

June 2020

# Next-Generation Self-Organizing Communications Networks: Synergistic Application of Machine Learning and User-Centric Technologies

Chetana V. Murudkar  
*University of South Florida*

Follow this and additional works at: <https://digitalcommons.usf.edu/etd>



Part of the [Artificial Intelligence and Robotics Commons](#), and the [Electrical and Computer Engineering Commons](#)

---

## Scholar Commons Citation

Murudkar, Chetana V., "Next-Generation Self-Organizing Communications Networks: Synergistic Application of Machine Learning and User-Centric Technologies" (2020). *USF Tampa Graduate Theses and Dissertations*.

<https://digitalcommons.usf.edu/etd/8973>

This Dissertation is brought to you for free and open access by the USF Graduate Theses and Dissertations at Digital Commons @ University of South Florida. It has been accepted for inclusion in USF Tampa Graduate Theses and Dissertations by an authorized administrator of Digital Commons @ University of South Florida. For more information, please contact [digitalcommons@usf.edu](mailto:digitalcommons@usf.edu).

Next-Generation Self-Organizing Communications Networks:  
Synergistic Application of Machine Learning and User-Centric Technologies

by

Chetana V. Murudkar

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
Department of Electrical Engineering  
College of Engineering  
University of South Florida

Co-Major Professor: Richard D. Gitlin, Sc.D.  
Co-Major Professor: Kwang-Cheng Chen, Ph.D.  
Nasir Ghani, Ph.D.  
Srinivas Katkoori, Ph.D.  
Gabriel Arrobo, Ph.D.

Date of Approval:  
March 30, 2020

Keywords: 5G and 6G Networks, Anomaly Detection, Quality of Experience,  
Load Balancing and Capacity Optimization, RAN-Based Notification Area Configuration

Copyright © 2020, Chetana V. Murudkar

## **Dedication**

To my family, friends, and well-wishers for embarking on this journey with me.....  
and making it through!

## **Acknowledgments**

I am immensely grateful to my advisor Dr. Richard D. Gitlin for being my toughest challenger and yet my greatest supporter throughout this accelerating doctoral journey and giving me the wings to take a leap of faith towards meeting my goals for the Ph.D. and more. I couldn't have asked for a better mentor whose teachings have showed me how to tune into the right frequencies of not only the wireless world of communications networks but also several other aspects of life that he has helped me explore and expand my ken.

I am very grateful to my co-advisor Dr. Kwang-Cheng Chen for providing his guidance and support and sharing his wisdom with me. I would like to express my sincere gratitude to Dr. Nasir Ghani, Dr. Srinivas Katkoori, and Dr. Gabriel Arrobo for serving in my doctoral committee and sharing their valuable advice and feedback to strengthen my research work.

I am thankful to the University of South Florida (USF) for giving me the opportunity to pursue my Ph.D. I am thankful to my employer, Sprint (New T-Mobile), for backing me up in my decision to moonlight. I am pleased to be affiliated with iWINLAB (Innovations in Wireless Information Networking Lab) and the Department of Electrical Engineering at USF.

I highly appreciate and thank everyone who inspired, motivated, and encouraged me. I feel blessed to receive the unconditional love and continued faith of my family and friends. I am thankful to the past for enriching me with experiences, to the present for making me who I am today, and to the future for inculcating the spirit and curiosity in me to keep looking forward towards exploring on what's coming next!

## Table of Contents

List of Tables .....	iii
List of Figures .....	iv
Abstract .....	vi
Chapter 1. Introduction .....	1
1.1. Research Motivation .....	1
1.2. Research Objective and Initiatives .....	3
1.3. Contributions and Organization of this Dissertation .....	5
Chapter 2. Literature Review .....	6
2.1. Introduction .....	6
2.2. Self-Organizing Networks .....	6
2.3. Machine Learning .....	16
2.4. User-Centric Technology .....	19
2.5. Simulation Toolkit .....	20
2.6. Concluding Remarks .....	21
Chapter 3. QoE-Driven Anomaly Detection in Self-Organizing Networks Using Machine Learning .....	22
3.1. Introduction .....	22
3.2. Anomaly Detection .....	23
3.3. Quality of Experience (QoE) .....	24
3.4. The Methodology .....	25
3.5. The ML Algorithms .....	28
3.5.1 Support Vector Machines .....	28
3.5.2 $k$ -Nearest Neighbor Algorithm .....	29
3.5.3 Decision Tree Methods .....	31
3.5.4 Neural Network .....	33
3.6. Performance Analysis and Evaluation .....	34
3.7. Concluding Remarks .....	39
Chapter 4. Optimal-Capacity, Shortest Path Routing in Self-Organizing Networks Using Machine Learning .....	41
4.1. Introduction .....	41
4.2. Load Balancing and Capacity Optimization .....	42

4.3. The Methodology.....	43
4.4. Performance Analysis and Evaluation.....	52
4.5. Concluding Remarks.....	55
Chapter 5. Self-Configuration of Radio Access Network-Based Notification Areas (RNAs) in Self-Organizing Networks Using Machine Learning .....	56
5.1. Introduction.....	56
5.2. RRC State Handling and Transitions.....	57
5.3. Key RNA Configuration Factors .....	59
5.4. Performance Analysis and Evaluation.....	62
5.5. Future Research Directions.....	72
5.6. Concluding Remarks.....	75
Chapter 6. Summary .....	77
6.1. Summary of the Main Contributions .....	77
6.2. Concluding Summary .....	80
References.....	83
Appendix A: Copyright Permissions .....	87
Appendix B: Abbreviations .....	91
Appendix C: Glossary.....	95
About the Author .....	END PAGE

## List of Tables

Table 3.1 Simulation parameters and values .....	35
Table 4.1 $Q$ -learning algorithm, assuming deterministic rewards and actions .....	48
Table 4.2 A correspondence table of the network mapping in US-OCSP with $Q$ -learning parameters .....	50
Table 4.3 Simulation set up parameters .....	52
Table 5.1 Characteristic differences and similarities of the 5G RRC states .....	59
Table 5.2 Simulation parameters and values .....	65
Table 5.3 Key performance indicators for RNA cluster performance monitoring and optimization .....	73

## List of Figures

Figure 1.1 The three-layered approach for the development of next-generation communications networks .....	3
Figure 1.2 The process flow for the research initiatives.....	4
Figure 2.1 SON structures based on the location of the SON algorithms .....	8
Figure 2.2 SON framework .....	9
Figure 2.3 C-SON view .....	9
Figure 2.4 D-SON view .....	10
Figure 2.5 H-SON view .....	11
Figure 2.6 5G network architecture embedded within the SON framework .....	12
Figure 2.7 Management data analytics service and SON functions .....	13
Figure 2.8 Taxonomy of self-organizing networks.....	16
Figure 2.9 Basic structures of ML categories .....	18
Figure 2.10 Classification of ML algorithms utilized in this dissertation .....	19
Figure 3.1 The process flow for QoE-driven anomaly detection in SON using ML.....	27
Figure 3.2 One hidden layer MLP .....	33
Figure 3.3 Training and testing accuracy scores for the ML algorithms implemented for QoE prediction .....	36
Figure 3.4 QoE prediction accuracy with different RF propagation models.....	37
Figure 3.5 The average accuracy results for QoE predictions against multiple independent simulation runs .....	38
Figure 4.1 Example network topology to illustrate the US-OCSP methodology.....	45



Figure 4.2 Representation of a reinforcement learning system .....	47
Figure 4.3 A graphical representation of the $Q$ -learning curve converging towards the optimal solution. ....	54
Figure 5.1 The RRC state transitions in a 5G network .....	58
Figure 5.2 RAN-initiated paging procedure for a 5G network.....	60
Figure 5.3 CN-initiated paging procedure for a 5G network.....	61
Figure 5.4 Process flow diagram for the demonstration and evaluation of the proposed RNA clustering mechanism .....	63
Figure 5.5 The network model used for clustering (The arrows represent the connectivity status of the mobile UEs with the network nodes as they move around in the network.).....	66
Figure 5.6 Performance evaluation for the selection of $kc$ .....	69
Figure 5.7 Resulting RNA clusters for the simulated network model.....	72
Figure 5.8 The proposed RNA configuration and management framework .....	74
Figure 6.1 Basic building blocks providing a high-level framework to deploy next-generation SON functions and use-cases with the synergistic application of ML and UC technologies.....	81

## **Abstract**

The telecommunications industry is going through a metamorphic journey where the 5G and 6G technologies will be deeply rooted in the society forever altering how people access and use information. In support of this transformation, this dissertation proposes a fundamental paradigm shift in the design, performance assessment, and optimization of wireless communications networks developing the next-generation self-organizing communications networks with the synergistic application of machine learning and user-centric technologies.

This dissertation gives an overview of the concept of self-organizing networks (SONs), provides insight into the “hot” technology of machine learning (ML), and offers an intuitive understanding of the user-centric (UC) technology that form the foundation of the research initiatives conceived, implemented, and validated in this dissertation. A three-layered approach based on the synergistic application of SON, ML, and UC technologies is applied for anomaly detection, load balancing and capacity optimization, and radio access network-based notification area (RNA) configuration and management.

In the first research initiative, ML is applied to learn and predict a UC key performance indicator that imports the effect of the end-user perception of the quality of service to achieve end-to-end service assurance and proactively detect dysfunctional network nodes enabling automatic detection and remediation of failing network nodes to mitigate network degradation in self-healing SON systems.

In the second research initiative a UC and ML based methodology called US-OCSP (i.e. user-specific optimal capacity and shortest path) is developed that can be integrated with an auto or personal navigation system to provide routing that avoids congested network traffic and effects resource optimization enabling load balancing and capacity optimization in self-optimizing SON systems.

In the third research initiative, a UC and ML-embedded clustering mechanism is developed for dynamic configuration and management of RAN-based notification areas (i.e. RAN-based paging areas) that can help achieve improved signaling and paging load to attain reduced latency and improved network capacity, while lowering power consumption supporting emerging 5G/6G applications and services that generate an extensive amount of random aperiodic and keep-alive data traffic in self-configuring SON systems.

Finally, a high-level framework consisting of several core building blocks is provided to support UC and ML-infused network standardization that the network operators can adopt to shape the network of tomorrow.

## **Chapter 1. Introduction**

The rapid increase of the capabilities of communications networks increases the likelihood of realizing the vision of a ubiquitously interconnected society, albeit in a virtual sense. In particular, the stunning evolution/revolution of wireless communications networks from the First Generation (1G) to the current initial deployment of the Fifth Generation (5G), and with high expectation of future networks, such as 6G, promises the fulfillment of this vision. Every generation witnessed big leaps of technological advancements from analog in 1G to digital in 2G to IP-based broadband in 3G to mobile packet-mode broadband in 4G to enhanced mobile broadband, massive machine-type communications, and ultra-reliable and low latency communications in 5G. In the 5G and beyond 5G world, the expectations and goals have tremendously increased targeting never before achieved network dimensions that will allow communicating immense amount of data at virtual zero latencies and unprecedented throughput supporting data-intensive applications or services with gigabit wireless and pervasive connectivity enabled via extensive cognitive capabilities.

### **1.1. Research Motivation**

History has shown that the mobile industry undergoes a (managed) major technology shift roughly once every decade and as network traffic growth continues to grow given the vast arrays of technology developments on the horizon such as enhanced mobile broadband, tactile internet, vehicle-to-everything, massive spatial processing [e.g. massive MIMO], etc., network operators

will need more and more innovative functionalities in their network to fully realize the performance and application targets [1]. 5G/6G networks are more complex than previous generations and will greatly benefit from automation via enhanced self-organizing network (SON) mechanisms to support network densification, co-existence of multiple radio access mobile networks, end-to-end service assurance, dynamic network optimization, and minimization of operational and capital expenditures. It is a central theme of this dissertation that SON platforms based on machine learning (ML) can leverage the extensive amount of data generated across the network, identify patterns and correlations, and make efficient decisions to better address network performance challenges and create new opportunities for data monetization. To increase revenue, it is critical that the end-users receive excellent service, and this can be achieved by developing user-centric (UC) technologies, where users are no longer mere end-points but rather are an integral and active part of the network such that the network strategies and solutions are tailored as per user needs and timely feedback.

Given the importance of the above-mentioned concepts towards developing the next-generation communications networks via enhanced automation, enhanced intelligence, and enhanced user experience, the research in this dissertation proposes a fundamental paradigm shift in the design, performance, and optimization of communications networks by developing novel network strategies for user-centric, machine learning-based self-organizing 5G/6G and beyond wireless communications network evolution. A three-layered approach is taken towards developing such next-generation communications networks by the synergistic integration of SON, ML, and UC technologies as depicted in Figure 1.1.

It is noted that the proposed approach generally applies to both wireless and wireline networks; however, the focus of this dissertation is on wireless networks.

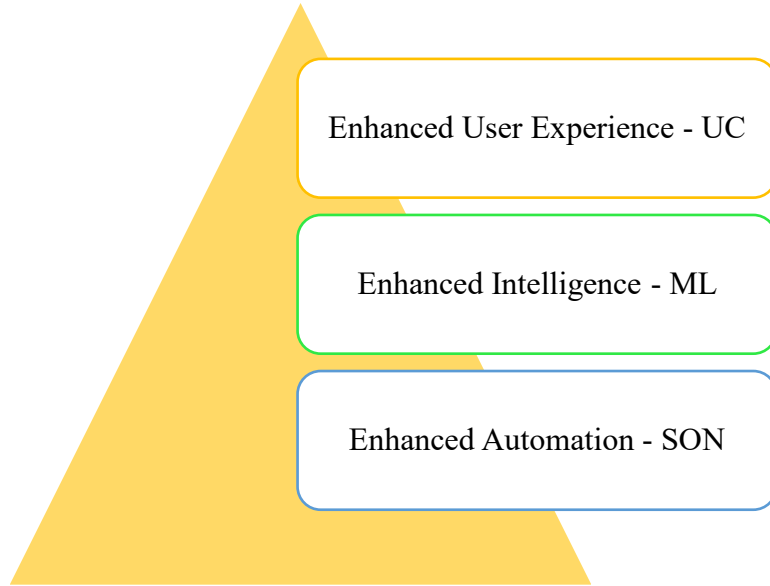


Figure 1.1 The three-layered approach for the development of next-generation communications networks

## 1.2. Research Objective and Initiatives

The research in this dissertation is directed towards developing innovative technologies for application in next-generation self-organizing wireless communications networks that are based on the synergistic application of machine learning and user-centric technologies.

The research initiatives include a user-centric anomaly detection methodology for self-organizing networks using machine learning that learns and predicts a user-centric key performance indicator, quality of experience (QoE), using machine learning to detect dysfunctional network nodes (e.g. base stations). In the next research initiative, a user-centric optimal capacity shortest path routing technique for load balancing and capacity optimization in self-organizing networks using machine learning is developed. The proposed methodology is called user-specific optimal capacity and shortest path (US-OCSP) where the optimization begins at the end-user level to alleviate network congestion and improve user experience by recommending a tailored path between the end-user's source and destination. The next research

initiative is a user-centric, adaptive mechanism for self-configuration of radio access network-based notification areas (RNAs) in 5G networks using machine learning where RNA clusters are formed by applying machine learning to user-centric network data.

The process flow followed to demonstrate and evaluate the performance of the research initiatives is described in Figure 1.2. The process begins by simulating a cellular network to test a particular SON functionality followed by the collection of network measurements and statistics. Data preprocessing is performed to apply a UC methodology and generate the input dataset for ML followed by the implementation of a suitable ML algorithm. Once the final output is obtained, its performance is evaluated to validate the accuracy and understand the values and benefits attained.

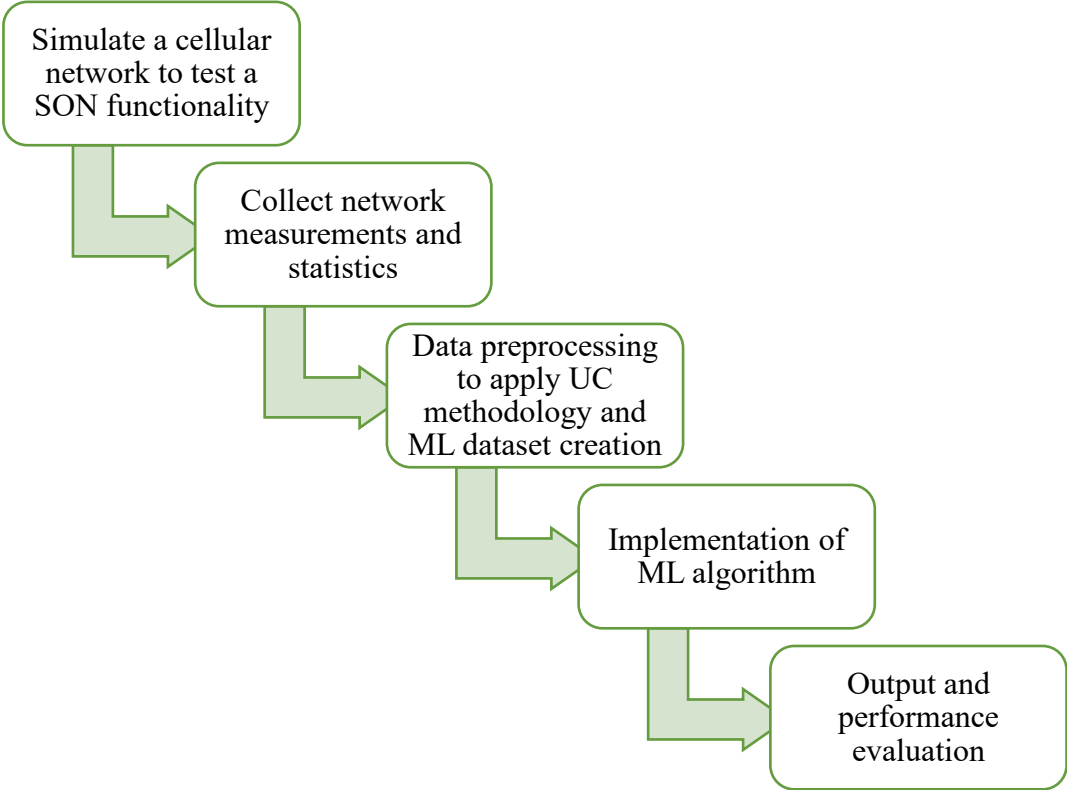


Figure 1.2 The process flow for the research initiatives

### **1.3. Contributions and Organization of this Dissertation**

The rest of the dissertation is organized as follows: Chapter 2 presents a literature survey of SON and ML and introduces the concept of UC technology. It also introduces the simulation and tools used in this dissertation. Chapter 3 proposes and evaluates a UC methodology that uses supervised machine learning to learn and predict the QoE level of the end-user experiences and uses this information to detect anomalous behavior of dysfunctional network nodes (base stations/ eNodeBs/ gNodesBs) in self-organizing networks providing an intelligent, self-learning decision making mechanism that imports the effect of end-user perception of the quality of service for automatic detection and remediation of failures helping network operators understand the end-user needs and identify network elements that are failing and need attention and recovery [2], [3]. Chapter 4 proposes and demonstrates a UC methodology, user-specific optimal capacity and shortest path (US-OCSP) routing, that uses reinforcement learning to determine the resource-based optimum-capacity shortest path for a user between source and destination such that the optimization begins at the end-user level to find the shortest path available that traverses through non-congested network nodes and recommends that path to the end-user given its source and destination thus, facilitating effective resource allocation that will optimize end-user satisfaction [4]. In chapter 5, a UC methodology that uses unsupervised machine learning to form RNA clusters is developed such that it learns about the user characteristics (e.g. connectivity status, mobility status), examines radio conditions and network load, tracks the paging load improvement, and applies this knowledge towards intelligently and adaptively constituting and dynamically evolving the RNAs in a SON network. Chapter 6 concludes this dissertation by summarizing the research contributions made in this dissertation and provides future research directions.



## **Chapter 2. Literature Review**

### **2.1. Introduction**

This chapter reviews the core technology components that provide the foundation of the research presented in this dissertation. The objective of this chapter is to give an overview of the concept of self-organizing networks (SONs), provide insight into the “hot” technology of machine learning (ML), and offer an intuitive understanding of the user-centric (UC) technology. An overview of the simulation tools used for experimental and evaluation purposes in this dissertation is also provided.

### **2.2. Self-Organizing Networks**

Network operators are under constant pressure of deploying denser networks that can sustain the tremendous growth of connected devices, types of services and applications, and mobile data traffic volume at acceptable levels of capital expenditures (CAPEX), operational expenditures (OPEX), and energy consumption that has driven significant momentum to realize network automation [5]. The umbrella concept of SON refers to the automation of network functions and capabilities that can realize a network that autonomously configures its entities, self-optimizes, and self-heals with little to no human intervention, thus minimizing the capital and operational expenditures, while providing enhanced network performance and efficiency.

Self-organization functions for wireline networks laid a baseline that provided the network infrastructure for self-organization in wireless networks. A few typical examples [6] of the

emergence of self-organized functions for wireline networks are the introduction of dynamic host configuration protocol (DHCP) and the standardization of Internet Protocol version 6 (IPv6) enabling IP auto configuration eliminating the need of dedicated address servers and administrators, and the introduction of the Transport Control Protocol (TCP) that implements a decentralized mechanism to handle congestion in the Internet without explicit management of network resources. Additional examples [6] can be found in the area of mobile *ad hoc* and sensor networks where *ad hoc* routing protocols are implemented for self-organized packet delivery and the notion of self-organization appears in the context of failure resilience and network restoration to develop self-healing and self-stabilizing networks that react to link and node failures or physical damages to re-route the affected traffic in a self-organized manner.

Introduced in 4G (3GPP Rel8-TS 32.500), with limited deployment, SONs for wireless networks are currently used to mechanize parallel operations of 4G and 5G (3GPP TR 28.861). Based on the location of the SON algorithm, SON is categorized as a centralized SON, a distributed SON and a hybrid SON [3] as shown in Figure 2.1 (NFs are the Network Functions, CN is the core network, and RAN is the Radio Access Network) and an overview of the SON framework [7], as illustrated by 3GPP, is shown in Figure 2.2. The 3GPP standards cited above have specified all three modes centralized, distributed, and hybrid SONs. It is the SON algorithm that is not standardized by 3GPP and is left to the implementer as an innovative opportunity.

A centralized SON has two variants, viz., cross domain-centralized SON and domain-centralized SON. In the cross domain-centralized SON, the SON algorithm is located in the 3GPP cross management domain layer such that the 3GPP cross management domain monitors the network via management data, analyzes the management data, makes decisions on the SON actions, and executes those actions, while in the domain-centralized SON, all these functionalities

are executed in the management domain. The 3GPP cross management domain is responsible for the management and control of the domain-centralized SON functions where the responsibilities may include switching on/off a domain-centralized SON function, making policies for a domain-centralized SON function, and/or evaluating the performance of a domain-centralized SON function. A centralized SON (C-SON) view [7] is illustrated in Figure 2.3 where the 3GPP management system monitors the network via management data that may include service-level agreement requirements, performance measurements of the network, alarm information, etc. depending on a case-by-case basis. The 3GPP management system then analyzes this data, makes decisions on the SON actions, and executes them in and across multiple network domains such as RAN and CN.

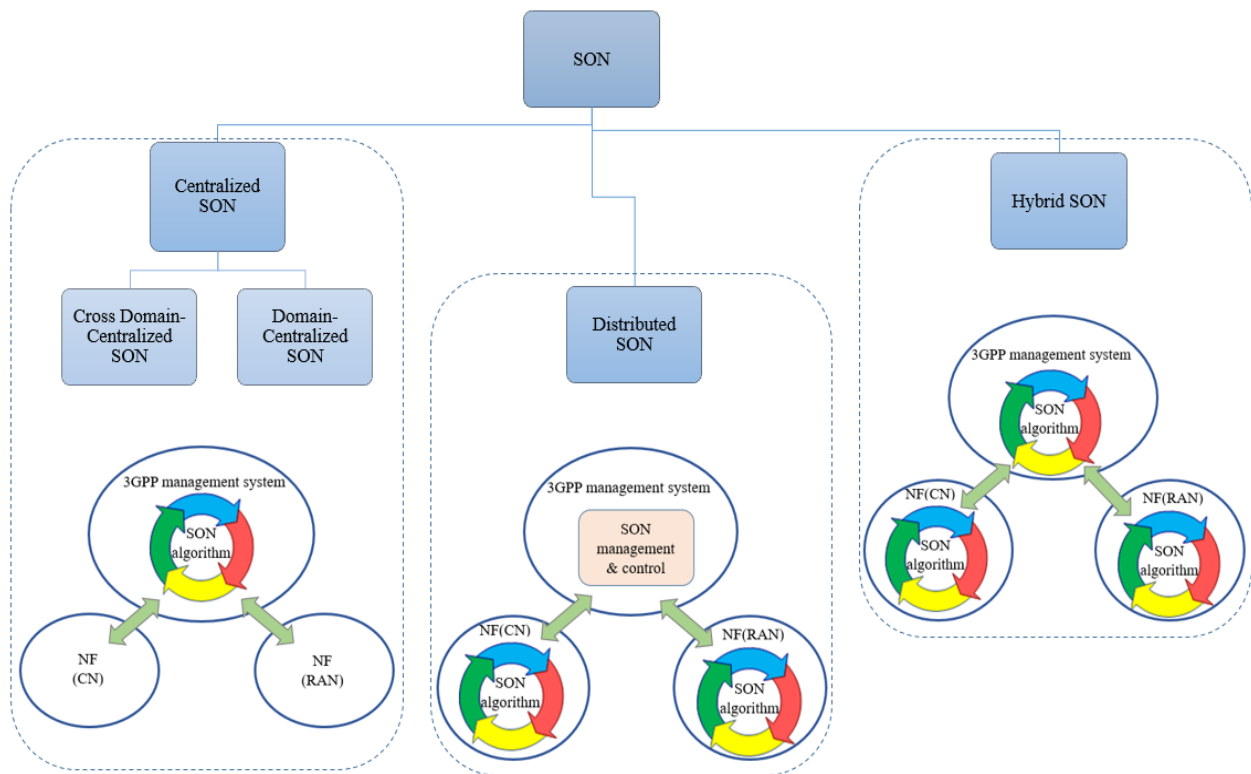


Figure 2.1 SON structures based on the location of the SON algorithms [7]

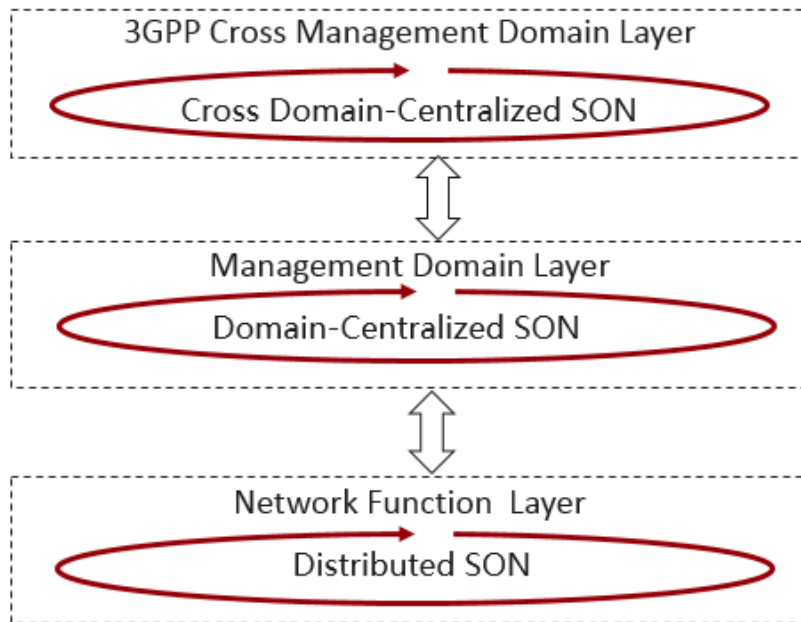


Figure 2.2 SON framework [7]

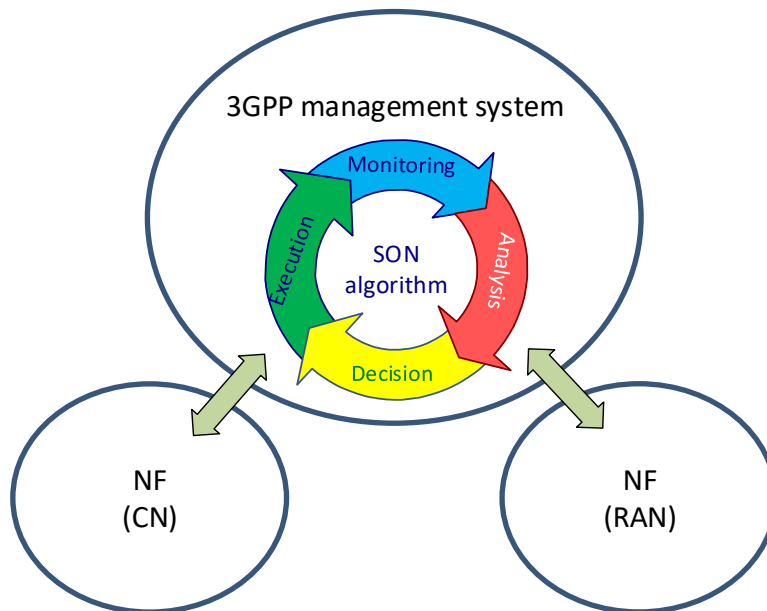


Figure 2.3 C-SON view [7]

In a distributed SON (D-SON) [7] as depicted in Figure 2.4, the SON algorithm is located in the NFs such that the NFs monitor the network events, analyzes the network data, makes decisions on the SON actions and executes the SON actions in the network nodes. The 3GPP management system is responsible for the management and control of the D-SON functions where the responsibilities may include switching on/off a D-SON function, making policies for a D-SON function, providing supplementary information such as the value range of an attribute to a D-SON function, and/or evaluating the performance of a D-SON function.

In a hybrid SON (H-SON) [7] as shown in Figure 2.5., the SON algorithm is partially located in the 3GPP management system and partially located in the NFs. The 3GPP management system and NFs work together, in a coordinated manner, to build up a complete SON algorithm. The decisions on the SON actions may be either made by the 3GPP management system or the NFs, depending on the specific cases.

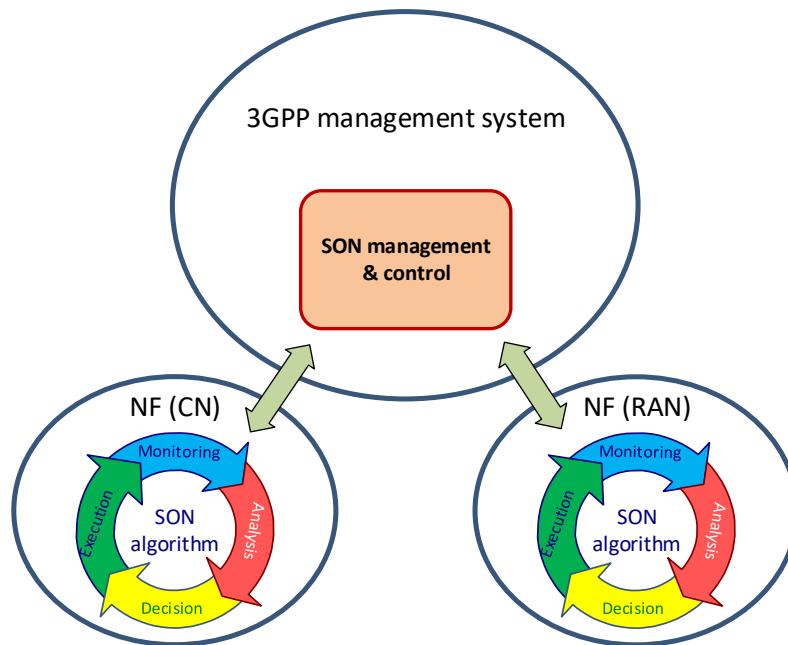


Figure 2.4 D-SON view [7]

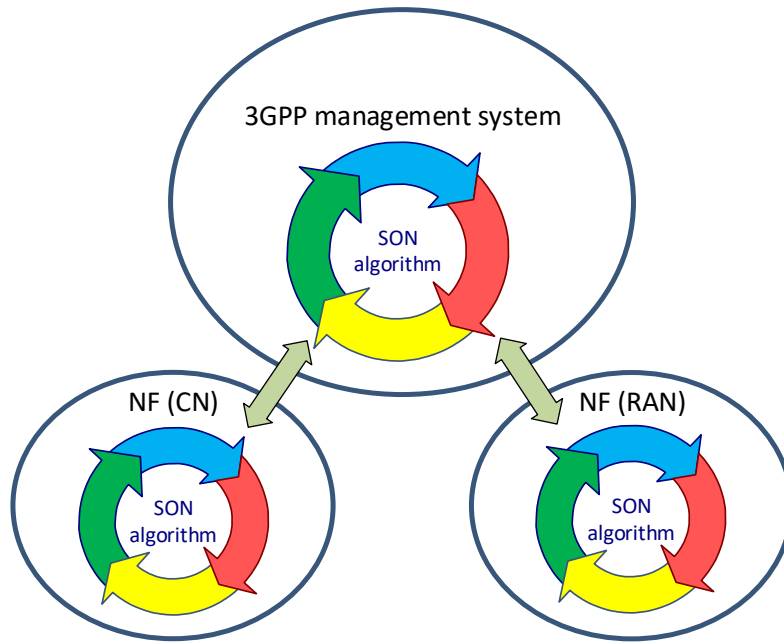


Figure 2.5 H-SON view [7]

An illustration of the 5G network architecture embedded within the SON framework described above is given in Figure 2.6 where the RAN domain entities include 5G base stations/NG-RAN nodes<sup>1</sup> (gNB and ng-eNB), and the CN domain entities include access and mobility management function (AMF), user plane function (UPF), session management function (SMF), unified data management (UDM), and policy control function (PCF). Additional details on the 5G network architecture and network functions can be found in 3GPP TS 23.501 [8] and 3GPP TS 38.300 [9]. If the SON decisions are made at the NF layer, the SON framework can be characterized as a D-SON. If the SON decisions are made at the management domain or the cross management domain layer, the SON framework can be characterized as a C-SON. If the SON decisions are coordinately made by both management and NF entities, the SON framework can be characterized as an H-SON.

<sup>1</sup> The gNBs and ng-eNBs are interconnected with each other by means of the Xn interface. The gNBs and ng-eNBs are also connected by means of the NG interfaces to the 5G core network (5GC).

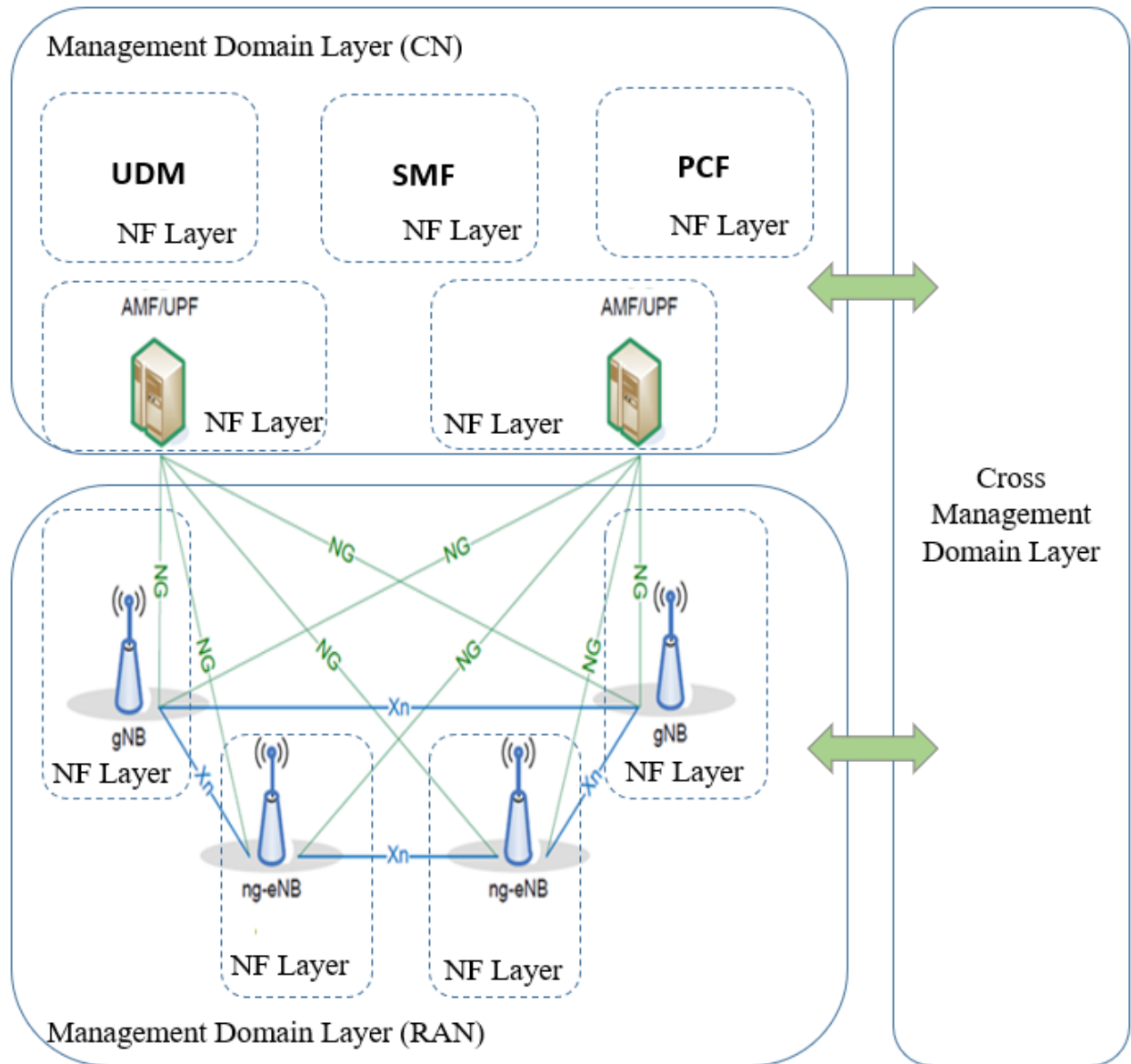


Figure 2.6 5G network architecture embedded within the SON framework

The raw performance data of NFs of the mobile network, along with other management data such as alarm information, configuration data, etc., forms what is referred to as the management data analytics service [MDAS] and utilizes this information for the analysis and correlation of the overall performance data of the mobile network to diagnose ongoing issues impacting the performance of the mobile network and predict any potential issues (e.g., potential failure and/or performance degradation) [10]. The MDAS services can be made available and

consumed by other management and SON functions [7] as illustrated in Figure 2.7 such that the SON functions may utilize the services provided by MDAS to conduct their functionalities and control actions.

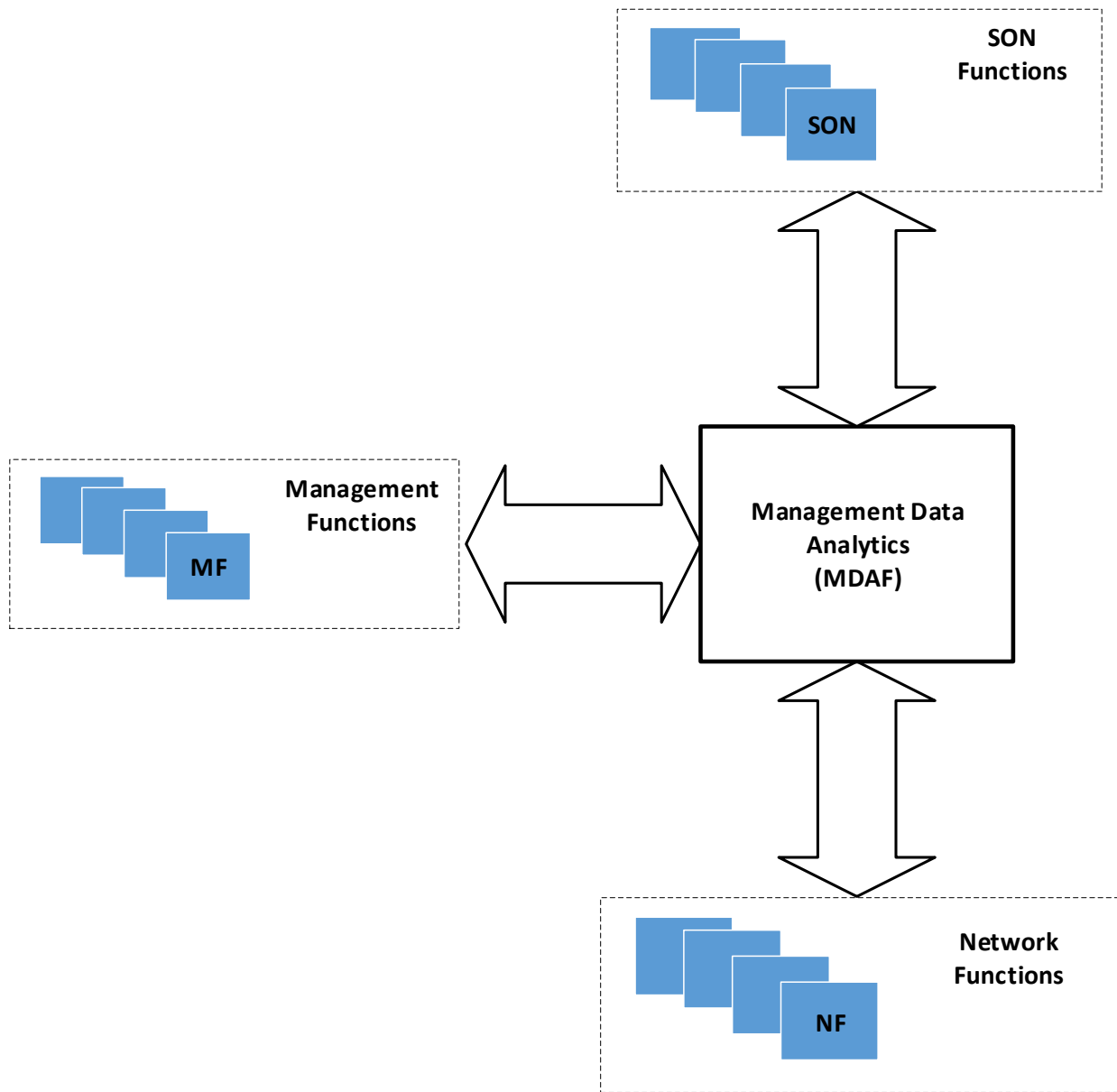


Figure 2.7 Management data analytics service and SON functions [7]



SON functions can be broadly classified into three categories: self-configuration, self-optimization, and self-healing each of which can be described below [5], [11], [12].

Self-configuration refers to the automatic configuration of network nodes and parameters. This may include the dynamic plug-and-play configuration of newly deployed network nodes where a network node will, by itself, configure operational parameters, radio parameters, and neighbor relations (for mobility management and hand overs). The self-configuration network functions may incorporate dynamic assignment of physical cell identity, transmission frequency, and power, dynamic connections to IP backhaul, dynamic configuration of paging areas such as RAN-based notification areas (RNA), automated neighbor relations and other such functions that are required for a newly deployed network node or sub-network to become fully operable. This initial configuration of network parameters may successfully be able to manage a network in a static environment, but since the real-world environment is not static, there is a need for further optimization.

Self-optimization refers to constant monitoring of network parameters and environment to dynamically update system parameters in order to ensure efficient network performance. Self-optimization involves functions such as load balancing optimization, where network nodes exchange information about load level and available capacity by means of radio resource status reports in order to transfer load or part of the user traffic from congested cells to other cells that may have spare resources, mobility robustness optimization that performs mobility management and handover parameter optimization for automatic detection and correction of errors in the mobility configuration, random access channel (RACH) optimization where a UE can be polled by a network node to obtain RACH statistics that can be used to minimize the number of attempts on the RACH channel reducing interference, interference coordination to keep inter-cell

interference under control by managing radio resources, and energy efficiency to enable a greener network where some network nodes can be switched off during off-peak-traffic situations when capacity is not needed. While these optimization strategies can help improve the network performance, partial or full outages may occur due to various faults and failures that can degrade the overall performance of the network and require self-healing.

Self-healing refers to the automatic detection and remediation of failures in order to ensure fast and seamless recovery. Self-healing includes functions such as anomaly detection that is automatically able to detect faults and failures that have occurred in the network, fault diagnosis or classification that can determine the causes of the problems to find the correct solution, and cell outage management to implement compensation mechanisms in order to minimize the disruption caused in the network until the completion of recovery operations. The self-healing function in future networks is expected to proactively predict the faults and anomalies and to take the necessary measures to mitigate network degradation before a fault or failure actually happens.

The full automation of SON is desirable to maximally reduce the OPEX of the networks, and to achieve the fastest reaction to the network issues, but in order to prevent any major negative network impact due to improper SON actions, it is critical that the network operators build confidence about the SON functions step by step before allowing the SON process to run fully autonomously, thus human intervention of the SON process needs to be allowed [5], [7]. In accordance with this observation, the SON process can be categorized as open loop or closed loop. Network operators have the flexibility to stop, resume, and cancel the SON process and make adjustments to the network as needed in an open loop SON process and once the network operators have built adequate confidence, they may convert the open loop SON process to a closed loop SON process that will be completely autonomous [5], [7].

The taxonomy of self-organizing networks [5] is illustrated in Figure 2.8. The research initiatives in this dissertation have developed novel methods to process and execute SON functions such as anomaly detection, load balancing optimization, and dynamic configuration of paging areas. Initially, SON functions can be implemented as open loop where the SON updates take effect based on the response by the network operator and eventually can be converted to closed loop after enough confidence is gained such that the response from the network operator is no longer required.

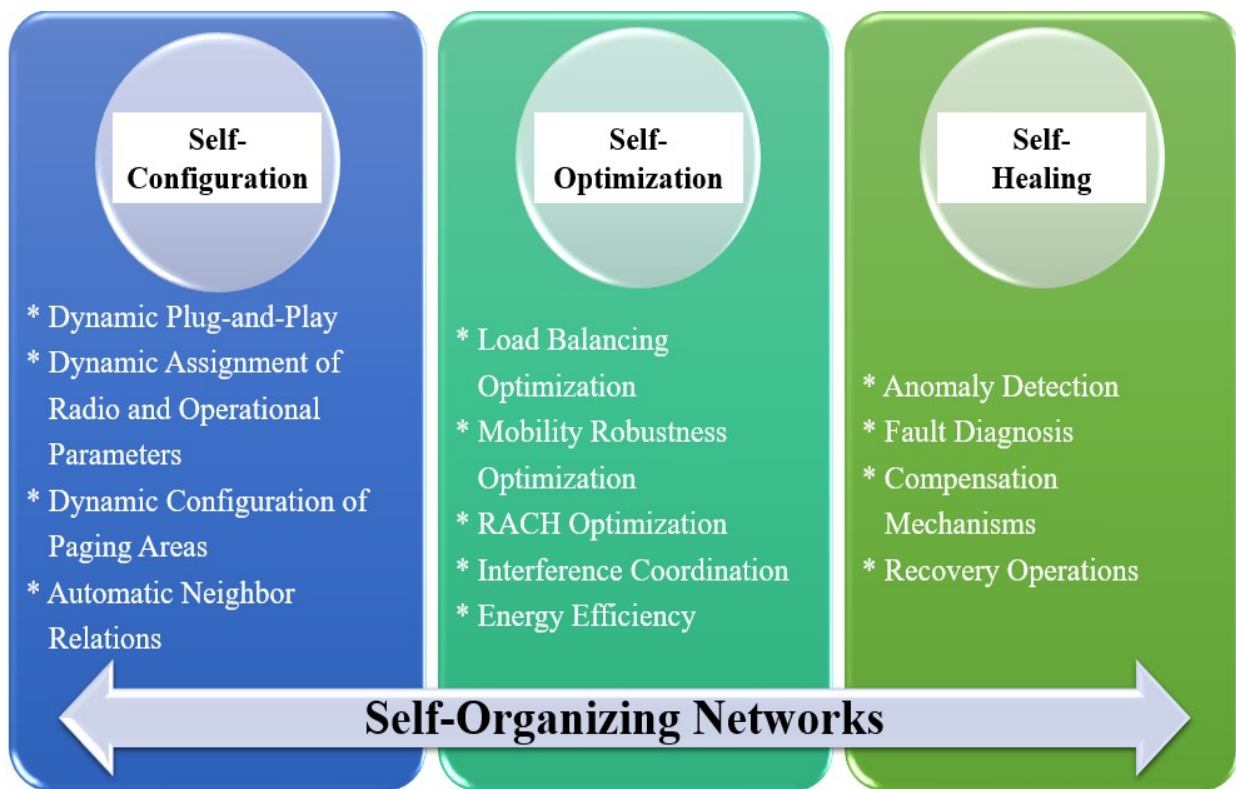


Figure 2.8 Taxonomy of self-organizing networks

### 2.3. Machine Learning

Machine Learning (ML) is the ability of systems to acquire and continuously improve their own knowledge, by extracting patterns from raw data to address problems involving knowledge

of the real world and make decisions that appear to be subjective and mimic human "cognitive" functions [13]. As opposed to well-established mathematical models, ML is a data-driven paradigm shift where a machine learning algorithm learns from experience while working on some tasks and determines if the performance improves with experience.

ML is broadly classified into three categories, namely, supervised learning, unsupervised learning, and reinforcement learning and can be explained as given below [5], [12].

Supervised Learning (SL), as the name implies, requires a supervisor in order to train the system. This supervisor tells the system, for each input, what is the expected output and the system then learns from this guidance. Unsupervised Learning (UL), on the other hand, does not have the luxury of having a supervisor. This occurs, mainly when the expected output is not known, and the system will then have to learn by itself. Reinforcement Learning (RL) works similarly to the unsupervised scenario, where a system must learn the expected output on its own, but in this case, a reward mechanism is applied such that the system receives a reward or a penalty depending on the type of decision it makes. The basic structures of these ML categories are depicted in Figure 2.9.

In addition to these categories, deep learning is a particular kind of machine learning that achieves great power and flexibility by representing the target system as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations are computed in terms of less abstract ones [13]. Deep learning is based on neural networks and can be applied to supervised learning, unsupervised learning, as well as reinforcement learning algorithms.

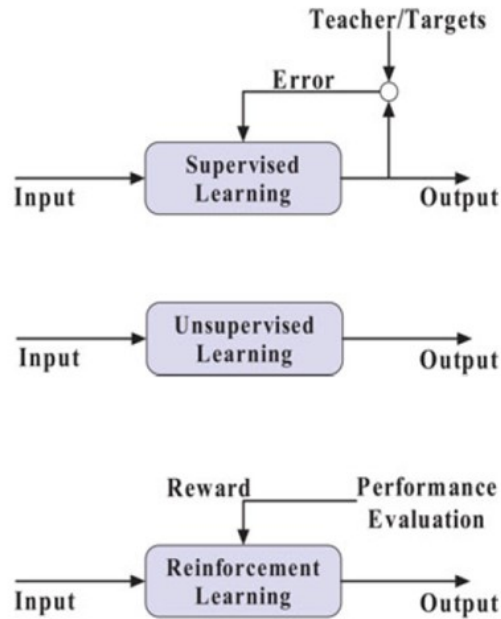


Figure 2.9 Basic structures of ML categories [14]

ML can be used to solve various problems across a variety of applications and emphasizes thinking outside a single issue and beyond established boundaries. ML is expected to deeply transform system design and optimization for the existing as well as next-generation wireless communications networks and will play a pivotal role in implementing the SON functions, thus helping the network operators to transition from the existing human management models to self-driven automatic management. ML-based SON networks will not only achieve the fastest reaction to the network issues but will also be able to take proactive measures based on ML-based predictions.

The research initiatives taken in this dissertation involves the application of different types of supervised, unsupervised, as well as reinforcement learning algorithms [13], [15], [16], [17], [18], [19], [20], [21], [22], [23] listed in Figure 2.10.

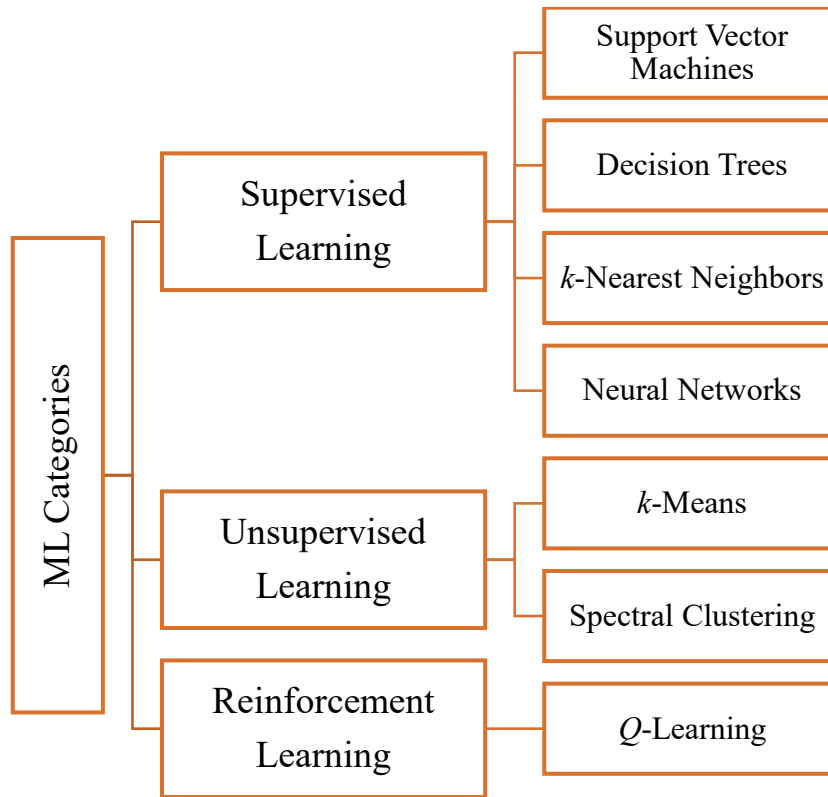


Figure 2.10 Classification of ML algorithms utilized in this dissertation

## 2.4. User-Centric Technology

It is well-known that capturing and utilizing the end-user or customer experience is one of the most vital aspects of any industry or business domain. A good and satisfying experience leads the users to spend more time on the network, which in turn drives demand and increase revenues [2]. This can be achieved by developing user-centric technologies, where the network strategies are based on user needs, the network optimization is based on user feedback, and the network performance is monitored via user-focused key performance indicators (KPIs).

The network operators are expected to deliver significantly increased operational performance (e.g. increased spectral efficiency, higher data rates, low latency), as well as superior user experience (approaching that of wireline, fixed networks but offering full mobility and

coverage) and the need to deploy massive deployments of Internet of Things (IoT) networks, while still offering acceptable levels of energy consumption, equipment cost and network deployment and operation cost making it extremely significant that the next-generation network developments are based on use cases that are more user-centric opening up new revenue streams for network operators [5]. This dissertation proposes new user-centric technologies for next-generation network development and enhancement.

## **2.5. Simulation Toolkit**

Simulation modeling allows a practical and effective way to test novel methodologies and algorithms without having to make a scale copy of the entire network. The network simulator used in this dissertation is ns-3 [24], which provides a network simulation platform based on 3GPP standards [24] to design and test SON algorithms and solutions. It supports the evaluation of radio level performance and end-to-end QoE (QoE can be defined as the overall acceptability of an application or service, as perceived subjectively by the end-user). The ns-3 simulator is a discrete-event network simulator intended primarily for research and educational use. In brief, ns-3 provides models of how packet data networks work and perform and provides a simulation engine for users to conduct network experiments. Some of the reasons to use ns-3 include the ability to perform studies that are more difficult or not possible to perform with real systems, to study system behavior in a highly controlled, reproducible environment, and to get insight into the workings of a particular network. The programming language used to write the ns-3 simulation programs in this dissertation is C++.

The ns-3 output consists of network metrics and measurements that are extracted and processed to generate the input datasets to test the ML algorithms applied in this dissertation. The

programming language used to implement ML algorithms is Python and the ML programs are run via the Anaconda Distribution [25] that provides an open-source platform to execute ML algorithms. It comprises of an extensive set of ML-focused libraries and modules such as scikit-learn [26], [27] and TensorFlow. It supports scientific libraries such as SciPy, NumPy, and pandas providing high-performance and easy-to-use data structures, and data analysis and manipulation methods used for scientific computing. It also consists of a 2D plotting library, Matplotlib to visualize and interpret results.

## **2.6. Concluding Remarks**

This chapter laid out the background for the research presented in this dissertation. It included a literature review for self-organizing networks and machine learning and introduced the concept of user-centric technologies. It covered an overview of the simulation toolkit used in this research for demonstration and evaluation purposes.



## **Chapter 3. QoE-Driven Anomaly Detection in Self-Organizing Networks using Machine Learning<sup>2</sup>**

### **3.1. Introduction**

With the exponential growth in mobile traffic data, network operators are in a dire need of a technology that can meet the demanding requirements of ever-increasing network usage marked by a radical change in user behavior that has been triggered by the proliferation of bandwidth-hungry applications. As the emerging bandwidth-intensive technologies such as 5G, Internet of Things (IoT), and Virtual Reality (VR) begin to be deployed, the network operators need to ensure that networks are intelligent, scalable and robust enough to provide an excellent user experience that consumers expect. One promising solution to address these concerns is the deployment of self-organizing networks enhanced by the machine learning technology. There is little doubt that ML will be a foundation technology that will permeate next-generation wireless networks to provide ground-breaking levels of flexibility and intelligence.

This chapter applies the three-layered approach represented by the synergistic integration of SON, ML, and UC technologies to the first research initiative presented in this dissertation where a UC KPI (key performance indicator) uses ML to predict and detect the anomalous behavior of dysfunctional network nodes (base stations) by importing the effect of the end-user

---

<sup>2</sup> The contents of this chapter have been published in [2], [3], [5]. Permissions are include in Appendix A.

perception of the quality of service for automatic detection and remediation of failures in self-healing SON systems.

### **3.2. Anomaly Detection**

One of the salient functions of self-organizing networks is to be able to correctly detect anomalies, e.g., dysfunctional nodes (e.g., base stations) or sites that cause outages and degradation in the network. Automatic detection of network node failures and outages is crucial to ensure fast and seamless recovery from such failures. The occurrence of failures in a network element, such as a base station, may cause deterioration of this network element's functions and/or service quality and will, in severe cases, lead to the complete unavailability of the respective network element [28]. Consequently, anomaly detection is crucial to minimize the effects of such failures on network users. In case of green communication, energy-efficient network planning strategies include networks designed to meet peak-hour traffic such that energy can be saved by partially switching off base stations when they have no active users or simply very low traffic [29]. This makes anomaly detection even more critical as the detection methods must be well-equipped not to falsely detect partially switched off base stations in energy saving mode as dysfunctional.

Currently, alarm monitoring, routinely performed checks on the configuration parameters and counters, collecting traffic data to profile the behavior of the network in normal vs. abnormal conditions, measuring reference signal received power (RSRP) and signal-to-noise-and-interference ratio (SINR), analyzing incoming handover measurements from neighboring cells, keeping track of customer complaints, and analyzing key performance indicators (KPIs) to detect any degradation are some of the widely used detection methods that network engineers follow to detect dysfunctional nodes [30]. These procedures may not always provide timely or accurate

determination of the network state. Alarm monitoring and configuration parameter checks may not necessarily be able to detect sleeping cells. Drive test data, RSRP, and SINR measurements can be affected by poor radio frequency (RF) conditions due to temporary reasons like ducting or external interference which may not be due to faulty network nodes. The number of handover (HO) attempts made could unduly be affecting the results of the detection method based on HO measurements. Customer complaints may provide limited technical information. On balance, the state-of-the-art approaches for anomaly detection lack the knowledge of the end-user perception of the quality of a provided service. KPI analysis is crucial in anomaly detection and needs more user-centric KPIs such as the quality of experience (QoE) to evaluate and detect dysfunctional nodes.

### **3.3. Quality of Experience (QoE)**

QoE is defined by the Telecommunication Standardization Sector of International Telecommunication Union (ITU-T) as “the overall acceptability of an application or service, as perceived subjectively by the end-user” [31]. In other words, QoE describes the degree of the end-user’s “delight or annoyance” while using a product or service. Unlike the quality of service (QoS), QoE incorporates user-centric network decision mechanisms and processes such that it takes into account not just the technical aspects regarding a service but also incorporates any kind of human-related quality-affecting factors reflecting the impact that the technical factors have on the user’s quality perception [32]. There are different types of approaches that can be used for QoE assessment. These approaches can be classified into subjective tests, objective tests and hybrid tests methods. There are various types of evaluation models based on these approaches proposed for QoE estimation in the literature. Parametric QoE estimation models are currently the most

popular candidates for quantifying QoE levels in an indirect and user-transparent way in mobile networks. Parametric models use network parameters and metrics for QoE estimation [32], [33]. Parametric QoE models are derived by performing subjective experiments that may include laboratory tests or crowdsourcing and by performing statistical analysis on the results. The derived models may then be used to generate formulas which can be used to compute QoE given specific input parameters [34].

### **3.4. The Methodology**

The proposed methodology, QoE-driven anomaly detection in SON using ML, can be explained using the process flow described in Figure 3.1. An end-to-end network scenario is created using the network simulator ns-3 [24], where end users interact with a remote host that is accessed over the Internet to run the most commonly used applications like file downloads and uploads. The transmit power on a few network nodes is altered intentionally to test the methodology.

The data obtained from the ns-3 simulation serves as the input dataset for the machine learning model where a parametric QoE model and ML algorithms are implemented to predict QoE scores of end users that are further used to identify dysfunctional network nodes. There are multiple parametric QoE models which can be used for a reliable estimation of QoE for various types of services. One of the most commonly used application by users is the file download application. The protocol used by this application is the file transfer protocol (FTP). File transfer services are considered to be elastic services, whose utility function is an increasing, strictly concave, and continuously differentiable function of throughput [35]. The principal characteristic of FTP services is that there is no need for a continuous and in-sequence packet arrival. Taking

into account that the delay expected by the end-user is proportional to the size of the downloaded file, the most dominant factor that affects the QoE level is the data rate. A parametric model that is commonly used to provide the mean opinion score (MOS) for FTP services used to generate the QoE scores is expressed in Equation (3.1)

$$MOS_{FTP} = \begin{cases} 1 & u < u^- \\ b_1 \cdot \log_{10}(b_2 \cdot u) & u^- \leq u < u^+ \\ 5 & u^+ \leq u \end{cases} \quad (3.1)$$

where  $u$  represents the user data rate (user throughput) of the correctly received data. The values of the  $b_1$  and  $b_2$  coefficients are obtained from the upper ( $u^+$ ) and lower rate ( $u^-$ ) expectations for the service [32], [34], [35]. Either the parametric model or acquiring direct feedback from users can be used to analyze the differences and similarities of the results. It would be even more beneficial to create the QoE target values based on a combined method that acquires QoE scores as rated directly by the end-users and supplement that with user-impacting network metrics to analyze and assign appropriate weights to each metric generating a holistic approach for QoE evaluation. This will ensure that the direct user feedback as well as important network metrics are considered in the QoE calculation.

In selecting ML algorithms, the first step is to understand the problem at hand and the associated data available to find out what general category the ML algorithm belongs, viz., supervised learning, unsupervised learning, or reinforcement learning. ML learns from examples and the preferred approach is to train a model by making the best possible use of the data available. If the expected output information is available for training, supervised learning is preferable. For QoE prediction, it is important and (also possible) that the expected QoE output is selected to get as close to the end-user perception of the quality of service experienced as possible so that the ML algorithm learns using this reference. Consequently, the type of machine learning algorithms

implemented in this research are supervised machine learning algorithms. Four supervised learning algorithms, support vector machines (SVM),  $k$ -nearest neighbors ( $k$ -NN), decision trees (DT), and neural networks (NN) were implemented each of which are explained in Section 3.4.

To perform anomaly detection, the predicted QoE score for every user, using the machine learning models described above, is used to filter out all the users with poor QoE scores ( $QoE \leq 1$ ). If the majority of users attached to a particular network node have poor QoE scores, the node is declared to be dysfunctional. In other words, if the mode of the QoE scores of all the users connected to that particular network node is less than or equal to the threshold which in this case is set to 1, then the network node is declared as dysfunctional but if the mode of the QoE scores of all the users connected to a particular network node is greater than the threshold, the network node is declared as functional.

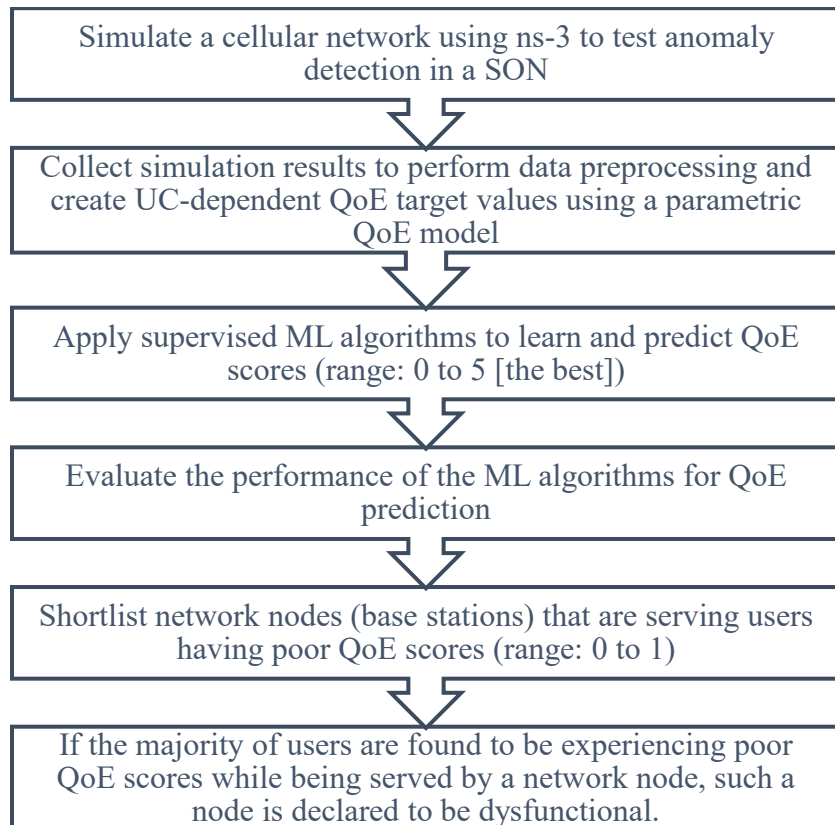


Figure 3.1 The process flow for QoE-driven anomaly detection in SON using ML

### 3.5. The ML Algorithms

#### 3.5.1 Support Vector Machines

A support vector machine [26], [16], [19] constructs a hyperplane or set of hyperplanes in a large or infinite dimensional space, which can be used for classification, regression or other tasks. If sufficient separation is achieved by the hyperplane with the largest distance to the nearest training samples of any class, the algorithm will generally be effective. The training samples that are the closest to the decision surface are called support vectors. The SVM algorithm finds the largest margin (i.e., “distance”) between the support vectors to obtain optimal decision regions. The type of SVM algorithm used in the proposed method is SVM regression. In SVM regression, the input vector  $\mathbf{x}$  is first mapped<sup>3</sup> onto an  $m$ -dimensional feature space using a fixed (nonlinear) mapping i.e. by using kernel functions, and then a linear model is constructed in this feature space to separate the training data points. The linear model in the feature space  $f(\mathbf{x}, \omega)$  is given by

$$f(\mathbf{x}, \omega) = \sum_{j=1}^m \omega_j g_j(\mathbf{x}) + b \quad (3.2)$$

where  $g_j(\mathbf{x})$ ,  $j = 1, \dots, m$  denotes a set of nonlinear transformations (e.g. radial basis function) and  $b$  is a bias term. A loss function [16] often used by an SVM to measure the quality of estimation is called the  $\varepsilon$  – insensitive loss function and is given below.

$$\mathcal{L}_\varepsilon(y, f(\mathbf{x}, \omega)) = \begin{cases} 0, & \text{if } |y - f(\mathbf{x}, \omega)| \leq \varepsilon \\ |y - f(\mathbf{x}, \omega)| - \varepsilon, & \text{otherwise} \end{cases} \quad (3.3)$$

The SVM performs linear regression in the high-dimension feature space using  $\varepsilon$  – insensitive loss and, at the same time, tries to reduce model complexity by minimizing  $\|\omega\|^2$ . This can be described

---

<sup>3</sup> In SVM, the input space is transformed into a new feature space using kernel functions where it becomes easier to process the data such that it is linearly separable. Hard margin SVM works well when the data is completely linearly separable. Hard margin SVM is very sensitive to errors (noise/outlier), in which case, soft margin SVM is preferred. Soft margin SVM uses slack variables ( $\xi$ ) to soften the constraints that determine the decision boundaries by skipping a few outliers.

by introducing (non-negative) slack variables  $\xi_i, \xi_i^*$  where  $i = 1, \dots, n$ , to measure the deviation of training samples outside the  $\varepsilon$  – insensitive zone. Thus, SVM regression is formulated as the minimization of the following function [19]:

$$\min \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \quad (3.4)$$

$$\text{subject to } \begin{cases} \mathbf{y}_i - f(\mathbf{x}_i, \omega) \leq \varepsilon + \xi_i^* \\ f(\mathbf{x}_i, \omega) - \mathbf{y}_i \leq \varepsilon + \xi_i \\ \xi_i, \xi_i^* \geq 0, i = 1, \dots, n \end{cases}$$

where  $C$  is a regularization parameter that determines the tradeoff between the model complexity and the degree to which deviations larger than  $\varepsilon$  are tolerated in optimization formulation,  $\mathbf{x}_i$  represents the input values,  $\omega$  represents the weights, and  $\mathbf{y}_i$  represents the target values. This optimization problem can be transformed into the dual problem and its solution is given by

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) K(\mathbf{x}_i, \mathbf{x}) \quad (3.5)$$

subject to  $0 \leq \alpha_i^* \leq C, 0 \leq \alpha_i \leq C$ , where  $n$  is the number of support vectors,  $\alpha_i$  is the dual variable,

and the kernel function is given by

$$K(\mathbf{x}, \mathbf{x}_i) = \sum_{j=1}^m g_j(\mathbf{x}) g_j(\mathbf{x}_i) \quad (3.6)$$

The SVM performance (estimation accuracy) depends on the optimized setting of meta-parameters  $C, \varepsilon$  and the kernel parameters.

### 3.5.2 $k$ -Nearest Neighbor Algorithm

The basic idea behind the  $k$ -nearest neighbor algorithm [26], [20] is to base the estimation on a fixed number of observations  $k$  which are closest to the desired data point. A commonly used metric measure for distance is the Euclidean distance. Given  $X \in \mathbb{R}^q$  and a set of samples  $\{X_1, \dots, X_n\}$ , for any fixed point  $x \in \mathbb{R}^q$ , it can be calculated how close each observation  $X_i$  is to



$x$  using the Euclidean distance  $\|x\| = (x'x)^{\frac{1}{2}}$  where “ ’ ” denotes the vector transpose. This distance is given as

$$D_i = \|x - X_i\| = ((x - X_i)'(x - X_i))^{\frac{1}{2}} \quad (3.7)$$

The order statistics for the distances  $D_i$  are  $0 \leq D_{(1)} \leq D_{(2)} \leq \dots \leq D_{(n)}$ . The observations corresponding to these order statistics are the “nearest neighbors” of  $x$ . The observations ranked by the distances or “nearest neighbors”, are  $\{X_{(1)}, X_{(2)}, X_{(3)}, \dots, X_{(n)}\}$ . The  $k^{\text{th}}$  nearest neighbor of  $x$  is  $X_{(k)}$ . For a given  $k$ , let

$$R_x = \|X_{(k)} - x\| = D_{(k)} \quad (3.8)$$

denote the Euclidean distance between  $x$  and  $X_{(k)}$ .  $R_x$  is the  $k^{\text{th}}$  order statistic on the distances  $D_i$ . In  $k$ -NN regression, the label<sup>4</sup> assigned to a query point is computed based on the mean of the labels of its nearest neighbors. The weights used in the basic type of  $k$ -NN regression are uniform where each point in the local neighborhood contributes to the classification of a query point. In some cases, it can be beneficial to weigh points such that nearby points contribute more to the regression than points that are far away. The classic  $k$ -NN estimate is given as

$$\tilde{g}(x) = \frac{1}{k} \sum_{i=1}^k 1(\|x - X_i\| \leq R_x) y_i \quad (3.9)$$

This is the average value of  $y_i$  among the observations that are the  $k$  nearest neighbors of  $x$ . A smooth  $k$ -NN estimator is a weighted average of the  $k$  nearest neighbors and is given as

$$\tilde{g}(x) = \frac{\sum_{i=1}^k \omega\left(\frac{\|x - X_i\|}{R_x}\right) y_i}{\sum_{i=1}^k \omega\left(\frac{\|x - X_i\|}{R_x}\right)} \quad (3.10)$$

---

<sup>4</sup> In supervised machine learning, the task of the ML model is to predict target values from labelled data. The input is referred to by terms such as independent variables or features. The output is referred to by terms such as dependent variables or target labels or target values.

### 3.5.3 Decision Tree Methods

The primary idea for decision tree methods [26], [21], [36] is that, based on the original data, a set of partitions are created so that the best class (in classification problems) or value (in regression problems) can be determined by creating decision rules which could be a set of if-then-else rules deduced from the data features. The type of decision tree algorithm used in this dissertation is an optimized version of the classification and regression trees (CART) algorithm which can be explained as follows: Given training vectors  $\mathbf{x}_i \in R^n$ ,  $i = 1, \dots, l$  and a label vector  $\mathbf{y} \in R^l$ , a decision tree recursively partitions the space such that the samples with the same labels are grouped together. Let the data at node  $m$  be represented by  $Q$ . For each candidate split  $\theta = (j, t_m)$  consisting of a feature  $j$  and threshold  $t_m$ , partition the data into  $Q_{left}(\theta)$  and  $Q_{right}(\theta)$  subsets. This means the data represented by  $Q$  is now divided into two subsets  $Q_{left}(\theta)$  and  $Q_{right}(\theta)$  that are computed using the equations given below:

$$Q_{left}(\theta) = (x, y) | x_j \leq t_m \quad (3.11)$$

$$Q_{right}(\theta) = Q \setminus Q_{left}(\theta) \quad (3.12)$$

where the division operator ‘\’ is used to denote left division<sup>5</sup>. The impurity<sup>6</sup> at  $m$  is computed using an impurity function,  $H ( )$  the choice of which depends on the task being solved (classification or regression)

$$G(Q, \theta) = \frac{n_{left}}{N_m} H(Q_{left}(\theta)) + \frac{n_{right}}{N_m} H(Q_{right}(\theta)) \quad (3.13)$$

The parameters are selected such that they minimize the impurity

$$\theta^* = \operatorname{argmin}_{\theta} G(Q, \theta) \quad (3.14)$$

---

<sup>5</sup> The left division operator, also known as backslash operator, is generally used in computer programming languages and performs the reverse of right division such that  $X \setminus Y = X^{-1}Y$ .

<sup>6</sup> The impurity refers to the quantification of the error between the predicted value of the machine learning model and the actual target values.

The subsets  $Q_{left}(\theta^*)$  and  $Q_{right}(\theta^*)$  are recursively computed until the maximum allowable depth is reached. The maximum allowable depth is a computational choice specified to control the complexity of the tree and prevent overfitting via pruning. The classification and regression trees are used for constructing prediction models from data. These models are obtained by recursively partitioning the data space and fitting a prediction model within each partition. The recursive partitioning can be represented graphically as a decision tree which is easy to visualize and interpret. The classification trees are designed for dependent variables that take a finite number of unordered values, with prediction error measured in terms of misclassification cost and regression trees are designed for dependent variables that take continuous or ordered discrete values, with prediction error typically measured by the squared difference between the observed and predicted values [36]. The proposed method uses the regression criteria for determining the locations for future splits such that for a node  $m$  in a region  $R_m$  with  $N_m$  observations, the common criteria used to minimize are mean squared error (MSE) and mean absolute error (MAE). The MSE minimizes the L2 error using mean values at terminal nodes and can be expressed as follows:

$$\bar{y}_m = \frac{1}{N_m} \sum_{i \in N_m} y_i \quad (3.15)$$

$$H(X_m) = \frac{1}{N_m} \sum_{i \in N_m} (y_i - \bar{y}_m)^2 \quad (3.16)$$

The MAE minimizes the L1 error using median values at terminal nodes and can be expressed as follows:

$$\bar{y}_m = \frac{1}{N_m} \sum_{i \in N_m} y_i \quad (3.15)$$

$$H(X_m) = \frac{1}{N_m} \sum_{i \in N_m} |y_i - \bar{y}_m| \quad (3.17)$$

where  $X_m$  is the training data in node  $m$ .

### 3.5.4 Neural Network

A neural network (multi-layer perceptron) [26], [18] algorithm learns a function  $f(\cdot): R^m \rightarrow R^o$  by training on a dataset, where  $m$  is the number of dimensions for input and  $o$  is the number of dimensions for output. Given a set of features  $X = x_1, x_2, \dots, x_m$  and a target  $y$ , it can learn a non-linear function approximator for either classification or regression. In logistic regression, there can be one or more non-linear layers called hidden layers between the input and output layers. A one hidden layer multi-layer perceptron (MLP) with scalar output is illustrated in Figure 3.2.

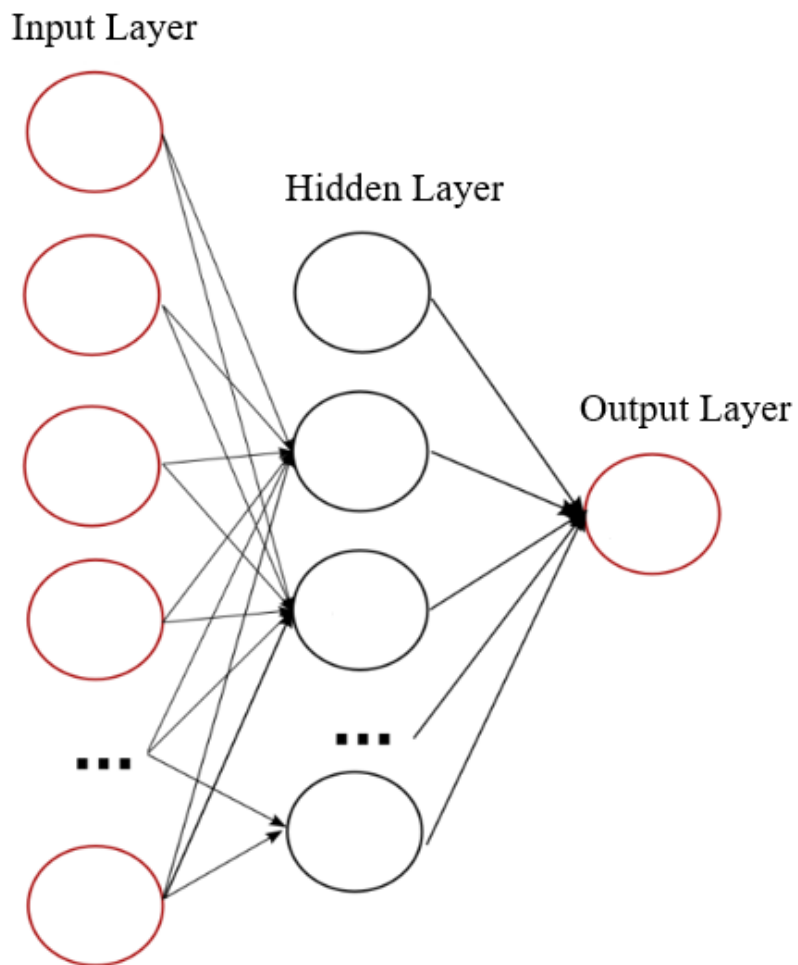


Figure 3.2 One hidden layer MLP

The input layer (the leftmost layer) consists of a set of neurons  $\{x_i | x_1, x_2, \dots, x_m\}$  that represent the input features. Each neuron in the hidden layer transforms the values from the previous layer with a weighted linear summation  $\omega_1 x_1 + \omega_2 x_2 + \dots + \omega_m x_m$  shifted over biases  $b_1, b_2, \dots, b_m$ , followed by a non-linear activation function  $g(\cdot): R \rightarrow R$  (e.g. logistic sigmoid function). The denotations  $\omega_1, \omega_2, \dots, \omega_m$  represent the weights. The output layer (the rightmost layer) receives the values from the last hidden layer and transforms them into output values.

MLP regression implements a multi-layer perceptron that trains using backpropagation with no activation function in the output layer. In other words, it uses the identity function as the activation function. It uses the square error as the loss function and the output is a set of continuous values. It uses L2 regularization that helps avoid overfitting by penalizing weights with large magnitudes. Starting from initial random weights, MLP minimizes the loss function by repeatedly updating these weights. After computing the loss, a backward pass propagates it from the output layer to the previous layers, providing each weight parameter with an updated value meant to decrease the loss. The algorithm stops when it reaches a preset maximum number of iterations; or when the improvement in loss is below a certain, small number.

### **3.6. Performance Analysis and Evaluation**

The values of the primary parameters used to configure the network scenario created in the ns-3 simulation are given below in Table 3.1. The output generated from the ns-3 simulation is fed as an input to the four supervised ML algorithms to study their effectiveness and the scalability of the proposed methodology.

Table 3.1 Simulation parameters and values

Parameters	Value
Number of network users	50 (scalable)
Number of network nodes	5 (scalable)
Channel bandwidth	20 MHz
Transmission power of network nodes	46 dBm
Transmission power of dysfunctional network nodes	30 dBm
Application type	FTP

The performance of the ML algorithms is investigated as their accuracy in predicting the QoE scores determines the ability of the methodology to correctly detect the dysfunctional network nodes. The SVM performance depends on the type of kernel function and the setting of meta parameters C, epsilon, and the kernel parameter, gamma. C is a regularization parameter that determines the tradeoff between the model complexity and the degree to which deviations larger than epsilon are tolerated, epsilon specifies the epsilon-tube within which no penalty is associated in the training loss function with points predicted within a distance epsilon from the actual value, and gamma specifies how far the influence of a single training example reaches and is the inverse of the radius of influence of samples selected by the model as support vectors [26], [19]. The optimal SVM solution for the dataset obtained from the ns-3 simulation is found using the kernel function, radial basis function<sup>7</sup> (rbf) with C = 5, gamma ( $\gamma$ ) = 0.001, and epsilon = 0.01. The  $k$ -

---

<sup>7</sup> Radial basis function can be expressed as rbf:  $\exp(-\gamma \cdot ||x - x'||^2)$ ; where  $||x - x'||^2$  is the squared distance (Euclidean distance) between two samples  $x$  and  $x'$ ; gamma ( $\gamma$ ) must be greater than 0.

NN performance depends on the value of  $k$  and the optimum value of  $k$  for the given dataset is observed to be 4. The decision tree performance for MSE and MAE criteria at varying values of maximum allowable depth is tested and it is observed that MSE at maximum depth value 3 gives the most optimum performance for the given dataset. The optimal performance for neural networks was achieved for the given dataset by using the logistic sigmoid function  $f(x) = 1/(1 + \exp(-x))$  and applying the solver limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (L-BFGS). L-BFGS is a solver that approximates the Hessian matrix which represents the second-order partial derivative of a function. Further it approximates the inverse of the Hessian matrix to perform parameter updates. The training and testing accuracy scores for all four algorithms are compared and the accuracy results are summarized in Figure 3.3.

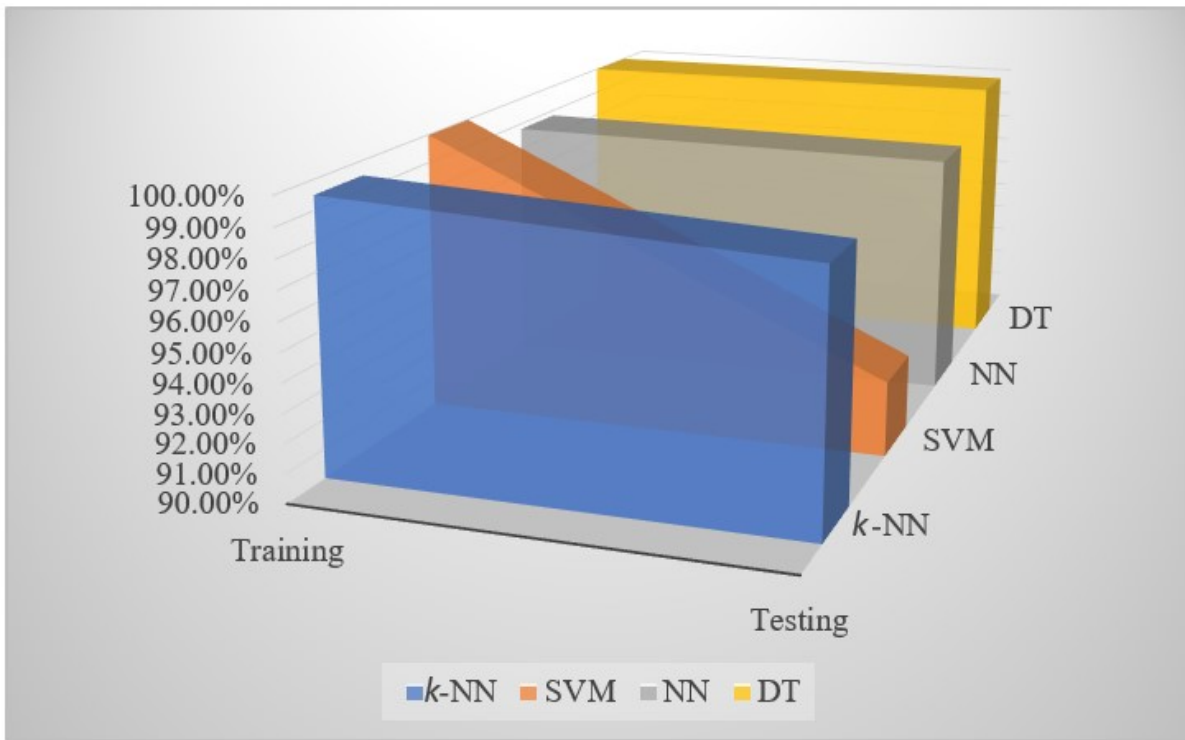


Figure 3.3 Training and testing accuracy scores for the ML algorithms implemented for QoE prediction

The performance and scalability of these algorithms are further evaluated by creating different network scenarios by embedding three different propagation path loss models (Friis propagation, Log-Distance propagation, and Cost 231 propagation [24]) in the ns-3 simulation. The average accuracy results for QoE predictions are shown in Figure 3.4.

To test the scalability even further, the ns-3 simulation was extended as a sequence of independent trials, so as to compute statistics on multiple independent runs and the average resulting performance of the ML algorithms is depicted in Figure 3.5. The dysfunctional network nodes were successfully detected based on the QoE scores predicted by each of the ML algorithms making them a viable choice.

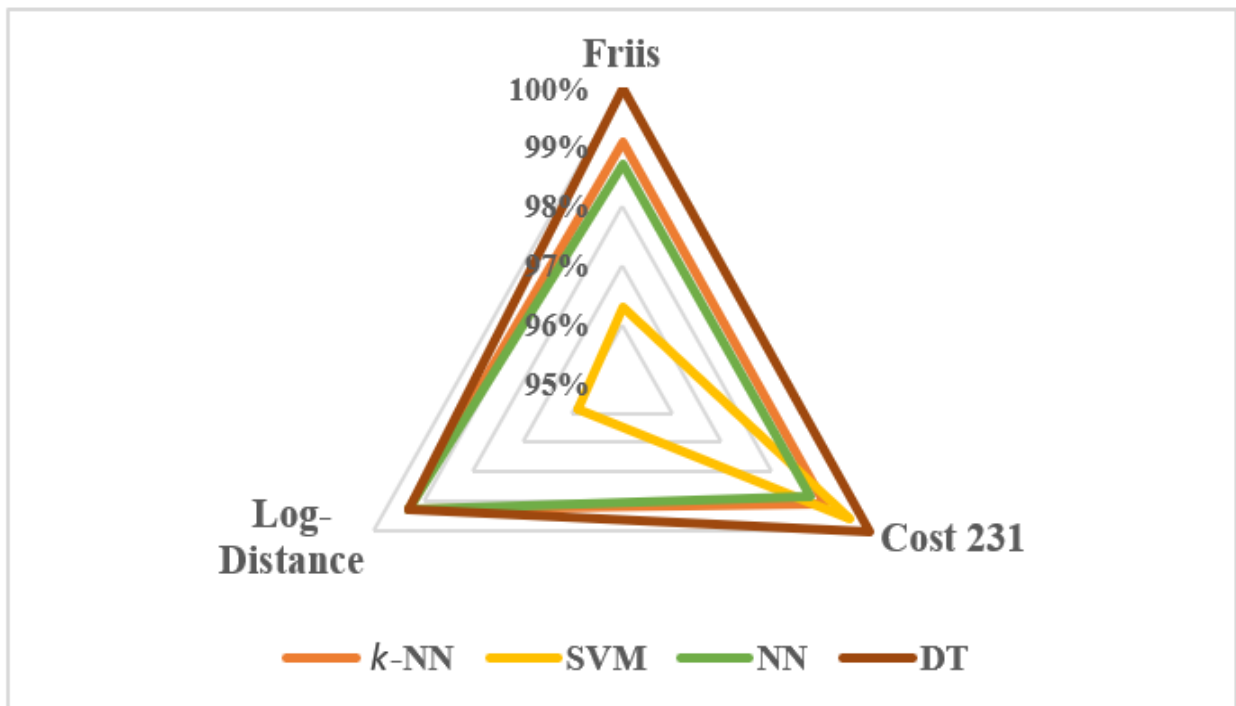


Figure 3.4 QoE prediction accuracy with different RF propagation models



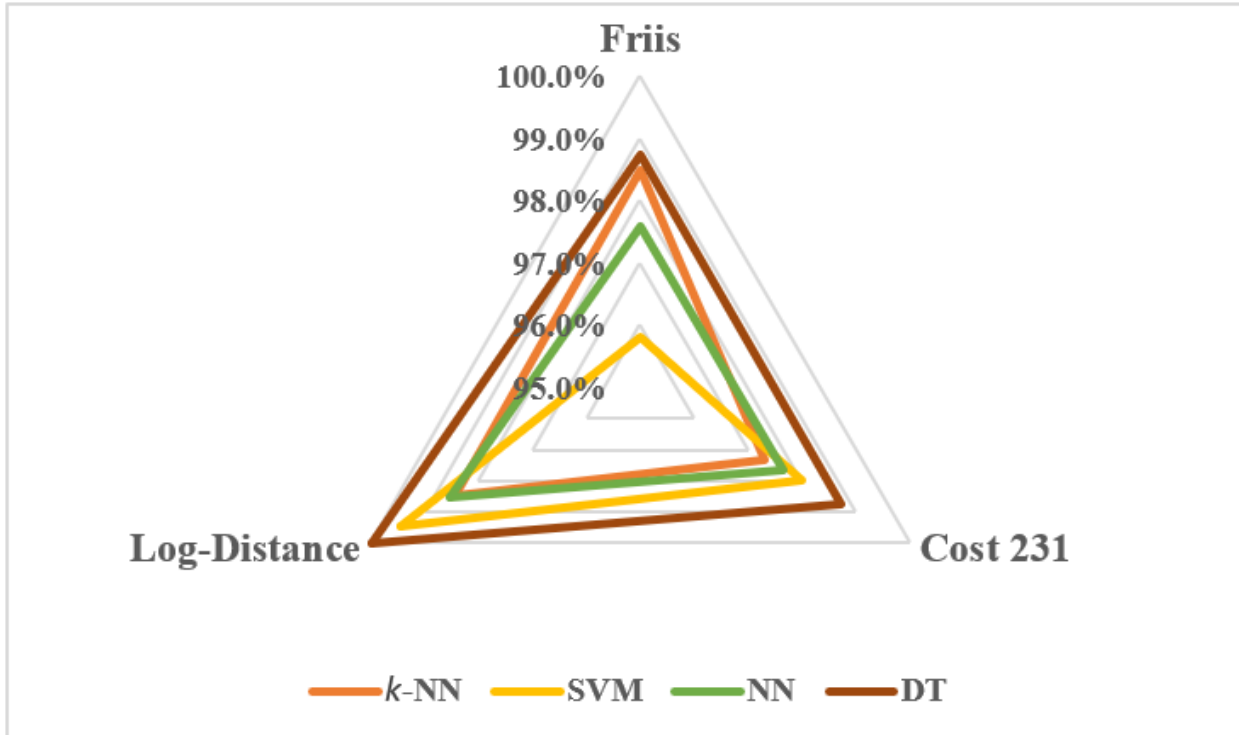


Figure 3.5 The average accuracy results for QoE predictions against multiple independent simulation runs

The application of the optimized version of the decision tree algorithm with the regression criteria MSE gave relatively better results than the rest of the algorithms. It is important to note that the choice of the ML algorithms depends on the nature of the dataset. Each ML algorithm has a few limitations that need to be considered carefully. For example, SVM complexity is very high and a wrong choice of kernel can lead to an increase in error percentage.  $k$ -NN is sensitive to localized data such that the localized anomalies can affect the outcomes significantly. The black-box nature of NN makes it difficult to interpret how the results were derived and troubleshoot. DT has a high probability of overfitting and needs pruning for larger datasets.

### 3.7. Concluding Remarks

The proposed methodology and its demonstration in this chapter realizes a novel method, QoE-driven anomaly detection in self-organizing networks using machine learning that will facilitate self-healing networks. The methodology is a user-centric approach that imports the effect of the end-user perception of the quality of a provided service by learning and predicting QoE scores. The performance of four supervised learning algorithms was investigated where the QoE scores of network users were predicted and were used to identify dysfunctional serving nodes in the network. It is a resource-efficient method as it not only provides information about the user experience, which is extremely crucial for a network operator, but also uses this information to identify nodes that are not functioning well enough to serve the users in their vicinity. It avoids over-engineering as network operators can prioritize their recovery operations on the network nodes that need immediate attention versus the ones that may still be manageable given their QoE scores.

The proposed method can play an important role in supporting future networks based on green communication network design with features like enhanced mobile broadband, extreme densification, and energy efficiency where a large number of serving nodes may be partially turned off for certain intervals to attain energy efficiency as it can distinguish dysfunctional network nodes from partially switched off nodes in energy saving mode i.e. even when a network node is partially switched off to save energy, it would not be falsely detected as dysfunctional unless the QoE scores of the users in the vicinity goes down making it then a real issue that needs to be addressed. Furthermore, the existing network-centric techniques, for example, alarm monitoring can be supplemented by the proposed UC-technique such that any unnecessary troubleshooting actions against falsely generated alarms can be prevented as the QoE-driven anomaly detection

technique can validate if a network node that triggered the alarm needs attention based on the QoE levels predicted. Consequently, combining this method with the existing techniques for anomaly detection can provide highly robust and reliable methods for anomaly detection supporting the ultra-dense 5G/6G networks with the expected benefits of an improved understanding of end-users' perspective and resource-efficiency by effectively prioritizing recovery operations supporting high-density networks.

## Chapter 4. Optimal-Capacity, Shortest Path Routing in Self-Organizing Networks using Machine Learning<sup>8</sup>

### 4.1. Introduction

A central challenge for emerging wireless communication networks, beyond the promise to deliver faster speeds and greater connectivity, is to optimize the ability of wireless network service providers to efficiently deliver the required user capacity over the available spectral resources. A self-organizing network (SON) is recognized as central to capacity optimization of mobile networks [37] and one of the promising ways to autonomously and intelligently manage SONs is by integration with machine learning and UC (user-centric) technology.

This chapter applies the three-layered approach represented by the synergistic integration of SON, ML, and UC technologies to propose a methodology called user-specific optimal capacity and shortest path (US-OCSP) for load balancing and capacity optimization in self-optimizing SON systems. US-OCSP performs user-specific dynamic routing to find the shortest path with optimal capacity given a source and destination. In this dissertation the routing is considered optimal when the routing makes the most efficient use of the resources available given the constraints and limitations related to the available network nodes and routes going from the source to destination. The methodology uses the percentage of allocated physical resource blocks (PRBs) to evaluate the available capacity of 4G/5G network nodes (eNodeB/gNodeB) and uses  $Q$ -learning, an ML

---

<sup>8</sup> The contents of this chapter have been published in [4]. Permissions are included in Appendix A.

reinforcement learning technique, to determine the shortest path that meets the capacity needs of a user in a SON network.

US-OCSP can dynamically route the user traffic through non-congested network nodes, with minimal compromise on the subscriber's capacity, thus facilitating effective resource management to attain customer satisfaction and alleviate network congestion by developing an in-built application that is coordinated with an automobile or mobile phone's navigation system (GPS) where a network route provided by US-OCSP is driven by the topography based on a GPS mapping system linking GPS and optimal network routing.

#### **4.2. Load Balancing and Capacity Optimization**

Load balancing and capacity optimization in a SON is critical to efficiently deliver the required user capacity over the available spectrum resources. Load balancing optimization is a SON function where cells in a congested state can transfer part of the user traffic to other cells that have spare resources [7]. The radio resource status reports can help determine the physical resource block<sup>9</sup> (PRB) utilization and the available nodal capacity that can be further used for load balancing and capacity optimization in the network [11].

The prior art methods to achieve load balancing and capacity optimization include a channel borrowing mechanism, handover-based approaches, and antenna tilt optimization [38]. In channel borrowing, a cell can borrow a fixed number of channels from adjacent cells, but if the adjacent cells do not have enough resources to share, this can lead to even more congestion. In handover-based approaches, the user equipment (UEs) are handed off between the serving and

---

<sup>9</sup> In LTE and NR, one subcarrier and one OFDM symbol forms a resource element. A physical resource block (PRB) is a group of resource elements such that it consists of 12 consecutive subcarriers across one slot. A PRB is the smallest radio resource unit used for resource allocation in 4G and 5G networks.

neighboring cells, but there is a possibility that this can give rise to a “ping-pong” effect<sup>10</sup> causing instability and an increase in the occurrence of handover drops. In antenna tilt optimization, tilt adjustments are made to optimize coverage areas of the serving and neighboring cells, but these generally have a limited range and in case of equipment failure, it may require several days for repairing or replacing the physical parts. These network-centric approaches can be further enhanced by implementing a user-centric approach, where the shortest path with available optimal capacity is pre-determined based on the current network state and recommended to the end-user given its source and destination (much like a GPS offers recommended routes).

### **4.3. The Methodology**

The legacy methods lack a user-centric approach while transferring load from congested network nodes (base stations) to other nodes, which may not always lead to an improved user experience. Non-UC approaches would move users from one network node to other to reduce network congestion and while doing this some users may receive improved service, but some may face degraded service, but a UC-approach will strive to ensure every user gets good service. In US-OCSP, which is a user-centric methodology, the optimization begins at the end-user level and strives to find the shortest path available that traverses through non-congested network nodes and recommends that route to the end-user given its source and destination. An in-built application that is coordinated with an automobile or mobile phone’s navigation system (GPS) can be developed where a network route provided by US-OCSP is driven by the topography based on a GPS mapping system linking GPS and optimal network routing.

---

<sup>10</sup> When UEs are in a loop where they are repeatedly handed off between the source and the target base stations, the resulting effect is called the “ping-pong” effect.

The proposed methodology takes a UC-approach that tailors the capacity needs of the end-user to find the shortest path with optimal capacity for a given source and destination. Capacity is measured by the availability of resources (i.e., PRBs) at all possible serving network nodes between the source and destination. A reinforcement machine learning algorithm implemented (i.e.,  $Q$ -learning) determines the shortest path avoiding congested network nodes so as to achieve the required throughput and/or bit rate. In other words, under the assumption that a user will be served by multiple network nodes while moving from its source to destination no matter what route it takes, the proposed methodology will give the user optimal throughput by selecting a path with the least viable distance that goes through the network nodes to reach that destination and that have adequate availability of resources (PRBs) to serve the user. It avoids selecting a path that goes through congested network nodes that have very high PRB utilization. So, if the user takes the recommended path, the user will be able to achieve an optimal throughput/ bit rate, consistent with the definition of “optimal” on page 1 of this chapter.

An example scenario could work as follows: the driver’s GPS, or mobile phone, provides options with multiple paths going from a source to destination that are relayed to the wireless network and then the US-OCSP algorithm finds the shortest network node path with non-congested nodes recommending a routing consistent with at least one of the GPS recommended paths. The application in autonomous vehicles could be quite important. The methodology can be used to further enable dynamic network path optimization, such that if the user changes its inputs, such as GPS route, the US-OCSP would recompute the network path in response to the changes in the user inputs. Thus, US-OCSP can help build a navigation system that will allow users to pick a route with less congested network traffic and help network operators with resource optimization.

The visual map depicted in Figure 4.1 is used to illustrate US-OCSP. Given a source and a destination of a user, US-OCSP first determines the available capacity of all 4G/5G base stations i.e. eNodeBs (eNBs)/ gNodeBs (gNBs) in the region of service that could potentially serve the user. This is done by calculating the PRB utilization of each eNB/gNB. PRB utilization is a performance measurement typically used for 4G/5G networks that provides the total usage (in percentage) of physical resource blocks, per node, and is defined as [39]

$$\text{PRB utilization} = M(T) = \frac{M1(T)}{P(T)} * 100 \quad (4.1)$$

where  $M(T)$  is the percentage of PRBs used, averaged during a time period  $T$  with value range: 0 – 100%,  $M1(T)$  is the count of all PRBs used,  $P(T)$  is the total number of PRBs available during time period  $T$ , and  $T$  is the time period during which the measurement is performed. The periodicity of the radio resource status report can be typically requested in the range of 1 to 10s [11].

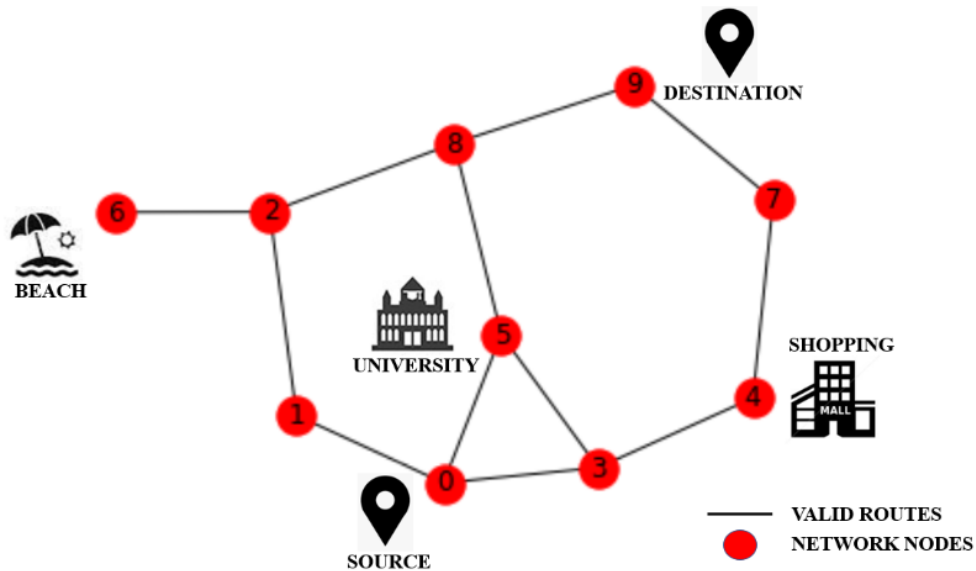


Figure 4.1 Example network topology to illustrate the US-OCSP methodology

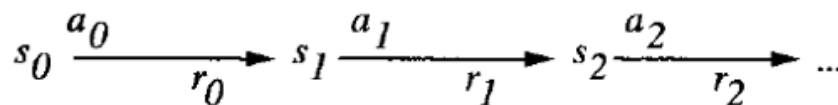
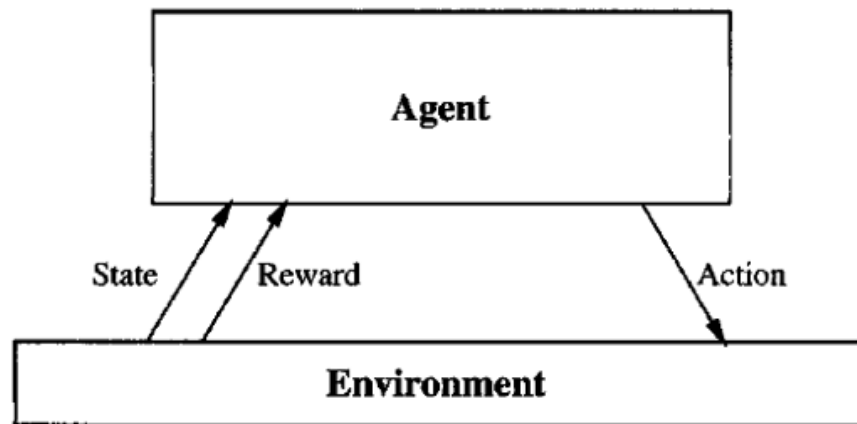


The network scenario described in Figure 4.1 was simulated using the network simulator ns-3 [24]. One hundred UEs that generate constant bit rate traffic are randomly placed in the network and are connected to the closest of the 10 network nodes. Areas with high user concentration are denoted by beach, university and shopping mall symbols in the visual map. PRB utilization is calculated for every network node to evaluate its capacity. A threshold of 70% is set such that a network node with PRB utilization above the threshold is declared to be “busy” while a network node with PRB utilization below the threshold is declared to be “available.”

US-OCSP uses a machine learning algorithm called  $Q$ -learning, a form of reinforcement learning, to determine the shortest path to be taken by the end-user from its source to destination. In selecting ML algorithms, the first step is to understand the problem at hand by assessing the relevant data available to determine the general category of ML algorithm that will be used - supervised, unsupervised, or reinforcement. Since ML learns from examples, the approach is to train a model by making the best possible use of the data available. If expected output information is available for training, supervised learning is preferable. But when the target labels/values are not available the next best option is to find out if a reward mechanism can be applied to the data. In case of US-OCSP, the expected/desired output is not available, but a reward/penalty can be associated for every state-action pair in accordance with some performance criterion, hence allowing the use of reinforcement learning.

Reinforcement learning addresses how an autonomous agent that senses and acts in its environment can learn to choose optimal actions to achieve its goals as described in Figure 4.2 and explained as given below [17]. Each time the agent performs an action  $a_j$  from a set of possible actions  $A$  in some state  $s_j$  in an environment described by a set of possible states  $S$ , the agent receives a reward or penalty  $r_j$  that represents an immediate value of the state-action transition

indicating the desirability of the resulting state. This generates a sequence of states  $s_i$ , actions  $a_i$ , and immediate rewards  $r_i$  as shown in Figure 4.2, where  $s_0, s_1, s_2$  represent states,  $a_0, a_1, a_2$  represent actions,  $r_0, r_1, r_2$  represent immediate rewards, and  $\gamma$  represents the discount factor. The task of the agent is to learn a control policy,  $\pi : S \rightarrow A$ , that would maximize the expected sum of rewards, with future rewards discounted exponentially by their delay. The discount factor is denoted by  $\gamma$ . It represents the degree to which the rewards hold their value over time. When  $\gamma$  is close to 0, the algorithm tends to consider only immediate rewards but as  $\gamma$  gets closer to 1, the algorithm strives for a long-term high reward.



**Goal: Learn to choose actions that maximize**

$$r_0 + \gamma r_1 + \gamma^2 r_2 + \dots, \text{ where } 0 \leq \gamma < 1$$

Figure 4.2 Representation of a reinforcement learning system [17]

The  $Q$ -learning algorithm is described in Table 4.1 and can be explained as follows [17]:

Table 4.1  $Q$ -learning algorithm<sup>11</sup>, assuming deterministic rewards and actions [17]. The discount factor  $\gamma$  may be any constant such that  $0 \leq \gamma < 1$ .

For each  $s, a$  initialize the matrix entry  $\hat{Q}(s, a)$  to zero.

Observe the current state  $s$

Do forever:

- Select an action  $a$  and execute it
- Receive immediate reward  $r$
- Observe the new state  $s'$
- Update the matrix entry for  $\hat{Q}(s, a)$  as follows:

$$\hat{Q}(s, a) \leftarrow r + \gamma \max_{a'} \hat{Q}(s', a')$$

- $s \leftarrow s'$

In  $Q$ -learning, learning the  $Q$  function corresponds to learning the optimal policy. The evaluation function the agent attempts to learn is  $Q(s, a)$  such that the value of  $Q$  is the maximum discounted cumulative reward that can be achieved starting from state  $s$  and applying action  $a$  as the first action. In this algorithm, the learner represents its hypothesis  $\hat{Q}$  by a large matrix that consists of a separate entry for each pair of state and action. The matrix entry for  $(s, a)$  stores the value for  $\hat{Q}(s, a)$ , the learner's current hypothesis about the actual, but unknown, value  $Q(s, a)$ . The initial values of the matrix are set to zero. The agent recurrently observes its current state  $s$ ,

---

<sup>11</sup> The training rule in the algorithm can also be written as an iterative equation where the  $Q$  value at iteration " $n + 1$ " is :

$$\hat{Q}_{n+1}(s, a) = r + \gamma \max_{a'} \hat{Q}_n(s', a')$$

that could be interpreted as:

$$\hat{Q}_{n+1}(\text{current state, current action}) = r(\text{current state, current action}) + \gamma \max[\hat{Q}_n(\text{next state, next possible actions})]$$

chooses some action  $a$ , executes action  $a$ , then observes the resulting reward  $r = r(s, a)$  and the new state  $s' = \delta(s, a)$  where  $\delta$  denotes the state resulting from applying action  $a$  to state  $s$ . It further updates the matrix entry for  $\hat{Q}(s, a)$  following each such transition in accordance to the rule given by (4.2)

$$\hat{Q}(s, a) \leftarrow r + \gamma \max_{a'} \hat{Q}(s', a') \quad (4.2)$$

This training rule uses the agent's current  $\hat{Q}$  values for the new state  $s'$  to refine its estimate of  $\hat{Q}(s, a)$  for the previous state  $s$ .  $Q$ -learning propagates  $\hat{Q}$  estimates one step backwards i.e. each time the agent moves forward from a previous state to a new one,  $Q$ -learning propagates the computed  $\hat{Q}$  values backward from the new state to the old state. At the same time, the immediate reward received by the agent for the state-action transition is used to augment these propagated values of  $\hat{Q}$ .

After multiple iterations, the information that the agent collects will propagate from the transitions with non-zero rewards back through the entire state-transition space available to the agent, resulting eventually in a matrix that consists of the steady-state  $Q$  values. Using this algorithm, the agent's estimate  $\hat{Q}$  converges in the limit to the actual  $Q$  function, provided the system can be modeled as a deterministic and stationary Markov decision process, the reward function  $r$  is bounded, and actions are chosen such that every pair of state-action is visited infinitely often.

A significant aspect of  $Q$ -learning that makes it scalable is that it can be employed in an arbitrary environment where the agent or the learner has no prior knowledge of how its actions affect its environment. The agent is not required to be able to predict in advance the immediate

result for every possible state-action. Hence, the algorithm can be applied even if there are newly added states and actions.

Using the knowledge of PRB utilization gained from the output of the ns-3 simulation, US-OCSP has at its core the  $Q$ -learning algorithm, which was implemented in Python, and finds the shortest user path through the network that the user should take while in transit from a given source to destination to meet the capacity requirements of the user. A correspondence table of the network functions considered in US-OCSP with regards to the  $Q$ -learning parameters is given in Table 4.2.

Table 4.2 A correspondence table of the network mapping in US-OCSP with  $Q$ -learning parameters

<b><math>Q</math>-learning Parameters</b>	<b>Network Mapping</b>
$s$	A state corresponds to a network node (eNB/gNB).
$a$	An action corresponds to the agent's virtual movement from one network node to another.
$r(s, a)$	A reward is an immediate/instant value, or score received after every virtual move of the agent from one network node to another.
$Q(s, a)$	A $Q$ value is computed and refined recursively using the $Q$ -learning algorithm until the agent virtually reaches the network node that serves the destination.

Considering the network scenario described in Figure 4.1, the agent (virtual user) will explore different paths<sup>12</sup> going from the end-user's source to destination using US-OCSP to find the optimal path. Every time the agent moves from one network node to another, it will receive an immediate reward for the transition whose value depends on whether or not there is a valid link established between the two network nodes and how close or far the network nodes are from the destination. Initially the immediate reward matrix  $r$  is set to -1. If there is a valid path between two nodes, then the reward value is changed from -1 to 0 and if there is a direct connection from a node to the destination node, then the corresponding reward value is set to 100. The discount factor  $\gamma$  may be any constant such that  $0 \leq \gamma < 1$ . When  $\gamma$  is close to 0, the algorithm tends to consider only immediate rewards but as  $\gamma$  gets closer to 1, the algorithm strives for a long-term high reward. The  $Q$ -matrix is initialized to 0 and continues to refine the  $Q$  value until it reaches the network node that serves the destination.

The availability of network nodes is determined based on their PRB utilization derived from the ns-3 simulation. After multiple, recursive iterations, the information that the algorithm collects propagates from the transitions with non-zero rewards back through the entire state-transition space available, resulting eventually in the final  $Q$ -matrix obtained at the state of convergence. The rows of the  $Q$ -matrix represent the current state and the columns represent the possible actions leading to the next state. The algorithm finds the actions with the maximum reward values recorded in the matrix for the initial state and continues the hops until it reaches the destination state. The algorithm traces the best sequences of states by following the transitions associated with the highest values recorded in the final  $Q$ -matrix. The optimal path corresponds to

---

<sup>12</sup> In a real-world network, the network layout can be overlaid with a GPS highway/street map such that the node to node path recommended by the algorithm can be traced in consultation with a GPS mapping system.

selecting the links or routes with maximal  $Q$  values. In other words, the computed  $Q$  values<sup>13</sup> will help determine which node to node transitions should be selected to achieve the shortest path with optimal capacity.

#### 4.4. Performance Analysis and Evaluation

The example network model was simulated, and the network topology is fed as an input to the machine learning program. Table 4.3 provides the simulation parameters used to implement the network scenario described in Section 4.3 using ns-3. The output of the ns-3 simulation gives the modulation coding scheme (MCS) used and the transport block (TB) size for every user-network node pair per unit time. The output is further used to find the PRB utilization by referring to 3GPP standards [40] and implementing equation (4.1).

Table 4.3 Simulation set up parameters

Parameter	Value
Number of network users	100 (scalable) <sup>14</sup>
Number of network nodes	10 (scalable)
Channel Bandwidth	20 MHz (100 PRBs)
Scheduler	Token Bank Fair Queue Scheduler (TBFQ)
Traffic Type	Constant Bit Rate (CBR)

<sup>13</sup>  $Q$  is an evaluation function/utility function such that the value of  $Q$  for the current state and action summarizes in a single number all the information needed to determine the discounted cumulative reward that will be gained in the future if that state-action pair is selected [17].

<sup>14</sup> The term scalable in the table denotes that the number of users and networks nodes used to run the current network scenario can be changed to run a different network scenario. The simulator can scale up to tens of network nodes and hundreds of UEs. In a real network, the algorithm should be tested for a small county or area first and then expanded to validate the feasibility.

The ns-3 simulator supports QoS-aware packet scheduling where the fundamental unit for resource allocation is a physical resource block (PRB). The MAC scheduler generates specific structures called Data Control Indication (DCI) that are then transmitted by the physical layer of the network node to the connected UEs, in order to inform them of the resource allocation on a per subframe basis. In doing this in the downlink direction, the scheduler has to fill some specific fields of the DCI structure with all the information, such as the MCS to be used, the TB size, and the allocation bitmap which identifies which resource blocks will contain the data transmitted by the network node to each user. The scheduler implemented in this simulation is called Token Bank Fair Queue (TBFQ). TBFQ is a channel-aware/QoS-aware scheduler that is derived from the leaky-bucket mechanism, which guarantees the fairness by utilizing a shared token bank and can be explained as follows [8]: TBFQ maintains a shared token bank so as to balance the traffic between different flows. The user who contributes more to the token bank has a higher priority to borrow tokens, while the user who borrows more tokens from the bank has a lower priority to continue to withdraw tokens. In case of multiple users with same token generation rate, traffic rate, and token pool size, users suffering from severe interference and shadowing conditions get more opportunities to borrow tokens from the bank. TBFQ can police the traffic by setting the token generation rate to limit the throughput.

If the PRB utilization of a network node is found to be above 70%, it is declared as “busy” and if the PRB utilization of a network node is found to be below 70%, it is declared as “available” in terms of capacity. In accordance with this model, network nodes 4, 5, and 6 from Figure 4.1 are found as “busy” whereas network nodes 0, 1, 2, 3, 7, 8 and 9 are found as “available” based on the calculated PRB utilization values. The  $Q$ -learning algorithm is then implemented in Python for determining the shortest path between the end-user’s source and destination. When distance is the



only criterion used to determine the path that the end-user should take, the recommended path given by the algorithm goes via network nodes 0, 5, 8 and 9. This path is the shortest path that the end-user can take to reach the destination, but is not the most efficient path as it does not verify if all the serving nodes on this path have enough capacity available to serve the end-user. In order to find the most efficient path, the status of all the network nodes based on their PRB utilization derived from the ns-3 simulation is given as an input to the  $Q$ -learning algorithm. With this knowledge, the  $Q$ -learning algorithm takes into account not only distance but also the available nodal capacity (i.e., eNB or gNB availability of PRBs) while determining the most efficient or optimal path the end-user should take given a source and a destination. Subsequently, the most efficient path suggested by US-OCSP goes via network nodes 0, 1, 2, 8, and 9 based on the results shown in Figure 4.3. Thus, US-OCSP not only avoids all other possible paths that may be longer and more time-consuming, but it also avoids selecting paths that may be served by network nodes that due to congestion cannot meet the capacity (throughput and/or bit rate) needs of the end-user.

User-Specific Optimal Capacity Shortest Path:  
[0, 1, 2, 8, 9]

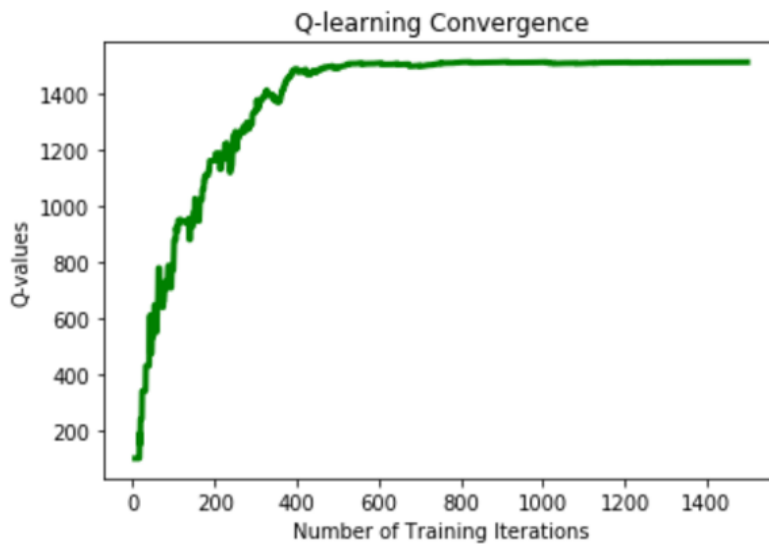


Figure 4.3 A graphical representation of the  $Q$ -learning curve converging towards the optimal solution.

#### 4.5. Concluding Remarks

This chapter introduced and proposed a methodology called US-OCSP that uses Machine Learning to determine the most efficient (“optimal”) path to be followed by an end-user in terms of distance and capacity in a SON network given its source and destination. Network capacity generally depends on the utilization of the network nodes and of the backhaul links. This research considers utilization (in terms of nodal PRB utilization) that is associated with the resource blocks available at the network nodes. It is assumed that the network has enough backhaul capacity available to support link utilization which is generally determined by the committed information rate (CIR) associated with the backhaul links. The ML algorithm used in this research is  $Q$ -learning, a form of reinforcement learning. The effectiveness of this methodology is demonstrated in a network scenario simulated in ns-3 followed by the ML implementation in Python. The results showed that the shortest path with optimum capacity is rapidly determined, so that network dynamics can be accommodated. This methodology can help network providers to meet the end-user demands by finding the most efficient path and to optimize network resource allocation. The proposed method assumes that some of the network nodes and links have available capacity.

This chapter demonstrates the potential for implementing US-OCSP in future networks that will be autonomous and user-centric by incorporating ML in SON networks where the system can provide the most optimal path for end-users while moving from a given source to destination. An in-built application for navigation based on this methodology can play a significant role in future networks where US-OCSP can help build a navigation system that will allow users to pick a route with less congested network traffic and help network operators with resource optimization.

## **Chapter 5. Self-Configuration of Radio Access Network-Based Notification Areas (RNAs) in Self-Organizing Networks using Machine Learning<sup>15</sup>**

### **5.1. Introduction**

The merging of the digital and physical worlds is giving rise to a Fourth Industrial Revolution [41] where a diverse range of applications and services will require a radically new communication network architectural design. Such networks, such as 5G and 6G wireless networks are expected to be self-organizing networks (SON) to minimize OPEX and CAPEX, and as these networks expand, it is anticipated that there will be a need to extend the scope of network automation functionalities. This will likely include integration of new capabilities, such as self-learning via machine learning (ML) and user-centric (UC) technologies that can leverage the data generated across the network and better meet user expectations and experiences enabling intelligent self-learning decision-making mechanisms that can manage network complexities and improve network performance and efficiency.

In order to comply with the challenging demands of emerging 5G/6G applications and services that require lower latency and improved capacity, a key design consideration is to minimize the dependencies between the radio access network (RAN) and the core network (CN). The 5G RAN evolution includes the introduction of a novel radio resource control (RRC) state called RRC inactive and a RAN-based notification area (RNA). An RNA may constitute cells covered by one or more 5G network nodes (base stations), referred to as gNodeBs (gNBs) and an

---

<sup>15</sup> A portion of this chapter is pending publication.

effective configuration of RNA clusters will be crucial in attaining optimum network performance by improving the signaling load, network capacity, latency, and power consumption.

This chapter proposes a conceptual framework to enable self-configuration and management of RNAs in a hybrid SON domain. This methodology applies the three-layered approach represented by the synergistic integration of SON, ML, and UC technologies towards developing an adaptive mechanism that can self-configure and reconfigure RNAs to improve network signaling load and capacity. The chapter provides an overview of the RRC state handling and transitions followed by the key RNA configuration factors considered in this dissertation. Performance analysis and evaluation of the proposed technique for RNA configuration is demonstrated using a case study. The study optimizes the RNA design by balancing the tradeoff between maximizing the paging load improvement and minimizing (the probability of) cluster variance. Future research directions are provided to further improve the RNA configuration technique to achieve more robustness and resilience in larger, more practical networks.

## **5.2. RRC State Handling and Transitions**

A typical LTE network has two RRC states, RRC connected and RRC idle. The RRC connected mode is activated during data transfer and the UE enters RRC idle mode when there is no data to be transmitted or received. A 5G network is expected to encounter a large amount of random aperiodic and keep-alive traffic generated by a plethora of autonomous applications and services supported by 5G that will cause several RRC state transitions, adversely affecting the signaling and paging load, latency, power consumption, and capacity of the network. A new RRC state, RRC inactive, has been introduced in the 3GPP standards to address these issues [42]. An overview of the RRC state transitions [42] in a 5G network is illustrated in Figure 5.1.

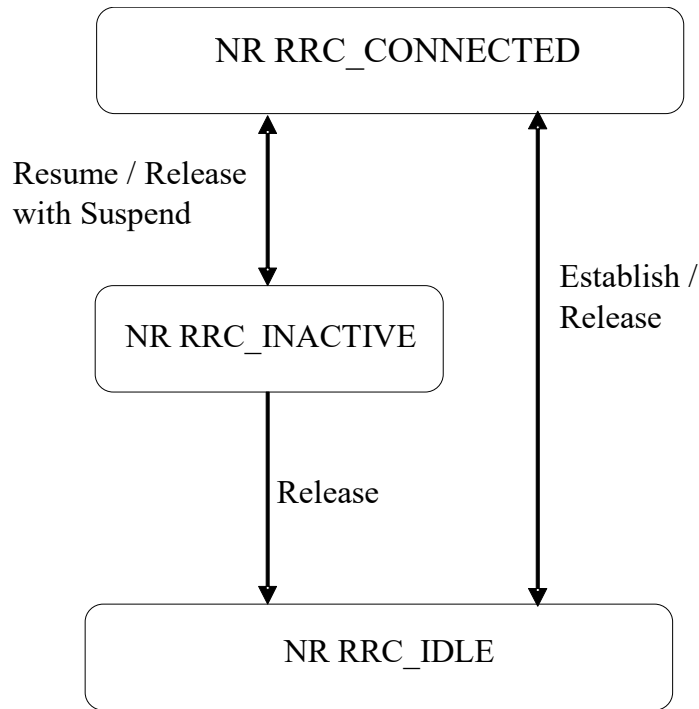


Figure 5.1 The RRC state transitions in a 5G network [42]

A UE is either in the RRC connected state or in the RRC inactive state when an RRC connection has been established, but if this is not the case, i.e. no RRC connection is established, the UE is in the RRC idle state [42]. The differences and similarities in the characteristics of the 5G RRC states [42] are noted in Table 5.1.

The transitions from the RRC idle to the RRC connected state are expected to occur mainly when a UE first attaches to the network, while the transitions from the RRC inactive to the RRC connected are expected to occur frequently and are optimized to be fast and lightweight in terms of signaling achieved by keeping the CN-RAN connection alive during the inactivity periods allowing the UE to move around within a pre-configured area (the RNA) without notifying the network [43].

Table 5.1 Characteristic differences and similarities of the 5G RRC states

<b>RRC Idle</b>	<b>RRC Inactive</b>	<b>RRC Connected</b>
UE controlled mobility based on network configuration.	UE controlled mobility based on network configuration.	Network controlled mobility within NR (New Radio) and to/from E-UTRA (Evolved Universal Terrestrial Radio Access).
The UE monitors a paging channel for CN paging.	The UE monitors a paging channel for CN paging and RAN paging. A RAN-based notification area is configured by RRC layer. The UE performs RAN-based notification area updates periodically and when moving outside the configured RAN-based notification area.	The UE monitors control channels associated with the shared data channel to determine if data is scheduled for it. The UE provides channel quality and feedback information.
UE's Access Stratum <sup>16</sup> context is discarded.	UE's Access Stratum context is stored.	UE's Access Stratum context is stored.
A UE specific DRX (Discontinuous Reception) <sup>17</sup> may be configured by upper layers.	A UE specific DRX may be configured by upper layers or by RRC layer.	At lower layers, the UE may be configured with a UE specific DRX.

### 5.3. Key RNA Configuration Factors

One of the most important factors to consider while configuring RNA clusters is analyzing the user activity by means of UE-gNB connections, as this has a direct impact on the signaling

<sup>16</sup> The Access Stratum is located between the edge node of the serving network domain and the UE domain and provides services related to the transmission of data over the radio interface and the management of the radio interface [44].

<sup>17</sup> A power saving feature where paging cycles can range from seconds to several hours, depending on the radio access technology [45].

load. The radio frequency (RF) conditions will help select the boundaries of the RNA clusters and hence, it is important to incorporate the reference signal received power (RSRP) and signal-to-noise-and-interference ratio (SINR) conditions of the user connections in the network. The RSRP measurements help in determining the path loss and SINR measurements can be used to ensure good cluster throughput. Another aspect that is critical in RNA cluster formation is the paging load. In LTE, paging is a CN function that is envisaged to be moved into the RAN in 5G by taking advantage of the RRC inactive state and the RNAs, thus allowing RAN controlled paging initiation procedures [43]. A RAN-initiated paging and a CN-initiated paging procedure for a 5G network can be described as shown in Figure 5.2 and Figure 5.3 respectively [46].

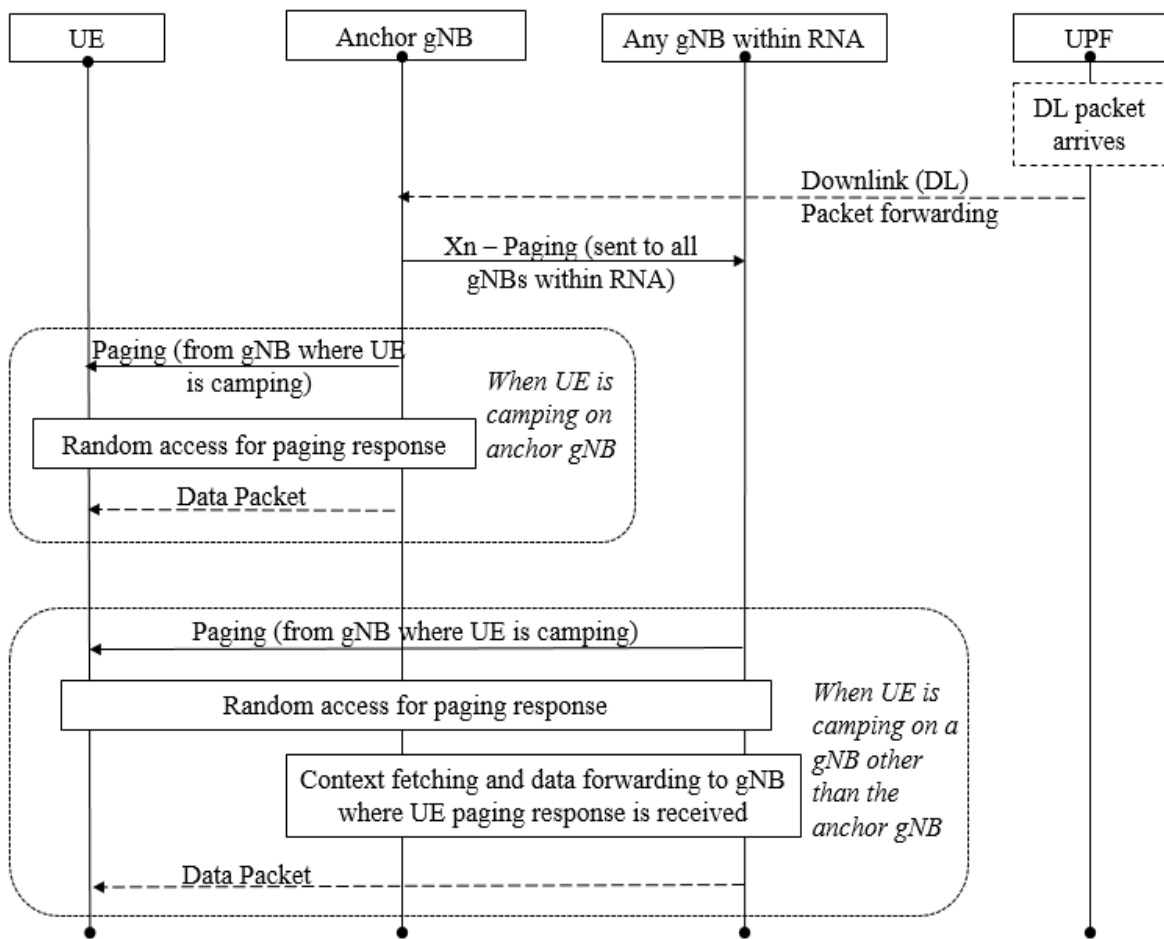


Figure 5.2 RAN-initiated paging procedure for a 5G network

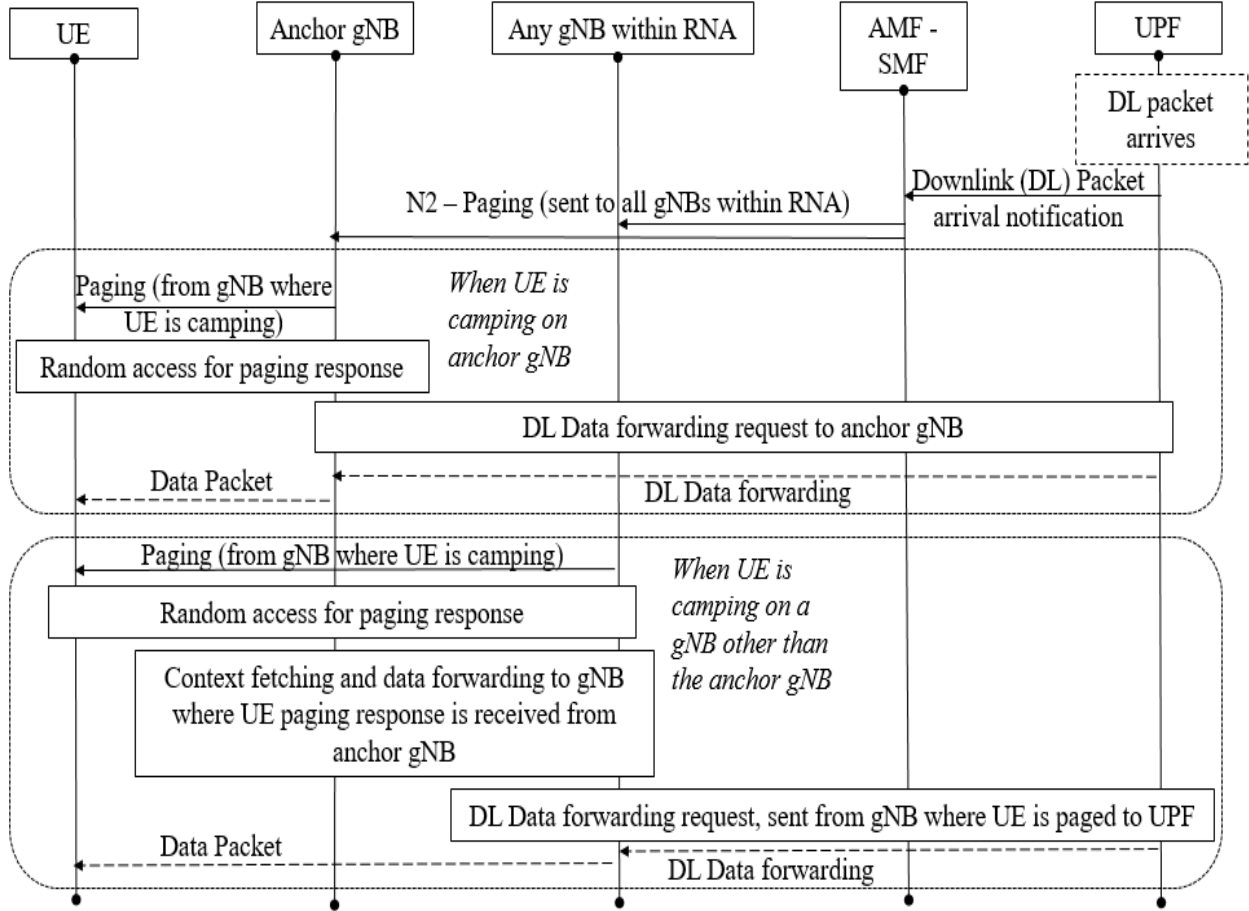


Figure 5.3 CN-initiated paging procedure for a 5G network

As a generic example consider a network with  $M$  cells and  $N$  gNBs per RNA. The paging load (in terms of the number of messages) in the RAN-initiated paging and the paging load in the CN-initiated paging are given below [46]:

$$\text{RAN initiated paging load} = \underbrace{M}_{\text{messages over radio}} + \underbrace{(N - 1)}_{\text{messages over } Xn} \quad (5.1)$$

$$\text{CN initiated paging load} = \underbrace{M}_{\text{messages over radio}} + \underbrace{(N + 3)}_{\text{messages over } N2} + \underbrace{3}_{\text{messages over } N4, N11} \quad (5.2)$$

where  $Xn$  is the interface between gNBs,  $N2$  is the interface between the RAN and AMF (Access and Mobility Management Function),  $N4$  is the interface between the SMF (Session Management Function) and UPF (User Plane Function), and  $N11$  is the interface between the AMF and SMF.



#### **5.4. Performance Analysis and Evaluation**

The network dynamics in 5G that have paved a way to a variety of new services and applications that expect efficient management of the signaling load, bandwidth, latency, and power requirements making it more and more critical to minimize the inter-dependencies of CN and RAN where a converged CN with a common CN-RAN interface integrates multiple RAN networks [8] supporting independent functioning of the RANs enabling a significant reduction of the signaling overheads. The improved modularity and reduced signaling load enable a more resourceful and efficient use of network resources.

The RAN has long been the most complex and dynamic part of the mobile wireless communications network and an effective configuration and management of RNA via the application of ML will help curb the compounding network complexities introduced by the emerging 5G services and demands. That said, RNA configuration is a good candidate for network operators to apply ML as an ML-fueled RNA solution will have the ability to learn about the user characteristics and locations and study the impact of radio conditions and network load and use this knowledge towards intelligently and adaptively constituting and dynamically evolving the RNAs to improve the overall network capacity.

This section discusses a case study performed to demonstrate and evaluate the performance, feasibility, and the potential benefits of the proposed RNA clustering mechanism using simulation as depicted in the process flow diagram shown in Figure 5.4. For this study, a simulated network using ns-3 [24], consisting of several users being served by multiple network nodes representing 4G/5G base stations (network nodes), is configured as specified in Table 5.2.

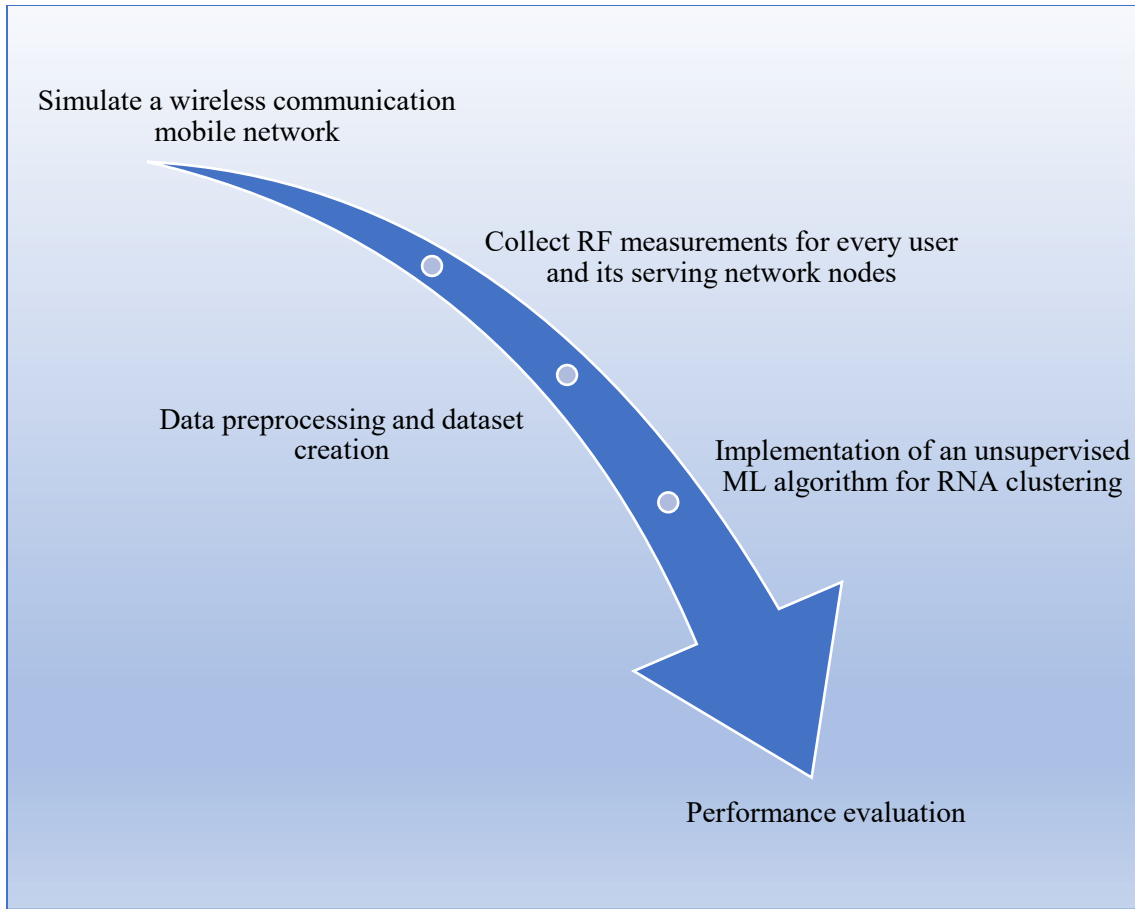


Figure 5.4 Process flow diagram for the demonstration and evaluation of the proposed RNA clustering mechanism<sup>18</sup>

The maximum transmission power of the outdoor network nodes is up to 40 watts and the transmission power of the network nodes that are located indoors is up to 20 watts. The channel bandwidth considered is 20 MHz and the testing frequencies<sup>19</sup> include 700 MHz and 2.6 GHz. The proportional fair algorithm is used in the simulation for scheduling purpose [47].

<sup>18</sup> Ideally, data preprocessing would be performed by extracting network metrics and measurements from a real-world cellular network. In this study, the results from a network-simulator are used.

<sup>19</sup> It is expected that the proposed mechanism would support millimeter wave (mmW) frequencies, but this needs to be tested.

The radio propagation model used is Cost231 [48] that is designed to cover a broad range of frequencies to predict path loss for outdoor scenarios in urban areas and is expressed in (5.3), (5.4), and (5.5)

$$L = 46.3 + 33.9 \log f - 13.82 \log h_b + (44.9 - 6.55 \log h_b) \log d - F(h_m) + C \quad (5.3)$$

such that  $F(h_m)$

$$= \begin{cases} (1.1 \log(f)) - 0.7 h_m - (1.56 \log(f) - 0.8) & \text{for medium and small size cities} \\ 3.2 (\log(11.75 h_m))^2 & \text{for large cities} \end{cases} \quad (5.4)$$

$$C = \begin{cases} 0 \text{ dB} & \text{for medium - size cities and suburban areas} \\ 3 \text{ dB} & \text{for large cities} \end{cases} \quad (5.5)$$

where  $f$  is the frequency (megahertz),  $h_b$  is the height of the base station (meters),  $h_m$  is the height of UE,  $d$  is distance (kilometers), and  $\log$  is a logarithm in base 10.

The indoor scenarios are mimicked by creating a building with user-defined dimensions and attributes and applying hybrid buildings propagation loss model that is a combination of several well-known path loss models. The users are allocated random positions and the mobility model used is the 2D random walk mobility [49], where each user moves with certain speed and direction chosen at random until a certain amount of time after which the users randomly change their positions and directions.

The standardized A3 RSRP handover algorithm [50] implemented utilizes the RSRP measurements and the A3 event is triggered when the UE perceives that a neighbor cell's RSRP value is better than the serving cell's RSRP value by an offset at which point the handover occurs. The RSRP and SINR statistics are collected during the entire simulation run for every user and its serving network nodes.

Table 5.2 Simulation parameters and values<sup>20</sup>

<b>Parameters</b>	<b>Values</b>
Number of network nodes	10 (scalable)
Number of users (UEs)	25 (scalable)
Transmission power of network nodes	Up to 40 watts (if outdoors), Up to 20 watts (if indoors)
Channel bandwidth	20 MHz
Frequency	700 MHz and 2.6 GHz
Scheduling algorithm	Proportional Fair
Radio propagation models	Cost231 (for outdoor), Hybrid building (for indoor)
Distribution of UEs	Randomly distributed
Mobility Model	2D random walk mobility
Handover algorithm	A3 RSRP

During data preprocessing, thresholds for RSRP and SINR conditions utilize measurements that are within an acceptable range of radio conditions so that the RNA clusters are configured to meet minimum allowable range of signal strength and throughput requirements and reduce the ping-pong effect at cluster boundaries. The processed data defines the relationship between every network node and users (e.g., the connectivity status for every UE-network node connection) and the size of the dataset is defined by the number of network nodes and users. The network model used for clustering is depicted in Figure 5.5.

<sup>20</sup> These values are applicable for 4G and 5G (non mmW) nodes.

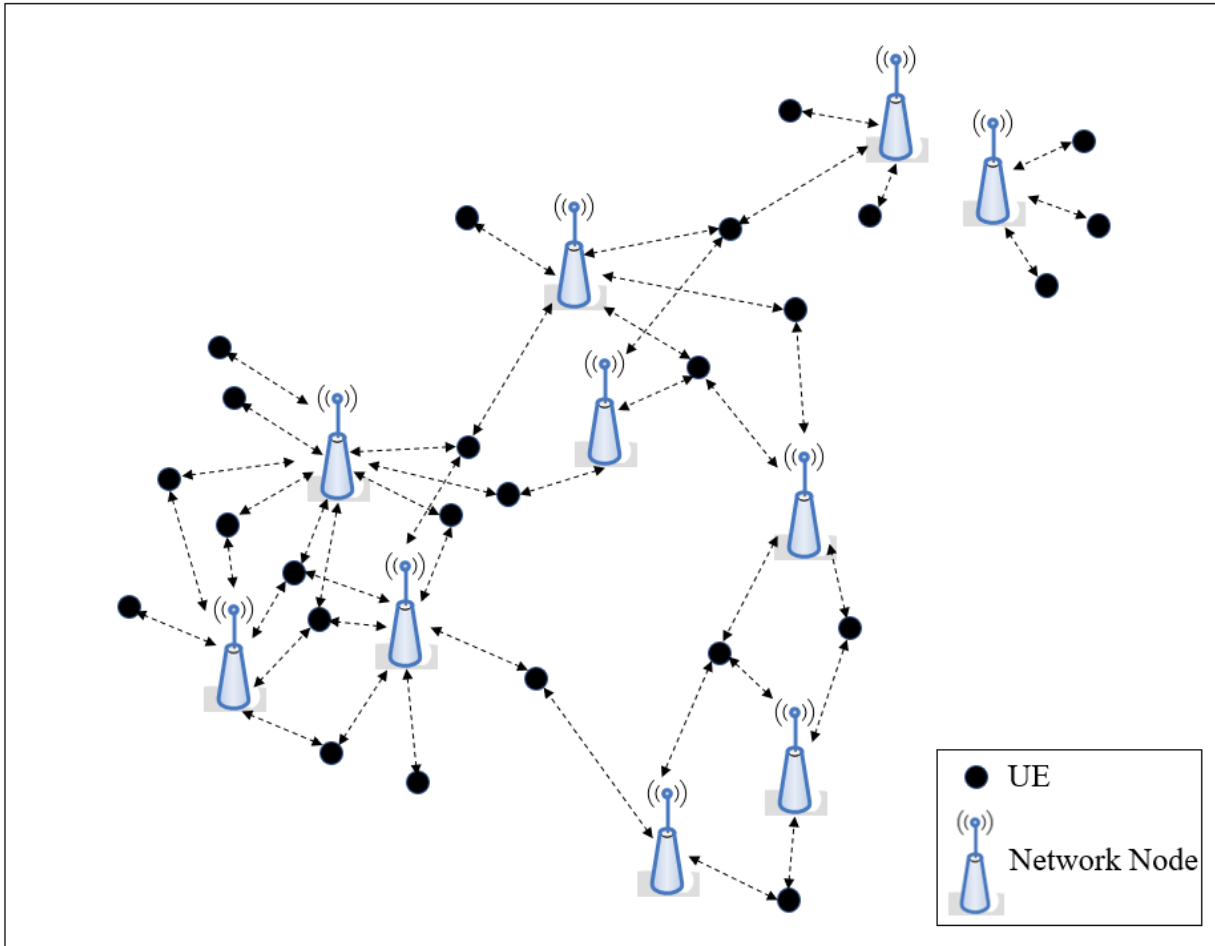


Figure 5.5 The network model used for clustering (The arrows represent the connectivity status of the mobile UEs with the network nodes as they move around in the network.)

During the ns-3 simulation, the mobile users are connected to the closest network node as they move around in the network. The UE-network node association for every user and network node in the network model during the entire simulation is captured in the input dataset prepared for the ML application in order to take a UC approach towards RNA configuration, since the UE-network node association accounts for the user connectivity and mobility status. The resulting RNA clusters represent the averaged best results obtained based on the UE to network node associations captured during the network simulation. The clusters are formed such that each network node belongs to only one cluster.

In selecting ML algorithms, the first step is to understand the problem at hand by assessing the relevant data available to determine the general category of ML algorithm that will be used - supervised, unsupervised, or reinforcement. Since ML learns from examples, the approach is to train a model by making the best possible use of the data available. If the expected output information is available for training, supervised learning is preferable. But when the target labels/values are not available the next best option is to find out if a reward mechanism can be applied to the data in which case, reinforcement learning is applied. But if none of these options are available, the next preferred option is to identify patterns and derive correlations between the data samples to make the best possible sense of the data and make predictions using unsupervised learning which is what is applicable here.

The  $k$ -means algorithm [23], an unsupervised machine learning algorithm is implemented in Python to operate on the dataset for cluster analysis. The  $k$ -means algorithm clusters data by dividing a set of samples of a dataset into  $k$  disjoint clusters, each described by the centroid (mean) of the samples in the cluster. The algorithm as described below strives to choose centroids that minimize the inertia or within-cluster sum-of-squares criterion. The algorithm works as follows:

- Create a dataset, or matrix, of dimensions defined by the number of network nodes and the number of users to represent the relationship (i.e., the connectivity status) between every network node and user under nominal conditions for a given time period. (The matrix entries are populated over time and are set to zeroes and ones to show the association between every network node and user and capture the user mobility status.)

- For an initial setting of  $k$  clusters, choose  $k$  samples from the dataset to select the initial centroids (i.e. the initial cluster centers are selected using the  $k$ -means++ initialization method<sup>21</sup>).
- Repeat the steps below until convergence, which occurs when the centroids stop changing.
  - For all the data samples, find the closest (in the sense of Euclidean distance) centroid.
  - Create new centroids by taking the mean value of all of the samples assigned to each preceding centroid and compute the difference between the previous and new centroids.

In this study, the determination of the “optimum” number of clusters is performed by using the silhouette analysis, and the comparison between the RAN-initiated and CN-initiated paging loads calculated via (5.1) and (5.2). The optimum number of clusters in this case study are determined with an objective to maximize the paging load reduction, subject to the constraints of the silhouette scores. The silhouette analysis [52] measures the quality of clustering by studying the separation distance between the resulting clusters to validate the consistency within them. The silhouette coefficients have a range of  $[-1, +1]$  such that the worst value is  $-1$  and the best value is  $+1$ . A value of  $+1$  indicates that the sample is far away from the neighboring clusters. A value of  $0$  indicates that the sample is on or very close to the decision boundary between two neighboring clusters and a value of  $-1$  and all other negative values indicate that those samples might have been assigned sub-optimally to a cluster. The silhouette coefficient is calculated using the mean intra-cluster distance  $a(i)$  and the mean nearest-cluster distance  $b(i)$  for each sample  $i$ .

$$s(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}} \quad (5.6)$$

---

<sup>21</sup> The  $k$ -means++ algorithm can be explained using the steps below [51].

- Select the first center such that it is chosen uniformly at random from all the data points.
- Select a second center with probability  $\frac{D(x)^2}{\sum_{x \in X} D(x)^2}$  where  $D(x)$  denotes the shortest distance from a data point to the first center chosen. Repeat this step until  $k$  cluster centers are chosen.

The average silhouette score provides a measure of clustering validity and is used to select an ‘appropriate’ number of clusters<sup>22</sup>. The silhouette analysis and the paging load reduction obtained after comparing the RAN-initiated paging over CN-initiated paging were plotted and the optimum value for the number of clusters,  $k_c$ , is found to be equal to 4 such that it gives a balanced tradeoff between maximizing the average paging load reduction and maximizing the average silhouette score in this study as depicted in Figure 5.6. The silhouette analysis prevents the number of clusters from becoming arbitrarily large and the paging load reduction prevents the number of clusters from becoming arbitrarily low. So, in this study, the optimum number of clusters, in the sense defined above, is determined and for this number of clusters that design of the clusters is also determined.

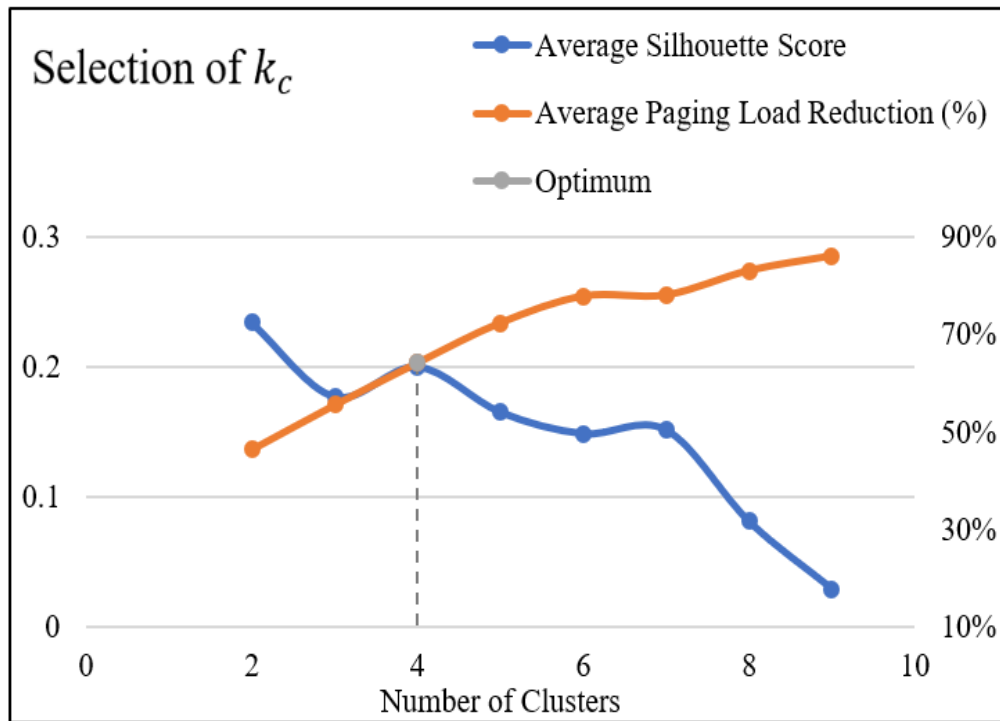


Figure 5.6 Performance evaluation for the selection of  $k_c$

<sup>22</sup> The silhouette analysis validates if there are data samples that are sub-optimally assigned to clusters and verifies if the clustering is within the acceptable limits where the silhouette scores are within a range of 0 to 1.



A limitation of the  $k$ -means algorithm is that it may not succeed in optimizing the centroid locations globally and can get stuck at a local minimum [53]. To address this, a more powerful ML algorithm, spectral clustering [18] [22] is implemented that allows to take a more general and practical approach for clustering. In spectral clustering, there are no issues of getting stuck in local minima or restarting the algorithm for several times with different initializations [22].

Instead of clustering in the original space, the data is first mapped to a new space such that similarities are made more apparent using Laplacian Eigenmaps<sup>23</sup> [18] (spectral<sup>24</sup> embedding) to place the data instances in such a way that the similarities between neighboring instances are preserved and clustering is applied to a projection of the normalized Laplacian. In the original space, a local neighborhood is created such that the instances in the same neighborhood are defined by creating an affinity matrix using the kernel radial basis function (RBF) [16].

The matrix value of a similar pair of data instances  $\rightarrow 0$  and the value of a dissimilar pair of data instances  $\rightarrow 1$ . This has the effect that instances that are nearby in the original space, probably located within the same cluster, will be placed very close in the new space, whereas those that are some distance away, probably belonging to different clusters, will be placed far apart. The  $k$ -means clustering or a discretization approach to search for a partition matrix (clustering) that is closest to the eigenvector embedding is then run with the new data coordinates in the new space. The following steps describe the kernel-based clustering algorithm based on spectral embedding.

---

<sup>23</sup> Laplacian Eigenmaps is a feature embedding method; that is, it finds a low dimensional representation of the data such that it projects similar data instances nearby in the new space by capturing and preserving the local information.

<sup>24</sup> The main tools for spectral clustering are graph Laplacian matrices. There exists a whole field dedicated to the study of those matrices, called spectral graph theory. One of the main goals in graph theory is to deduce the principal properties and structure of a graph from its graph spectrum (or from a short list of easily computable invariants). The eigenvalues are closely related to almost all major invariants of a graph, linking one extremal property to another. Spectral graph theory starts by associating matrices to graphs, computes the eigenvalues of such matrices, and relates the eigenvalues to structural properties of graph [22], [54], [55].

- Create an affinity matrix using the kernel, radial basis function to transform the input dataset into graph representation.
- Construct the normalized graph Laplacian and solve the eigenvalue problem.
- Perform eigenvalue decomposition on the graph Laplacian to define a new subspace.
- Apply  $k$ -means/discretization to form clusters in this subspace.

In spectral clustering, although the similarities are local, they propagate [18]. Consider three instances,  $a$ ,  $b$ , and  $c$  where the instances  $a$  and  $b$  lie in the same neighborhood and so do  $b$  and  $c$ , but not  $a$  and  $c$ . As  $a$  and  $b$  will be placed close to each other and  $b$  and  $c$  will be placed close to each other,  $a$  will lie close to  $c$  too, and they will likely be assigned to the same cluster. Consider now when  $a$  and  $d$  are not in the neighborhood with too many intermediate nodes between them; these two will not be placed nearby and it is very unlikely that they will be assigned to the same cluster.

The  $k$ -means algorithm would work effectively for simple cluster formations, but spectral clustering can serve as an extension to  $k$ -means and would be preferred for more general problems. The application of  $k$ -means and spectral clustering both provided identical resulting clusters for the example network model used in this case study. For a more complex network, it is recommended to combine these unsupervised learning algorithms with deep learning<sup>25</sup>. The resulting RNA clusters formed in the simulated network model are depicted in Figure 5.7.

---

<sup>25</sup> Deep learning is a particular kind of machine learning that achieves great power and flexibility by representing the target system as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations are computed in terms of less abstract ones [13].

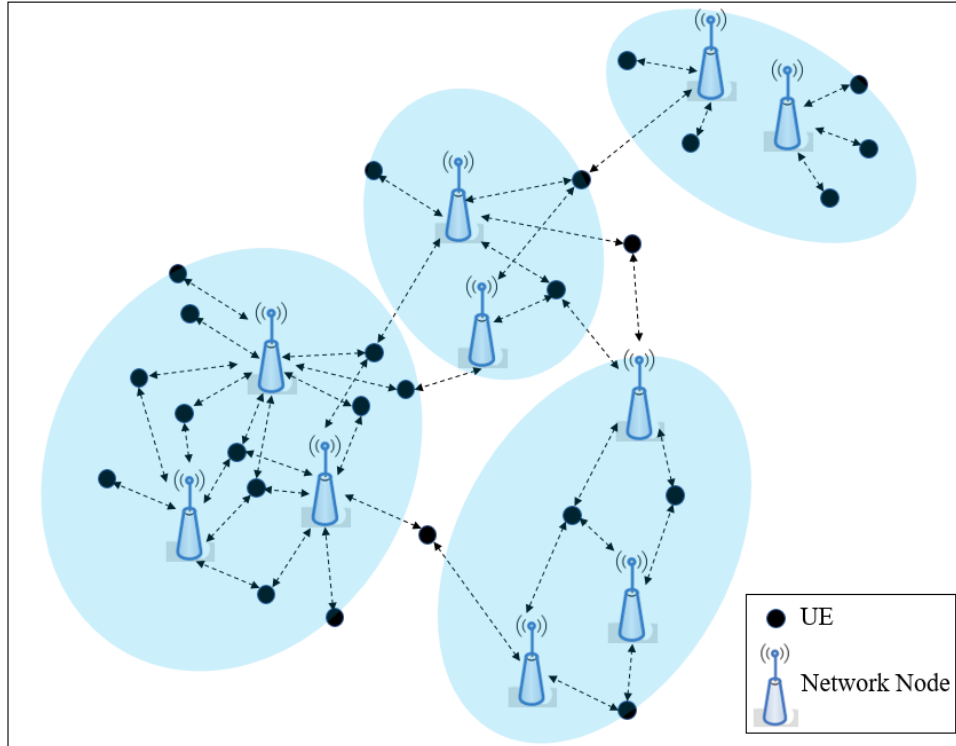


Figure 5.7 Resulting RNA clusters for the simulated network model

### 5.5. Future Research Directions

To improve network resilience and robustness, it is recommended that the RNA clusters should be monitored periodically and fine-tuned post initial clustering. This can be achieved by monitoring KPIs such as experienced user throughput, traffic volume density, end-to-end latency, reliability, availability, and retainability as described in Table 5.3 [56].

These KPIs are defined to enable network operators to support the needs of 5G use cases and services to facilitate a fair assessment and comparability of the different technical concepts considered for 5G. They are heavily dependent on the network conditions, such as available infrastructure and related radio resources, number of users, radio conditions, etc. that need to be considered. The computational capabilities and network scalability required to effectively embed these KPIs can be achieved by adopting an ML-based SON framework.

Table 5.3 Key performance indicators for RNA cluster performance monitoring and optimization

KPI	Description
Experienced user throughput	<ul style="list-style-type: none"> <li>• Experienced user throughput refers to an instantaneous data rate between Layer 2 and Layer 3.</li> <li>• Experienced user throughput is calculated as:  <math display="block">U_{Tput} = \frac{S}{T},</math>                     where <math>S</math> is the transmitted packet size and <math>T</math> is the packet transmission duration calculated as the difference between the time when the entire packet is correctly received at the destination and the time when packet is available for transmission.</li> </ul>
Traffic volume density	<ul style="list-style-type: none"> <li>• Traffic volume density is defined as the aggregated number of correctly transferred bits received by all destination UEs from source radio points (DL traffic) or sent from all source UEs to destination radio points (UL traffic), over the active time of the network to the area size covered by the radio points belonging to the RNA(s) where UEs can be deployed.</li> </ul>
E2E latency	<ul style="list-style-type: none"> <li>• E2E latency, or one trip time (OTT) latency, refers to the time it takes from when a data packet is sent from the transmitting end to when it is received at the receiving entity, e.g., internet server or another device.</li> <li>• Another latency measure is the round-trip time (RTT) latency which refers to the time from when a data packet is sent from the transmitting end until acknowledgements are received from the receiving entity.</li> </ul>
Reliability	<ul style="list-style-type: none"> <li>• Reliability accounts for the percentage of packets properly received within the given maximum E2E latency (OTT or RTT depending on the service).</li> </ul>
Availability	<ul style="list-style-type: none"> <li>• Availability in percentage is defined as the number of places (related to a predefined area unit which in this case would refer to an RNA) where the QoE level requested by the end-user is achieved divided by the total coverage area of an RNA.</li> </ul>
Retainability	<ul style="list-style-type: none"> <li>• Retainability is defined as the percentage of time where transmissions meet the target experienced user throughput or reliability.</li> </ul>

A conceptual framework to enable self-configuration and management of RNAs is proposed as depicted in Figure 5.8 that is analogous to a hybrid SON structure [7] where the centralized management system represented by the RNA controlling unit and the element management system represented by the anchor gNBs work together, in a coordinated manner, to build up a complete SON algorithm. The decisions on SON actions may be either made by the RNA controlling unit or the anchor gNBs<sup>26</sup>, depending on the specific cases. The RNA controlling unit consists of the RNA configurator and the RNA monitor. The initial RNA clustering mechanism illustrated in the previous section is implemented in the RNA configurator.

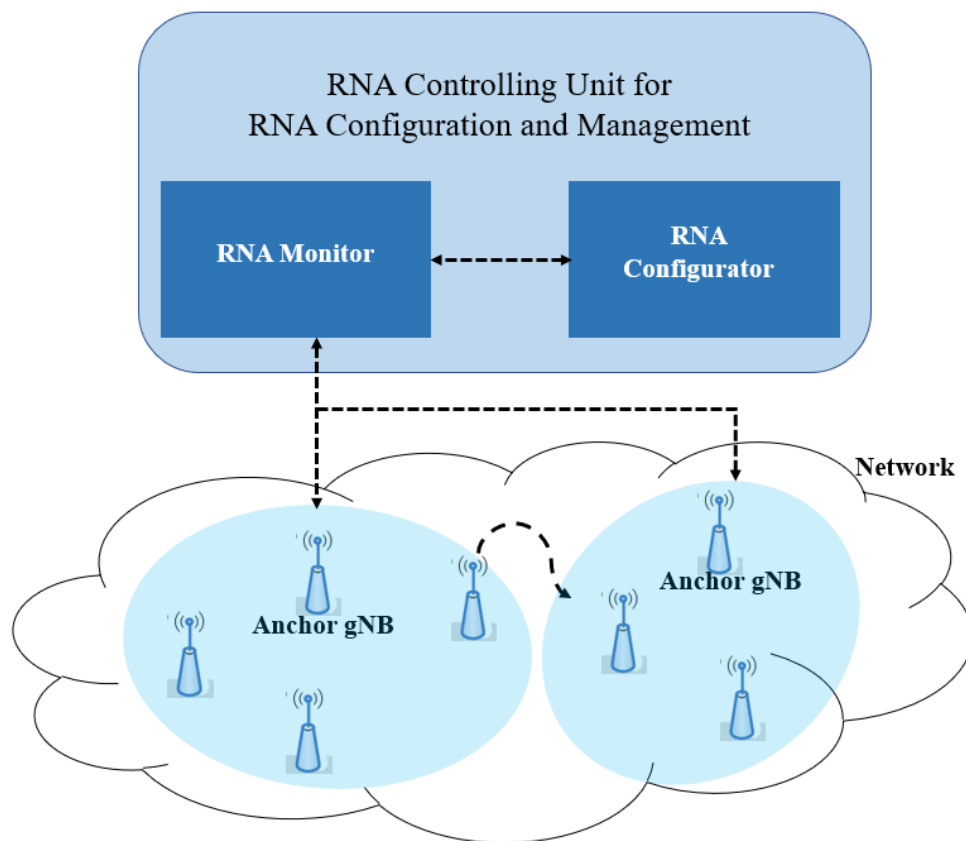


Figure 5.8 The proposed RNA configuration and management framework

<sup>26</sup> An anchor gNB is the network node that is aware of or has the list of all the gNBs that are a part of that RNA. It is the anchor gNB that maintains the CN-RAN connection and the UE context as the UE moves around within the RNA.

Once the initial clusters are formed, the RNA monitor would monitor the KPIs for each cluster. A cluster-level threshold margin can be set such that it would trigger either an addition or removal of a gNB from a cluster, or trigger re-initiation of RNA clustering depending upon the tolerance limit set for the threshold variations. If a gNB is moved from one cluster to its adjacent cluster in order to maintain an acceptable level of cluster performance, the anchor gNBs of the clusters that underwent the changes would relay this information to the RNA monitor so that the modified clusters are taken into account for future monitoring. When the RNA monitor detects threshold variations exceeding the tolerance limit set, the RNA monitor would trigger the RNA configurator to re-initialize and form new RNA clusters in the network.

## **5.6. Concluding Remarks**

In the 5G and beyond era, improved signaling and paging load to achieve reduced latency and improved capacity are key requirements for emerging use cases where appropriate configuration of the RAN-based notification areas will play a significant role as it will have a direct impact on controlling the RRC state transitions improving the network capacity and efficiency. An effective configuration and management of RNAs in a SON will be crucial in attaining optimum network performance by improving the signaling load and network capacity as the transitions from the RRC inactive to RRC connected states are expected to occur frequently in the emerging 5G/6G applications and services and are optimized to be fast and lightweight in terms of signaling achieved by keeping the CN-RAN connection alive during the inactivity periods allowing the UE to move around within an RNA. In this chapter, a self-learning mechanism that can dynamically configure RNA clusters is proposed, demonstrated, and evaluated enabling a user-centric smart RAN paging technique. Additional recommendations are made to optimize the RNA clusters by

means of performance monitoring of key performance indicators. A conceptual framework for effective RNA configuration and management in a hybrid SON system is proposed as a reference model to facilitate the development of the next-generation self-organizing 5G/6G networks.

## **Chapter 6. Summary**

The substantial increase in data-intensive applications and services have motivated significant industry efforts to develop the next-generation 5G/6G networks with advanced automation, intelligence, and user experience focused capabilities. This will require network operators to significantly alter the traditional models and technologies used in the previous generations and integrate machine learning (ML) and user-centric (UC) technologies to address the complexities of the next-generation self-organizing network (SON) deployment, performance assessment, and optimization. This dissertation proposed, demonstrated, and evaluated a three-layered approach for three important scenarios expected in next-generation wireless communications networks that are based on the synergistic integration of SON, ML, and UC technologies. This chapter summarizes the main contributions of this dissertation.

### **6.1. Summary of the Main Contributions**

The research initiatives in this dissertation focused on developing the next-generation self-organizing communications networks that are capable of drawing insights from the user-centric network-generated data and yielding predictions via machine learning to proactively configure, manage, assess, and optimize various network functions and operations.

In the first research initiative covered in Chapter 3, QoE (Quality of Experience)-driven anomaly detection for self-organizing networks (SONs) using machine learning (ML) was applied to learn and predict a UC key performance indicator that imports the effect of the end-user



perception of the quality of service to achieve an increased level of end-to-end service assurance and proactively detect dysfunctional network nodes. This advance will enable automatic detection and remediation of failing network nodes (i.e., base stations and their associated systems) to mitigate network degradation in self-healing SON systems. The proposed methodology was demonstrated and evaluated by creating an end-to-end network scenario using the ns-3 network simulator, where end users interacted with a remote host that was accessed over the Internet to run the most commonly used applications, and applying ML to generate and validate QoE score predictions using a parametric QoE model. The performance of four supervised ML algorithms was investigated as well as their performance accuracy in making correct QoE predictions in detecting dysfunctional network nodes. The performance of these algorithms was further evaluated by creating varied network scenarios for different propagation path loss models. A sequence of multiple independent simulation runs was executed to test the methodology and detect the dysfunctional network nodes. The resulting average accuracy scores obtained for correct QoE prediction of the functional and dysfunctional network nodes ranged between 95% and 100%. This resource-efficient technique enables identification and prioritization of faulty network nodes based on an improved utilization of the end-users' perception of the state of network node performance.

In Chapter 4, the research initiative of determining and assigning optimal-capacity, shortest path routing in self-organizing networks was investigated using a user-centric, machine learning methodology called US-OCSP (i.e. user-specific optimal capacity and shortest path). This approach enables load balancing and capacity optimization in self-optimizing SON systems. US-OCSP can dynamically route the user traffic through non-congested network nodes, with minimal compromise on the subscriber's capacity, thus facilitating effective resource management to attain customer satisfaction and alleviate network congestion by developing an in-built application that

is coordinated with an automobile or mobile phone's navigation system (GPS) where a network route provided by US-OCSP is driven by the topography based on a GPS mapping system linking GPS and optimal network routing. This scenario could work as follows: the driver's GPS, or mobile phone, provides options with multiple paths going from a source to destination that are relayed to the wireless network and then the US-OCSP algorithm finds the shortest network node path with non-congested nodes recommending a routing consistent with at least one of the GPS recommended paths. The application in autonomous vehicles could be quite important. The methodology can be used to further enable dynamic network path optimization, such that if the user changes its inputs, such as GPS route, the US-OCSP would recompute the network path in response to the changes in the user inputs. Thus, US-OCSP can help build a navigation system that will allow users to pick a route with less congested network traffic and help network operators with resource optimization. The effectiveness of this methodology was demonstrated using a simulated network model whose output was used to calculate PRB utilization of network nodes followed by a reinforcement ML application. The results showed that the shortest path with optimal capacity was rapidly determined.

In the research initiative of Chapter 5, which is directed towards self-configuration of radio access network-based notification areas (RNAs<sup>27</sup>) in self-organizing networks using machine learning, a UC and ML-embedded clustering mechanism was developed for dynamic configuration and management of RNAs in self-configuring SON systems. The impact of the network load (in terms of UE to network node connections), radio conditions, and paging load was factored in and an unsupervised learning algorithm was applied to a user-centric dataset that represented the relationship between the network users and the network nodes to form RNA clusters. Performance

---

<sup>27</sup> As discussed in Chapter 5, an RNA constitutes a group of cells covered by one or more network nodes (base stations) enabling efficient paging and load management in SONs.

analysis and evaluation of the proposed technique for RNA configuration was demonstrated using a case study. For this study, a simulated network model was configured where several mobile users were connected to different network nodes as they moved around in the network during the simulation. A UC network dataset was created using the simulation output and ML was applied to form RNA clusters. The study optimized the RNA design by balancing the tradeoff between maximizing the paging load improvement and minimizing (the probability of) cluster variance. Additional recommendations were made to optimize the RNA clusters by means of performance monitoring of key performance indicators and a conceptual framework for effective RNA configuration and management in a SON system was presented. Appropriate configuration and management of RNAs using the proposed mechanism will help achieve improved signaling and paging load to attain reduced latency and improved network capacity, while lowering power consumption supporting emerging applications that generate an extensive amount of random aperiodic and keep-alive data traffic.

The ML algorithms applied to the prototypes developed and tested in this dissertation are forms of supervised learning, reinforcement learning, and unsupervised learning. For larger, more complex networks, more effective results may be obtained by combining these with deep learning that will require higher computational and processing powers.

## **6.2. Concluding Summary**

This dissertation conceived, implemented, and validated innovative methodologies for anomaly detection, load balancing and capacity optimization, and dynamic configuration of RAN-based notification areas (i.e. RAN-based paging areas) that are directed towards application in the next-generation of self-organizing communications networks. Each of these methodologies

represent the synergistic application of machine learning and user-centric technologies. Network operators capture an abundance of data about their subscribers, and ML and UC based technologies can help exploit and utilize that data in a variety of ways to improve customer experience. In order to receive the most value from ML, an appropriate selection of UC methodologies and ML algorithms on a case-by-case basis is critical so that the network operators can employ algorithms that can scale resources as per customer demand meeting service assurance within acceptable level of capital and operational expenditures. It is important to develop a certain degree of standardization in order to effectively facilitate UC and ML-enabled network configuration, management, and optimization of SON systems. Several core building blocks to support UC and ML-infused network standardization that the network operators can adopt to realize potential key features, functions, and use cases for next-generation SON networks are proposed and described in Figure 6.1.

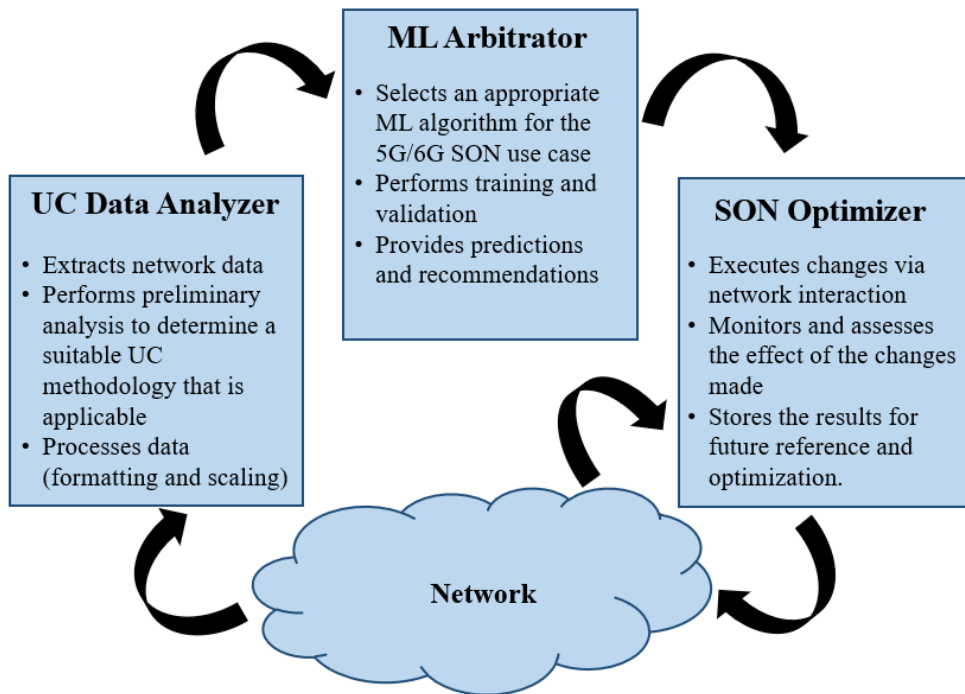


Figure 6.1 Basic building blocks providing a high-level framework to deploy next-generation SON functions and use-cases with the synergistic application of ML and UC technologies

The UC data analyzer extracts the data generated across the network, performs preliminary analysis to determine a suitable UC methodology that is applicable, and processes the data to create the input dataset for the next-generation network function or use-case to be implemented. The ML arbitrator selects an appropriate ML algorithm and performs training and validation so that the ML arbitrator can start making predictions. The SON optimizer will apply the recommendations made by the ML arbitrator via network interaction and monitors to assess the effect of the changes made in the network. The results are stored for future reference and optimization. These building blocks provide a generic guideline that can be utilized by network operators to take a holistic approach towards developing the network of tomorrow that will be tasked with delivering unprecedented levels of performance efficiencies with exceptional user experience.

## References

- [1] 3GPP, “3GPP System Standards Heading into the 5G Era.” [https://www.3gpp.org/news-events/1614-sa\\_5g](https://www.3gpp.org/news-events/1614-sa_5g).
- [2] Chetana V. Murudkar and Richard D. Gitlin, “QoE-Driven Anomaly Detection in Self-Organizing Mobile Networks using Machine Learning,” in *2019 Wireless Telecommunications Symposium (WTS)*, Apr. 2019, pp. 1–5.
- [3] Chetana V. Murudkar and Richard D. Gitlin, “Machine Learning for QoE Prediction and Anomaly Detection in Self-Organizing Mobile Networking Systems,” *Int. J. Wirel. Mob. Netw. IJWMN*, vol. 11, no. 2, p. 1, Apr. 2019.
- [4] Chetana V. Murudkar and Richard D. Gitlin, “Optimal-Capacity, Shortest Path Routing in Self-Organizing 5G Networks using Machine Learning,” in *2019 IEEE 20th Wireless and Microwave Technology Conference (WAMICON)*, Apr. 2019, pp. 1–5.
- [5] Chetana V. Murudkar and Richard D. Gitlin, “User-Centric Approaches for Next-Generation Self-Organizing Wireless Communication Networks Using Machine Learning,” in *2019 IEEE International Conference on Microwaves, Antennas, Communications and Electronic Systems (COMCAS)*, Nov. 2019, pp. 1–6.
- [6] C. Prehofer and C. Bettstetter, “Self-Organization in Communication Networks: Principles and Design Paradigms,” *IEEE Commun. Mag.*, vol. 43, no. 7, pp. 78–85, Jul. 2005.
- [7] 3GPP TR 28.861, “Telecommunication management; Study on the Self-Organizing Networks (SON) for 5G networks.” V16.0.0, Dec. 2019.
- [8] 3GPP TS 23.501, “System architecture for the 5G System (5GS).” V16.3.0, Dec. 2019.
- [9] 3GPP TS 38.300, “NR; NR and NG-RAN Overall Description.” V16.0.0, Dec. 2019.
- [10] 3GPP TS 28.550, “Management and orchestration; Performance assurance.” V16.3.0, Dec. 2019.
- [11] 3GPP, “SON.” <https://www.3gpp.org/technologies/keywords-acronyms/105-son>.

- [12] P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks," *IEEE Commun. Surv. Tutor.*, vol. 19, no. 4, pp. 2392–2431, Fourthquarter 2017.
- [13] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*. MIT Press, 2016.
- [14] S. Wang, W. Chaovallitwongse, and R. Babuska, "Machine Learning Algorithms in Bipedal Robot Control," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 42, no. 5, pp. 728–743, Sep. 2012.
- [15] Scikit-learn developers, "Scikit-Learn User Guide." Release 0.22.1, Jan. 08, 2020.
- [16] Sergios Theodoridis, *Machine Learning*. Academic Press, 2015.
- [17] Tom M. Mitchell, *Machine Learning*. McGraw Hill, 1997.
- [18] Ethem Alpaydin, *Introduction to Machine Learning*. MIT Press, 2014.
- [19] "Support Vector Machine Regression." <http://kernelsvm.tripod.com/>.
- [20] Bruce E. Hansen, "Nearest Neighbor Methods." <https://www.ssc.wisc.edu/~bhansen/718/NonParametrics10.pdf>.
- [21] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Chapman and Hall/CRC, 1984.
- [22] U. von Luxburg, "A tutorial on spectral clustering," *Stat. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.
- [23] Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, and Vipin Kumar, *Introduction to Data Mining*. Pearson, 2005.
- [24] "Network Simulator ns-3." <https://www.nsnam.org/>.
- [25] "Anaconda Software Distribution." <https://www.anaconda.com/distribution/>.
- [26] "Scikit-Learn." <https://scikit-learn.org/stable/#>.
- [27] F. Pedregosa *et al.*, "Scikit-Learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, no. Oct, pp. 2825–2830, 2011.
- [28] 3GPP TS 32.111-1, "Fault Management; Part 1: 3G fault management requirements." V15.0.0, Jun. 2018.
- [29] J. G. Andrews *et al.*, "What Will 5G Be?," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.








- [30] R. Barco, P. Lazaro, and P. Munoz, “A Unified Framework for Self-Healing in Wireless Networks,” *IEEE Commun. Mag.*, vol. 50, no. 12, pp. 134–142, Dec. 2012.
- [31] ITU-T Recommendation P.10/G.100 Amendment 2, “Vocabulary for Performance and Quality of Service.” Jul. 2008.
- [32] E. Liotou, D. Tsolkas, and N. Passas, “A Roadmap on QoE Metrics and Models,” in *2016 23rd International Conference on Telecommunications (ICT)*, May 2016, pp. 1–5.
- [33] E. Liotou, D. Tsolkas, N. Passas, and L. Merakos, “Quality of Experience Management in Mobile Cellular Networks: Key Issues and Design Challenges,” *IEEE Commun. Mag.*, vol. 53, no. 7, pp. 145–153, Jul. 2015.
- [34] Dimitris Tsolkas, Eirini Liotou, Nikos Passas, and Lazaros Merakos, “A survey on parametric QoE estimation for popular services,” *J. Netw. Comput. Appl.*, vol. 77, no. C, pp. 1–17, Jan. 2017, doi: 10.1016/j.jnca.2016.10.016.
- [35] S. Thakolsri, S. Khan, E. G. Steinbach, and W. Kellerer, “QoE-Driven Cross-Layer Optimization for High Speed Downlink Packet Access,” *JCM*, vol. 4, pp. 669–680, 2009.
- [36] Wei-Yin Loh, “Classification and Regression Trees.”  
<https://www.stat.wisc.edu/~loh/treeprogs/guide/wires11.pdf>.
- [37] L. Jorguseski, A. Pais, F. Gunnarsson, A. Centonza, and C. Willcock, “Self-Organizing Networks in 3GPP: Standardization and Future Trends,” *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 28–34, Dec. 2014.
- [38] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, “A Survey of Self Organisation in Future Cellular Networks,” *IEEE Commun. Surv. Tutor.*, vol. 15, no. 1, pp. 336–361, First 2013.
- [39] 3GPP TS 28.552, “Management and orchestration; 5G performance measurements.” V16.4.0, Dec. 2019.
- [40] 3GPP TS 36.213, “LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures.” V12.3.0, Oct. 2014.
- [41] G. Aceto, V. Persico, and A. Pescapé, “A Survey on Information and Communication Technologies for Industry 4.0: State-of-the-Art, Taxonomies, Perspectives, and Challenges,” *IEEE Commun. Surv. Tutor.*, vol. 21, no. 4, pp. 3467–3501, Fourthquarter 2019.
- [42] 3GPP TS 38.331, “NR; Radio Resource Control (RRC) protocol specification.” V15.6.0, Jun. 2019.
- [43] P. Marsch *et al.*, “5G Radio Access Network Architecture: Design Guidelines and Key Considerations,” *IEEE Commun. Mag.*, vol. 54, no. 11, pp. 24–32, Nov. 2016.




- [44] 5G PPP 5G Architecture White Paper, “View on 5G Architecture,” Version 2.0, Dec. 2017. [Online]. Available: <https://5g-ppp.eu/wp-content/uploads/2018/01/5G-PPP-5G-Architecture-White-Paper-Jan-2018-v2.0.pdf>.
- [45] 3GPP TR 21.905, “Vocabulary for 3GPP Specifications.” V16.0.0, Jun. 2019.
- [46] S. Hailu and M. Säily, “Hybrid paging and location tracking scheme for inactive 5G UEs,” in *2017 European Conference on Networks and Communications (EuCNC)*, Jun. 2017, pp. 1–6.
- [47] S. Sesia, I. Toufik, and M. Baker, *LTE - The UMTS Long Term Evolution - from theory to practice*. Wiley, 2009.
- [48] “Digital Mobile Radio: COST 231 View on the Evolution Towards 3rd Generation Systems,” Commission of the European Communities, L-2920, Luxembourg, 1989.
- [49] T. Camp, J. Boleng, and V. Davies, “A survey of mobility models for ad hoc network research,” *Wirel. Commun. Mob. Comput.*, vol. 2, no. 5, pp. 483–502, 2002.
- [50] K. Dimou *et al.*, “Handover within 3GPP LTE: Design Principles and Performance,” in *2009 IEEE 70th Vehicular Technology Conference Fall*, Sep. 2009, pp. 1–5.
- [51] D. Arthur and S. Vassilvitskii, “k-means++: The Advantages of Careful Seeding,” *Proc. Eighteenth Annu. ACM-SIAM Symp. Discrete Algorithms*, Society for Industrial and Applied Mathematics 2007.
- [52] Peter J. Rousseeuw, “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis,” *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, Nov. 1987.
- [53] P. Fränti and S. Sieranoja, “How much can k-means be improved by using better initialization and repeats?,” *Pattern Recognit.*, vol. 93, pp. 95–112, Sep. 2019.
- [54] Fan Chung Graham, *Spectral Graph Theory*. American Mathematical Society, 1996.
- [55] Bogdan Nica, *A Brief Introduction to Spectral Graph Theory*. European Mathematical Society, 2018.
- [56] David Kennedy, “Euro-5G Deliverable D2.6 Final report on programme progress and KPIs,” *5G PPP*, Oct. 2017. [https://5g-ppp.eu/wp-content/uploads/2017/10/Euro-5G-D2.6\\_Final-report-on-programme-progress-and-KPIs.pdf](https://5g-ppp.eu/wp-content/uploads/2017/10/Euro-5G-D2.6_Final-report-on-programme-progress-and-KPIs.pdf).

## Appendix A: Copyright Permissions

The permissions below are for the use of the material in chapter 3.



Home   Help   Email Support   Sign In   Create Account



**Requesting permission to reuse content from an IEEE publication**

**QoE-driven Anomaly Detection in Self-Organizing Mobile Networks using Machine Learning**

Conference Proceedings: 2019 Wireless Telecommunications Symposium (WTS)  
Author: Chetana V. Murudkar  
Publisher: IEEE  
Date: April 2019

*Copyright © 2019, IEEE*

**Thesis / Dissertation Reuse**

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

*Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:*

- 1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
- 2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
- 3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

*Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:*

- 1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
- 2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
- 3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to [http://www.ieee.org/publications\\_standards/publications/rights/rights\\_link.html](http://www.ieee.org/publications_standards/publications/rights/rights_link.html) to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW

© 2020 Copyright - All Rights Reserved | Copyright Clearance Center, Inc. | Privacy statement | Terms and Conditions  
Comments? We would like to hear from you. E-mail us at [customer@copyright.com](mailto:customer@copyright.com)

**Re: Copyright Permission**

wire Mobil <ijwmn@airccse.org>

Tue 5/26/2020 1:09 AM

To: Murudkar, Chetana <cvm1@usf.edu>

This email originated from outside of USF. Do not click links or open attachments unless you recognize the sender or understand the content is safe.

H  
i

Thanks for your email and permission is granted to use your article for PhD dissertation. Have a great day!

**Cheers!**

\*\*\*\*\*

Janelle Zara  
Editorial Secretary ,  
International Journal of Wireless & Mobile Networks (IJWMN)  
<http://airccse.org/journal/ijwmn.html>

WhatsApp : +91 9940054805

AIRCC Publishing Corporation  
<http://www.airccse.org/>

\*\*\*\*\*  
\*\*\*\*\*

**Note:**  
AIRCC's International Journal of Wireless & Mobile Networks (IJWMN) is dedicated to strengthen the field of Wireless and Mobile Networks and publishes only good quality papers. IJWMN is highly selective and maintains less than 15% acceptance rate. All accepted papers will be tested for plagiarism manually as well as by Docoloc.  
Papers published in IJWMN has received enormous citations and has been regarded as one of the best Journal in the Wireless & Mobile Network research field.

On Tuesday, May 26, 2020, 12:04:48 AM GMT+1, Murudkar, Chetana <cvm1@usf.edu> wrote:

Dear Publisher,

I would like to request copyright permission to reuse the content of my paper titled 'Machine Learning for QoE Prediction and Anomaly Detection in Self-Organizing Mobile Networking Systems' that was published in the International Journal of Wireless & Mobile Networks (IJWMN) in Vol. 11, No. 2 in April 2019 for my Ph.D. dissertation.

Thanks & Regards,  
Chetana V. Murudkar

### User-Centric Approaches for Next-Generation Self-Organizing Wireless Communication Networks Using Machine Learning



Conference Proceedings:  
2019 IEEE International Conference on Microwaves, Antennas, Communications and Electronic Systems (COMCAS)

Author: Chetana V. Murudkar

Publisher: IEEE

Date: Nov. 2019

Copyright © 2019, IEEE

#### Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

*Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:*

- 1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
- 2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
- 3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

*Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:*

- 1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
- 2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis online.
- 3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to [http://www.ieee.org/publications\\_standards/publications/rights/rights\\_link.html](http://www.ieee.org/publications_standards/publications/rights/rights_link.html) to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK

CLOSE WINDOW

The permission below is for the use of the material in chapter 4.



RightsLink®



Home



Help



Email Support



Sign in



Create Account



### Optimal-Capacity, Shortest Path Routing in Self-Organizing 5G Networks using Machine Learning

Conference Proceedings:

2019 IEEE 20th Wireless and Microwave Technology Conference (WAMICON)

Author: Chetana V. Murudkar

Publisher: IEEE

Date: April 2019

Copyright © 2019, IEEE

#### Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

*Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:*

- 1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
- 2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
- 3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

*Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:*

- 1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
- 2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis online.
- 3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to [http://www.ieee.org/publications\\_standards/publications/rights/rights\\_link.html](http://www.ieee.org/publications_standards/publications/rights/rights_link.html) to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK

CLOSE WINDOW

## Appendix B: Abbreviations

3GPP	3rd Generation Partnership Project
4G/5G/6G	Fourth Generation/Fifth Generation/Sixth Generation
AI	Artificial Intelligence
AMF	Access and Mobility Management Function
CAPEX	Capital Expenditures
CART	Classification And Regression Trees
CBR	Constant Bit Rate
CIR	Committed Information Rate
CN	Core Network
C-SON	Centralized Self-Organizing Network
DCI	Data Control Indication
DHCP	Dynamic Host Configuration Protocol
DRX	Discontinuous Reception
D-SON	Distributed Self-Organizing Network
DT	Decision Tree
E2E latency	End-to-End latency
eNB	Evolved Node B
E-UTRA	Evolved Universal Terrestrial Radio Access

FTP	File Transfer Protocol
gNB	Next-Generation Node B
GPS	Global Positioning System
HO	Handover
H-SON	Hybrid Self-Organizing Network
IoT	Internet of Things
IP	Internet Protocol
ITU	International Telecommunication Union
$k$ -NN	$k$ -Nearest Neighbor
KPI	Key Performance Indicator
L-BFGS	Broyden–Fletcher–Goldfarb–Shanno algorithm
MAC	Multiple Access Control
MAE	Mean Absolute Error
MCS	Modulation Coding Scheme
MDAS	Management Data Analytics Service
MIMO	Multiple Input Multiple Output
ML	Machine Learning
MLP	Multi-Layer Perceptron
MOS	Mean Opinion Score
MSE	Mean Squared Error
NF	Network Function
NG	Next Generation
ng-eNB	Next-Generation Evolved Node B

NN	Neural Networks
NR	New Radio
ns-3	Network Simulator-3
OPEX	Operational Expenditures
OTT	One Trip time
PCF	Policy Control Function
PRB	Physical Resource Block
QoE	Quality of Experience
QoS	Quality of Service
RACH	Random Access Channel
RAN	Radio Access Network
RB	Resource Block
RBF	Radial Basis Function
RF	Radio Frequency
RL	Reinforcement Learning
RNA	Radio Access Network-Based Notification Area
RRC	Radio Resource Control
RSRP	Reference Signal Received Power
RTT	Round Trip Time
SINR	Signal-to-Noise-and-Interference Ratio
SL	Supervised Learning
SMF	Session Management Function
SON	Self-Organizing Network



SVM	Support Vector Machine
TB	Transport Block
TBFQ	Token Bank Fair Queue Scheduler
TCP	Transport Control Protocol
TR	Technical Report
TS	Technical Specification
UC	User-Centric
UDM	Unified Data Management
UE	User Equipment
UL	Unsupervised Learning
UPF	User Plane Function
US-OCSP	User-Specific Optimal Capacity and Shortest Path
VR	Virtual Reality

## Appendix C: Glossary<sup>28</sup>

**Access Stratum:** The access stratum is located between the edge node of the serving network domain and the UE domain and provides services related to the transmission of data over the radio interface and the management of the radio interface.

**Anchor gNB:** An anchor gNB is the network node that is aware of or has the list of all the gNBs that are a part of that RNA (radio access network-based notification area). It is the anchor gNB that maintains the CN-RAN connection and the UE context as the UE moves around within the RNA.

**Channel bandwidth:** The RF bandwidth supporting a single RF carrier with the transmission bandwidth configured in the uplink or downlink of a cell.

**Connection:** A communication channel between two or more end-points.

**Core network:** An architectural term relating to the part of 3GPP System which is independent of the connection technology of the terminal (e.g. radio, wired).

**Coverage area:** Area over which a mobile cellular service is provided with the service probability above a certain threshold.

**Discontinuous reception (DRX):** Discontinuous reception is a power saving feature where paging cycles can range from seconds to several hours, depending on the radio access technology.

**Downlink:** A unidirectional radio link for the transmission of signals from a base station to a UE.

---

<sup>28</sup> Much of the contents of the glossary are taken from [45].

Handover: The transfer of a user's connection from one radio channel to another (can be the same or different cell).

Machine learning (ML): Machine Learning is the ability of systems to acquire and continuously improve their own knowledge, by extracting patterns from raw data to address problems involving knowledge of the real world and make decisions that appear to be subjective and mimic human "cognitive" functions.

Medium access control: A sub-layer of radio interface layer 2 providing unacknowledged data transfer service on logical channels and access to transport channels.

Mobility: The ability for the user to communicate whilst moving independent of location.

Network element: A discrete telecommunications entity which can be managed over a specific interface.

Network node (Base station): A network node or a base station is a network element in radio access network responsible for radio transmission and reception in one or more cells to or from the user equipment.

Network operator: The entity which offers telecommunications services over an air interface.

New radio (NR): Fifth generation radio access technology

Node B: A logical node responsible for radio transmission / reception in one or more cells to/from the user equipment.

Paging: The act of seeking a user equipment.

Parametric QoE models: Parametric QoE models are derived by performing subjective experiments that may include laboratory tests or crowdsourcing and by performing statistical analysis on the results. The derived models may then be used to generate formulas which can be used to compute QoE given specific input parameters.

**Physical resource block (PRB):** A physical resource block is a group of resource elements such that it consists of 12 consecutive subcarriers across one slot. A PRB is the smallest radio resource unit used for resource allocation in 4G and 5G networks.

**Ping-pong effect:** When UEs are in a loop where they are repeatedly handed off between the source and the target base stations, the resulting effect is called the “ping-pong” effect.

**Power saving mode:** A mode of operation similar to power-off, allowing a UE to greatly reduce its power consumption while remaining registered with the network, without the need to re-attach or to re-establish packet data network (PDN) connections.

**Protocol:** A formal set of procedures that are adopted to ensure communication between two or more functions within the same layer of a hierarchy of functions.

**Quality of experience (QoE):** QoE can be defined as the overall acceptability of an application or service, as perceived subjectively by the end-user.

**Radio resource control:** A sublayer of radio interface Layer 3 existing in the control plane only which provides information transfer service to the non-access stratum. RRC is responsible for controlling the configuration of radio interface Layers 1 and 2.

**RAN-based notification area (RNA):** An RNA constitutes a group of cells covered by one or more network nodes (base stations) enabling efficient paging and load management.

**Self-organizing network (SON):** SON automates network functionality to realize a network that can autonomously configure its entities, self-optimize, and self-heal with little to no human intervention thus minimizing operational and capital expenditures.

**Signaling:** The exchange of information specifically concerned with the establishment and control of connections, and with management, in a telecommunications network.

Throughput: A parameter describing service speed. The number of data bits successfully transferred in one direction between specified reference points per unit time.

Transport block: Transport Block is defined as the basic data unit exchanged between Layer 1 and MAC.

Uplink: A unidirectional radio link for the transmission of signals from a UE to a base station.

User equipment (UE): A UE allows a user access to network services.

User-centric (UC) technology: A user-centric technology is where users are not mere end-points but are an integral part of the network such that the network strategies are based on user needs, network optimization is based on user feedback, and network performance is monitored and assessed via user-focused key performance indicators.

## **About the Author**

Chetana V. Murudkar is a graduate student pursuing Ph.D. in Electrical Engineering at University of South Florida (2018-present) under the supervision of Dr. Richard D. Gitlin where she is a part of the Advanced Wireless Networking group investigating new technologies that address research challenges directed towards emerging 5G and future 6G cellular systems. She is an Engineer at Sprint/New T-Mobile (2015-present) where she has led and/or worked on several projects that involve the design, deployment, performance assessment, and optimization of its multi-technology, multi-band, and multi-vendor network. Her past work experience includes working at AT&T Labs (2012) where she worked in the Radio Technology and Strategy group whose main task was to conduct applied research and develop new technology/algorithms in the area of cellular networks and working with Ericsson (2012-2014) where she worked in the RAN department supporting vendor-specific deployment and optimization projects for AT&T, Sprint, and Clearwire. She received a Master of Science degree from Southern Methodist University in Telecommunications Engineering (2011-2013) and a Bachelor of Engineering from University of Mumbai in Electronics and Telecommunication Engineering (2007-2011).