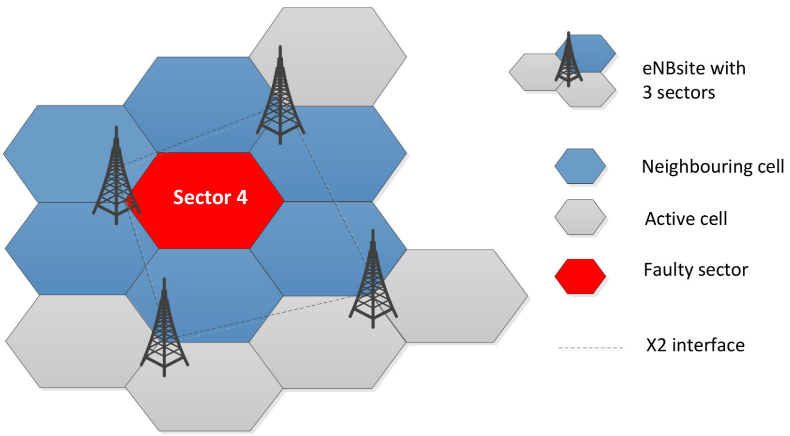

Figure 3.1: COC scenario

In particular, the surrounding eNBs automatically and continuously adjust, based on interactions with the environment carried out by a learning process, their antenna tilt and downlink transmission power levels, till the coverage gap is filled. In order to avoid inter-cell interference at the new cell borders when the cells extend their coverage area, eNBs negotiate over the X2 interface the scheduling criteria that they are following to allocate the new outage users. We propose a modified FFR scheme, which, at the beginning of each subframe divides each cell into two regions, and assigns a certain group of RBs to the inner region and another, orthogonal, to the outer region, at the border of the cell. The inner regions of all the cells in the scenario are covered by the same spectrum, but the outer regions are assigned orthogonal RBs to reduce the effect of inter-cell interference generated by the COC coverage enhancement scheme. The overall resource allocation scheme is designed taking into account the spectral efficiency and in particular, the corresponding CQI of all the users in the scenario. Moreover since the eNB antenna tilt is an important parameter for system performance, the vertical radiation pattern of a cell sector is obtained according to [2], building an adjustable electrical tilt in the vertical plane, with the same structure as for the horizontal component, where the gain in the horizontal plane is given by

$$G_h(\phi) = \max\left(-12\left(\frac{\phi}{HPBW_h}\right)^2, SLL_h\right) \tag{3.1}$$

Here ϕ is the horizontal angle relative to the maximum gain direction, $HPBW_h$ is the half power beam-width for the horizontal plane, SLL_h is the side lobe level for the horizontal plane. Similarly, the gain in the vertical plane is given by,

$$G_v(\theta) = \max\left(-12\left(\frac{(\theta - \theta_{tilt})}{HPBW_v}\right), SLL_v\right) \tag{3.2}$$

All eNBs in the scenario have the same antenna model where, θ is the vertical angle relative to the maximum gain direction, θ_{tilt} is the tilt angle, $HPBW_v$ is the vertical half power beam-width and SLL_h is the vertical side lobe level. Finally, the two gain components are added by,

$$G(\phi, \theta) = \max\{G_h(\phi) + G_v(\theta), SLL_0\} + G_0 \tag{3.3}$$

where SLL_0 is an overall side lobe floor and G_0 the antenna gain.

3.2 AC in the context of COC

In this section, we design the AC algorithm to implement the automatic transmission power and antenna tilt adjustment for the COC. We define the state and action spaces and the reward function as follows:

- *State*: The state is defined based on the result of the scheduling scheme, which defines: (1) the allocation of users to RBs $(RB_1, RB_2, ..., RB_R)$ to the N users, (2) the values of CQI of each user in the corresponding RB.
- *Actions*: The set of eligible actions are:
  - The finite set of downlink transmission power levels, which can be allocated to the RBs assigned to the users. The selected values are: 0 to 46 dBm per RB with 0.5 dBm granularity.
  - The finite set of available tilt values, which can be assigned to the gain of the vertical plane of the antenna model. Those values range from 0 to 15° with $0.1°$ granularity.
- *Reward*: The reward function is defined based on the SINR of the users as follows:

$$r(s_t, a_t) = \begin{cases} 1\,, & \text{if } SINR \geq -6 \\ 0\,, & \text{otherwise} \end{cases}$$

  Where the threshold is set in order to support the lowest Modulation and Coding Scheme (MCS) [3].

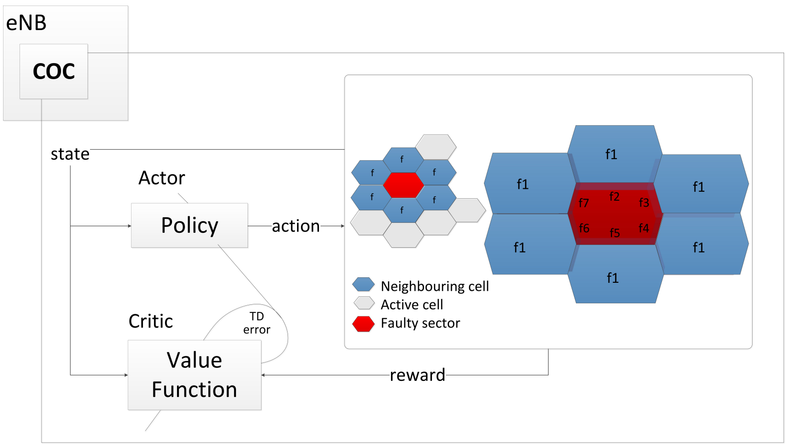

Figure 3.2: High level vision of the COC algorithm.

In order to mitigate the inter-cell interference that the adjustment of the eNB power levels may generate in the outage zone, in each neighbouring cell a certain amount of frequency band-width denoted by $\mathcal{F} = \{f_1, \ldots, f_c\}$ is reserved for users moving around the outage zone. The learning algorithm is executed every 1 msec, so every time that the scheduling is performed and users are allocated. The CQI is obtained from the spectral efficiency (η) provided in [4]. A high level vision of the COC algorithm is given in Figure 3.2.

3.3 Simulation platform

The proposed algorithm has been evaluated on the ns3 LENA platform based on 3GPP component LTE. In LTE systems each frame has a duration of 10 msec, divided into equally sized Transmission Time Interval (TTI), which have a duration of 1 msec. The granularity of the simulator for resource allocation is the Resource Block Group (RBG). The bandwidth is divided into 180 kHz physical RBs which are grouped in RBG of different size determined as a function of the transmission bandwidth configuration in use. In LENA ns3-model, the eNB physical layer handles the start and end of frames and subframes. Every TTI the UE physical layer sends to the eNB a feedback message with information related to the channel quality for each RB. In order to inform about the resource allocation, the scheduler is in charge of generating the Data Control Indication (DCI) structures, which are transmitted by the eNB to the connected UEs. Based on the tilt angle and/or the power levels selected by the algorithm per RB, the spectral efficiency per user u, $\eta_u = log2(1 + \frac{SINR_i}{\Gamma})$, is mapped onto the corresponding CQI value, using the mapping function reported in [4], where, Γ depends on the target Bit Error Rate (BER), $\Gamma = \frac{-ln(5BER)}{1.5}$. Once the UE computes these CQI values, the reward function described in Section 3.2 can be obtained. The scenario that we set up consists of 4 eNB sites, each one with three sectors, which results in 12 cells, as depicted in Figure 3.1. N Users are randomly distributed in the scenario, and after the related traffic session ends, the user appears in another location and starts a new session. The beginning and duration of sessions are based on Poisson scheme. The parameters used in the simulations, for both the cellular scenario and the learning algorithm, are given in Table 4.1.

Table 3.1: COC solution-Simulation parameters

Simulation parameters	Value
LTE	
No. Cells	12
No. UEs	100
Transmission Power	46 dBm
Path loss model	Friis spectrum propagation
Mobility model	pedestrian, speed 3 Km/h
Scheduler	FFR
Shadow Fading	Log-normal, std=8dB
AMC model	4-QAM, 16-QAM, 64 QAM
Cell layout	radius:500m
Bandwidth	5MHz
No. of RBs	25; RBs per RBG:2
Antenna parameters	
Horizontal angle ϕ	-180° $\leq \phi \leq$ 180°
HPBW	Vertical 10°: Horizontal 70°
Antenna gain G_0	18 dBi
Vertical angle θ	-90° $\leq \phi \leq$ 90°
SLL	Vertical -18 dB : Horizontal -20 dB
SLL_0	-30 dB
RL	
SINR threshold	-6 dB
Actions (power)	$0 - 46$ dBm per RB
Actions (tilt)	$0° - 15°$
τ, β, γ	$0.1; 0.5; 0.98$
Simulation time	10s

3.4 Results

The fault to self heal occurs when sector 4 fails to provide service to the associated users, generating a deep outage. We assume a cell outage detection function, which is out of the scope of this thesis, has already detected the problem and we focus on the COC solution. Here, the neighbouring cells are in charge of adjusting its power transmission in order to fill the coverage gap. We start analysing the behaviour of a particular user attached to the faulty sector, user 79. We observe that once the COC algorithm starts being operational, the user gets associated to one of the neighbour cells, to which we will refer in the following as the *compensating sector*. The CQI associated to the user in this moment is zero. As it was explained in Section 4.5, the scheduling scheme minimizes the interference, which can be generated among compensating sectors, by reserving a certain RBG to user 79. This information is shared with neighbour eNBs through the X2 interface. The performance to determine the capacity of the AC algorithm adjusting both Tx power and antenna tilt to which we refer in the following as AC(p+θ) is compared to the following approaches.

1 The Iterative Water-Filling (IWF) optimal solution formulated in [5], where taking into account any feasible power allocation, for each RBG, the algorithm calculates the optimal strategy of each eNB, assuming the power allocation of the other eNBs,

$$\max \sum_{r=1}^{R} log \left(1 + \frac{p_r^k h_{mm,r}}{\sum_{k=1,k\neq m} p_r^k h_{km,r} + \sigma^2} \right)$$

where,

$$p_r^m = \max \left(\frac{1}{\lambda^m} - \frac{\sum_{k=1,k\neq m} p_r^k h_{km,r} + \sigma^2}{h_{mm,r}}, 0 \right)$$

s.t.

$$\sum_{r=1}^{R} p_r^m \leq P_{max}^M \quad p_r^m \geq 0.$$

where λ is used in order to satisfy the power constraint, $h_{mm,r}$ indicates the link gain between eNB m and the user it serves in RB r, and $h_{km,r}$ is the link gain between eNB k and the user served by eNB m in RB r.

2 Fuzzy logic is proposed in [6] as the driving technique to fill a coverage gap. The authors vary both tilt angle and transmit power, with final values of tilt and flat transmission power of $11.8°$ and 31.60 dBm, respectively.

3 The AC algorithm where we compensate the outage by adjusting only the transmission power levels per RB, and which we indicate as AC(p).

4 The same AC algorithm where we compensate the outage by adjusting only the antenna tilt, which we indicate as AC(θ).

Figure 3.3, depicts the time evolution of the SINR of user 79 since when it is in outage, at the beginning of the simulation, till when it is recovered and correctly associated to the compensating sector. We assume the user is out of outage when its SINR is above the threshold of -6 dB, as explained in the previous section [3].

The behaviour followed by the AC(p+θ) scheme is that during the first part of the simulation it learns through interactions with the environment the proper policy to recover the user from outage. After approximately 500 msec the user's SINR is above the threshold and this result is

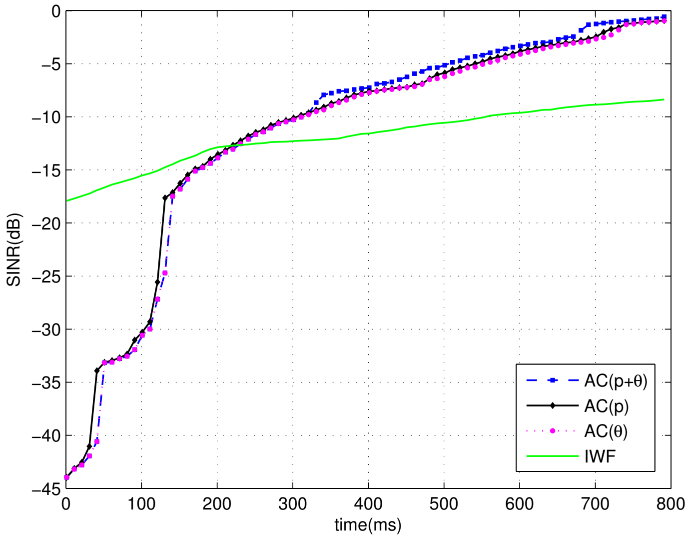

Figure 3.3: Time evolution of the average SINR of user 79, which is recovered from outage by the COC solution based on AC.

maintained during the rest of the simulation, as the correct policy has been learnt and it can be maintained independently of the variability of the proposed scenario. AC(p) and AC(θ) follow a similar behaviour taking just few msec more to reach the threshold. On the other hand, we observe that the IWF scheme never really recovers the user from the outage, as its driving principle is to maximize the total rate of the agent. In general once the AC algorithm has learnt the proper policy to make decisions, our learning approach provides better results in terms of coverage than the traditional IWF. Figure 3.4 describes the performance of SINR of users in the scenario.

We compare the CDF of the SINR of each user before and after COC actions. The NO COC implementation depicts the performance of each user when COC actions are not taken. Here, the number of users below the threshold is equal to 30%. When COC actions are taken it can be observed that the IWF is available to recover at least 10$\%$ of the scenario, similar results can be observed in case of fuzzy logic approach, while the AC approaches provide better results. The AC(p) fails to compensate 5% of users and a comparable performance is achieved by AC(θ). The proposed AC(p+θ) scheme, finally, offers the best results providing, on average, service to 98% of users.

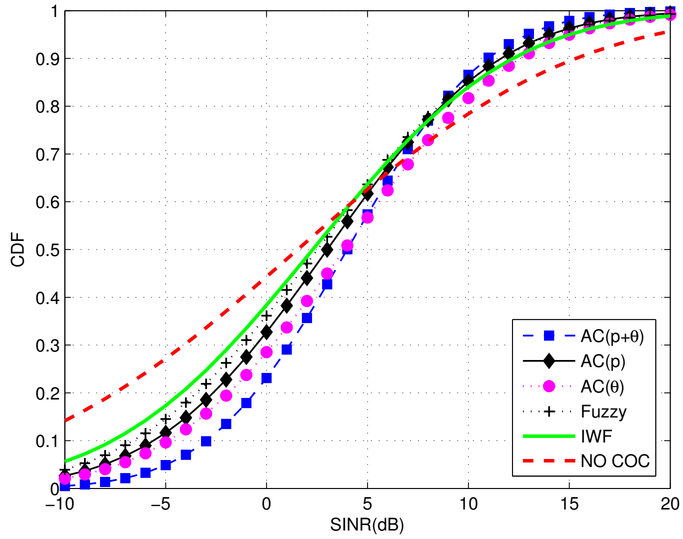

Figure 3.4: CDF of the SINR of all the users, which are recovered from outage due to different COC actions. i.e., finding the optimal tilt value and power transmission level per RB via the learning process.

3.5 Conclusion

We have presented in this chapter a self healing solution for COC, based on a reinforcement learning scheme. We assume a sector in the macro cellular scenario fails to provide service to its associated users, so that neighbour eNBs have to adjust their transmission power levels and electrical tilt in order to extend their coverage and fill the coverage and capacity gap generated by the fault. Due to the complexity of the proposed wireless cellular scenario, where UEs move around in random directions and at random speeds, and the channel is affected by fading and shadowing, as well as path loss, we propose a form of RL, a TD method referred to as AC, to provide a solution to our coverage problem. This kind of algorithm allows to learn from experience and from interactions with the surrounding environment an optimal policy to maximize the reward received by the eNBs over time. In order to avoid the inter-sector interference that can be generated by the surrounding eNBs extending their coverage, the AC algorithm is overlaid to a scheduling approach inspired to the FFR, which allocates a specific bandwidth to the users allocated in the outage zone. This reduces additional inter-cell interference in the outage zone and oscillations of the multi-agent learning scheme. Performance is evaluated on a ns3 LTE Release 10 platform and in comparison to the fuzzy logic approach and the traditional IWF transmission power allocation approach, which aims at maximizing the capacity of the node implementing it. Results demonstrate the ability of the proposed algorithm to compensate 98% of outage users and to provide them with service, outperforming the referenced benchmark in terms of SINR.

Bibliography

[1] O. Onireti, A. Zoha, J. Moysen, A. Imran, L. Giupponi, M. Ali Imran, A. Abu-Dayya. “A Cell Outage Management Framework for Dense Heterogeneous Networks , IEEE Transactions on Vehicular Technology”. *IEEE Transactions on Vehicular Technology* 64 (2015), pp. 2097–2113.

[2] Fredrik Gunnarsson, Martin N. Johansson, Anders Furuskar, Magnus Lundevall, Arne Simonsson, Claes Tidestav and Mats Blomgren. *VTC Fall’08* (2008), pp. 1–5.

[3] M. Amirijoo, L. Jorguseski, R. Litjens and R. Nascimento. “Effectiveness of Cell Outage Compensation in LTE Networks”. *The 8th Annual IEEE Consumer Communications and Networking Conference - Wireless Consumer Communication and Networking* (2011).

[4] Giuseppe Piro, N. Baldo, M.Miozzo. “An LTE module for the ns-3 network simulator”. *Wns3 in conjunction with SimuTOOLS* (2011).

[5] Gesualdo Scutari, Daniel Palomar, and Sergio Barbarossa. “Asynchronous Iterative Water-Filling for Gaussian Frequency-Selective Interference Channels”. *IEEE Transactions on Information Theory* 54.7 (July 2008), pp. 125–128.

[6] Saeed, A. and Aliu, O.G. and Imran, M.A. “Controlling Self Healing Cellular Networks using Fuzzy Logic”. *IEEE Wireless Communications and Networking Conference (WCNC)* (2012).

Chapter 4

SON conflict in D-SON architectures: a MDP framework

Life is 10% what happens to you and 90% how you react to it. **Charles R. Swindoll**

Contents

As we mentioned in chapter 2, self-organization as applied to cellular networks is a key driver for reducing the cost of installation and management by simplifying operational tasks through the capability to configure, optimise and heal itself. This trend, and in particular the self-healing function has been our motivation for the first chapter of this thesis. In this chapter, we focus on the 3GPP SON functions, where, in order to deal with the conflicts generated by the concurrent execution of multiple SON functions, we proposed a functional architecture and a theoretical framework based on the theory of MDPs for the self-coordination of different actions taken by different SON functions. In order to cope with the complexity of the overall SON problem, we subdivide the global MDP modelling the LTE eNB onto simpler sub-MDPs modelling the different SON functions. Each sub-problem is defined as a sub-MDP and solved independently by means of RL, and their individual policies are combined to obtain a global policy. This combined policy can execute several actions per state but can introduce policy conflicts.

As a particular case, in this chapter we focus on the coordination of two specific SON functions, the CCO and the ICIC already described in chapter 2, and we model them through two sub-MDPs, which are solved independently through RL. The two selected SON functions incur in the so called output parameter SON conflict [1] when e.g., the CCO function increases the transmission power levels to decrease the outage probability at the cell edge, while the ICIC decreases the power transmission levels to minimize interference. These two policies may generate a resource conflict as each one requires modifying the eNB transmission power in a way that may cancel the actions that the other one intends to take. We propose then a self-coordination approach modelled by means of a coordination game [2], where the players are the conflicting SON functions, the actions are the solutions to the specific sub-MDPs, i.e., the output of the Actor-Critic algorithm, and the rewards are those provided by the solution of the individual sub-MDPs. Coordination games have been proven to correctly model the coordination problems that arise when there is a conflicting interest, i.e., when two or more persons prefer different equilibrium outcomes. This is why we consider they are appropriate to model the problems that the self-coordination function has to face. The self-coordination framework aims then at founding a Nash equilibrium through the coordination game, maximizing the average reward.

We consider that focusing on only two functions does not result in a loss of generality to our approach, as also other SON functions, considering their automatic characteristic, can be solved through RL, e.g., [3] [4], so that our theoretical model can be extended to n SON functions. The generality of our solution for 3GPP SON architecture is proven later in the chapter. Performance evaluation is carried out in a ns3 LTE system simulator and it shows that our self-coordination approach provides satisfying solutions in terms of system performances for both the conflicting SON functions.

The outline of the chapter is organised as follows. Section 4.1 provides the details of the system model. Section 4.2 presents a functional architecture for the solution of SON conflicts. Section 4.3 defines the global SON problem modelling it through a MDP, and its decomposition through sub-MDPs modelling the different SON functions. Section 4.4 describes the CCO and

ICIC conflict case study. Section 4.5 describes details of the simulation platform and scenarios, as well as meaningful simulation results. Finally, section 4.6 concludes the chapter.

4.1 System model

We consider a heterogeneous wireless network composed of a set of $\mathcal{M}$ macro-cells that co-exist with $\mathcal{F}$ small cells. The $M{=}|\mathcal{M}|$ macro-cells form a regular hexagonal network layout with inter-site distance D, and provide coverage over the entire network, comprising both in-door and outdoor users. The $F{=}|\mathcal{F}|$ small cells are placed indoors within the macro-cellular coverage area following the 3GPP dual strip deployment model. Both macro and small cells operate in the same frequency band, which allows to increase the spectral efficiency per area through spatial frequency reuse.

An Orthogonal Frequency Division Multiple Access (OFDMA) downlink is considered, where the system bandwidth BW is divided into B RBs. A RB represents one basic time-frequency unit that occupies the bandwidth BW_{RB} over time T. In particular, in LTE systems each frame has a duration of 10ms, divided into equally sized TTI, which have a duration of 1ms. The bandwidth B is divided into $B_{\mathrm{RB}} = 180$ kHz physical RBs which are grouped in RBG of different size determined as a function of the transmission bandwidth configuration in use. Associated with each macro and small cell Base Station (BS) are U^{M} macro and U^{F} small cell users, respectively. The multi-user resource assignment that distributes the B RBs among the U^{M} macro and U^{F} femto-users, is carried out by a proportional fair scheduler.

We denote by $\mathbf{p}_t^n = (p_{1,t}^n, \ldots, p_{B,t}^n)$ the transmission power vector of BS n at time t, with $p_{r,t}^n$ denoting the downlink transmission power of RB r. The maximum transmission power for small and macro BSs are $P_{\max}^{\mathrm{F}}$ and $P_{\max}^{\mathrm{M}}$, with $P_{\max}^{\mathrm{F}} {\ll} P_{\max}^{\mathrm{M}}$, such that $\sum_{r=0}^{B} p_{r,t}^m \leq P_{\max}^{\mathrm{M}}$, $m{\in}\mathcal{M}$ and $\sum_{r=0}^{B} p_{r,t}^f \leq P_{\max}^{\mathrm{F}}$, $f{\in}\mathcal{F}$.

We analyse the system performance under different perspectives. First of all, we consider the SINR. Assuming perfect synchronization in time and frequency, the SINR of macro-user u^m who is allocated RB b of macro-cell $m{\in}\mathcal{M}$ amounts to:

$$\gamma_{b,t}^m = \frac{p_{b,t}^m h_{b,t}^{mu}}{\sum\limits_{n\in\mathcal{M}, n\neq m} p_{b,t}^n h_{b,t}^{nu} + \sum\limits_{f\in\mathcal{F}} p_{b,t}^f h_{b,t}^{fu} + \sigma^2} \tag{4.1}$$

where $h_{b,t}^{mu}$ accounts for the link gain between the transmitting macro BS m and its macro-user u^m; while $h_{b,t}^{nu}$ and $h_{b,t}^{fu}$ represent the link gain of the interference that BSs n and f imposes on macro-user u^m, respectively. Finally, σ^2 denotes the thermal noise power.

Likewise, the SINR of small cell user v^f who is allocated in RB b by small cell $f \in \mathcal{F}$ is in the form:

$$\gamma_{b,t}^{f} = \frac{p_{b,t}^{f} h_{b,t}^{fv}}{\sum_{m \in \mathcal{M}} p_{b,t}^{m} h_{b,t}^{mv} + \sum_{n \in \mathcal{F}, n \neq f} p_{b,t}^{n} h_{b,t}^{nv} + \sigma^2} \tag{4.2}$$

where $h_{b,t}^{nv}$ and $h_{b,t}^{mv}$ indicate the link gain between BSs n and m and small cell user v^f, respectively.

We also use as a meaningful indicator of quality perceived by users the CQI. This is computed based on the spectral efficiency per user, using the mapping function as indicated in [5], where the Block Error Rate (BLER) $BLER = 1 - \exp(log\frac{(1-BLER)}{TBS})$, should be smaller or equal to 10%, and the Transport Block Size (TBS) for the estimated CQI is calculated as reported in [5]. As a system metric, we also use the RSRQ from the serving cells, which is defined as the number of RBs multiplied by the RSRP over the system bandwidth BW multiplied by Reference Signal Strength Indication (RSSI). Finally, the throughput per user, is achieved by User Datagram Protocol (UDP) Client application.

4.2 Functional Architecture

In this section, we model the self-organised decision making process of the eNB, characterized by the multiple parallel SON functions, by means of a MDP. The problem, involving all the radio access autonomous functions, is so complex that cannot be handled by means of classical approaches.

In order to reduce the complexity of MDPs, the literature proposes three approaches: factorization [6], abstraction [7] and decomposition [8]. The idea behind the *factorization* approach is to address the complexity of the problem by identifying variables, which determine the state of the environment and the specific actions which have an effect on them, under certain conditions. In this framework, a state is implicitly described by an assignment to a set of state variables $\mathbf{x} = \{x_1, ..., x_n\}$, where the state at time t is now represented as a vector $\mathbf{x}_t = \{x_{1,t}, ..., x_{n,t}\}$, where $\mathbf{x}_{i,t}$ denotes the ith state variable at time t [9]. Furthermore, the rewards can often also be decomposed as a sum of rewards related to individual variables. The *abstraction* approach, in turn, creates an abstract model that aims at generating an equivalent simplified MDP by mapping a group of states, sharing a local behaviour, onto a single state. Finally, the *decomposition* approach subdivides the complex problem into smaller tasks. Each task is modelled by means of a sub-MDP and the value function and optimal policy for the MDP associated to each subtask are computed and then combined.

In this work, we rely on the so called decomposition approach [10], which subdivides the autonomous decision making process into multiple tasks represented by the individual SON functions. This results in a MDP organised onto multiple tasks which are theoretically modelled by different sub-MDP. Each sub-MDP is solved independently through AC describes in chapters 2 and 3, and the resulting policies are combined to obtain a global solution. This results in a MDP organised onto multiple tasks which are theoretically modelled by different sub-MDP. Each sub-MDP is solved independently and their policies are combined to obtain a global solution, such that the actions of each sub-MDP can be executed. This is illustrated in Figure 4.1.

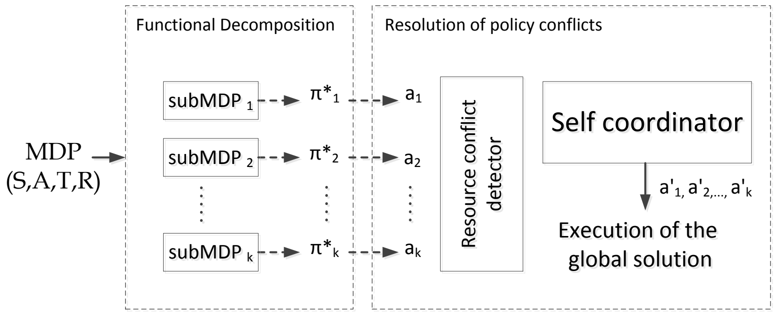

Figure 4.1: General Architecture.

If the tasks are independent, the policies can be executed without incurring into conflicts. However, if it is not the case, and the selected actions for each task are executed concurrently and not serially, conflicts among local policies may arise, which may result in undesirable behaviours. In order to solve SON conflicts we propose a functional architecture based on two main functions:

- *Functional Decomposition*: It is the function in charge of breaking the complex problem into tasks. The complex MDP is subdivided into k sub-MDPs, which are solved locally. k Optimal policies $\pi_1^*, \ldots, \pi_k^*$ are obtained so that at each time step t actions $a_1, \ldots, a_k$ can be selected.

- *Resolution of Policy Conflicts*: It is the function in charge of detecting and solving potentially conflicting policies, which are to be executed concurrently. The *resource conflict detector* entity, represented in Figure 4.1, evaluates whether two or more of the actions $a_1, \ldots, a_k$ aim at modifying the same parameter. In this case, the conflict is detected and the *self-coordinator* entity is activated to solve the conflict, the result is the execution of the global solution $a_1', \ldots, a_k'$.

The next sections describe these two functions with further details.

4.2.1 Functional Decomposition

This module is responsible for breaking the global MDP $\{\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}\}$ into k tasks. Each task is characterized by a specific objective and is modelled by a sub-MDP, which is solved independently to find a policy π that maps states to actions to maximize the expected reward. Each sub-MDP i is characterized by its own state, action set, transition probability and reward functions and is denoted by $sub - MDP_i = \{\mathcal{S}_i, \mathcal{A}_i, \mathcal{T}_i, \mathcal{R}_i\}$.

The set $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$ of states of the global problem is modelled by means of a set of m variables of possible states of the environment $\mathbf{x} = \{x_1, x_2, \ldots, x_m\}$. For each sub-MDP we consider a decomposed representation, so that the sub-MDP state space is modelled by a set of v state variables: $\mathbf{x}_i = \{x_1, x_2, \ldots, x_v\}$, where $v < m$. The state space of the global problem is fully modelled when the k sub-MDPs include all the m variables, i.e., $\mathbf{x} = \cup_{i=1}^{k} \mathbf{x}_i$. If the global problem is not modelled in decomposed representation, the union of all space states of the sub-MDPs must be equal to the state space of the global problem, i.e., $\mathcal{S} = \{\mathcal{S}_1 \cup \mathcal{S}_2 \cup \ldots \cup \mathcal{S}_k\}$. Each sub-MDP is solved independently to obtain the value function V_i^* for any π_i^*. The global problem is then defined by k sub-MDPs, $sub - MDP_1, sub - MDP_2, ..., sub - MDP_k$, such that,

- The global state space $\mathcal{S}$ is modelled in decomposed form, in such a way that the total state variable $\mathbf{x}$ is the union of the k sets of state variables $\mathbf{x} = \mathbf{x}_1 \cup \mathbf{x}_2 \cup ... \cup \mathbf{x}_k$.
- The action space $\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_k\}$.
- The transition function $\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_k\}$
- The reward function $\mathcal{R} = \{R_1 + R_2 + \ldots + R_k\}$

4.2.2 Resolution of policy Conflicts

In this section we deal with the conflicts, which may arise from the combination of local solutions of individual sub-MDPs. We focus on the parameter conflict generated by different sub-MDPs, which occurs when two or more sub-MDPs may request different values for the same parameter. If there are no conflicts, the set of actions to be selected in a generic state s, $\mathbb{A} = \{\pi_1^*(s) = a_1, \pi_2^*(s) = a_2, ..., \pi_k^*(s) = a_k\}$, is executed simultaneously and the solution is optimal. Otherwise, the sub-MDPs with conflicts are detected and solved. In the following we propose a solution based on the theory of coordination games, which have already been proven in literature good to solve coordination problems which arise when there is a conflicting interest. The classic example of application is the "battle of the sexes" game, where the man prefers to attend a baseball game and the woman prefers to attend an opera, but both would rather do something together than go to separate events. The question is, if they cannot communicate, where they would go. We consider that this game perfectly fits our coordination game in case of conflicting interests between two SON functions. This game has two pure strategy Nash equilibria, one where both go to the opera and another where both go to the football game. There is also a mixed strategies Nash equilibrium, where the players go to their preferred event more often than the other.

For the sake of simplicity, we model the conflict between two sub-MDPs, $sub-MDP_1$ and $sub-MDP_2$, by means of a two player coordination game. However, the conflict between n sub-MDPs is scalable to a n-player coordination game.

- $\mathbf{S}$ is the set of possible states of the environment, which is the same as the one defined for the global MDP $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$.
- $\mathbf{A} = \{\alpha, \beta\}$ is the action set, where α and β are the actions selected by $sub-MDP_1$ and $sub-MDP_2$, respectively, and they belong to the corresponding action sets, i.e., $\alpha \in \{a_{sub-MDP_{11}}, ..., a_{sub-MDP_{1q1}}\}$ and $\beta \in \{a_{sub-MDP_{21}}, ..., a_{sub-MDP_{2q2}}\}$.
- The reward matrix associated with each state at time t is denoted by $\mathbf{R}$.

The self-coordinator module receives as input the actions selected by each sub-MDP, and based on that, the possible situations to face are the following:

1 The sub-MDPs choose the same action ($\alpha = \beta$).

2 The sub-MDPs choose different actions, with different rewards, i.e., $\alpha \neq \beta$ with $r_\alpha \neq r_\beta$.

3 The sub-MDPs choose different actions, but with the same reward, i.e., $\alpha \neq \beta$ with $r_\alpha = r_\beta$.

If the actions are the same, the coordinator just executes the action, otherwise the conflict is solved by mixed strategies through the reward matrix depicted inside the coordination box in Figure 4.2, where A, a, are the rewards of $sub-MDP_1$ and $sub-MDP_2$, respectively, when executing for both sub-MDP α; B, b, are the rewards of $sub-MDP_1$ and $sub-MDP_2$, respectively when executing action α for $sub-MDP_2$ and action β for $sub-MDP_1$; C, c, are the rewards of $sub-MDP_1$ and $sub-MDP_2$, respectively, when executing action α for $sub-MDP_1$ and action β for $sub-MDP_2$; D, d, are the rewards of $sub-MDP_1$ and $sub-MDP_2$,

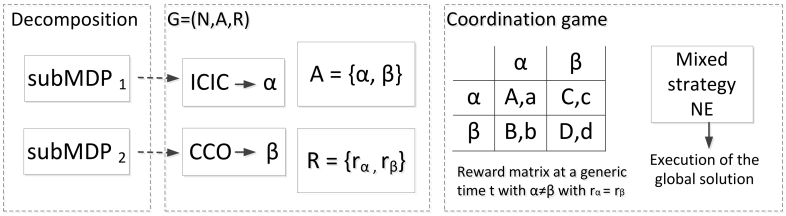

Figure 4.2: Self Coordination based on Coordination game.

respectively, when executing for both sub-MDPs action β. This game has mixed strategy Nash Equilibria given by probabilities $p = (d - b)/(a + d - b - c)$ to play α and $1 - p$ to play β, for player 1(rows) and $q = (D - C)/(A + D - B - C)$ to play α and $1 - q$ to play β, for player 2 (columns). Hence, each player is not actually choosing α, β directly, but choosing a probability with which a player will play α. A given number p means that player 1 will play α with probability p and β with probability $1 - p$. Similar considerations can be done for player 2. Since $d > b$ and $d - b < a + d - b - v$, p and q are always between 0 and 1, so the existence is assured.

The algorithm for solving resource conflict is described in Algorithm 4.1.

Algorithm 4.1. Coordination game $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R})$

Let $sub - MDP_1$= player 1
Let $sub - MDP_2$= player 2
$\mathcal{A} = \{\alpha, \beta\}$
if $\alpha = \beta$ **then**
  compute and execute $a' = \alpha = \beta$
end if
if $\alpha \neq \beta$ and $r_\alpha > r_\beta$ **then**
  compute and execute $a' = \alpha$
else
  compute and execute $a' = \beta$
end if
if $\alpha \neq \beta$ and $r_\alpha = r_\beta$ **then**
  if $A > B, D > C$ and $a > c, d > b$ **then**
    compute and execute
    Function mixedStrategies **CG** $(\mathcal{A}, R_t, p, q)$

    return a'
    end Function
  end if
end if

return $\{a'\}$

4.3 Functional decomposition of the general 3GPP self-optimisation use case

In this section we provide an example about how the proposed functional architecture can be used to deal with the conflicts generated by the concurrent execution of multiple SON functions. The objective of this section is to show that the proposed approach is general enough to model all the SON functions, and their derived conflicts. For this purpose, we focus on the self-optimisation functionality and on all the associated SON functions, as defined in [1] and [5], i.e., MLB, MRO, CCO, ICIC, COC, ES, and RACH optimisation. We first introduce these SON functions in the context of the general SON architecture, together with high level examples of how they may interfere. Then, we define the state and action spaces of the global MDP that models the self-optimisation procedure of the overall RAN segment. Finally, we show that the global self-optimisation problem can be decomposed onto as many sub-MDPs as SON functions. We define the different sub-MDPs, with state and action spaces and tentative proposals for reward functions. References in literature will show that these self-optimisation problems can be solved using reinforcement learning functionalities. We consider that this demonstrates the generality of our approach to solve conflicts in 3GPP SON architectures.

4.3.1 Overview of 3GPP self-optimisation functions

In the following we quickly described the main self-optimisation functions. We describe the main information these functions rely on and the main parameters the aim to tune.

The MLB is a SON function where cells with congestion can transfer load to other cells. The main objective is to improve end-user experience and achieve higher system capacity by distributing user traffic across system radio resources. The implementation of this function is generally distributed and supported by the load estimation and resource status exchange procedure. The messages containing useful information for this SON function (resource status request, response, failure and update) are transmitted over the X2 interface [11]. MLB can be implemented by tuning the CIO parameter. The CIO contains the offsets of the serving and the neighbour cells that all UEs in this cell must apply in order to satisfy the A3 hand-over condition [5].

The MRO is a SON function designed to guarantee proper mobility, i.e. proper hand-over in connected mode and cell re-selection in idle mode. Among the specific goals of this function we have the minimization of call drops, the reduction of RLFs, the minimization of unnecessary hand-overs, ping pongs, due to poor hand-over parameters settings, the minimization of idle problems. Its implementation is commonly distributed. The messages containing useful information are e.g., the S1AP hand-over request or X2AP hand-over request, the hand-over report, the RLF indication/report. MRO operates over connected mode and idle mode parameters. In connected mode, it tunes meaningful hand-over trigger parameters, such as the event A3 offset (when referring to intra-RAT, intra-carrier hand-overs), the TTT, or the Layer 1 and Layer 3 filter coefficients. In idle mode, it tunes the offset values, such as the Qoffset for the intra-RAT, intra-carrier case. CCO is a SON function, which aims to provide capacity and coverage optimisation. The targets that can be optimised may be vendor dependent and include coverage, cell throughout, edge cell throughput, or a weighted combination of the above.

CCO reacts to changes in the environment depending on diverse origins: seasonal changes, changes in the surrounding infrastructures, changes in the network planning, daily variations of traffic, etc. It can be implemented in both centralized (in the network manager or element manager) and distributed architectures. Useful information is generally extracted from UE measurements. Parameters that may be tuned are the transmission power, the pilot power and antenna parameters (azimuth and tilt).

ICIC is a SON function, which aims to minimize interference among cells using the same spectrum. It involves the coordination of physical resources between neighbouring cells to reduce interference from one cell to another. ICIC can be done in both uplink and downlink for the data channels PDSCH, and PUSCH, or uplink control channel PDCCH. ICIC can be static, semi-static or dynamic. Dynamic ICIC relies on frequent adjustments of parameters, supported by signalling among cells over X2 interface. To support proactive coordination among cells the HII and the RNTP indicators have been defined, while to support reactive coordination, the OI has been introduced [11]. Parameters that may be tuned are the transmission power, the pilot power, antenna parameters (azimuth and tilt), and the support of coordinated Almost Blank Subframes (ABS).

COC is applied to alleviate the outage caused by the loss of a cell from service. For this use case an adequate reaction is vital for the continuity of the service, so vendor specific COD schemes have to be designed. Parameters to tune, to try to compensate the outages are the transmission power and antenna parameters of the cells neighbouring the fault.

ES aims at providing the quality of experience to end users with minimal impact on the environment; the objective is to optimise the energy consumption, by designing NEs with lower power consumption and temporarily shutting down unused capacity when not needed. The most common action is to switch on/off the appropriate cells.

RACH optimisation aims at optimising the random access channels in the cells based on UE feedback and knowledge of its neighbouring eNBs RACH configuration. RACH optimisation can be done by adjusting the Pc parameter or change the preamble format to reach the set target access delay.

The independent execution of these individual SON functions affects parameters or performances that can end up in conflict. For example, the ICIC may decide to reduce the transmission power to reduce inter-cell interference, while the CCO may decide on increasing it, to improve coverage. These conflicting actions affect the borders of the cell, and consequently the performances of the MLB function of the same cell and its neighbours. To compensate for the actions taken by ICIC and CCO, the MLB may decide to modify some hand-over parameters, which then have impact on the hand-over and MRO performances, etc.

4.3.2 Definition of the global MDP and of the decomposed sub-MDPs

We consider a heterogeneous wireless network composed by M 3GPP (H)eNBs, as defined in section 4.1. Each eNB has to be capable of executing the N standardized SON functions, where $N = \{N_{CCO}, N_{ICIC}, N_{COC}, N_{MLB}, N_{MRO}, N_{ES}, N_{RACH}\}$, and where $N_{CCO}, N_{ICIC}, N_{COC}, N_{MLB}, N_{MRO}, N_{ES}, N_{RACH}$ are, respectively, the number of CCO, COC, ICIC, MLB, MRO, ES, RACH instances across the M eNBs and the Network Management (NM)/Ensemble Methods (EM). We can model the global self-optimisation problem defined by the N SON functions through a MDP, as it consists of a multi-objective, multi-parameter decision making/optimisation process where the outcomes are partly random and partly under the control of the decision maker. The global problem is then defined by,

- *State*. The state space $\mathcal{S}$ is defined by a set of state variables X defined, among others, by: (1) the allocation of users to RBs, (2) the values of CQI, (3) UE measurements in terms of RSRP, the values of RSRQ, (4) resource status information,(5) hand-over and RLF statistics and information, (6) interference coordination information in terms of HII, OI, RNTP.

- *Actions*. The action set $\mathcal{A}$ consists of all the possible actions that can be taken by tuning, among others, the following parameters: (1) transmission power, (2) pilot power, (3) antenna parameters, in terms of tilt and azimuth, (4) CIO, (5) hand-over parameters in terms of event offsets, TTT, Qoffset, etc.

- *Reward*. The reward $\mathcal{R}$ is defined based on the following rationale. If the combination of the selected actions gives e.g., an intercell interference below a threshold or an outage probability above a threshold, or RLF statics above a threshold or throughput performances below objectives, or pilot pollution above threshold, etc., the reward is negative, otherwise the reward is weighted function of multiple objectives, such as the network throughput and the users fairness.

The global MDP including the N SON functions and their related instances, is extremely complex. To solve it, we should rely on multi-objective optimisation frameworks, which do not provide real-time solutions [12]. As a result, the global SON problem is subdivided into multiple tasks represented by the simpler $N = 7$ SON functions described before. Other functional decomposition approaches may be possible but they would not be aligned with the 3GPP SON architecture, and consequently they are not interesting for our problem. Here each $sub-MDP_i$ is characterized by its own state, action set, transition probability and reward functions, and is denoted by $sub - MDP_i = \{\mathcal{S}_i, \mathcal{A}_i, \mathcal{T}_i, \mathcal{R}_i\}$.

We define the state and action spaces, together with one of the possible reward functions of each sub-MDP as follows:

1. CCO

   - *State*: The state $\mathcal{S}_1$ is defined based on the result of the scheduling scheme, which defines (1) the allocation of users to RBs, (2) the values of CQI of each user in the corresponding RB, (3) UE measurements (e.g., RSRP, RSRQ, etc.)
   - *Actions*: The action set $\mathcal{A}_1$ is based on (1) the set of eligible actions are a finite set of downlink transmission power levels, which can be allocated to the RBs assigned to the users, (2) the finite set of available tilt, and azimuth values, which can be assigned to the gain of the vertical plane of the antenna model.
   - *Reward*: If the CQI is greater or equal than 1, the reward will be positive, otherwise will be negative. The threshold is set in order to support the SINR values for Multiple-input Multiple-output (MIMO) transmissions.

   The sub-MDP representing this SON function can be solved through reinforcement learning, as for example has been done before in [13].

2. ICIC

- *State*: The state $\mathcal{S}_2$ is defined based on the result of the scheduling scheme, which defines (1) the allocation of users to RBs, (2) the values of SINR of each user in the corresponding RB, (3) UE measurements (e.g., RSRP, RSRQ, etc.)
- *Actions*: The set of eligible actions $\mathcal{A}_2$ are defined based on (1) the finite set of downlink transmission power levels, which can be allocated to the RBs assigned to the users, (2) the finite set of available tilt and azimuth values, which can be assigned to the gain of the vertical plane of the antenna model.
- *Reward*: If the SINR is greater or equal than 0 dB, the reward will be positive, otherwise will be negative. The threshold is set in order to support the SINR values for MIMO transmissions.

Also the sub-MDP modelling this SON function can be solved through reinforcement learning as it is done for example in [14].

3. COC

- *State*: The state $\mathcal{S}_3$ is defined based on the result of the scheduling scheme, which defines (1) the allocation of users to RBs, (2) UE measurements (e.g., RSRP, RSRQ, etc.)
- *Actions*: The set of eligible actions $\mathcal{A}_3$ are defined based on (1) the finite set of downlink transmission power levels, which can be allocated to the RBs assigned to the users, (2) the finite set of available tilt, and azimuth values, which can be assigned to the gain of the vertical plane of the antenna model.
- *Reward*: If the SINR is greater or equal than 6 dBs, the reward is positive, otherwise is negative. The threshold is set in order to support the lowest MCS.

A complete solution for COC by adjusting the gain of the antenna due to the electrical tilt and the downlink transmission power of the surrounding eNBs can be founded in [15], and this solution is based on reinforcement learning tools.

4. ES

- *State*: The state $\mathcal{S}_4$ is defined based on (1) the resource status information of the cell and its neighbours, (2) the expected demand of traffic, (3) the energy available to the network element.
- *Actions*: The action set $\mathcal{A}_4$ consists of switching on/off the cell.

- *Reward*: If the resource usage and the energy available for consumption (in case we are in an energy constrained system) are below a certain threshold, the reward is negative, otherwise it is positive.

The sub-MDP modelling this use case can also be solved through reinforcement learning, as it is demonstrated in [16].

5. MLB

- *State*: The state $\mathcal{S}_5$ is defined based on the resource status information of each cell, and of that of its neighbours.
- *Actions*: The action set $\mathcal{A}_5$ is defined based on the update of the CIO value, by a finite set of fixed values.
- *Reward*: If the change of CIO removes overload with minimal negative Handover (HO) effects the reward will be positive, otherwise will be negative.

A solution for the sub-MDP modelling this function can also be based on reinforcement learning tools [4].

6. MRO

- *State*: The state $\mathcal{S}_6$ is defined based on hand-over and RLF reports and statistics.
- *Actions*: The action set $\mathcal{A}_6$ is defined based on the update of the cell's hand-over event offsets, TTT and layer1/layer 3 filter coefficients.
- *Reward*: If the users affected by the RLF in the cell, after the HO parameter setting is completed, are lower than a threshold, the reward will be positive, otherwise will be negative.

A solution for the sub-MDP modelling this function can also be based on reinforcement learning tools [4].

7. RACH optimization

- *State*: The state $\mathcal{S}_7$ is defined based on the resource status information.
- *Actions*: The action set $\mathcal{A}_7$ is defined based on the update of the power control parameter or of the preamble format to reach the set target access delay.
- *Reward*: If the users achieve lower data rate than the agreed Guaranteed Bit Rate (GBR), the reward will be negative, otherwise, will be positive.

This can also be solved through reinforcement learning strategies, as shown in [17].

If n polices are in conflict the coordination is handled through a n-player coordination game, as discuss in section 4.2.2.

4.4 Case study: Self-coordination for ICIC and CCO function conflict

Among the different SON functions defined by 3GPP, we focus our attention on CCO, and ICIC. The CCO is in charge of optimizing the capacity and coverage of the area of influence of the particular eNB. As a result, it aims to decrease the outage probability at the border of the cell. The ICIC is in charge of minimizing the interference among different cells. We will focus on the conflict generated by ICIC and CCO SON functions, as both of them aim at modifying the transmission power. In particular, while the CCO may decide to increase the power e.g., to improve the coverage or the capacity, the ICIC may decide to decrease the power to reduce the interference. Figure 4.3 shows the actions taken by the two different SON functions, in the D-SON architecture implemented in a heterogeneous scenario. The impact of the concurrent execution of two conflicting actions is highlighted for cell boundaries of one cell. In this section we describe how to solve the independent sub-MDPs characterizing the ICIC and CCO functions through TD-learning, and the Actor Critic algorithm in particular. Once each SON function has found the optimal policy by means of the AC algorithm, if the actions are different, ICIC and CCO play a game with conflicting interests. We define a two-player game $\mathbf{G} = \{\mathbf{N}, \mathbf{A}, \mathbf{R}\}$, where the $\mathbf{N} = 2$ players are the SON functions, $\mathbf{A} = \{\alpha, \beta\}$ is the action set consisting of the actions selected by CCO, $\alpha = \{p_{CCO_1}, ..., p_{CCO_R}\}$, and ICIC, $\beta = \{p_{ICIC_1}, ..., p_{ICIC_R}\}$. The reward matrix associated with each state is denoted by $\mathbf{R}$, represented in Figure 4.2, inside the coordination game box. Here, the rows correspond to ICIC and the columns to CCO.

4.4.1 AC-based solution for CCO function

The CCO SON function aims to provide capacity and coverage optimization. In this scenario, the uncovered planned cell area is the coverage holes that need to be optimised by the coverage and capacity optimization [18]. We measure this by decreasing the outage probability. As an indicator, we consider the CQI, which is a measurement of the communication quality of wireless channels. We define the state and action spaces and the reward function as follows.

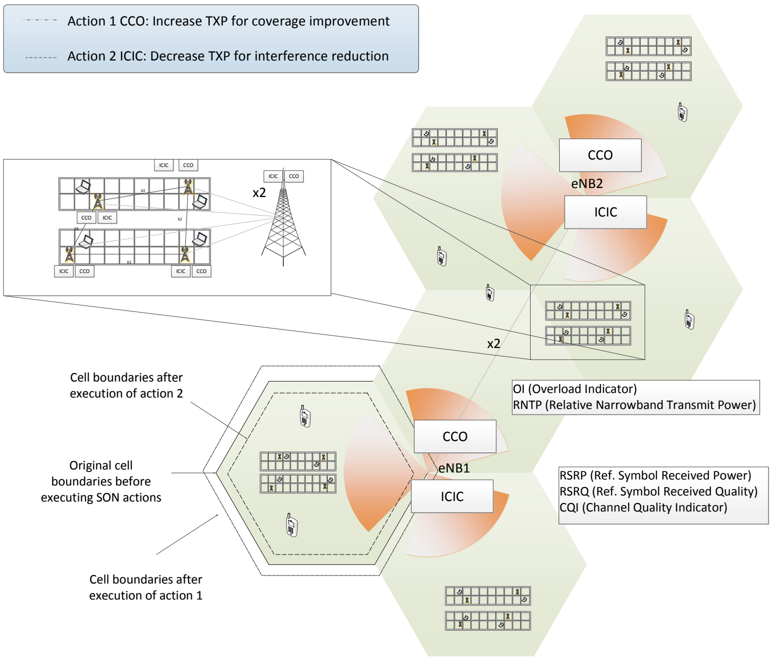

Figure 4.3: Impact of ICIC, and CCO parameter conflict

- *State*: The state is defined based on the result of the scheduling scheme, which defines: (1) the allocation of users to RBs $(RB_1, RB_2, ..., RB_R)$ to the N users, (2) the values of CQI of each user in the corresponding RB.

- *Actions*: The set of eligible actions are the finite set of downlink transmission power levels, which can be allocated to the RBs assigned to the users. The selected values are: 0 to 46 dBm per RB with 0.5 dBm granularity.

- *Reward*:

$$r(s_t, a_t) = \begin{cases} 1\,, & \text{if } CQI \geq 1 \\ 0\,, & \text{otherwise} \end{cases}$$

  The threshold is set based on the CQI. The reason behind this value, is that one of the possible causes of bad BLER is bad coverage. This one should be smaller or equal than 10%, which is the requirement from the LTE standard. As for the particular CQI values associated to the modulation schemes and channel coding rates, we refer to [5].

4.4.2 AC-based solution for ICIC function

The ICIC SON function aims to minimize interference among cells using the same spectrum. In this scenario [18], we define the state and action spaces and the reward function as follows.

- *State*: The state is defined based on the result of the scheduling scheme, which defines:

(1) the allocation of users to RBs $(RB_1, RB_2, ..., RB_R)$ to the N users, (2) the values of SINR measured for each user in the corresponding RB.

- *Actions*: The set of eligible actions are the finite set of downlink transmission power levels, which can be allocated to the RBs assigned to the users. The selected values are: 0 to 46 dBm per RB with 0.5 dBm granularity.

- *Reward*:

$$r(s_t, a_t) = \begin{cases} 1\,, & \text{if } SINR \geq 0 \\ 0\,, & \text{otherwise} \end{cases}$$

  Where the threshold is set in order to support the SINR values for multi user MIMO transmission mode [5]. MIMO can be used to increase the SINR, i.e, the capacity increases logarithmically with the SINR.

4.5 Simulation platform, scenario and results

The proposed algorithms have been evaluated on the ns3 LENA platform based on LTE Release 8 [19]. The Self Coordinator framework given in Figure 4.2 is executed every time a new action is selected by one of the functions, i.e., every time a CQI or the UE measurements are reported to the (H)eNB. This happens, in periodic reporting, every 2-160 msec for the CQI, and every $120 - 60$ msec for the UE measurements [20]. The parameters used in the simulations, for CCO and ICIC are given in Table 4.1.

The scenario that we set up consists of 2 eNBs, each one with three sectors, which results in 6 cells and 38 UEs. The small cell network is based on the dual stripe scenario with 1 block of 2 buildings. Each building has one floor, with 20 apartments, which results in 40 apartments per block, as depicted in Figure 4.4. The number of blocks is equal to 5. The Home eNodeB (HeNB) activation factor is 0.5 and the deployment ratio is 0.2, which results in 20 HeNBs, each one located in an independent apartment. Each HeNB provides service to one user in the scenario, which results in 20 HeNB users.

In the scenario, users are randomly distributed, and after the related IP traffic session ends, the UE appears in another location and starts a new session. In addition, in order to test the QoS performance we use a UDP Client application, which takes care of the generation of Radio Link Control (RLC) Protocol Data Units (PDUs) allowing multiple flows belonging to different QoS classes.

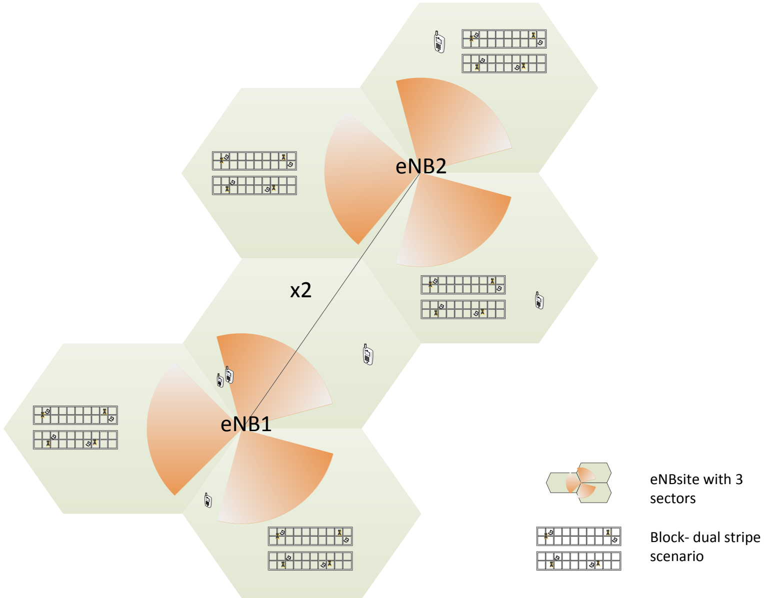

Figure 4.4: Scenario-SON conflict

4.5.1 Results

We first show in Figure 4.5 the time evolution of the average SINR reported by UEs when the two SON functions are executed independently, and for different values of the CQI and UE measurements reporting periodicity (i.e., CQI feedback every $2/120/160$ msec and RSRQ feedback every 120 msec). We observe that both the AC algorithms implementing the two SON functions - even if the proposed scenario is characterized by the dynamism typical of realistic wireless networks, as the UEs move around, the HeNBs are characterized by random activation factors, the channel model include shadow and fading effects, etc. - after a first training phase, converge in a stable manner to a situation where no user is in outage. We observe as well that the time of convergence depends on the periodicity of feedback to the (H)eNBs from the users.

On the other hand, in Figure 4.6, we compare performances in terms of time evolution of the SINR provided by the proposed coordination game framework and by the approach that is suggested by 3GPP in [21]. Here, it is proposed that every time that a SON function is willing to modify a transmission parameter, asks for permission to the self-coordination entity, which handles the queues of requests for parameter modifications. We consider then an implementation of ICIC and CCO characterized by similar reporting periodicity, i.e., 120 msec for each SON function. We observe in Figure 4.6 that while the self-coordination framework based on the proposed coordination game, actually selects the most appropriate action to execute based on a compromise between conflicting interests, the 3GPP proposed approach handles

Table 4.1: SON conflict-Simulation parameters

Simulation parameters	Value
Parameter	
Path loss model	Friis spectrum propagation
Mobility model	Pedestrian, speed 3 Km/h
Shadow fading	Log-normal, std = 8 dB
Scheduler	Proportional Fair Scheduler (PF)
AMC model	LteAmc::MiErrorModel
Transport protocol	UDP
Macro cell scenario	
Number of cells	6
Number of UEs	38
eNB Tx power	46 dBm
Small cell scenario	
Number of cells	20
HeNBs per block	4
Number of home UEs	20
HeNB Tx power	23 dBm
LTE	
Cell layout	Radius: 500 m
Bandwidth	5 MHz
Number of RBs	25; RBs per RBG: 2
RL	
Actions (power)	0 to 46 dBm per RB
$\tau; \beta; \gamma$	$0.1; 0.5; 0.98$
ICIC threshold	SINR $>$ 0 dB
CCO threshold	CQI $\geq$ 1
Simulation time	10 s

the conflict by properly scheduling in time the different actions, but due to the execution of both actions, it generates unnecessary oscillations and poorer performances in terms of average achieved SINR for the UEs in the scenario.

We now further analyse the results provided by the independent implementation of ICIC and CCO, in comparison to the results obtained when the self-coordination framework is active. Figure 4.7 represents the CDF of the SINR of the UEs, at the end of simulation time. We observe that when performing the SON functions independently, the CCO offers better performances than the ICIC, for low values of SINR, i.e., at the border of the cell, as it aims at optimizing the capacity and coverage features of the scenario. On the other hand, the ICIC function performs better than the CCO for higher values of SINR, as it aims at minimizing the

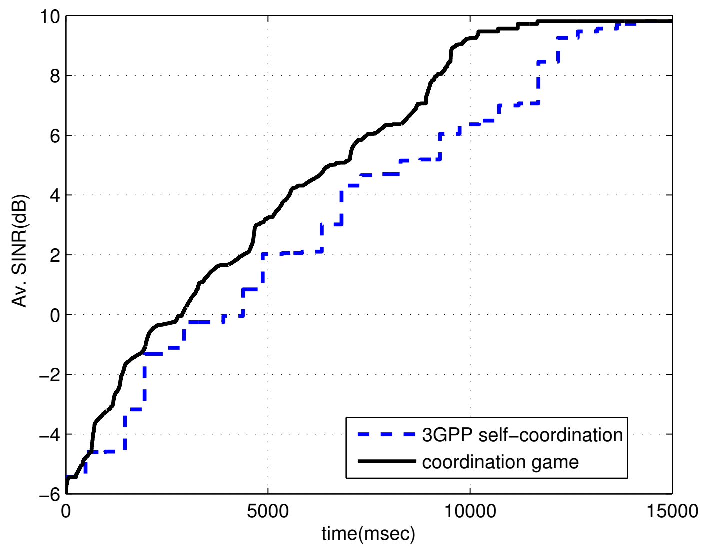

Figure 4.5: Time evolution of the average SINR of UEs in the scenario, considering different values of reporting periodicity for the CQI and the UE measurements.

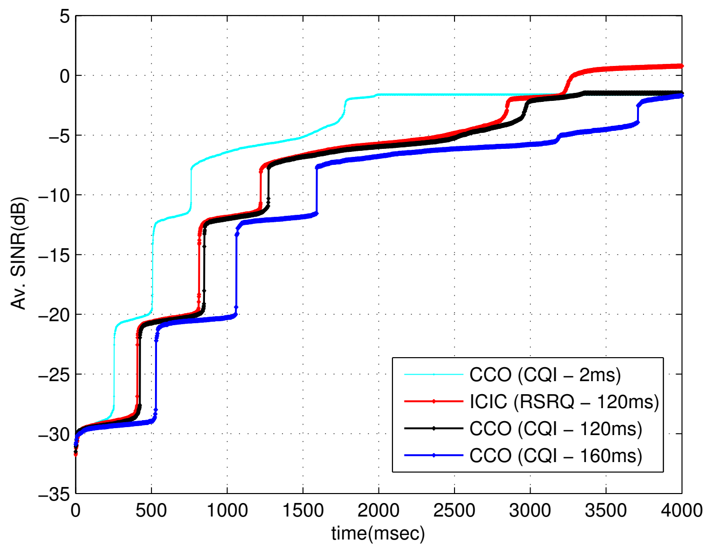

Figure 4.6: Time evolution of the average SINR of UEs in the scenario, for the self-coordination frameworks based on Coordination game and 3GPP approaches.

effect of inter-cell interference in the whole scenario, thus improving interference performances for all users. When executing the two SON functions in parallel, conflicts may arise, so we need the support of a self-coordination function. When implementing it, we achieve a compromise between the conflicting objectives of the two SON functions. On the one hand, at the cell edge, we are reducing the outage probability with respect to the results obtained otherwise with ICIC, while we are maintaining the outage with respect to results obtained by CCO. On the other hand, inside the cell, the self-coordination framework obtains better performances in terms of outage, compared to previous results of the CCO, while it increases the outage compared to ICIC independent results. The reason behind this behaviour is to found in the compromise achieved by means of the mixed strategy equilibrium, which consists in an equilibrium where there is a percentage of time during which ICIC gets less reward than CCO, and the rest of the time when the CCO achieves a higher reward than ICIC.

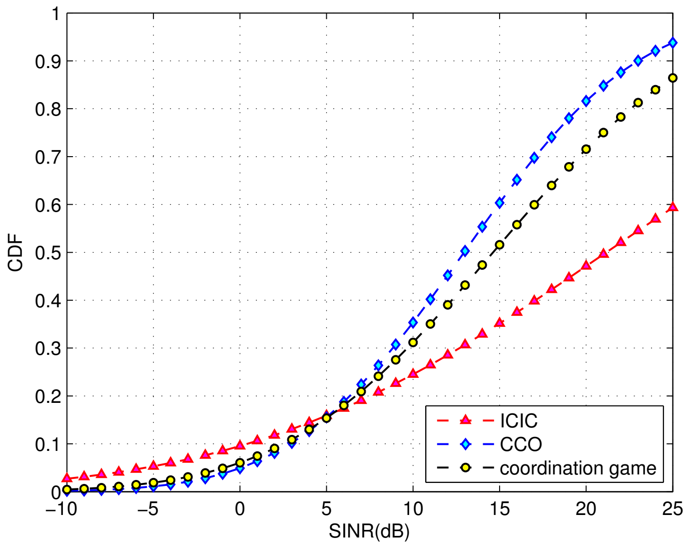

Figure 4.7: CDF of the SINR of UEs in the scenario.

Figure 4.8 shows the RSRQ from the serving cell, which is one of the UE measurements periodically reported by the same UE indicating its performance rate. We observe a similar behaviour as discussed for Figure 4.7. However, here the self-coordinator performs more similarly to ICIC inside the cell, and more similarly to CCO at the cell edge, thus managing to get the best out of each SON function. Finally, the same desirable behaviour is also confirmed in Figure 4.9, which depicts the CDF of each UE average throughput. The traffic considered uses a RLC Saturation Mode (SM), which takes care of the generation of RLC PDUs allowing multiple flows belonging to different QoS classes.

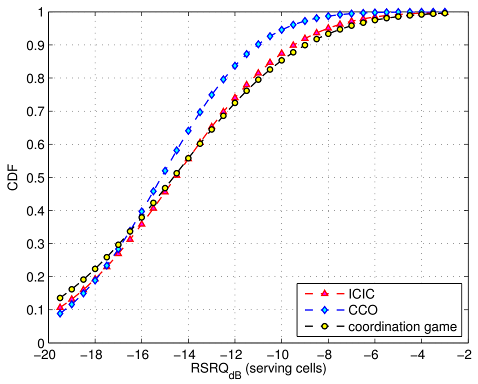

Figure 4.8: CDF of the RSRQ from the UEs.

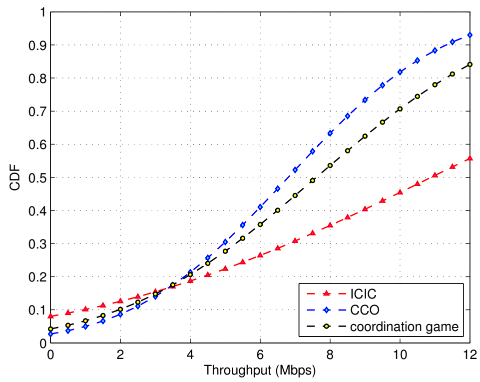

Figure 4.9: CDF of the UE Average THR.

4.6 Conclusions

In this chapter, we have discussed the challenging problem that arises when multiple concurrent SON functions are executed by the same node, or different instances of the same or different SON functions are executed in neighbouring cells. Without loss of generality, we have focused on the conflicts between two different SON functions, which aims at updating the same (H)eNB transmission parameter in a D-SON architecture, more suitable for a small cell scenario. We have proposed then a general framework to support the modelling of SON functions and their conflicts when they are executed in parallel. We have shown that the global SON problem can be modelled through a MDP, which can be organised onto simpler sub-problems, to favour scalability, and modelled by means of sub-MDP. Due to the dynamic nature of the wireless environment and to the autonomous characteristic of the SON functions, we solve the sub-MDPs by means of RL. RL algorithms provide solution policies to the different SON functions which can be in conflict, so that require a self-coordinator framework. We have shown that this framework can be modelled by means of a coordination game, where the sub-MDPs are the players, and their solution policies the actions. Simulation results obtained in a LENA platform LTE network simulator demonstrate that the proposed scheme provides a convenient compromise among conflicting actions, taking the best result among the conflicting solution policies.

Bibliography

[1] Seppo Hämäläinen, Henning Sanneck, Cinzia Sartori. *LTE Self-Organising Networks (SON): Network Management Automation for Operational Efficiency*. John Wiley and Sons, 2012.

[2] S. Weidenholzer. “Coordination Games and Local Interactions: A Survey of the Game Theoretic Literature”. *Games* (2010), pp. 551–585.

[3] H. Claussen, L. T. W. Ho, and L. G. Samuel. “Self-optimization of coverage for femtocell deployments”. *Wireless Telecommunicacion Symposium (WTS)* (2008), pp. 278–285.

[4] Iacoboaeia, O. and Sayrac, B. and Ben Jemaa, S. and Bianchi, P. “SON Coordination for parameter conflict resolution: A reinforcement learning framework”. *IEEE Wireless Communications and Networking Conference 2014 (IEEE WCNC 2014)* (2014).

[5] 3GPP. *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures*. Tech. rep. TS 36.213, version 10.2.0 Release 10.

[6] Craig Boutilier, Richard Dearden, and Moises Goldzmidt. “Exploting Structure in Policy Construction” (1995), pp. 1104–1113.

[7] Nicholas K. Jong and Peter Stone. “State Abstraction Discovery from Irrelevant State Variables”. *Joint Conference on Artificial Intelligence* (2005), pp. 752–757.

[8] Pierre Laroche, Yann Boniface, and René Schott. “A New Decomposition Technique for Solving Markov Decision Process”. *Symposium on Applied Computing (SAC)* (2001), pp. 12–16.

[9] Thomas Dean and Robert Givan. “Model Minimization in Markov Decision Process”. *14th National Conference on Artificial Intelligence and ninth conference on Innovative applications of Artificial Intelligence (AAAI)* (1997), pp. 106–111.

[10] Elva Corona-Xelhuantzi, L. Enrique Sucar and Eduardo F. Morales. “Executing Concurrent Actions with Multiple Markov Decison Process”. *Adaptative Dynamic Programming and Reinforcement Learning* (2009), pp. 531–579.

[11] 3GPP. *Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access Network (EUTRAN); X2 Application Protocol (X2AP) (Release 11)*. Tech. rep. TS 36.423, version 10.7.0. 2013.

[12] C. A. Coello, G. B. Lamont, and D. A. Van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*. 2nd. Springer: Genetic and Evolutionary Computation Series, 2007.

[13] M.N. ul Islam and A. Mitschele-Thiel. “Reinforcement Learning Strategies for Self-Organized Coverage and Capacity Optimization”. *Wireless Communication and Networking Conference (WCNC)* (2012).

[14] A. Galindo, L. Giupponi. “Distributed Q-learning for Interference Control in OFDMA-based Femtocell Networks”. *IEEE 71st Vehicular Technology Conference: VTC-Spring* (2010).

[15] Jessica Moysen and Lorenza Giupponi. “A Reinforcement Learning based solution for Self-Healing in LTE networks”. *in proc. of the 80th IEEE Vehicular Technology Conference (VTC Fall), Vancouver, Canada* (2014).

[16] M.Miozzo, L. Giupponi, M. Rossi, P. Dini. “Distributed Q-Learning for Energy Harvesting Heterogeneous Networks”. *IEEE ICC 2015 Workshop on Green Communications and Networks with Energy Harvesting, Smart Grids, and Renewable Energies* (2015).

[17] Lawal Mohammed Bello, Paul Daniel Mitchell, and David Grace. “Frame based back-off for Q-learning RACH access in LTE networks”. *Australasian Telecommunication* (2014).

[18] 3GPP. *Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Requirements (Release 10)*. Tech. rep. TR 32.521, version 10.2.0. 2012.

[19] Centre Tecnològic de Telecomunicacions de Catalunya (CTTC). *The LENA-EPC Network simulator*. URL: `http://networks.cttc.es/mobile-networks/software-tools/lena/`.

[20] 3GPP. *Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol Specification (Release 10)*. Tech. rep. TS 36.331, version 10.7.0. 2010.

[21] 3GPP. *TSG SA WG5 (Telecom Management) Meeting 85; Study of implementation alternative for SON coordination*. Tech. rep. S5-122330.

Chapter 5

A Mobile Network Planning Tool based on Data Analytics

The best way to predict the future is to create it. **Abraham Lincoln**

Contents

As we already mentioned in chapter 1, nowadays, we are assisting to the definition of what 5G networks will look like. 3GPP has started multiple work items which lead to the definition of novel 5G radio and architecture [1]. The main vendors are publishing numerous white papers presenting their view on 5G networks and architectures. EU commission has put in place an important 5GPPP program to fund research for 5G networks [2]. What looks clear from all these converging visions, is that in network management beyond 4G and future 5G networks has to face a whole new set of challenges due to high densification of nodes and heterogeneity of layers, applications, RAT, among others.

In this chapter, among many network management problems, we focus on designing a smart network planning, which gives particular emphasis to the QoS offered to the users. The state of the art in network planning, in literature and in the market, offers a wide range of platforms systems and applications oriented to the industry, as well as for research purposes. From the industry perspective and the research community, the market of commercial planning tools aims at providing a complete set of solutions to design and analyse networks [3–6]. In [7], the authors present a network planning tool capable to meet the requirements of the academia and the industry. They propose different planning algorithms to analyse the network under different failures and energy efficiency schemes. However, these works in general focus on several configuration scenarios, RF coverage planning, network recovery test, traffic load analysis, forecasting traffic, among others, and not directly on QoS offered to end-users and the resources the operator needs to offer it. Other works are more targeted to QoS estimation, but not in the area of network planning. The literature already offers different works targeting the problem of QoS prediction, and verification, such as, [8, 9]. In our preliminary work [10], we focus on more complex multi-layer heterogeneous networks, where we predict QoS independently of the physical location of the UE. Preliminary results show that by abstracting from the physical position of the measurements, we can provide high accuracies in the estimations of QoS in other arbitrary regions. Furthermore, our results obtained in [11] show that data analysis has better performance in a reduced space rather than in the original one.

This chapter presents a smart network planning tool based on 2 steps. First, we estimate the QoS in every point of the network based on measurements collected in different moments in time, and from other regions of the heterogeneous network. Second, we close the loop by adjusting network performances to target network objectives. To do this, we apply Machine Learning (ML) and Genetic Algorithms (GAs) techniques. We believe that ML can be effectively used to learn from experience while improving performance. The indicator that we want to estimate is expressed in terms of PRB per MB in an arbitrary point of the network. We focus on this specific QoS indicator because it combines information that is relevant for the operator (PRBs) and other that is relevant to the end user (Mb of data) into a single metric. We perform this estimation through supervised learning tools. Second, we close the loop by adjusting the network parameters in order to target network objectives in terms of PRB per Mb. The minimization of this indicator allows to serve users with an improved spectral efficiency and offered QoS. To carry out this optimization we take advantage of GAs, which are stochastic search algorithms useful to implement learning and optimization tasks [12].

In order to evaluate the performance of the proposed scheme, we consider 2 use cases, the first one in a densified indoor scenario, and the second one in a more traditional macrocell scenario. In the first use case, we focus on how to plan a dense small cell deployment inspired by a typical 3GPP dual stripe scenario [13], where the aspects to optimise are the number of small cells and their position. In the second use case we deal with self-healing aspects in macrocellular scenarios. We focus on readjusting the network planning to quickly solve an outage problem by automatically optimising the antenna tilt parameters.

We will show, through network simulation results obtained over the 3GPP compliant, full protocol stack LTE ns-3 module, that the proposed ML enabled network planning, differently from other closed optimisation approaches, is extremely flexible to the application problem and scenario. It is equally valid and brings great benefits in both the studied use cases, characterized by very different optimisation problems and scenarios.

The outline of the chapter is organised as follows. The general approach is described in Section 5.1. It introduces the criteria that we use to construct the proposed method. In Section 5.2, we select the learning methods that the network planning tool follows to reach the operators targets. In Section 5.3, we present the details of 2 use cases of application, the simulation platform, and meaningful simulations results. Finally, Section 6.1 summarizes with the conclusions.

5.1 ML-based network planning tool description

We propose to design a smart network planning tool, which involves 3 main processes: 1) Data preparation; 2) Data analysis; and 3) Optimised network planning. We model and estimate the QoS in every point of the network through analysing data extracted from UE measurements. This modelling consist on the first 2 process already mentioned. The 3 main processes are depicted in Figure 5.1 and discussed in the following.

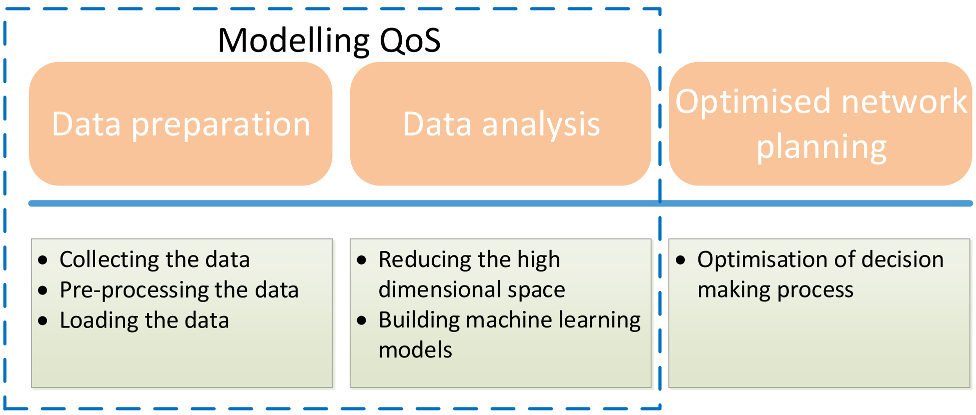

Figure 5.1: Architecture of the network planning tool.

1. *Data preparation*. This process aims at transforming data into an understandable format. The target is to integrate and prepare large volumes of data over the network to provide a unified information base for analysis. To do this, we follow the Extract-Transform-Load

Table 5.1: Relevant sources of information in mobile networks

Source	Information	Usage
Control info for short-term network operation.	Call/session setup, release, maintenance QoS, RRC, idle and connects mode mobility.	Discarded after usage.
Control info for SON functions.	Info on Radio link failure, inter-cell interference, UE measurements, MDT measurements, radio resource status, cell load signalling, etc.	Heuristic algorithms typically discard info after.
Management information for long-term operation.	Fault configuration, accounting, performance and security management (FCAPS), Operations and Management, e.g. in OAM aggregated statistics per eNB, on network performances, # users, successful/failed HO, active bearers, information from active probing.	Mainly used for triggering engineer intervention.
Customer Relationship Information.	Complaints about bad service quality, churn info.	Only used by customer service.

(ETL) process, which is responsible for pulling data out of the source and placing it into a database. It involves 3 main steps: Data extraction (E), whose objective is to collect the data from different sources; Data transformation (T), which prepares the data for the purpose of querying and analysis; Data loading (L), which loads the data into the main target, most of the cases into a flat file. This process plays an important role for the design and implementation of planning future mobile networks. The objective is to create a data structure, which is able to provide meaningful insights and decisions. Some examples of the kind of sources available in mobile networks are shown in Table 5.1 [14]. Here the data are classified based on the purpose for which are generated in the network. The usage that is given nowadays in the network is also suggested in the last column. For the purpose of network planning, we plan to extract data reported by the UEs to the network in the form of UE measurements, in terms of received power, received quality and perceived QoS.

Once the data has been collected, we prepare the data for storing, using the proper structure for the querying.

2. *Data analysis*. The objective of this process is to discover patterns in data that can lead to predictions about the future. This is done by finding this information/correlation among the radio measurements extracted from the network. We do this by applying ML techniques.

3. *Optimised network planning*. The objective of this process is to find the configuration parameters for the optimised network planning based on the information extracted from the previous data analysis process. In the complex cellular context, we need to deal with several challenges which introduce high complexity, e.g., the very large number of parameters, the strong cross-tier interference, due to aleatory introduced by random propagation patterns, like fast fading, shadowing, mobility of users, etc. In order to deal with these issues, and guarantee the capacity requirements of the network, in this work, we propose to use the tool of GAs, for many advantageous characteristics, which allow to avoid more typical closed optimisation techniques, which would be infeasible in a complex and dynamic 5G scenario as the one we are considering. In particular, GAs perform parallel search from a population of points. They face the ability to avoid local minima. They use probabilistic, and not deterministic search rules. They work as chromosomes, which are the encoded versions of potential solutions parameters, rather than parameters themselves, and they use fitness scores, obtained from the objective function [15].

5.1.1 Modelling the QoS

We estimate the QoS in every point of the network based on measurements collected in different moments in time, and from other regions of the heterogeneous network. To do this, we consider the *Data preparation* and *Data analysis* processes of the network planning tool. As mentioned previously, the objective of these 2 processes is to extract, prepare and analyse the information already available in the network to provide insightful information from the analysis of it. As we stated earlier, in this kind of problems ML techniques can be very effective to make predictions based on observations. We then take advantage of ML techniques to create a model that allows to estimate the QoS by finding patterns in PHY layer measurements reported by the users. We propose to use SL, since among many applications, it offers tools for estimation and prediction of behaviours. In particular, we focus on a regression problem, since we want to analyse the relationship between a continuous variable (PRB per Mb), and data extracted from the network in the form of UE measurements. Many regression techniques have been developed in the SL literature, and criteria to select the most appropriate method include aspects such as the kind of relation that exists between the input and the output, or between the considered features, the complexity, the dimension of the dataset, the ability to separate the information from the noise, the training speed, the prediction speed, the accuracy in the prediction, etc. We focus on regression models, and we select the most representative approaches. We then use ensemble methods to sub-sampling the training samples, prioritizing criteria such as the low complexity and the high accuracy.

We build a dataset of user measurements, based on the same data contained in the MDT database. The MDT is a standardized database used for different 3GPP use cases. The dataset contains training samples (rows), and features (columns), and is divided in 2 sets. The training set to train the model, and the test set to make sure that the predictions are correct. That training data develops a predictive model, and evaluate the accuracy of the prediction, by inferring a function $f(\mathbf{x})$, returning the predicted output $\hat{y}$. The input space is represented by a n-dimensional input vector $\mathbf{x} = (x^{(1)}, \ldots, x^{(n)})^T \in \mathbf{R}^n$. Each dimension is an input variable. In addition, a training set involves m training samples $((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m))$. Each sample consists of an input vector $\mathbf{x}_i$, and a corresponding output y_i of one data point i. Hence $x_i^{(j)}$ is the value of the input variable $x^{(j)}$ in training sample i, and the error is usually computed via $|y_i - \hat{y}_i|$.

In addition to regression analysis, we exploit UL techniques for dimensionality reduction, whose output is fed into an ensemble method consisting of Bagging/AdaBoost to manipulate the training examples. The SL techniques under evaluation are then applied. Details for each step are given in the following, and the whole process is depicted in Figure 5.2.

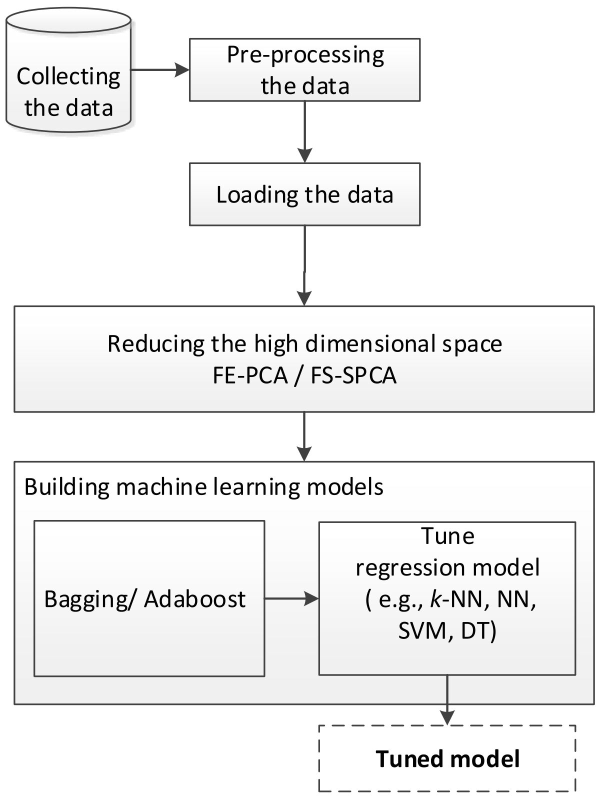

Figure 5.2: Modelling the QoS.

Collecting the data

The data we take into account comes from mobile networks, which generate data in the form of network measurements, control and management information (Table 5.1). As we mentioned previously, we focus on MDT functionality, which enables operators to collect UEs measurements together with location information, if available, with the purpose of optimising network management, while reducing operational effects and maintenance costs. This feature has been introduced by 3GPP since Release 10, among the targets there are the standardization of solutions for coverage optimisation, mobility, capacity optimisation, parametrization of common channels, and QoS verification. Since operators are also interested in estimating QoS performance, in Release 11, MDT functionality has been enhanced to properly dimension and plan the network by collecting measurements indicating throughput and connectivity issues [16]. Therefore, we collect for each UE: (1) the RSRP, and (2) the RSRQ coming from the serving and neighbouring eNBs. The size of the input space is $[l \times n]$. The number of rows is the number l of UEs in the scenario, and the number of columns corresponds to the number of measurements n. The size of the output space is $[l \times 1]$, which corresponds to the QoS performance associated to the measurements in terms of the PRB per transmitted Mb. In particular, these measurements gathered at arbitrary points of the network throughout its lifetime are exploited to plan other arbitrary future deployments.

Pre-processing the data

In order to obtain a good performance during the evaluation, the input variables of the different measurements must be on a similar scale and range. So, a common practice is to normalize every variable between $-1 \leq x^{(j)} \leq 1$ range, and replace $x^{(j)}$ with $x^{(j)} - \mu^{(j)}$ over the range (max-min), where $\mu^{(j)}$ is the average of the input variable (j) in the dataset. The normalized data is then split into training and test set. We create a random partition from the l sets of input. This partition divides the observations into a training set of m samples, and a test set $p = l - m$ samples. We randomly select approximately $p = \frac{1}{5} \times l$ observations for the test set.

Loading the data

This process varies widely. As we mentioned before, depending on the operator requirements, the data can be updated or new data can be added in a historical form at regular intervals. For our propose, in this particular work, we maintain a history of all changes to the data loaded in the network.

Reducing the high dimensional space

One of the problems that mobile operators have to face in this kind of networks is the huge amount of potential features we have as input. Therefore, to deal with the huge amount of features, we propose to apply regression techniques in a reduced space, rather than in the original one. The idea behind that, comes from our previous work [10], in which we observed that we lose information in the high dimensionality of measurements obtained from all the neighbours. Therefore, we suggest that the regression analysis has a better performance in a reduced space. As a result, we take advantage of dimensionality reduction techniques to reduce the number of random variables under consideration. These methods can be divided into FE and FS methods. Both methods seek to reduce the number of features in the dataset. FE methods do so by creating new combinations of features (e.g. PCA), which project the data onto a lower dimensional subspace by identifying correlated features in the data distribution. They retain the PCs with greatest variance and discard all others to preserve maximum information and retain minimal redundancy [17]. Correlation based FS methods include and exclude features present in the data without changing them. An example is SPCA, which extends the classic method of PCA for the reduction of dimensionality of data by adding sparsity constraint on the input features. Notice that, in a sparse matrix most of the elements are zero. By contrast, if most of the elements are nonzero, then the matrix is considered dense. In FS-SPCA the sparsity is promoted up to the selection of the f features that give us the most useful information. That is, by adding sparsity constraint on the input features, we promote solutions in which only a small number of input features capture most of the variance. Some preliminary work on this features was presented in [11].

Building the machine learning models

We select some representative regression models, prioritizing criteria such as the low complexity and the high accuracy: (1) k-NN, (2) Neural Networks (NN), (3) SVM, and (4) DT, and we analyse them by performing an empirical comparison of these algorithms, observing the impact on the prediction of the different kinds and amounts of UE measurements.

1. k-NN can be used for classification and regression [18]. The k-NN method has the advantage of being easy to interpret, fast in training, and the amount of parameter tuning is minimal.

2. NN is a statistical learning model inspired by the structure of a human brain where the

interconnected nodes represent the neurons to produce appropriate responses. NN support both classification and regression algorithms. NNs methods require parameters or distribution models derived from the dataset, and in general they are susceptible to over-fitting [19].

3. SVM can be used for classification and regression. The estimation accuracy of this method depends on a good setting of the regularization parameter C, ϵ, and the kernel parameters. This method in general shows high accuracy in the prediction, and it can also behave very well with non-linear problems when using appropriate kernel methods [20].

4. DT is a flow-chart model, which supports both classification and regression algorithms. Decision trees do not require any prior knowledge of the data, are robust, and work well on noisy data. However, they are dependent on the coverage of the training data as, as for many classifiers, and they are also susceptible to over-fitting [21, 22].

In order to enhance the performance of each learning algorithm described before, instead of using the same dataset to train we can use multiple data sets by building an ensemble method. Ensemble methods are learning models, which combine the opinions of multiple learners. This technique has been investigated in a huge variety of works [23, 24], where the most useful techniques have been found to be Bagging and AdaBoost [25]. Bagging manipulates the training examples to generate multiple hypothesis. It runs the learning algorithm several times, each one with different subset of training samples. AdaBoost works similarly, but it maintains a set of weights over the original training set, and adjusts these weights by increasing the weight of examples that are misclassified, and decreasing the weight of examples that are correctly classified [26].

At high level, once the extracted data has been processed and loaded into a flat file, is fed into a dimensionality reduction step, and successfully into an ensemble step, which manipulates the training set by applying Bagging/AdaBoost techniques. The learning algorithm is then applied to produce a regressor (see Algorithm 5.1).

Evaluation of accuracy

To evaluate the accuracy of the model, the performance of the learned function is measured on the test set. That is, we use a set of examples used to tune the regressor algorithm. For each test value, we predict the average QoS, and evaluate performance against the actual value in terms of the Root Mean Squared Error (RMSE) as follows $RMSE = \sqrt{\frac{\sum_{i=1}^{p}(y_i-\hat{y}_i)^2}{p}}$, where p is the length of the test set, $\hat{y}_i$ indicates the predicted value, and y_i is the testing value of one data point i. In order to compare the RMSE with different scales, the input and output variable values are normalized by, NRMSE=$\frac{RMSE}{y_{max}-y_{min}}$, where y_{max} and y_{min} represent the max and min values in the output space Y_{test} of size $[p \times 1]$ respectively.

Algorithm 5.1. Train regressor algorithm

Input: Input space of size $[m \times n](X_{train})$,
output space of size $[m \times 1](Y_{train})$,
Input space of size $[p \times n](X_{test})$,
output space of size $[p \times 1](Y_{test})$,
number of iterations $(niter)$

Output: *model*

// Given the training set (X_{train}, Y_{train})
$X_{train} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$
$Y_{train} = \{y_1, \ldots, y_m\}$
// Apply dimensional reduction step (if it is necessary)
obtain c-dimensional input vector
Apply regression analysis in a reduced space
// Set up the data for Bagging/Adaboost
for $k = 1$ to $niter$ **do**
// Call regressor algorithm
model(k):=regressor algorithm, namely k-NN, NN, SVM,DT
// Predict the average QoS
$QoS_{predicted}$:=predict(model(k),X_{test})
Evaluate performances against the actual value (Y_{test}) by NRMSE
end for
// Result of training base learning algorithm
model:=best(model)
return (*model*) }

5.1.2 Optimised network planning

As we mentioned before, the objective of this process is to close the loop by adjusting network performances through a GA to target network objectives in terms of PRB per Mb. This results in a GA organised onto different phases, as depicted in Figure 5.3.

Create S feasible solutions

We create a set of $\mathrm{S} = \{\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(p_{size})}\}$ random feasible solutions (also called chromosomes or individuals), where p_{size} is the starting population size. We denote by $\boldsymbol{\theta}^{(s)} = (\theta_1^{(s)}, \ldots, \theta_M^{(s)})$

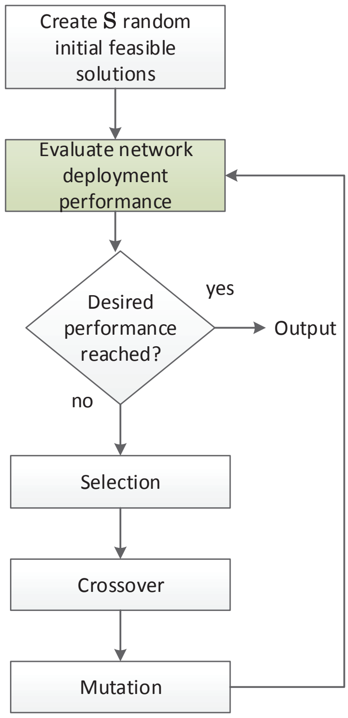

Figure 5.3: Optimised network planning.

the configuration parameters vector of an individual s, with $\theta_j^{(s)}$ denoting the parameter configuration, e.g., the transmitted power ($\theta_{j_{txp}}^{(s)}$) or the antenna tilt ($\theta_{j_{tilt}}^{(s)}$) value of the j-th eNB, or switch on-off ($\theta_{j_{sc}}^{(s)}$) the j-th small cell, among others.

Evaluate network deployment performance

This function is responsible for evaluating the network deployment performance. The objective is to define the evaluation function (fitness) of each individual. Given a particular configuration parameter vector $\boldsymbol{\theta}^{(s)}$, this function is responsible for returning the average offered QoS of a subset of random points in the scenario. This function takes as an input $\mathbf{S}$ feasible solutions, and the model produced by the regressor algorithm discussed in previous section. That is, the output of Algorithm 5.1.

The behaviour of this module is as follows. In each iteration we collect X_{val} measurements at some arbitrary point in the scenario. These measurements are obtained as a result of the evaluation of each $\boldsymbol{\theta} \in \mathbf{S}$ in the **evalNetPerformance** function (see Algorithm 5.2). These X_{val} measurements and the tuned model (*model*), are the inputs to the *predict* function, which gives us the QoS in any position of the scenario. As a result, for each $\boldsymbol{\theta} \in \mathbf{S}$, we obtain a QoS value predicted for each point of interest in the scenario.

As we mentioned before, we aim at minimizing the PRB per transmitted Mb to allow max-

Algorithm 5.2. Evaluate network deployment performance

Input: Configuration parameters vector ($\boldsymbol{\theta}$),
the tuned model (*model*)
Output: average QoS

---

evalNetPerformance := function($\boldsymbol{\theta}$, *model*){
// Call ns-3 network simulator
evaluate $\boldsymbol{\theta} = (\theta_1^{(s)}, ..., \theta_M^{(s)})$
// Collect measurements at q points in the scenario
$X_{eval} = \{\mathbf{x}_1, \ldots, \mathbf{x}_q\}$
// Predict the average QoS
$QoS_{predicted}$:= *predict* (*model*,X_{eval})
average QoS:=mean($QoS_{predicted}$)
return (average QoS)}

imizing the QoS of the users and the spectral efficiency of the operator. Therefore, the fitness function aims at finding the configuration of parameters for which the total PRB per transmitted Mb is minimized. The operator can target a desired value for the total PRB per transmitted Mb, and based on this the network planning tool can decide when the objective has been achieved and interrupt the operation.

Selection

We select the best fit individuals for reproduction based on their fitness, i.e., this function generates a new population of individuals from the current population. This selection is known as elitist selection. Elitism copies the best e fittest candidates, into the next generation.

Crossover

This function forms a new individual by combining part of the genetic information from their parents. The idea behind crossover is that the new individual may be better than both of the parents if it takes the best characteristics from each of them. We use an arithmetic crossover, which creates new individuals (β) that are the weighted arithmetic mean of two parents. If $\boldsymbol{\theta}^{(a)}$ and $\boldsymbol{\theta}^{(b)}$ are the parents, the function returns,

$$\beta = \alpha \times \boldsymbol{\theta}^{(a)} + (1 - \alpha) \times \boldsymbol{\theta}^{(b)} \tag{5.1}$$

where, α is a random value between $[0, 1]$.

Mutation

This function randomly selects a parameter based on a uniform random value between a minimum and maximum value. It maintains the diversity in the value of the parameters for subsequent generations. That is, it avoids premature convergence on a local maximum or minimum. For that, we set to δ the probability of mutation in a feasible solution $\boldsymbol{\theta}^{(s)}$. If δ is too high, the convergence is slow or it never happens. Therefore, most of the times δ tends to be small. Finally we replace the worst fit population with new individuals. The whole genetic algorithm is described in Algorithm 5.3.

Algorithm 5.3. GA scheme

Input: Initial population ($\mathbf{S}$) ,
    size of $\mathbf{S}$ (p_{size}),
    the tuned model (*model*),
    number of generations (g),
    rate of elitism e,
    rate of mutation δ
Output: solution $\boldsymbol{\theta}^{(*)}$

// Initialization
for $i = 1$ to g **do**
  // Return the value of average QoS describing the fitness of each individual $\boldsymbol{\theta} \in \mathbf{S}$
  for all $\boldsymbol{\theta} \in \mathbf{S}$ **do**
    average QoS := **evalNetPerformance**($\boldsymbol{\theta}$, *model*)
  end for
  // Elitism based selection
  select the best e solutions
  // Crossover
  number of crossover $n_c = (p_{size} - e)/2$
  for $j = 1$ to n_c **do**
    randomly select two solutions $\boldsymbol{\theta}^{(a)}$ and $\boldsymbol{\theta}^{(b)}$
    generate $\boldsymbol{\theta}^{(c)}$ by arithmetic crossover to $\boldsymbol{\theta}^{(a)}$ and $\boldsymbol{\theta}^{(b)}$
  end for
  // Mutation
  for $j = 1$ to nc **do**
    mutate each parameter of $\boldsymbol{\theta}^{(j)}$ under the rate δ and generate a new solution
  end for
end for

return the best solution $\boldsymbol{\theta}^{(*)}$

5.2 ML tool to estimate the QoS

This section defines our model in terms of dimensional reduction, ensemble and regression methods. We consider different options for: 1) dimensionality reduction (i.e., FE-PCA, FS-SPCA); 2) ensemble methods (i.e., Bagging and AdaBoost); and 3) regression models (i.e., k-NN, NN, SV, DT), as anticipated in section 5.1.1.

Table 5.2 summarizes the accuracy performance of each learning algorithm in terms of $(1 - NRMSE) \times 100$. From this table, we observe the following:

Table 5.2: Overall model accuracy

Approach	Regression model	Bagging	AdaBoost
1. FE-PCA	1.1 k-NN	90.33%	91.98%
	1.2 NN	93.44%	92.28%
	1.3 SVM	**94.70%**	94.07%
	1.4 DT	92.84%	93.60%
2. FS-SPCA	2.1 k-NN	89.69%	90.88%
	2.2 NN	92.41%	91.78%
	2.3 SVM	93.62%	92.87%
	2.4 DT	91.22%	92.08%

1. FS-SPCA is a very useful approach if we are interested in excluding features to retain minimal redundancy. In this context, we can reduce the dimensionality of the data up to 92 features and still maintain almost the same accuracy. This can be observed in Figure 5.5, which shows the NRMSE as a function of different number of features selected by the SPCA. As a consequence, we promote sparsity down to 92 features that give us the most useful information. In this respect, FE-PCA would present less inputs to the SL step of the data processing chain at the cost of a prior processing of features. As we observed in Figure 5.4, the results of the FE-PCA implementation suggest, that we can obtain the 70$\%$ of cumulative variance if we consider only the first 10 PCs. That is, with the first 10 PCs we already capture the main variability of the data. As a result, FS-SPCA would simplify the initial feature processing, since selected features are taken as they are, but the cost would be the higher storage need (i.e., 92 inputs vs. 10 inputs to the SL

step).

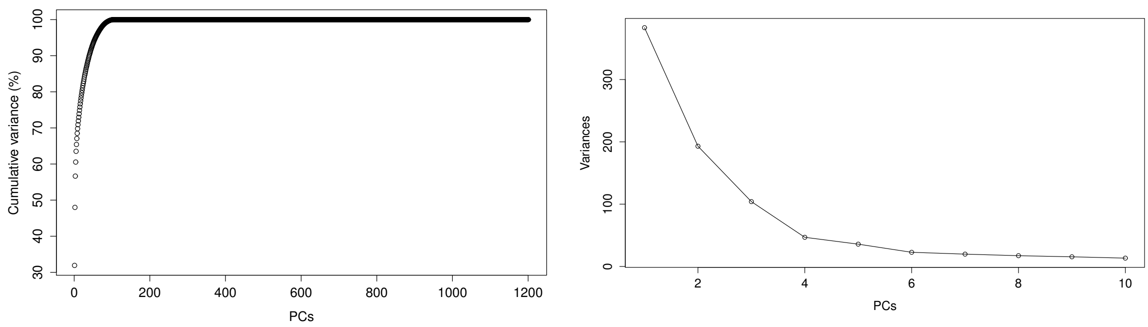

Figure 5.4: The left side, shows the cumulative contribution of each PC to the original data's variance. The right side, shows the variability of the data set as a function of the $c = 10$ PCs. That is, the figure shows the variances (y-axis) associated with the PCs (x-axis).

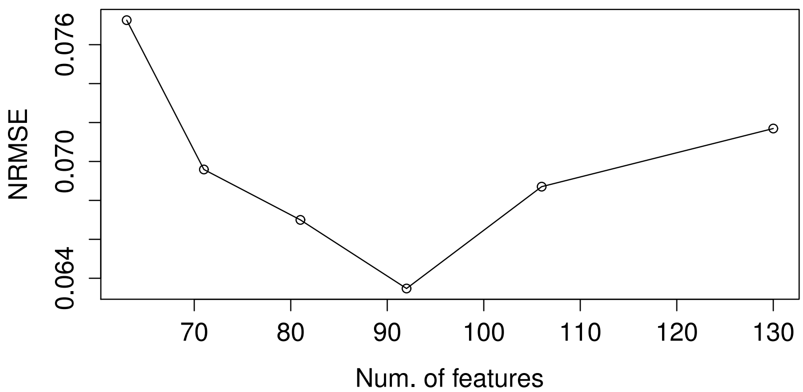

Figure 5.5: NRMSE a function of different number of features selected by the SPCA.

Once the kinds and amounts of measurements that each approach takes into account, have been explained, from Table 5.2, we observe that if we consider the 92 features, we lose only 1% of accuracy with respect to FE-PCA. Therefore, it will depend on the specific network and operator to select whether computing or storage should be optimised and to decide whether the price paid in terms of accuracy is acceptable.

2. When we build ensemble methods, SVM and NN regression models perform better when they are bagged than when they are boosted. This was to expect, as Bagging combines many weak predictors (i.e., the predictor is only slightly correlated with the true prediction) to produce a strong predictor (i.e., the predictor is well-correlated with the true prediction). This works well for algorithms where by changing the training set, the output changes.

The opposite behaviour can be found in k-NN and DT regression models, i.e., when these algorithms are boosted the models tend to provide better results than when they are bagged. That is, in order to improve the performance of AdaBoost, we use sub-optimal values, k for k-NN, and T for DT, i.e., we use values that are not that good, but at least better than random. As a result AdaBoost does its job properly. This is not the case for SVM and NN. Since these learning algorithms do not have an input parameter that we can adjust to obtain a weak predictor without affecting the accuracy of the model, the probability that these algorithms provide better performance when are boosted than when are bagged is lower. Some initial results can be found in [27].

3. By applying different regression models and in particular, when the SVM regression model is bagged, we improve by 5% the overall accuracy of the prediction with respect to the k-NN model. More specifically, in terms of the NRMSE, k-NN exhibits an error of 10%, while SVM halves this value to 5%.

Results suggest that while in overall, all the regression models exhibit high accuracy, bagged-SVM learning model is the one that better fits our needs, and exhibits more accurate predictions. Therefore, in this work we focus on Bagged-SVM to build the best model that fit the data. The whole network planning process is depicted in Figure 5.6.

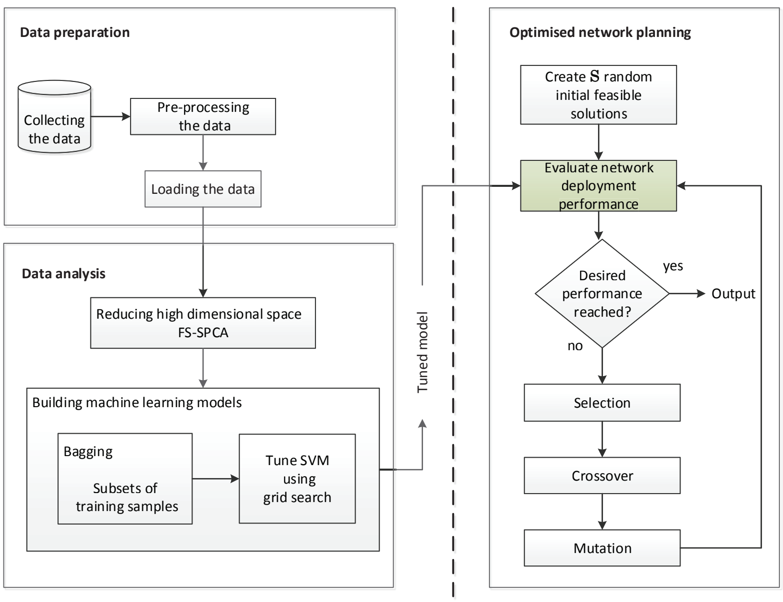

Figure 5.6: Bagged-SVM network planning tool architecture.

5.3 Network planning results

In order to evaluate the performance of the proposed scheme, we consider two cases of study.

A. *Case study* $\#1$*-Deployment planning in a dense small cell scenario*. We propose to exploit the experience gained in other sectors of the network, to properly dimension and deploy heterogeneous nodes in an indoor small cell deployment. We aim at providing a smart network planning to improve the QoS offered to end-users, and to increase the spectral efficiency of the operators

B. *Case study* $\#2$*-Self-healing to compensate faults*. COC is applied to alleviate the outage caused by the loss of service from a faulty cell. For this use case, an adequate reaction is vital for the continuity of the service. As a result, vendor specific COD schemes have also to be designed [28]. In this case of study we assume the outage has already been detected, and we focus on readjusting the network planning to quickly solve the outage problem.

The planning tool aims at guaranteeing that the network meets the operators needs. We proceed to design the appropriate planning tool by defining the following aspects:

1. Data. We define the radio measurements we extract and analyse from the network.
2. Parameter. We define the network parameters we aim to optimise.
3. Action. We define the possible actions to take to optimise the network
4. Objective. We define the system level target.

Table 5.3 presents these information for each case of application.

The results of the use cases of application are described in the rest of the section. We first present the simulation scenario and then the simulation results for each use case.

5.3.1 Case of study $\#1$-Deployment planning in a small cell scenario

We aim at providing a network planning of a small cell indoor deployment to improve the QoS offered to end-users, and to increase the resource efficiency of our planning by our approach.

Table 5.3: Information relevant for the network planning tool

Information	Case study $\#1$	Case study $\#2$
Data	RSRP and RSRQ coming from the serving and neighbouring cells	RSRP and RSRQ coming from the serving and neighbouring cells
Parameter	$\boldsymbol{\theta_{sc}} = (\theta_{sc_1}, \ldots, \theta_{sc_{Nsc}})$, which is a binary vector that denotes if the small cells are switch on-off	$\boldsymbol{\theta_{tilt}} = (\theta_{tilt_1}, \ldots, \theta_{tilt_M})$, which is a vector that denotes the tilt value associated to each of the surrounding cells that are trying to fill the outage gap
Action	Switch on-off each small cell	Adjusting the antenna tilt parameter
Objective	Increasing the resource efficiency of our planning by designing the dimension of a LTE indoor small cell deployment	Readjusting the network planning to quickly solve an outage problem

Simulation scenario

The scenario that we set up consists of 1 eNB, with 3 sectors. We need to plan the deployment of the small cell network defined as the standard dual stripe scenario based on 1 block of 2 buildings. The building has 1 floor, with 20 apartments, which results in 40 apartments, as depicted in Figure 5.7. We consider that 1 small cell is located in each apartment, and the planning will decide which one will be switched ON or OFF. The parameters used in the simulations and the learning parameters are given in Table 5.4.

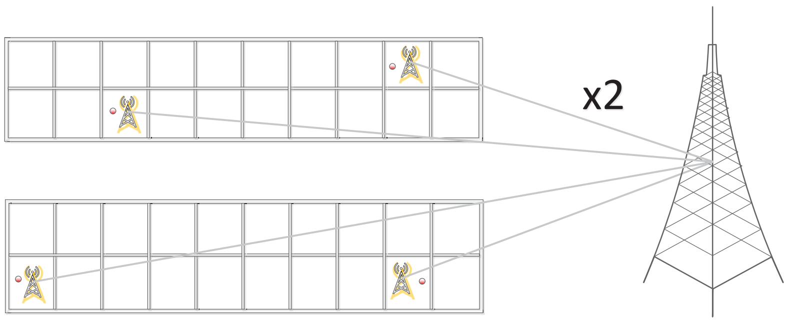

Figure 5.7: Scenario

Simulation results

The simulation starts by an initial deployment, where each apartment of the building has randomly deployed a small cell. This initial deployment is depicted in Figure 5.8(a), which represents the SINR in each point of the scenario. The idea is to determine the most effective location for the small cells by evaluating the performance of each $\boldsymbol{\theta_{sc}}$ configuration. There-

Table 5.4: Case of study #1-Simulation parameters

Simulation parameters	Value
Propagation loss model	HybridBuildings
Shadow fading	Log-normal, std = 8 dB
Scheduler	Proportional Fair Scheduler (PF)
AMC model	LteAmc::MiErrorModel
Transport protocol	UDP
Traffic model	Constant bit rate
Layer link protocol	RLC
Mode	Unacknowledged Mode (UM)
Macro cell scenario	
Number of cells	3
eNB Tx power	46 dBm
Small cell scenario	
Initial number of small cells (Nsc)	40
Small cell Tx power	23 dBm
LTE	
Cell layout	Radius: 500 m
Bandwidth	5 MHz
Number of RBs	25
TTI	1 ms
GA	
Type	binary-valued
Size of **S** ($psize$)	100
Elitism e	2
Mutation change δ	0.1
Bagged-SVM	
Num. of iteration (γ)	1000
Epsilon ϵ	0.1
Kernel	Radial Basis Function (RBF)
Simulation time	0.25 s

fore, the proposed network planning tool first takes advantage of the data preparation and data analysis processes to estimate the QoS at any random point in the scenario. Then, based on the offline evaluation, it online learns, through the *optimised network planning* tool the $\boldsymbol{\theta}_{\boldsymbol{sc}}^{(*)}$ best configuration to deploy. The configuration $\boldsymbol{\theta}_{\boldsymbol{sc}}^{(*)}$ is represented by a binary string, with dimension N_{sc}. This process is referred here after as $GA_{\boldsymbol{\theta}_{\boldsymbol{sc}}}$.

We evaluate the system performance of each deployment on the ns-3 platform based on

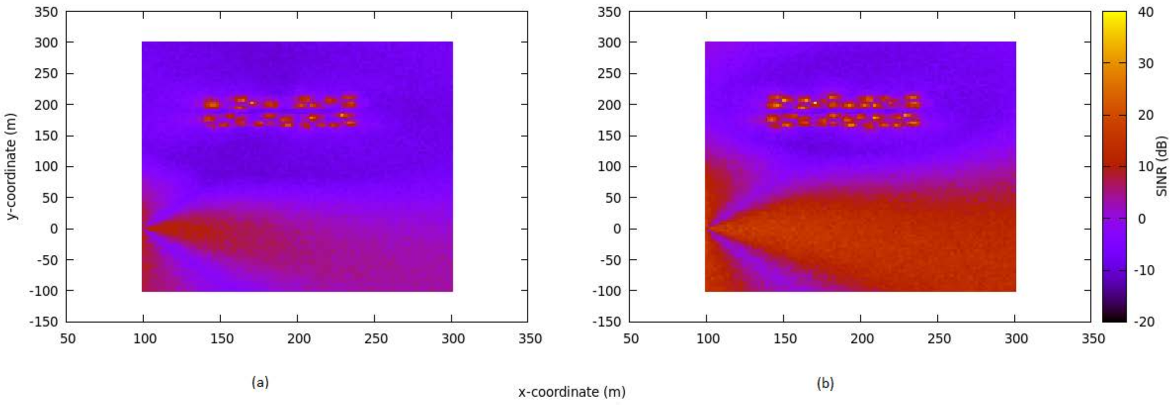

Figure 5.8: Figure (a), shows the initial deployment, in which we consider one small cell located in each apartment. Figure (b), shows the final deployment, in which the $GA_{\boldsymbol{\theta}_{sc}}$ scheme finds the number of small cells to deploy

3GPP component LTE. We consider a binary string (with dimension N_{sc}). Each number in this binary string represents either the j-th small cell is switch ON or OFF. A value of 1 means that the power transmission of the j-th small cell is set to 23 dBm, while a 0 means that the j-th small cell is switch-off. At each iteration, the $GA_{\boldsymbol{\theta}_{sc}}$ process evaluates different strings, and when the evaluation is done, the $GA_{\boldsymbol{\theta}_{sc}}$ process provides a new configuration of small cells to deploy. Once we know the new configuration, we modify the deployment to get the new performance, and the process starts again, until the network topology structure guarantees the operator's requirements (i.e., until the target of the operators has been reached).

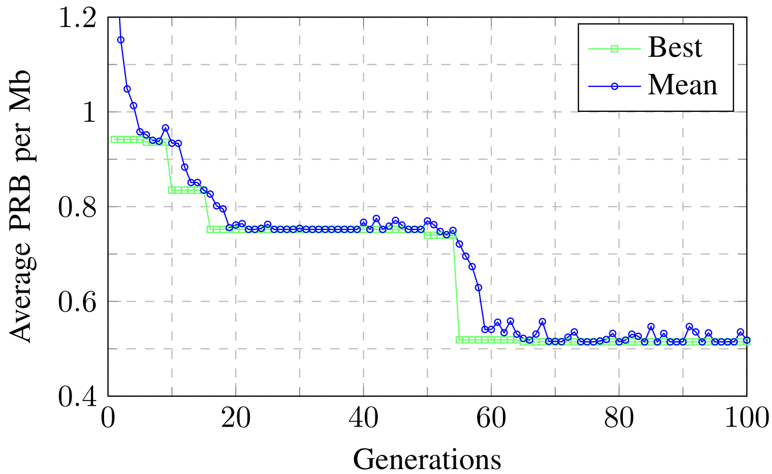

Figure 5.9: Evolution of the average PRB per Mb.

Figures 5.9 and 5.10, show the fitness of the best individuals found in each generation (Best), and the mean of the fitness values across the entire population (Mean). We observe that, as the generations proceed, and the $GA_{\theta_{sc}}$ evolves, the PRB/Mb decreases as it was expected, and the efficiency of the planning increases.

Figure 5.9, depicts the time evolution of the average PRB per Mb in the scenario. At 60-th generation, the $GA_{\theta_{sc}}$ finds the configuration of small cells able to increase the resource efficiency of our planning.

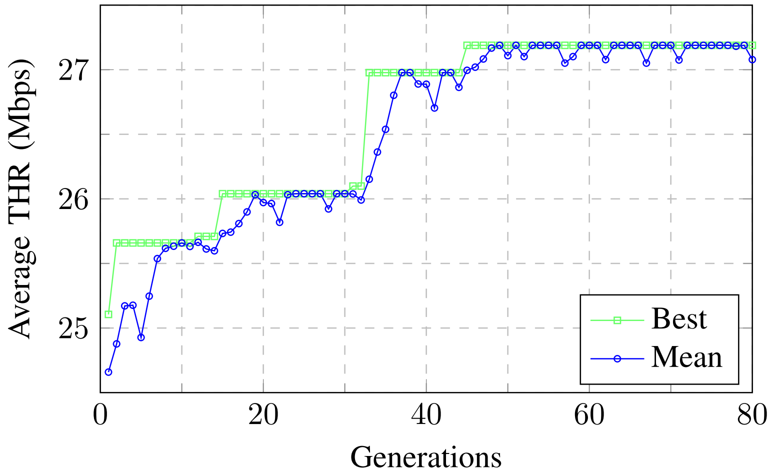

Figure 5.10: Evolution of the average throughput in the whole network.

Figure 5.10, shows the evolution of the $GA_{\theta_{sc}}$ scheme in terms of the average throughput in the whole network. We observe that as the generations proceed, the $GA_{\theta_{sc}}$ scheme finds the number of small cells which maximize the throughput. Figure 5.11, shows the CDF of the SINR in the building. From this figure we observe that taking into account the new configuration of small cells deployed, which results on 28 small cells in the whole building, we increase the SINR inside the building (see Figure 5.8(b)). The performance to determine the capacity of the $GA_{\theta_{sc}}$ scheme is compared in this figure to the greedy algorithm. The greedy algorithm is one of the algorithms often used in combinatorial optimisation [29]. The greedy algorithm searches for the best configuration, by finding the best combination. The algorithm generates a random string vector, and returning the QoS indicating how important a given configuration is, and choose the best one. And the process starts again. The selection of the best node is repeated approximately m_{iter} number of times in case no better node found. While the greedy search algorithm rarely outputs optimal solutions, it often provides some kind of approximation. This can be observed in Figure 5.11, where the number of small cells deployed results on 33 in the whole building. We observe in this figure, that in general the $GA_{\theta_{sc}}$ tends to work effectively since it makes a much deeper estimation of the state of the environment through the regression analysis approach than the greedy scheme, which takes the best it can get at any moment, i.e., the closest solution that seems to provide optimum solution is chosen.

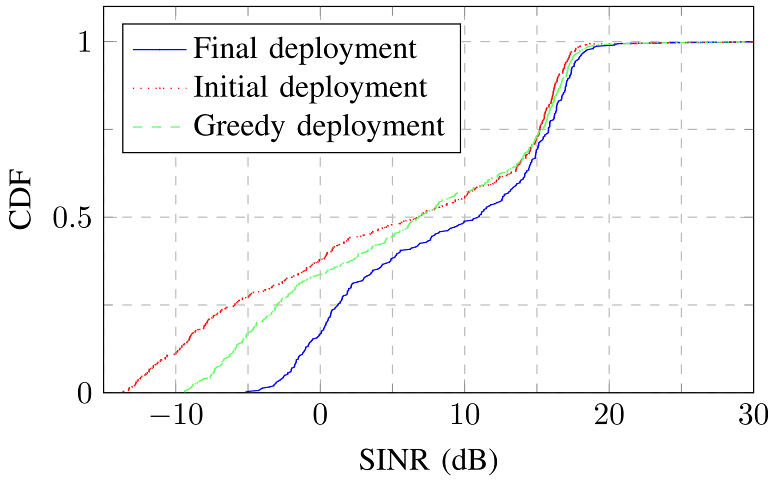

Figure 5.11: CDF of the SINR (dB) in the building.

5.3.2 Case of study $\#2$-Self-healing to compensate faults

As we already mentioned, in this case of application, we focus on readjusting the network planning to quickly solve an outage problem by automatically optimising the antenna tilt parameter. We consider the hypothesis of a sector fault in a typical macro cellular deployment. That is, during a certain period of time a sector is not able to provide a service to its users. Therefore, the proposed network planning tool allows optimisation of the antenna tilt for each compensating sector to automatically alleviate the outage caused by the loss of service from a faulty sector [30].

Simulation scenario

We consider a LTE cellular network composed of a set of $\mathcal{M}$ eNBs. The $\mathcal{M}$ eNBs form a regular hexagonal network layout with inter-site distance D, and provide coverage over the entire network. We assume that a sector in the scenario is down (see Figure 5.12). The parameters to tune are the antenna tilts of the cells neighbouring the affected sector. In particular, the surrounding M cells automatically and continuously adjust their antenna tilt, till the coverage gap is filled. The vertical radiation pattern of a cell sector is obtained according to [31], where the gain in the horizontal plane is given by

$$G_h(\phi) = \max\{-12\left(\frac{\phi}{HPBW_h}\right)^2, SLL_h\} \tag{5.2}$$

Here ϕ is the horizontal angle relative to the maximum gain direction, $HPBW_h$ is the half power beam-width for the horizontal plane, SLL_h is the side lobe level for the horizontal plane.

Similarly, the gain in the vertical plane is given by,

$$G_v(\rho) = \max\{-12\left(\frac{(\rho - \theta_{tilt})}{HPBW_v}\right), SLL_v\} \tag{5.3}$$

All eNBs in the scenario have the same antenna model where, ρ is the vertical angle relative to the maximum gain direction, θ_{tilt} is the tilt angle, $HPBW_v$ is the vertical half power beam-width and SLL_h is the vertical side lobe level. Finally, the two gain components are added by,

$$G(\phi, \rho) = \max\{G_h(\phi) + G_v(\rho), SLL_0\} + G_0 \tag{5.4}$$

where SLL_0 is an overall side lobe floor and G_0 the antenna gain. We consider a cellular network, whose system performance has been evaluated on the ns-3 LENA platform based on 3GPP component LTE. The parameters used in the simulations and the learning parameters

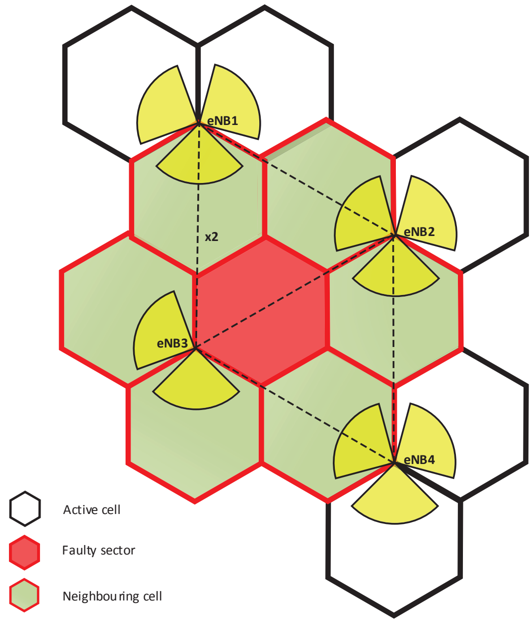

Figure 5.12: Scenario.

are given in Table 5.5. The macro cell scenario that we set up consists of 4 eNB with 3 sectors, which results in 12 cells, as depicted in Figure 5.12.

Therefore, each feasible solution corresponds to a vector of $M = 6$ tilt values (i.e., each tilt value is associated to one of the surrounding cells that are trying to fill the outage gap).

Simulation results

We analyse performance results obtained through the network planning tool described in Section 5.1. Figures 5.13, 5.15, and 5.16, show the fitness of the best individual found in each generation (Best), and the mean of the fitness values across the entire population (Mean).

Table 5.5: Case of study #2-Simulation parameters

Simulation parameters	**Value**
Propagation loss model	HybridBuildings
Shadow fading	Log-normal, std = 8 dB
Scheduler	PF
AMC model	LteAmc::MiErrorModel
Transport protocol	UDP
Traffic model	Constant bit rate
Layer link protocol	RLC
Mode	UM
Macro cell scenario	
Number of cells	12
eNB Tx power	46 dBm
LTE	
Cell layout	Radius: 500 m
Bandwidth	5 MHz
Number of RBs	25
TTI	1 ms
Antenna parameters	
Horizontal angle ϕ	-180° $\leq \phi \leq$ 180°
HPBW	Vertical 10°: Horizontal 70°
Antenna gain G_0	18 dBi
Vertical angle θ	-90° $\leq \phi \leq$ 90°
SLL	Vertical -18 dB : Horizontal -20 dB
SLL_0	-30 dB
GA	
Type	real-valued
Tilt values (min $-$ max)	0° $-$ 15°
Size of **S** ($psize$)	100
Elitism e	2
Mutation change δ	0.1
Bagged-SVM	
Num. of iteration (γ)	1000
Epsilon ϵ	0.1
Kernel	RBF
Simulation time	0.25 s

We observe that, in each generation, the population tends to get better as the generations proceed. Figure 5.13, depicts the time evolution of the PRB per offered Mb in the scenario during the evolution of the $GA_{\theta_{tilt}}$. We observe that as the $GA_{\theta_{tilt}}$ evolves, the efficiency of the planning increases. That is, as the generations proceed, the $GA_{\theta_{tilt}}$ finds the configuration parameters vector that minimize the value of the PRB per transmitted Mb.

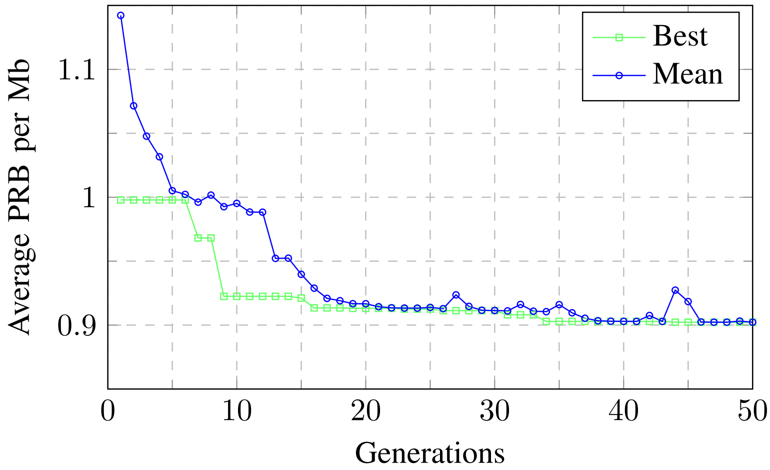

Figure 5.13: Evolution of the average PRB per Mb.

The performance to determine the capacity of the $GA_{\theta_{tilt}}$ is compared in Figure 5.14 to the self-organised RL based approach for COC proposed in [32], where in order to design the self-healing solution ($AC_{\theta_{tilt}}$), the antenna tilt is adjusted. The considered RL approach is an Actor Critic algorithm, which is already proven to outperform different solutions for COC available in literature in [32]. So we consider that the comparison to this benchmark is an exhaustive test for our approach.

Figure 5.14 depicts the CDF of the SINR of the network. We assume the user is out of outage when its SINR is above the threshold of -6 dB, as explained in [32]. Therefore, in this figure, we observe that the $AC_{\theta_{tilt}}$ is able to recover 95% of UEs, while the $GA_{\theta_{tilt}}$ is able to recover all the UEs. We observe in this figure, that in general the $GA_{\theta_{tilt}}$ tends to work effectively since it maintains the diversity in the values taken by the parameters, during the generations, and it makes a much deeper estimation of the state of the environment through the regression analysis approach than the $AC_{\theta_{tilt}}$ scheme, which considers only the SINR and CQI feedback from the UEs to determine the state of the environment and estimate the general behaviour of the network.

In order to analyse the QoS of the network in terms of the throughput, we analyse the simulations results considering the average throughput as the metric selected instead of the PRB per Mb of the neighbouring cells, as well as the whole network.

Figure 5.15, depicts the time evolution of the average throughput of the compensating sectors. That is, it shows the performance in terms of the throughput of the 6 neighbouring cells, which adjust their antenna tilt in order to compensate the faulty sector.

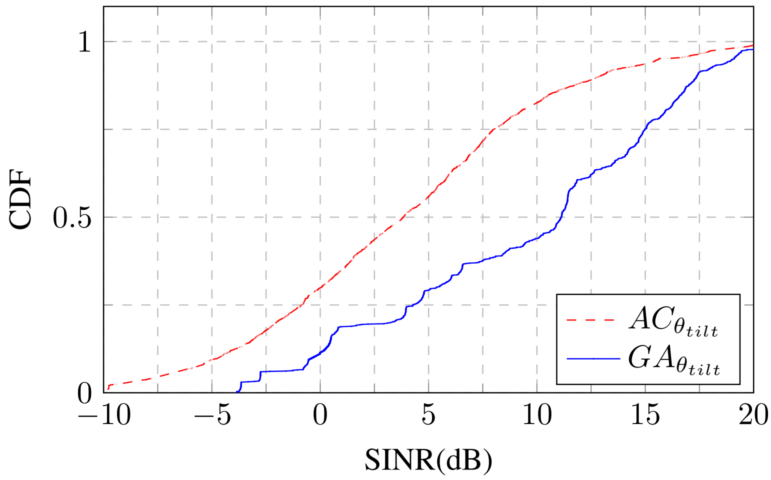

Figure 5.14: CDF of the average SINR values in the faulty sector by two different COC approaches. The GA based scheme is compared to a RL approach presented in [32].

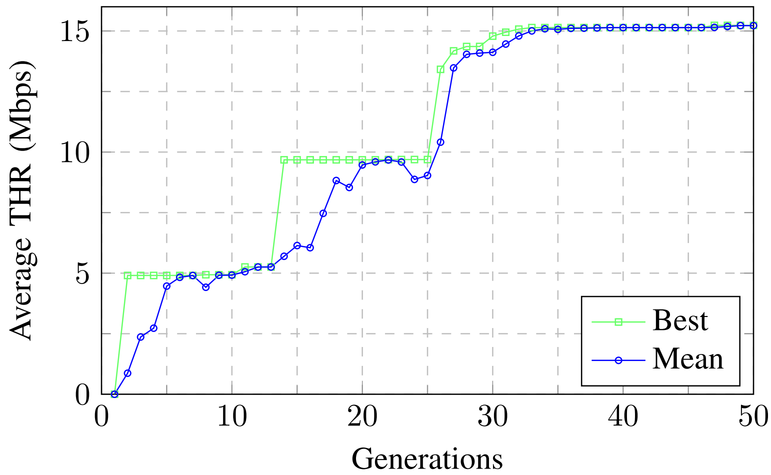

Figure 5.15: Evolution of the average throughput in the neighbouring cells.

Finally, Figure 5.16 describes the QoS of the whole network in terms of the average throughput. From this figure, we observe that the $GA_{\theta_{tilt}}$ scheme achieves at the 30-th generation 23 Mbps of average throughput in the whole network, while in the neighbouring cells (Figure 5.15), the achieved throughput is only 15 Mbps, which is reasonable, due to the challenging service conditions in this area.

From these figures, we observe that using the planning tool to optimise the antenna tilt parameter, we are able to automatic alleviating the outage caused by the loss of service from a faulty sector.

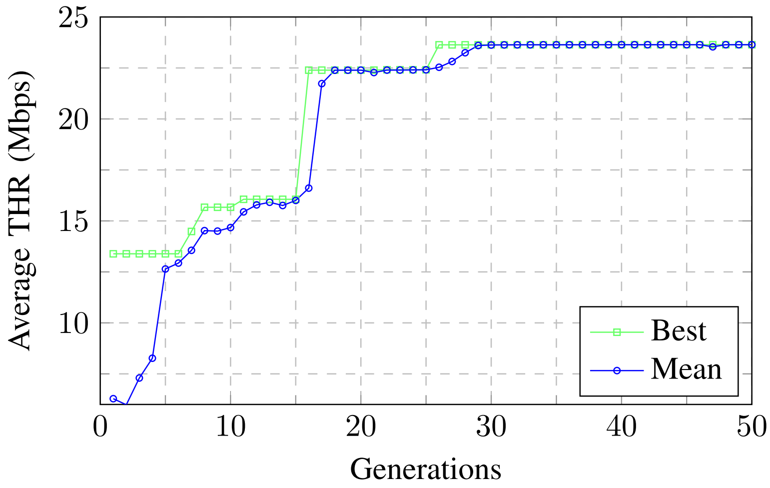

Figure 5.16: Evolution of the average throughput in the whole network.

5.4 Conclusions

In this chapter, we have defined a methodology and built a tool for smart and efficient network planning, based on QoS estimation derived by proper data analysis of UE measurements in the network. We collect UE measurements according to 3GPP MDT functionality. We apply proper regression analysis techniques to estimate the PRB per offered Mb, and based on the QoS that we can offer, we execute a GA_θ that over time increases the efficiency of resource allocation of the network.

We have applied our smart ML based planning tool to two different and independent use cases, 1) a small cell dense based scenario, and 2) a traditional macrocell deployment. For both cases our approach offers improved spectral efficiency to the operator and QoS to the end users. This demonstrates the efficiency of our proposed scheme to provide a design for multiple network planning challenges. The same technique can be applied to other scenarios identified by the operator. For the case of study $\#1$, results demonstrate the ability of the proposed scheme to deploy small cells in a network, in such a way that the QoS of the users is maximized and so the operator's spectral efficiency. Regarding the case of study $\#2$, results demonstrate the ability of the proposed scheme to compensate 100% of outage users in the scenario and provide them with service. A previously proposed RL scheme for the same scenario, where we apply intelligence in the configuration of parameters, but where we do not exploit the data available to properly estimate the QoS provided by the deployment, is demonstrated to recover only 95% of the users. In summary, we believe that the provided use cases demonstrate that the proper exploitation of data and experience through data analysis can bring a great added value to the operators.

Bibliography

[1] *3GPP on track to 5G*. URL: http://www.3gpp.org/news-events/3gpp-news/1787-ontrack_5g.

[2] 5GPPP. *The 5G Infrastructure Public Private Partnership*. URL: https://5g-ppp.eu/.

[3] Riverbed OPNET NetOne. URL: https://support.riverbed.com/content/support/software/steelcentral-npm/net-planner.html.

[4] Cariden MATE Design. *Adquired by CISCO*. URL: http://www.cisco.com/c/dam/en/us/td/docs/net_mgmt/wae/6-1/design/user/guide/MATE_Design_User_Guide.pdf.

[5] RSoft Design Group. *MetroWAND*. URL: https://optics.synopsys.com/rsoft/pdfs/RSoftProductCatalog.pdf.

[6] CelPlan. *CelTrace TM Wireless Global Solutions*. URL: http://www.celplan.com/products/indoor/celtrace.asp.

[7] Net2Plan. *The open-source network planner*. URL: http://www.net2plan.com/publications.php.

[8] Fedor Chernogorov and Timo Nihtilä. "QoS Verification for Minimization of Drive Tests in LTE Networks". *Proceedings of the 75th IEEE Vehicular Technology Conference, (VTC) Spring, Yokohama, Japan* (2012), pp. 6–9.

[9] Fedor Chernogorov and Jani Puttonen. "User Satisfaction Classification for Minimization of Drive Tests QoS verification". *24th IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communications, PIMRC 2013, London, United Kingdom* (2013), pp. 2165–2169.

[10] Jessica Moysen, Lorenza Giupponi, Nicola Baldo, Josep Mangues-Bafalluy. "Predicting QoS in LTE HetNets based on location-independent UE measurement". *in proc. of 20th IEEE Computer Aided Modelling and Design of Communication Links and Networks, Guildford, UK* (2015).

[11] Jessica Moysen, Lorenza Giupponi, Josep Mangues-Bafalluy. "On the Potential of Ensemble Regression Techniques for Future Mobile Network Planning". *in proc. of the Twenty-First IEEE Symposium on Computers and Communications (ISCC), Messina, Italy* (2016).

[12] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1989.

[13] 3GPP. *Radio performance and protocol aspects (system) - RF parameters and BS conformance*. Tech. rep. TSG RAN WG4 R4-092042. 2009.

[14] Nicola Baldo, Lorenza Giupponi, Josep Mangues-Bafalluy. "Big Data Empowered Self Organized Networks". *20th European Wireless (EW) Conference; Barcelona, Spain* (2014).

[15] Mitchell, Melanie. "An Introduction to Genetic Algorithms". *Cambridge, MA, MIT Press* (1996).

[16] Johansson, J., Hapsari, W.A., Kelley, S., Bodog, G. "Minimization of drive tests in 3GPP (Release 11)". *IEEE Communications Magazine* (2012).

[17] S. T. Roweis. "EM algorithms for PCA and SPCA". *Advances in Neural Information Processing Systems, The MIT Press* (1998).

[18] Altman, N. S. "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression". *The American Statistician 46* (1992).

[19] Bishop M.C. "Pattern Recognition and Machine Learning". *Busines Dia, Llc. Springer Science* (2006).

[20] A. J. Smola and B. Scholkopf. "A tutorial on support vector regression". *Statistics and Computing* (2004), pp. 199–222.

[21] Quinlan, J. R. "Induction of Decision Trees. Machine Learning ". *Kluwer Academic Publishers* (1986), pp. 81–106.

[22] Rokach, Lior, Maimon, O. "Data Mining with Decision Trees: theory and applications". *World Scientific Pub Co Inc.* (2008).

[23] Opitz, D., Maclin, R. "Popular Ensemble Methods: An Empirical Study". *Journal of Artificial Intelligence Research 11* (1999), pp. 169–198.

[24] Rokach, L. "Ensemble-based classifiers". *Artificial Intelligence* (2010), pp. 1–39.

[25] T. Dietterich. "An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting and Randomization". *Machine Learning* 40 (2000), pp. 139–157.

[26] Thomas G. Dietterich. "Machine-Learning Research: Four Current Directions". *American Association for Artificial Intelligence* (1997).

[27] Efrain Mayhua-Lopez, Vanessa Gomez-Verdejo, and Anibal R. Figueiras-Vidal. "A new boosting design of Support Vector Machine classifiers". *Information Fusion* 25 (2015), pp. 63–71.

[28] O. Onireti, A. Zoha, J. Moysen, A. Imran, L. Giupponi, M. Ali Imran, A. Abu-Dayya. "A Cell Outage Management Framework for Dense Heterogeneous Networks , IEEE Transactions on Vehicular Technology". *IEEE Transactions on Vehicular Technology* 64 (2015), pp. 2097–2113.

[29] Hazewinkel Michiel. "Greedy algorithm". *Encyclopedia of Mathematics, Springer* (2001).

[30] Jessica Moysen, Lorenza Giupponi, Josep Mangues-Bafalluy. "A Machine Learning enabled network Planning tool". *in proc. of The Twenty-Seven IEEE Personal Indoor and Mobile Radio Communications (PIMRC), Valencia, Spain* (2016).

[31] Fredrik Gunnarsson, Martin N. Johansson, Anders Furuskar, Magnus Lundevall, Arne Simonsson, Claes Tidestav and Mats Blomgren. *VTC Fall'08* (2008), pp. 1–5.

[32] J. Moysen and L. Giupponi. "A Reinforcement Learning based solution for Self-Healing in LTE networks". *in proc. of the 80th IEEE Vehicular Technology Conference (VTC Fall), Vancouver, Canada* (2014).

Chapter 6

Conclusions and future work

The future depends on what we do in the present. **Mahatma Gandhi**

Contents

This chapter completes the dissertation by summarizing our main contributions, while also providing some challenges for future investigation. In particular, Section 6.1, contains the most significant concluding remarks of each chapter, while Section 6.2, outlines the open research issues related to our contributions.

6.1 Concluding remarks

This dissertation has dealt with the open challenges of the SON problem. We have discussed the state of the art, and then focused on the aspects of SON, which, after the deep analysis carried out in chapter 2, we believe were still open for interesting research contributions. In our attempt to make networks self-aware, self-adaptive and intelligent, machine learning has been proven a valuable and appropriate tool. The major contributions of the thesis has been divided into two main parts: The first one includes Chapters 3 and 4, and the second one is reported in Chapter 5. The detailed contributions of each chapter are summarized in the following:

Chapter 3 In this chapter, we introduce the COC, to study the self-healing problem. The proposed scheme gives a solution based on reinforcement learning to compensate the huge coverage and capacity gap generated by a sector fault. Employing the ns3 LTE-EPC network simulator (LENA) platform, we have been able to evaluate the COC performance. In particular, we showed that the proposed algorithm has the ability to compensate 98% of outage users and to provide them with service. The self-healing solution has been compared in [1] to other state of the art COC solutions, and it outperforms them. As a result we consider that the proposed comparative results allow to make quite stable conclusions about the usefulness and effectiveness of reinforcement learning schemes to recover the network from network faults, by the gradual adjustment of neighbour nodes configuration parameters.

Chapter 4 In this chapter, we present a functional architecture and a theoretical framework based on the theory of MDPs to model and address the self-coordination problem of multiple interfering SON functions. Since the global problem consisting of all the different SON use cases and functions generates a too complex global MDP, we have proposed that the different SON functions are modelled by means of a sub-MDP, which are solved by means of RL through the theory of TD learning. This theoretical scheme is in line with the approach considered by 3GPP and industry in the area of SON. We have focused on two uses cases, the ICIC and the CCO, which generate a resource conflict, as both functions aim at modifying the transmission power to achieve their objectives. The two SON functions are solved by means of an actor critic scheme, while the coordination among functions is solved through a coordination game. The coordination game is guaranteed to have mixed strategy equilibria, which allows the self-coordinator framework to find a common solution for the conflicting SON functions, which achieves a trade off among their objectives. This solution, characterized by low complexity and execution costs, is compared to an artificial intelligence benchmark based on a policy iteration approach, which achieves a long run optimisation, but is more complex and difficult to implement in a real system. Furthermore, we provide an analysis about how the proposed functional architecture can be used to deal with the conflicts generated by the concurrent execution of multiple SON functions. We show that the proposed approach is general enough to model all the SON functions, and their derived conflicts.

Chapter 5 This chapter introduced a smart planning tool to deploy self-adaptive networks. Motivated by the huge amount of data generated in current 4G networks and more to come in 5G, we have considered the problem for QoS prediction, estimation and verification in dense heterogeneous network deployments. We have proposed a network planning tool, based on the prediction of the QoS offered to the users is able to adjust the network configuration parameters in order to meet certain targets. The contribution focuses on improving the network planning based on MDT and QoS prediction in terms of the Physical Resource Block (PRB) per Megabit (Mb) offered in an arbitrary point of the network. In this chapter we have proven that the use of ensemble methods for regression analysis allows to predict the network performance in terms of QoS offered to end-users, independently of their geographic position, and only based on measurements available in data bases defined by 3GPP. In addition, we have shown that that the dimension of the input space has an impact in the error. In this context, we have studied different techniques to apply dimensional reduction to the input space. By integrating these techniques in a network planning tool, operators are capable of finding the most appropriate deployment layout, but also, they can reduced the resources (i.e., the cost) they need to offer a given QoS in a newly planned deployment, and/or an arbitrary point of the network. We have analysed this approach in different deployment use cases, validating its effectiveness.

6.2 Challenges and future work

Despite the progress of SON during the last years, we have seen in chapter 2 that still many aspects remain open and we have tackled some of them in this thesis. In particular, we have tried to bring intelligence and self-awareness in the network by adding ML to the most significant use cases, that we consider are still open of research contributions. From the work presented in this dissertation, we observe, that once engineers and researchers acquire expertise in ML to empower SONs, there are multiple challenges preventing this research line to proceed. In what we follows we summarize the most important open points that should be tackled in future works.

6.2.1 How to get real data

A very important aspect, when dealing with data, is how to get real network data to analyse and get experience from. To the best of the authors' knowledge it is possible to find databases related to signals and coverage data [2, 3], by using/designing applications that collect information such as RSRP and RSRQ. However, it is not easy to find contributions analysing real network data. Some work can be found in the context of 3G networks, but currently, in the context of 4G networks it is very hard to find works considering real data [4, 5]. In this dissertation, it has been proven that an alternative to real data, could be to use a high-fidelity network simulators like ns-3 LTE/LENA module, to generate realistic data [6]. This simulator has been built around industrial Application Programming Interface (API) defined by the small cell forum and offers high-fidelity models from Media Access Control (MAC) to application layers. It has also been designed with the requirement to simulate tens of eNBs and hundreds of UEs, and to specifically test RRM and SON algorithms. Consequently it could be a very useful tool to generate data to analyse, build algorithms based on this analysis and close the loop on the simulator to test the designed algorithms. In this context, it is also hard to find contributions where machine learning approaches are used not only in network simulators, but in real networking products. In general, it seems vendors are reluctant to test algorithms whose behaviour is not completely predictable. We are aware of some works carried out [7]. However, these approaches are quite simple and do not exploit enough the real potential of machine learning tools. We consider that it is extremely important for this research line to get to the next level, to get access to operator's network data. An important research line is then, the one able to find or generate meaningful network data, and find patterns in them to understand aspects that should be optimized in the network.

6.2.2 Theoretical research for decision making process

With respect to more theoretical research, and specifically to control decision problems that allow to continuously take RRM/SON decisions, in chapters 3 and 4, we have proposed an approach based on reinforcement learning to solve this problem. However, this kind of algorithms require a considerable amount of time before finding a solution, and it increases with the state and action spaces. So, reinforcement learning approaches dealing with these issues have to be investigated. Even though ML literature offers different algorithms that can find interesting solutions (e.g., NashQ [8]), the space of possible solutions is so big that this kind of approaches is not feasible to be used in a realistic network where the time constraints of RRM/SON problem have to be met. So, more research in the area of multi-agent systems, which are also compatible with real network requirements need to be investigated.

6.2.3 SON for multi-technologies and virtualised architectures

The work of SON related to other technologies, such as WiFi, or the combinations of the automation of heterogeneous networks including multiple different RATs, or different layers of the network, e.g., WiFi, mm wave, mobile network layer, transport layer, among others, is still immature. ML has still not been exploited to handle these networks with intelligence and self-awareness. In particular, the management of densified and heterogeneous, in both technologies and layers, architectures, requires the evolution of complex SON concepts which have traditionally been designed and standardized for LTE based networks. Before reaching this vision, multiple challenges need to be addressed, e.g., the self-coordination problem and the solution of conflicts among SON functions executed in different nodes, or networks, which put the network at risk of instability, or the most appropriate location of SON functions and algorithms, to solve properly the distributed vs. centralized SON implementation issue. Many aspects have to be considered when locating and designing a SON function, e.g., response time, complexity, size of data bases, computational capability of nodes, etc. Centralized (i.e., a large number of cells is involved), distributed (approx. 2 cells are involved, coordinating through X2) and local (only one cell is involved) implementations of SON functions have been proposed. No architecture can be claimed superior to the other. The growing complexity, dynamicity, and heterogeneity of 5G networks will substantially increase the number of scenarios to solve. So, there is the need for exploiting their complementarity by virtualising and dynamically deploying them.

To benefit of all the opportunities offered by centralized, distributed and local implementations, there is the need to further study SON over NFV architecture, where SON functions, aimed at tackling the main radio access and backhauling challenges of extremely dense deployments, are virtualised and run over generic purpose hardware. The NFV infrastructure is to be managed by an orchestrator entity (in coordination with the corresponding virtual network function and virtual infrastructure managers), as proposed in European Telecommunications Standards Institute (ETSI) architecture. Out of all the NFV architecture entities, this is the brain with the broadest view of the service characteristics and the resource availability in the network. Therefore, it coordinates the allocation of functions across the different segments of the dense, heterogeneous network.

Bibliography

[1] J. Moysen and L. Giupponi. “A Reinforcement Learning based solution for Self-Healing in LTE networks”. *in proc. of the 80th IEEE Vehicular Technology Conference (VTC Fall), Vancouver, Canada* (2014).

[2] Qualcomm. *OpenSignal App*. URL: `http://opensignal.com/`.

[3] Orange. *The D4D Challenge*. URL: `http://www.d4d.orange.com/en/Accueil`.

[4] Fabio Ricciato, Peter Widhalm, Massimo Craglia and Francesco Pantisano. “Estimating population density distribution from network-based mobile phone data” (2015).

[5] M. Laner, P. Svoboda, P. Romirer, N. Nikaein, F. Ricciato. "A Comparison Between One-Way Delays in Operating HSPA and LTE Networks". *8th Int'l workshop on Wireless Network Measurements (WINMEE'12), Pedeborn, Germany* (2012).

[6] Centre Tecnològic de Telecomunicacions de Catalunya (CTTC). *The LENA-EPC Network simulator*. URL: `http://networks.cttc.es/mobile-networks/software-tools/lena/`.

[7] Emil J. Khatib, Raquel Barco, Ana Gómez-Andrades, Pablo Muñoz, Inmaculada Serrano. "Data mining for fuzzy diagnosis systems in LTE networks". *Expert Systems with Applications, Elsevier* (2015).

[8] Junling Hu and Michael P. Wellman. "Nash Q-Learning for General-Sum Stochastic Games". *Journal of Machine Learning Research* 4 (2003), pp. 1039–1069.

Part I

Acronyms

List of Acronyms

AAS	Active Antenna Systems
ABS	Almost Blank Subframes
AC	Actor Critic
ANN	Artificial Neural Networks
ANR	Automatic Neighbour Relation
AP	Access Point
API	Application Programming Interface
BER	Bit Error Rate
BLER	Block Error Rate
BS	Base Station
CAPEX	Capital Expenditure
CCO	Coverage and Capacity Optimisation
CDF	Cumulative Distribution Function
CDR	Charging Data Records
CESC	Cloud-Enabled Small Cell
CET	Changes Electrical Tilt
CG	Coordination Game

CIO	Cell Individual Offset
COC	Cell Outage Compensation
COD	Cell Outage Detection
COM	Cell Outage Management
CoMP	Coordinated Multi Points
CQI	Channel Quality Indicator
CTTC	Centre Tecnològic de Telecommunications de Catalunya
CRF	Conditional Random Fields
C-SON	Centralized SON
DBM	Deep Boltzmann Machine
DBN	Deep Belief Network
DNN	Deep Neural Network
DCI	Data Control Indication
DL	Downlink
DP	Dynamic Programming
DSA	Dynamic Spectrum Access
D-SON	Distributed SON
DT	Decision Tree
E3	End-to-End Efficiency
EIRP	Equivalent Isotropically Radiated Power
EM	Ensemble Methods
eNB	Enhanced Node Base station
EPC	Evolved Packet Core
ES	Energy Saving

ETL	Extract-Transform-Load
ETSI	European Telecommunications Standards Institute
ETRI	Electronics and Telecommunications Research Institute
E-UTRAN	Evolved Universal Terrestrial Radio Access Network
FE	Feature Extraction
FFR	Fractional Frequency Reuse
FL	Fuzzy Logic
FS	Feature Selection
GA	Genetic Algorithm
GANDALF	Monitoring and Self-tuning of RRM parameters in a Multi-System Network
GBR	Guaranteed Bit Rate
GLM	Generalized Linear Models
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
GT	Game Theory
3GPP	3rd Generation Partnership Project
5GPPP	5G Infrastructure Public Private Partnership
HeNB	Home eNodeB
HetNet	Heterogeneous Network
HII	High Interference Indicator
HMM	Hidden Markov Models
HO	Handover
ICIC	Inter-Cell Interference Coordination
IEEE	Institute of Electrical and Electronics Engineers

IMS	IP Multimedia Subsystem
IoT	Internet of Things
IRAT	Inter-Radio Access Technology
IRP	Integration Reference Point
IS	Information Service
IWF	Iterative Water-Filling
k**-NN**	k-Nearest Neighbours
KPI	Key Performance Indicator
LAA	Licensed Assisted Access
LB	Load Balancing
LENA	LTE-EPC Network Simulator
LTE	Long Term Evolution
LTE-A	LTE-Advanced
LTE-U	LTE-Unlicensed
MAC	Media Access Control
M2M	Machine to Machine
Mb	Megabit
MC	Monte Carlo
MCS	Modulation and Coding Scheme
MDP	Markov Decision Process
MDT	Minimization of Drive Tests
MIMO	Multiple-input Multiple-output
ML	Machine Learning
MLB	Mobility Load Balancing

MRO	Mobility Robustness Optimisation
MWC	Mobile World Congress
NB	Naive Bayes
NE	Network Element
NFV	Network Functions Virtualisation
NGMN	Next Generation Mobile Networks
NM	Network Management
NMS	Network Management Systems
NN	Neural Networks
NRMSE	Normalized Root Mean Squared Error
NRM	Network Resource Model
OAM	Operations, Administration, and Maintenance
OFDMA	Orthogonal Frequency Division Multiple Access
OI	Overload Indicator
OMC	Operation and Maintenance Center
OPEX	Operational Expenditure
OSS	Operation and Support System
PC	Principal Component
Pc	Power control
PCA	Principal Component Analysis
PCI	Physical Cell ID
PDF	Probability Density Function
PDCCH	Physical Downlink Control Channel
PDSCH	Physical Downlink Shared Channel

PDU	Protocol Data Unit
PF	Proportional Fair Scheduler
PHR	Power Headroom Report
PItoRC	Policy Iteration to Resource Conflicts
PM	Performance Management
PRB	Physical Resource Block
PS	Packet Switched
PSD	Power Spectral Density
PUSCH	Physical Uplink Shared Channel
QAM	Quadrature Amplitude Modulation
QoE	Quality of Experience
QoS	Quality of Service
RACH	Random Access Channel
RAN	Radio Access Network
RAT	Radio Access Technologies
RB	Resource Block
RBF	Radial Basis Function
RBG	Resource Block Group
RBM	Restricted Boltzmann Machine
REM	Radio Environment Map
RET	Remote Electrical Tilt
RF	Random Forest
RL	Reinforcement Learning
RLC	Radio Link Control

RLF	Radio Link Failure
RMSE	Root Mean Squared Error
RNTP	Relative Narrowband Transmit Power
RRC	Radio Resource Control
RRM	Radio Resource Management
RS	Reference Signal
RSRP	Reference Symbol Received Power
RSRQ	Reference Symbol Received Quality
RSSI	Reference Signal Strength Indication
SDN	Software Defined Network
SG	Stochastic Games
SGSN	Serving GPRS Support Node
SH	Self Healing
SIMO	Single-input Multiple-output
SINR	Signal to Interference and Noise Ratio
SL	Supervised Learning
SLA	Service Level Agreement
SM	Saturation Mode
SO	Self-Organisation
SOCRATES	Self-Optimisation and self-ConfiguRATion in wirelEss networkS
SOFOCLES	Self-Organised FemtOCeLls for broadband sErviceS
SOM	Self Organising Map
SON	Self-Organising Network
SOS	Self Organised System

SPCA Sparse Principal Component Analysis

SS Solution Sets

sub-MDP Markov decision sub-processes

SVM Support Vector Machine

SVR Support Vector Regression

TA Timing Advance

TBS Transport Block Size

TCE Trace Collection Entity

TCP Transmission Control Protocol

TD Temporal Difference

TTI Transmission Time Interval

TTT Time to Trigger

TXP Transmission Power

UDN Ultra Dense Network

UDP User Datagram Protocol

UE User Equipment

UL Unsupervised Learning

UM Unacknowledged Mode

UMTS Universal Mobile Telecommunications System

UTRAN UMTS Universal Terrestrial Radio Access Network

WCQI Wide-band Channel Quality Indicator

WLAN Wireless Local Area Network